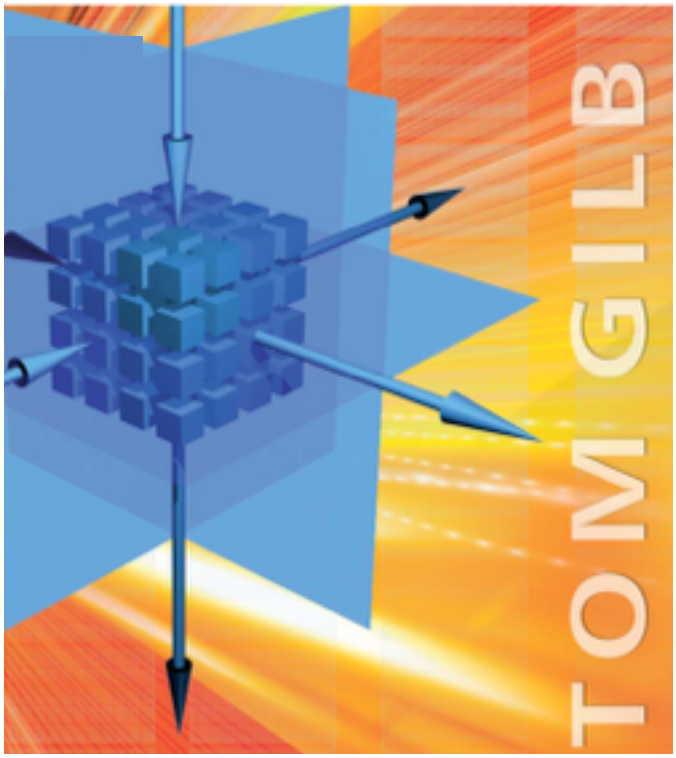
Version December 31 2015. Comments to tom@Gilb.com
tinyurl.com/ValuePlanning

Value Planning



Practical Tools
for
Clearer Management Communication

The One-Page 'Value Planning' Book.

Why? I believe your time is valuable. I believe that if someone is an expert or master of a subject, they can write it down in one page or less. So, to potentially save you the time, of reading the rest of the book, I'll try to do a 1-page version right here and now. If you need more detail later, you know where to find it.

Sound Bite

Deliver Real Stakeholder Value Now

The One Sentence Summary.

Value Planning (VP) means you will elicit and clarify critical stakeholder values quantitatively, and prioritize delivering those values, as soon as possible.

The One Paragraph Summary.

1. **STAKEHOLDERS**: Identify your most critical stakeholders.
2. **OBJECTIVES**: Identify the smart levels of their most critical value improvements.
3. **STRATEGIES**: Identify potential strategies for delivering planned value levels to stakeholders, at lowest cost and risk.
4. **SMALL STEPS**: Decompose strategies into suitably smaller deliverable increments.
5. **DELIVER VALUE**: Attempt to deliver measurable value to some stakeholders.
6. **LEARN**: Measure results and costs; then decide if you are on track, or need to change something. *Continue the process until all goals reached.*

The Rest-of-the-Page Summary.

1. We will make use of our Planning Language, called 'Planguage' ('PL').
2. The central capability of Planguage is that it can be used for *any system* of 'product' or 'service', at any level of abstraction or detail.
3. Planguage is capable of expressing *all results, improvements, values and qualities quantitatively.*
4. Planguage can help you plan, estimate and track delivery of *all costs* and resources.
5. Planguage will help you keep numeric accounts of *multiple critical values*, and corresponding *multiple critical resources*, so you can manage value for money; i.e. the *efficiency* of planning, decision-making and contracted result deliveries.
6. Planguage is extremely *risk* conscious at the level of every aspect of planning that might involve risk to your successful value delivery.
7. Planguage not only helps with planning values and costs, but is consequently used to manage practical *implementation*, learning and *feedback* from plan application.
8. Planguage will help you *align* and connect plans at many *related levels* of consideration, from top management to the most detailed level of planning you need.
9. Planguage enables you to *measure the quality of planning*, and to set a release threshold for plans.
10. Planguage has tools to *automate* plan specification, and to integrate your updated decisions and knowledge.

Technical Detail and Real Examples:

My TEDx Talk http://tinyurl.com/GilbTedx, "All Qualities Can Be Quantified". 18 minutes.

Dedicated

To

Solveig Grethe Schjelle Gilb

Who deserves to have this book dedicated to her, since she happily is planning to put up with me when I am writing it, and when I am not!

Introduction to the Book Structure and Intent.

FEEL FREE TO SKIP THIS AND DIVE INTO THE BOOK'S TECHNICAL CONTENT. You Can Start Anywhere At Random, Scan A Bit, Or Try The Beginning 'Part 0. Vision Engineering: Introduction'

Book Intent.
This book, with a very detailed appendix, is intended for fairly deep training, or self-training, in the Planning Language.

What type of People might benefit from this book.
'Planguage' [www.gilb.com/dl831], is useful for anyone doing **management**-level planning, at 'C' level or large-scale project decision-making.

- (as opposed to Engineering/Technical/Product planning, for which we have written 'Competitive Engineering', 2005, [1]).
- The technical and engineering manager (CTO) will find the 'Value Planning ('VP') level comfortable for communicating with other C-level executives.

The book assumes the reader has interest in planning for *organizational improvement*, and for the management responsibility for *large projects*, including *technical* projects (from the *manager* point of view). We assume that readers are open for learning about some more-powerful tools to plan with.

Book Organization.
The book is organized into a core presentation ('Vision Engineering') and an Appendix of 10 Chapters. Each Chapter is further divided into 10 Sections. Each Section is focused on a specific *principle* of management and planning. Each one of those principles is then explained in a few-page Section.
Where possible, each principle will be illustrated by practical examples of using Planguage, to articulate a planning aspect. We add small stories, and some 1 page Case Studies, from our practice.

Section Content and Structure.
Each section contains Case Studies with real numeric experience data when possible, from our own client experience.

We have a lot of *deep theory* (Deming called it 'Profound Knowledge') for those who want to become more expert in planning and communication.

By 'deep theory' I have tried to make points which will not be obvious to the reader, without considerable practice and thought.

But the deep theory will give useful immediate insights for the reader who is capable of appreciating and applying the deeper insights. Deep theory may give you prudent warnings about pitfalls, and useful tips about possible applications of the tools.

Deep theory can save you time getting to the expert level. Sometimes deep theory is more usefully learned *after* you have some practice with the methods. But if you read it immediately, it might just stick in your subconsciousness, ready to emerge when you need it. Or it will be ready to be re-read by you, and practiced.

We try to include frequent conceptual diagrams to aid understanding.

Each Section ends with a quotation of advice from some of the most interesting people on the planet.

Then each Section suggests a 'Practical Tip': ideas for something you might do 'today' to get going and try out the ideas.

Finally each Section ends with a suggestion for a management Policy of Planning, with a 'Why?' this policy justification.

The Main Book: 'Vision Engineering'.
The 'main book' is designed for one continuous reading (an evening?, a flight?), and gives a top-manager summary of the rest of the book. It can be read first and independently.

References
We point to extensive further writings by ourselves and others, for those who wish to go deeper. These references are mainly in the form of links to free papers, cases, videos, and our own teaching slides (all of which you are encouraged to reuse with credit).

Planguage Philosophy
As you progress in the book, we hope that in addition to the practical Planguage 'plan articulation' tool, you will pick up a set of practical 'planning **philosophy**' ideas.

We hope this will change your view of planning from 'conventional thinking' to a view of *dynamic measurable value delivery to stakeholders.*

Teaching Planguage with this base

We hope that the material will provide a solid toolset for teachers to teach these ideas, especially in private business, or in public sector organizational planning. If successful, we, as a community, will do less 'valueless' work, and *deliver more value to our fellows.*

It takes many decades to turn around the 'Titanics' of Conventional Thinking. But the good news is that our experience has shown, that we can do change and improvement to planning practices, one person at a time, one project at a time, one Corporation at a time, one country at a time. So I hope the reader will 'just do it'; because it is the right thing to do!

A Tough Read, but Hopefully Worth It.
This book is **not** a *lightweight* overview of our ideas. It is intended for serious expert study.

Many of the references can serve the purpose of lightweight introduction to some of the main subjects, and some specialities. (URL9 18 minute TEDx, or URL8 A. (short), or B. Longer).

We do have a plot to write a short 'top manager' book, which supports and points to this book, for practical detailed practices.

This text gives the detail needed to become an expert practitioner, consultant or teacher of Planguage.

Competitive Engineering: The Planguage Standards Basis.
The formal foundation textbook on Planguage, with a great many "standards' (Glossary of 671+ concepts, Rules, Processes, Principles, Templates), to back it up, is in the *Competitive Engineering* book [1]. We do not intend here to replicate that level of detail of Planguage 'Standards' (Rules, Processes, Glossary, Measures). But we do advise the expert or teacher to have a copy of CE for background reference.

Reading Sequence.
This book is intended as a sequential logical progression of subjects. But it should be possible to dip in anywhere of interest at random.

My Personal Motivation.
My personal reason for writing this book is to share my insights from almost six decades of practice (1958-2015).

I have *not* tried to make this book 'popular, fashionable, brief, or simple for simple minds'. It is not about sales, marketing, or income from spinoffs. It is only about sharing deep, practical, and useful knowledge with people mature enough to appreciate it.

You yourself should feel free to make Planguage, and any single tool set within it, simpler and more approachable to your own students, clients, listeners and readers. Several of our best clients have done that.

You are welcome to add any idea here to any set of ideas or methods that is already your basis. There is no all-or-nothing about Planguage. It is a toolset, and you can pick any tool you like, and use it for any useful purpose you like. But, do no evil! (please).

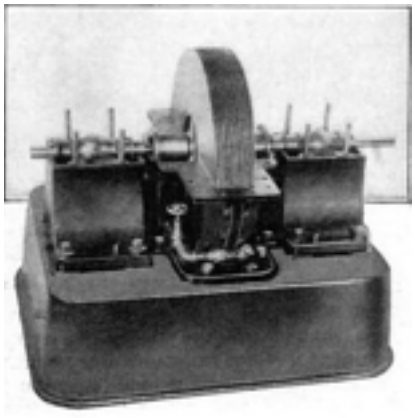
If you want to enhance your earning power that's great. But I hope you are most motivated by doing the *right thing*, and making the world a better place. You can do that by improving your ability to deliver more value for less cost.

I have tried to go into *depth* for the expert, the teacher, the coach, the writer, and the intellectually curious. I have tried to share ideas that will have a *very long half-life*. Way beyond our lifetime, would be nice.

"I would not give my rotating field discovery for a thousand inventions, however valuable... A thousand years hence, the telephone and the motion picture camera may be obsolete, but the principle of the rotating magnetic field will remain a vital, living thing for all time to come."

— Nikola Tesla
"A Famous Prophet of Science Looks Into the Future" (Popular Science Monthly, November, 1928)
http://www.teslauniverse.com/nikola-tesla/images

The 200-Hp. Tesla Turbine, with the Upper Section of the Case Removed to Show the Disks, etc.



Tesla in his office in 1916, demonstrating an electrical apparatus.
Photo date: 1916

Note: the rotating magnetic field is the heart of his induction motor and alternating current electricity. It is also a basis for MRI technology,

Enjoy, practice, share, and change the world, a little bit more.

The One-Page ‘Value Planning’ Book. 2

Introduction to the Book Structure and Intent. 4

Part 1. Our ‘Vision Engineering’ Objectives. 21

Part 2. Quality Control of Objectives 39

Part 3. Strategies. 52

Part 5. Evolutionary Deployment of Strategies 70

Value Planning Appendix: Chapters 1 to 10 76

Chapter 1. Objectives. 77

Section 1.1 Quantification of All Critical Performance Objectives, Especially Qualities.. 78

Section 1.2: On Limiting Your Project to 10 Critical Objectives, Initially. 89

Section 1.3 Clearly connecting the level of responsibility with objectives. 99

1.4 Making the degree of supporting planning for your objectives visible 104

Section 1.5 Alignment. Every plan, including your own, has to be clear about what other objectives it is supporting. The 12 Tough Questions tool. The 13th Question the ‘Record’ statement. 107

Section 1.7 Some Objectives are ‘Compound’: they consist of a ‘set’ of scalar variables. 121

Section 1.8 Object-Oriented Planning: Extracting Presentations from the Plan database 128

Section 1.9 Multidimensional Targets: ‘decomposition’ and ‘prioritization’ tools using ‘qualifiers’. Down-sizing plans for experiments and short high-value delivery cycles. 136

Section 1.10 Benchmarks: warning signals for planners. 147

Chapter 2. Strategies. 152

Section 2.1 How to know if a strategy will really work 154

Section 2.2 Why we have to worry about strategy side-effects, and how to worry. Responsibility for the whole effect picture. 163

Section 2.3 Looking at Multiple Impacts of A Strategy; And doing so ‘after the battle has begun’: Dynamic strategy selection. 178

Section 2.4. Evaluating the multiple cost aspects of strategies, in relation to project resource budgets. The simple 0 to 9 method. 188

SECTION 2.5 Strategy choice depends on residual resources. Dynamic Design to Cost. 195

SECTION 2.6 You cannot use a strategy that violates serious constraints. 202

SECTION 2.7. Divide big ideas into much smaller value-delivery increments. 205

SECTION 2.8 Delivering A Stream of Value Improvements to your 'bad old system', is likely to be more successful than building a big new system. 212

SECTION 2.9 Strategy Risks: what can go wrong with those great ideas? 216

SECTION 2.10 Ends and Means distinction. 224

Chapter 3. Levels of Interest and Levels of Control. 231

Section 3.1 The Plan Relationships: your links to others. 232

Section 3.2 'Who' decides , or 'how' do we decide, on which strategies to use? 238

Section 3.3 Focus on Improvements. 243

Section 3.4 Aligning Levels of Responsibility Numerically for Values and Costs. 250

Section 3.5 How many strategies do you need to reach your objectives? 259

Section 3.7 The problem of your Stakeholders' unclear communication about their objectives. 270

Section 3.8 Getting 100 times clearer objectives. 277

Section 3.9 Dealing with your many stakeholders 286

Section 3.10 It is not just about primary stakeholders, we need to include stakeholders with less power, but valid needs. 289

Chapter 4. Value Delivery. 292

Section 4.1 Value is Subjective. 293

Section 4.2 Tailoring Value Planning for Better alignment with the Stakeholder and Strategy Environment. 299

Section 4.3 Conflicts of Interest. 306

Section 4.4 Estimations of Strategy Impact: Uncertainty. 311

Section 4.5. Dynamic Strategy Planning, to Fit Costs. 317

Section 4.6 When the map differs from the terrain, believe the terrain. 331

Section 4.7 Focusing on real value-for-investment has rewards. 338

Section 4.8. How to protect the long-term values. 342

Section 4.9 Values conflict with each other because of the 'common purse'. 349

Section 4.10 The Never-Ending Quest for Value Improvement 356

Chapter 5. Decomposition (by value, by responsibility) 361

Section 5.1 Strategy decomposition for early value delivery. 363

Section 5.2 Decomposing strategies into independent implementations. 368

Section 5.3 The Impact Estimation Table as a value decomposition tool. 375

Section 5.4 The 'Scale' parameters as a tool for decomposition of strategies 380

Section 5.5 The value of the 'bad old system' as a starting point. 383

Section 5.6 Specific tactics to enable you to decompose strategies 389

Section 5.7 A simple checklist for practical strategy decomposition. 393

Section 5.8 The 'Project Startup Week' Process. 397

Section 5.9 Value delivery steps are also a 'hypothesis' we need to verify. 404

Section 5.10 Proving you can really deliver value: one step at a time. 409

Chapter 6. Prioritization, Evaluation 415

Section 6.1 A Better Way to Prioritize, and make decisions. 416

Section 6.2 Priority is dependent on many changing factors 423

Section 6.3 Articulate your priority policy, and adapt it. 429

Section 6.4 Quantify All Your Critical Objectives AS A BASIS FOR DECISION MAKING. 434

Section 6.5 Value Management Information for Prioritization. 439

Section 6.6 How to Prioritize in Both Long and Short Term 449
Section 6.7 Using the IE Table for Measurement and Feedback: The Meter. 454
Section 6.8 Our Priority Decisions Can be Computed 461
Section 6.9 The Five Levels of Priority 469
Section 6.10 Accepting a stakeholder Wish as a committed Goal. 474
Chapter 7. Risk Management 478
Section 7.1 Risk Management is universal and continuous. 480
Section 7.2 Reducing Risk 486
Section 7.3 Continuous learning, improving and grass roots power to change. 491
Section 7.4 Navigating projects over troubled waters 500
Section 7.5 Asking powerful questions about risk. 509
Section 7.6 The holistic systems view of risk planning. 518
Section 7.7 Spec Quality Control to Reduce Specification Defect Threats: ‘very upstream’ risk management 524
Section 7.8 Fighting the root causes of risks is good economics. 532
Section 7.9 The value of overwhelming value 540
Section 7.10 Focussing on overall value progress. 547
Chapter 8. Delegation, Outsourcing, Contracting 554
Section 8.1 Delegating Decisions 555
Section 8.2 Supporting People who you delegate decisions to. 562
Section 8.3 Smart ‘Value Contracting’. 569
Section 8.4 Managing by Results 573
Section 8.5 Removing barriers to delivering value 580
Section 8.6 Getting early short-term feedback. 586
Section 8.7 Crediting ideas and motivating responsibility. 593
Section 8.8 Delegation of planning activity and responsibility. 600
Section 8.9 Managing Stakeholder Awareness 606
Section 8.10 Making sure you get ‘value for money’ 612
Chapter 9. Communication 620

Section 9.1 Written Communication 622

Section 9.2 Capturing Background Information in your Plan 629

Section 9.3 Traceability: Document Relationships 636

Section 9.4 Manage the Degree of Value Improvement 643

Section 9.5 I'm from Missouri, you gotta show me! 653

Section 9.6 Dealing with the range of uncertainty pessimistically. 663

Section 9.7 Connecting Stakeholder Levels 673

Section 9.8 The incremental value delivery of a series of strategies; stepwise learning. 678

Section 9.9 How to play safe, with a Safety Factor. 686

Section 9.10 The 'Project Startup Week': Necessary Minimum Planning, then into battle. 695

Chapter 10. Quality Management. 709

Section 10.1 Defining which qualities any plan should have; the 'rule of rules'. 711

Section 10.2 Spec QC: Measure how bad your planning communication quality is. 718

SECTION 10.3. Prevent Planning Viruses Going Downstream Contagiously 728

Section 10.5 Simple Quantified Plan Quality- Control Process: 'Spec QC'. 745

Section 10.6: 'Quality' and other related fundamental concepts. 753

Section 10.7 Enriching the planning specification of quality values. 770

Section 10.8 The attributes of a planning language. 782

Section 10.9 WHAT PROFESSORS SHOULD REALLY TEACH: Lasting Wisdom. 791

Section 10.10 WHAT PROFESSORS SHOULD REALLY TEACH: Clear, deep, useful sets of professional concepts. 799

‘Vision Engineering’: Top Level Value Planning

This is a very short book, in 5 Parts. About 50 pages.

The following ‘Core’ book is suitable, we hope, for a manager, who wants to get a practical non-superficial feel, for the methods in this book.

For those who wish more detail, and also to provide a tool to the manager who wishes to spread these ideas in their organization, we provide the comprehensive 10-Chapter, 100 Section Appendix, in back of the book.

You have the author’s permission, and encouragement, to digitally copy or print out, the Core, ‘Vision Engineering’ and send it to anybody (like your own top management), as a *sample* of this set of Value Planning ideas and methods.

CONTENTS

Part 0. Vision Engineering: Introduction

Part 1. Our ‘Vision Engineering’ Objectives.

Part 2. Quality Control of Objectives

Part 3. Strategies.

Part 4. Quality Control of Strategies

Part 5. Evolutionary Deployment of Strategies

Part 0. Vision Engineering: Introduction

The most visionary visions of top executives and corporations can usefully be clarified, primarily through quantification [URL2].

Vision Engineering:
How to support your core business vision by much clearer practical plans and actions.

Introduction
In the best-selling management book 'Built To Last' [1] a convincing case is made that long-term successful organizations ("Visionary Companies") will derive their shorter-term objectives, strategies, and policies from the Core Values and Core Purpose (Core Ideology).

They need to make sure that all consequent activity is strongly aligned to these core values, and the core purpose.

This book will outline a set of specific practical 'management engineering' tools, for articulating the Core-supporting objectives and strategies, and for making sure that they are, in fact, aligned to the 'core'.

We hope the knowledgeable reader will find the tools *useful* and *practical*.

Here is a summary of the concepts of 'Core Ideology'.

Core Ideology (Core Values + Core Purpose)
Core ideology has the *highest priority* in planning, since "Once you're clear about the core ideology, you should feel free to change absolutely *anything* that is not part of the core ideology." [1, p.231]

Core Values
Core Values [1] are 'a handful of basic ingrained beliefs' about how to run a given organization. They are almost unchangeable for the lifetime of the organization. They are the strongest guidelines that all other activity in the organization must try to align with. They are the highest priority. The 'core values' can be viewed as **'constraints'** on planning and activity. Anything that violates these constraints is not acceptable; is out of 'alignment' with the core values.

Core Values Examples [1, 4]

Sony Core Values**:** [1, p.237]
- *Elevation of Japanese national culture and status.*
- *Being a pioneer – not following others, but doing the impossible.*
- *Respect and encouragement of individual ability and creativity.*

Merck Core Values: [1, p.236]
- *Corporate social responsibility.*
- *Unequivocal excellence in all aspects of the company.*
- *Science-based innovation.*
- *Honesty and integrity.*
- *Profit, but profit from work that benefits humanity.*

Core Purpose
A Core Purpose is the most fundamental 'why?' the organization exists.

A Core Purpose can be viewed as the most fundamental *objective*; the most fundamental *measure* of progress or improvement; the measure that all supporting strategies intend to improve.

We must always aim to improve in that dimension, while keeping loyal to our core value constraints.

Core **Purpose** Examples [1, 4]

Sony: *"To experience the sheer joy of innovation and the application of technology for the benefit and pleasure of the general public".*

Merck*: "To preserve and improve human life".*
All of our actions must be measured by our success in achieving this goal. [1, p.236. p.89]

Hewlett Packard: "To make technical contributions for the advancement and welfare of humanity"

Wal-Mart: "To give ordinary folk the chance to buy the same things as rich people"

Walt Disney: "To make people happy."

"It's not hard to make decisions when you know what your values are."
— Roy Disney (Famous for firing top executives based on these values)

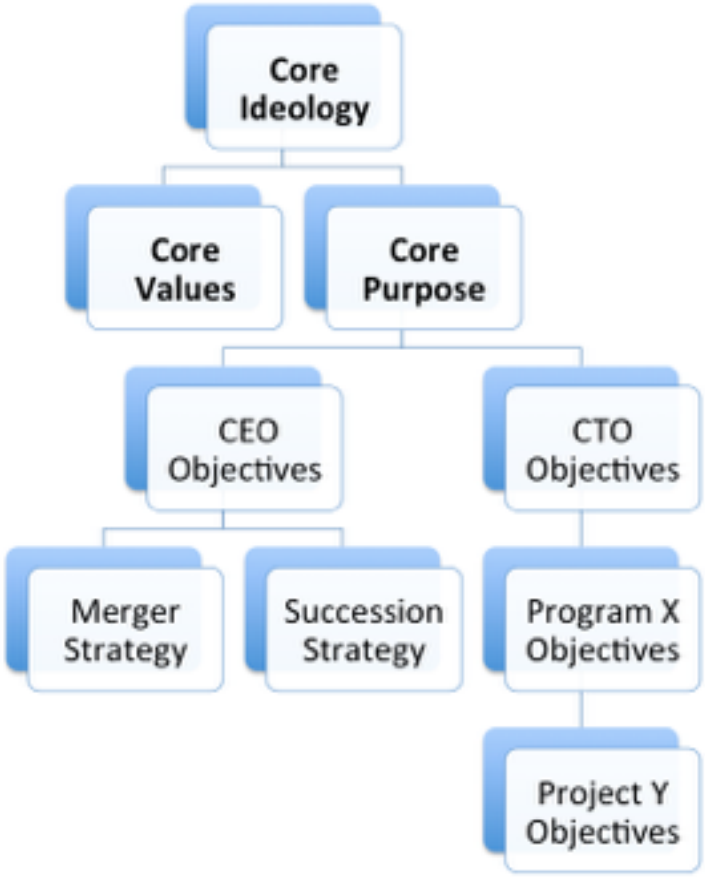


Figure A. Core Ideology.

Notice that the core purpose serves *two distinct areas*.

It defines a core **mission** (what we do) like 'make people happy'.

But it also implies that 'improvement is better', within constraints (core values). So it gives a potentially measurable '**performance** dimension'.

The more people made happier, the better. The more total happiness (people x degree of happiness) the better. The less pain or disease, the better.

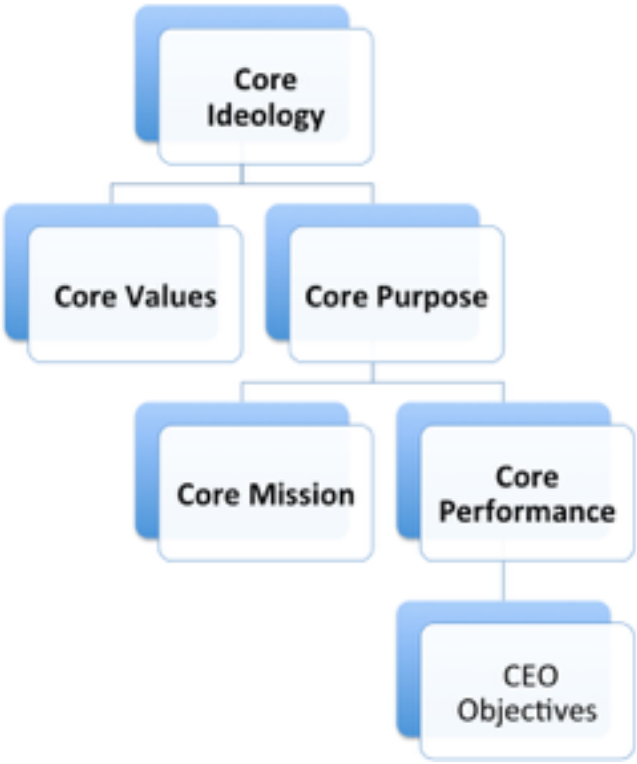


*Figure B The 2 components of Core Purpose. **What** you are (mission), and what you want to **improve** (performance).*

Summary: Core Ideology

The reader is referred to the 'Built To Last' [1, 4] literature for substantial detail on core ideology. We assume the reader grasps the basics, and is motivated to look at some potential tools for management planning and action, based on the platform or framework of a 'Core Ideology'.

Our main purpose in this book is

- to show some practical ways to make management plans clearer (so plan *readers 'cannot misunderstand'*),
- to show how to check the quality of planning (*are plans unambiguous, complete, consistent, powerful?*), and
- to show how to follow up the plans in practice (*do they work?)*

Part 1. Our 'Vision Engineering' Objectives.

Clarifying Core Ideology
We can begin with the core ideology statements themselves. It might be healthy to increase the *clarity* and *intelligibility* of the core *value* statements, and the core *purpose* statements, themselves.

This can either be done as a *clarification*, without changing the original statements themselves – they may be somewhat 'holy' and traditional – or you might find it advantageous to *directly modify* them for clarity.

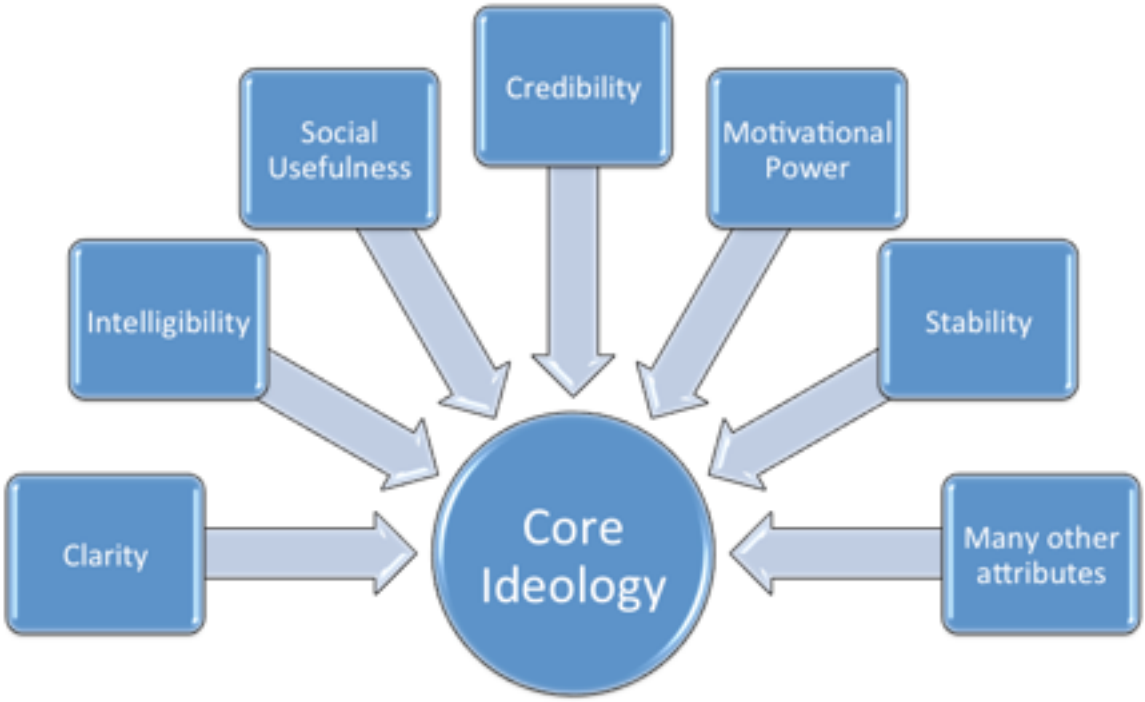


Figure 1 A. Some attributes of a Core Ideology.

Clarifying a Core Value Statement
Take for example a core *value* statement:

3M [1, p.68, p.152-3] **"Tolerance for Honest Mistakes"**

Anyone could reasonably ask:

- How much 'tolerance'?
- What does 'honest' mean?
- What is a 'mistake'?

No doubt, the corporate practice itself, and senior employees, can answer these questions in practice.

But let us say that we wanted to clarify the 3M Core Value *even better.* We might need to clarify it because of rapid corporate growth, in distant cultures – so people got it better, faster?

We might also want to clarify as a better basis for deriving more-detailed plans and practices.

We might rewrite the Core Value, or provide helpful interpretation and commentary.

Why?
A useful approach to clarification is to ask 'why?'

In this case the reasons (for the 3M 'tolerate honest mistakes') seem to be to *encourage experimentation*, so that improved ideas are more likely to emerge, than if people were afraid of being criticized for failed experiments.

So we could rewrite the core value, in order to get *nearer* the real 3M intent:

"Judge efforts on their <u>useful outcome</u>, not on necessary experiments to get there." *"Judge <u>results</u>, not process".*

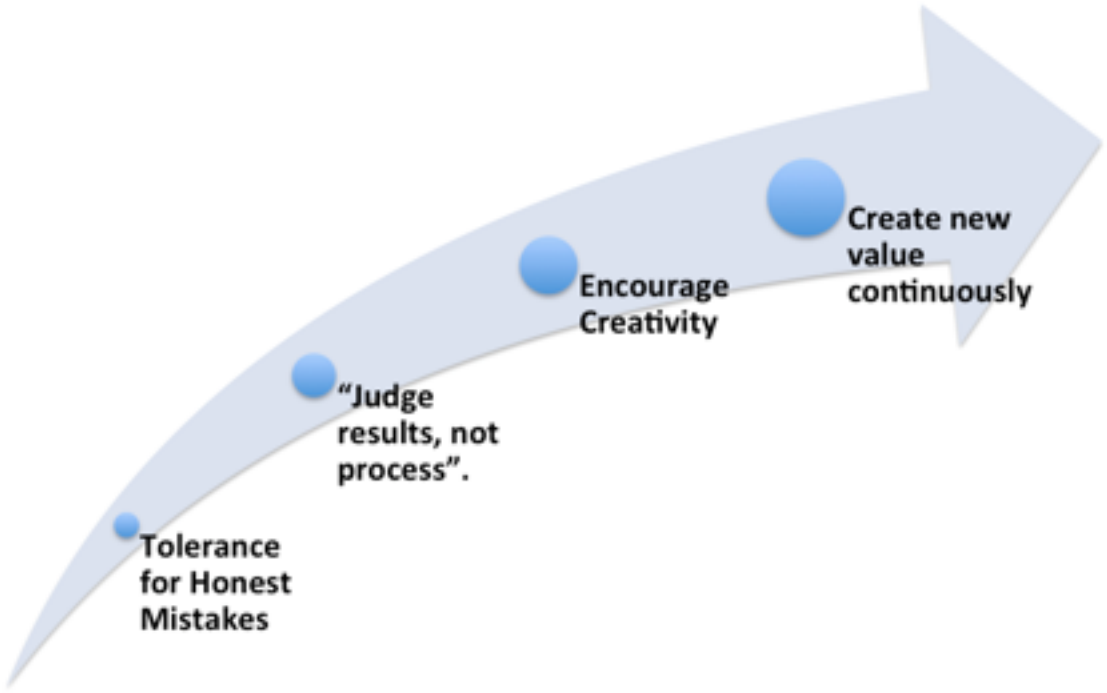


Figure 1 B Asking 'Why?', multiple times, is not only a practical way to 'clarify' a core value. It might easily lead to your own recognition, that you need to reformulate your core values at a higher level. What you had originally, might have been but one means (tolerate mistakes) to the real ends (create value). Here is a speculative example.

There are a large number of other possible methods for clarification of core values, and indeed any planning statement, at any level. More follows in the rest of this book, and its references.

The main point is that no matter how 'hallowed' the statement is ("All men are created equal.") you should consider as your first step, some clarification of the core statement itself, maybe a real 'elevation' to a higher level of purpose: it is 'core', right? So it may be worth it.

The penalty, if you fail to clarify, might be that **all other** critical planning will be based on *misinterpretations* of the core! The cost to get it right is small; like an hour to a day of effort.

Clarifying a Core Purpose
It is arguably *very* critical to have a rock-solid, crystal-clear Core *Purpose* Statement as the basis for further planning.

Take for example: Merck [1930s, 1, p.236] **"To preserve and improve human life"**

This is intended as the fundamental performance measure of all corporate activity for a pharmaceuticals company. It is of course constrained – and thus partly defined - by their core *values* (keywords: responsibility, excellence, science, honest, profit).

If someone found that their pharmaceutical technology could be used for *animals or plants* – does the 'human life' idea apply, or disqualify the product area?

If they could extend human life for people *living in a permanent coma*, does that count, as within their core purpose?

If they *found psychological, mechanical, electronic or religious means, or other 'services',* for improving the human life; are these means valid? Or is there some constraint about sticking to the drug business, even if other available means are more cost-effective?

I can't see where it says, *strategy constraint 'drugs only'.*

Let us look at some possible Merck 'clarifications' for
"To preserve and improve human life":

'To improve life quality by <u>any means</u>.'

'To <u>provide products</u> to improve life quality.'

' To develop <u>knowledge</u>, and apply it, to get improved human mental and physical life quality.'

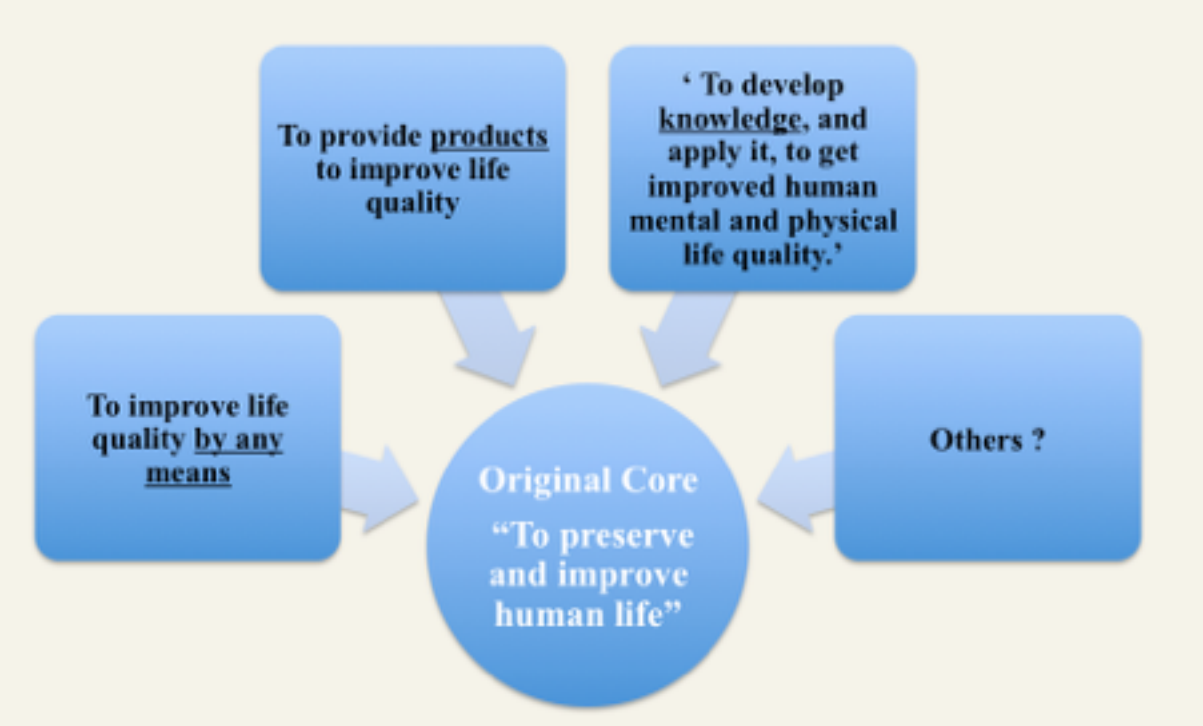


Figure 1 C. Some options for reformulating the original Merck Core purpose. The options would lead to very different business product options.

Each one of these is significantly different from the other. So consider a re-think of the articulation of your most fundamental purpose, before making it the touchstone of all other planning work.

Defining a Scale of measure:

One 'device' we will need, sooner or later, to *really* clarify performance objectives, is to define them, so that we can *quantify* them in practice.

The fact that we can then set numeric targets, and numeric constraints, and track them, is *powerful*; but in fact is *not* the main point.

The main purpose of 'quantification of performance objectives', is to force us to *think deeply*, and *debate* exactly, what we mean; so that others, later, cannot fail to understand us.

THE QUANTIFICATION PRINCIPLE

Performance objectives, ranging from *core objectives* to 'any' detailed performance objective – where 'getting better-and-better in time' is implied – can *always* be defined using 'scales of measure'.

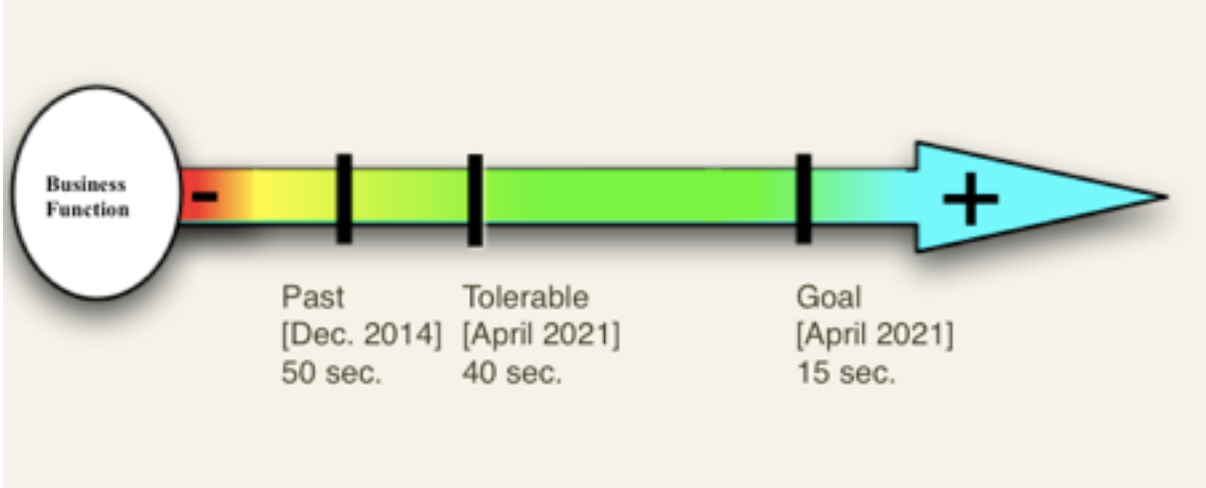


Figure 1 D. Any performance measure, for an organization, can be visualized as an arrow, with the performance itself varying from very bad (left hand side '-') to very good (right hand side '+'). The arrow itself is our standard symbol for a performance-defining variable, such as "average time needed to make a sale". Along this defined Scale of measure, we can then describe useful degrees of performance data. Past performance, competitor performance and future plans.

And once we have agreed on clearly defined 'scales of measure', we can apply a large useful set of devices, to exploit the fact that we have entered 'numeric territory'. Management plan accounting, for 'performance' values, not just financials.

Less poetry, more logic.

- Nothing wrong with poetry and the arts, in their place.

Let me introduce a 'planning language' method:
'Planguage', I call it [2,3]. It rhymes with 'plan' and 'language'.

- We write "**Scale**:… " in front of our defined **scale of measure**.
- Note that we are NOT defining a testing, tracking or measuring process (later called 'Meter: ---") yet. Just volts, not a voltmeter.
- We are just *enabling* ourselves to *think about* our most cherished core purpose *numerically*.
- Let us try with the example: "***To <u>provide products to</u> improve life quality."***
- *What is the 'scale' to quantify this, and to define what we mean numerically?*

<u>S1</u>: **Scale:** New Products Released Annually.

- *You can see the weakness with this draft, S1?*

<u>S2</u>: **Scale:** Annual Sales for all products that improve life quality.

- And the weakness with *this, S2*? For example, Merck is famous for giving away some drugs!

I would prefer this draft, because there is less to misunderstand:

<u>S3</u>: **Scale:** Estimated **B**etter **D**ays for defined [**Life Form**] as a direct result of defined [**Products**].

<u>Better Days</u>: days where the entity themselves, or another better judge, would judge their life to be better than without Our Product.

<u>Life Form</u>: {Human, Animal, Plant}.

<u>Products</u>: {Patents, Drugs, Machines, Licenses, Services, Distribution, Education, Motivation, Others}.

S1,S2 and S3 are arbitrary reference tags to the statements. Capitalized terms (‘**B**etter **D**ays’) are formally defined terms.

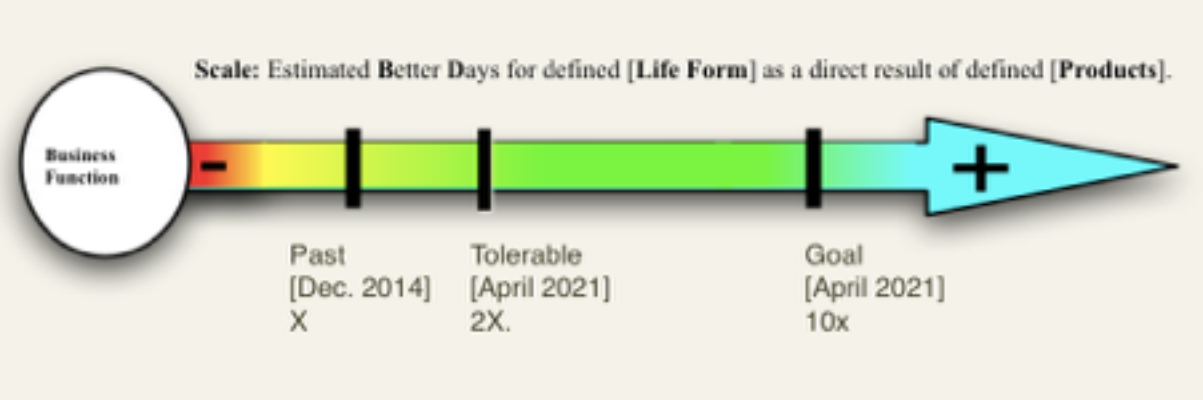


Figure 1 E. My suggested, draft, ‘scale of measure’, for my ‘improved’ variant of Merck’s original core purpose (‘To preserve and improve human life’) ***“To provide products to improve life quality.” .***

I would argue that 'S3' is a pretty good draft effort, as a powerful definition of our Core Purpose. The core purpose has not changed. But our ability to articulate it, and to discuss any related plans, is arguably improved.
I would argue that it can help us, in deriving relevant aligned plans, and help us to judge their effectiveness, for promoting our core purpose.

I would argue that 'S3' is a better top management tool than the 'poetic' phrases (S1, S2) that preceded it, even though poetry might *still* be useful for simple *emotive* presentation, in some circumstances.

Management can usefully distinguish between '**presentation** formats' (like

Ambition Level: To provide products to improve life quality.

and '**planning** formats' (like S3).

You probably need both formats, for different audiences and purposes.

Deriving Objectives from the Core Purpose and Core Values.

A defined 'Scale' gives us a 'numeric-scale definition' of core value and core purposes.

This enables us to move our planning from a 'poetic' to a 'numeric' basis.
(From Management BS to Management Engineering)

We can now plan, by *determining a useful set-of-points on that scale of measure.*

There are three major planning categories:

- **Benchmarks**: points for *comparison* with future plans.
- **Constraints**: borders, *worst acceptable* levels.
- **Targets**: levels of performance we are *aiming* for.

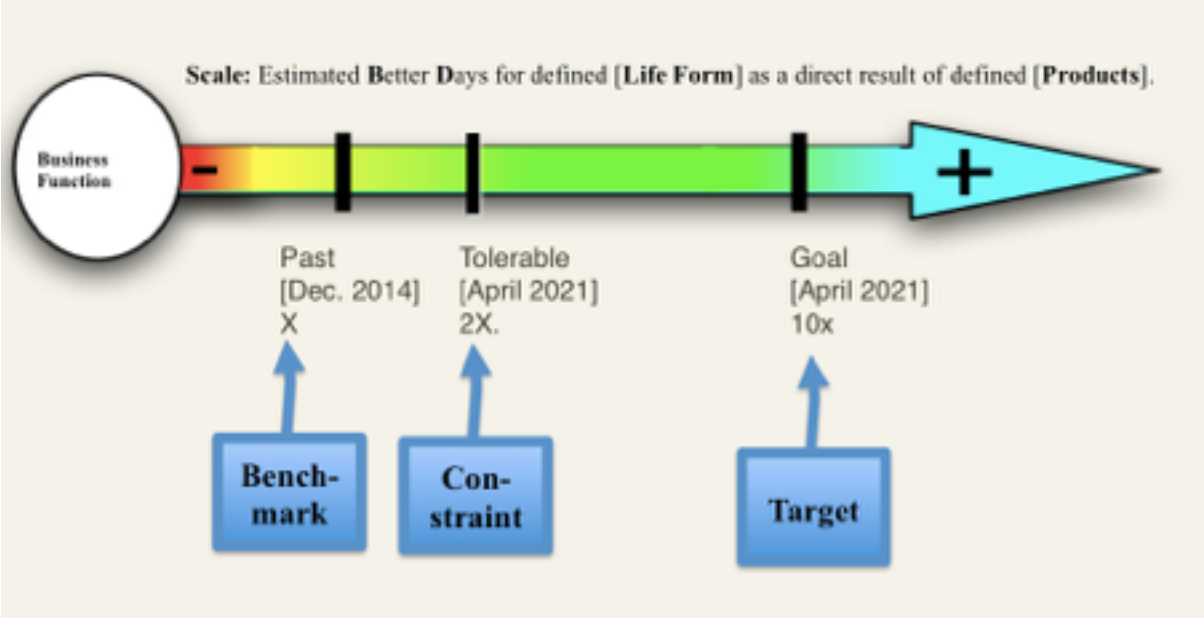


Figure 1 F. Three major categories of 'levels' of performance.

I have defined a number of these concepts in Planguage. Here is a useful set.

Benchmarks: levels of performance worth knowing about, in comparison with future planned levels. For us, for competitors, for the past and possible future.

Past: any estimated, or measured, level for us, or others, that is *interesting* to compare future plans to.

Trend: an estimation of the levels, good or bad, that will possibly be reached by us, or others, at defined times, and under defined circumstances.

Record: a state-of-the-art extreme, attained under defined conditions.

Constraints: less-than-successful areas we are trying to avoid.

Catastrophe: the edges of a numeric range of performance results that are disastrous in consequence, and possibly not recoverable.

Tolerable: the edges of a numeric range that is tolerable, just above Catastrophe, but still **failing** *to some degree* to satisfy, even at the OK level.

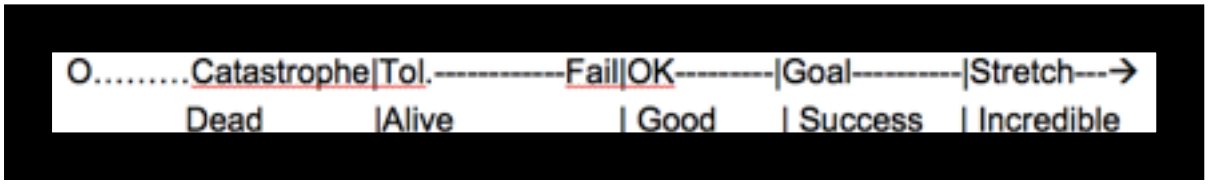


Figure 1 G: points and ranges on a scale of performance.

OK: a range just above the tolerable range.

- Not intolerable.
- Not failing.
- Pretty 'good',
- but not yet at an ambitious and competitive 'success' level, (called the Goal).

Targets: levels we are aiming to reach

Goal: a level which is both satisfactory, and considered feasible; you can promise it.

- 'Better than the Goal level' is a *range*, we can call the success range.
- But, there might not be any defined or planned value for getting better, in that range.

Stretch: a level that has stakeholder value, and which you will attempt to move towards, if resources remain, after all other critical objectives' Goal levels are reached. This means we are not fully committed initially to achieve this level: it *depends*.

Ideal: *(rarely used except to distinguish it from more practical targets)* a level of perfection unlikely to be achievable in practice, and not necessary achievable (since competitors cannot get to it either). But we can aim to 'tend towards' it. From another point of view, it is also a 'benchmark'.

- Examples 100% availability, zero time to learn to do a non-trivial task.

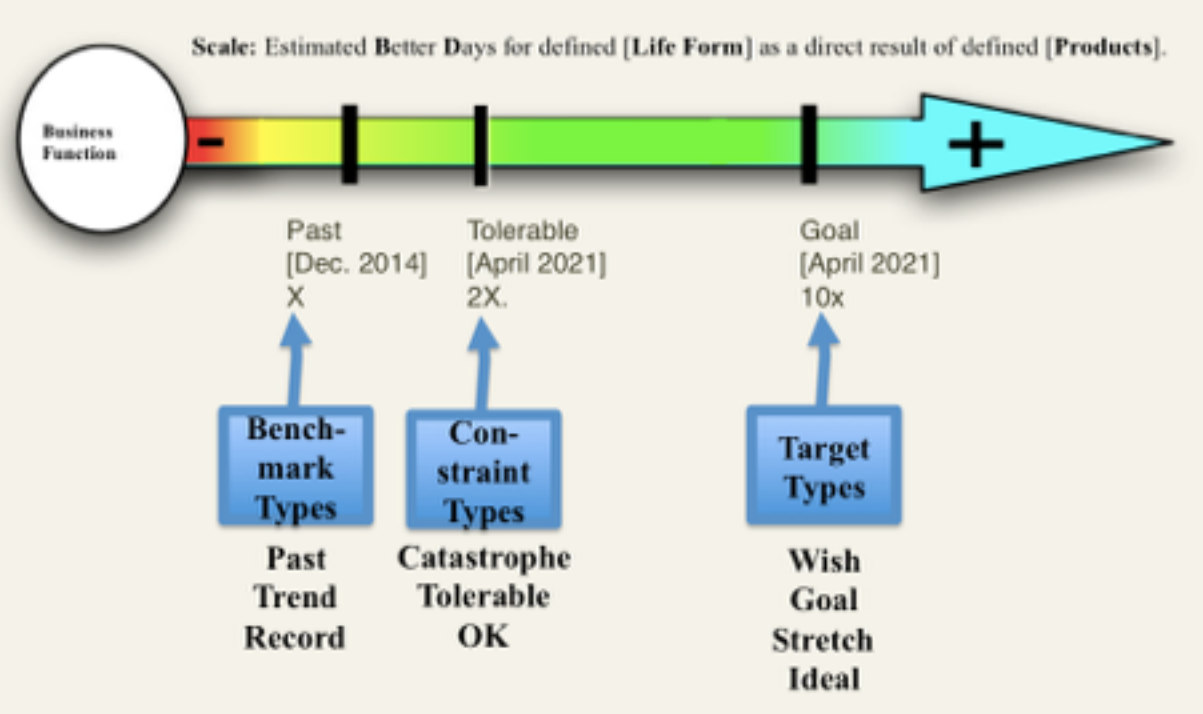


Figure 1 H : Specific instances of each type of planning-level on a scale.

Determining the numbers.
The next step is to determine some 'planning values' (some numbers on the scale that are valuable for our planning purposes), using any useful means to determine the numbers.

For example:

Merck Core Purpose:

S3: **Scale**: Estimated Better Days for defined [Life Form, default: Humans] as a direct result of defined [Product, Default: All].

Past: 100.

Goal: 1,000.

This makes the point that we plan to get 'ten times better'. But it would be more intelligible, if we added some 'implied but not stated *here* yet' defined conditions, in a 'qualifier' statement, in [*square brackets*], like this:

Merck Core Purpose:

Scale: Estimated Better Days for defined [**Life Form**, default: Humans] as a direct result of defined [**Products**, Default: Pharmaceuticals].

Past *[2014, Europe, Products = Tranquilizers, Life Form = {Humans, Animals}]* 100.

Goal *[2024, USA, Products = All Merck Products and Services]* 1,000.

The [qualifier] statement enables us to be more specific. We can have many such statements (many Goals, many Pasts) about different interesting levels of performance.

This means we can plan for a both **complex** enterprise (many connected parts), and **complicated** enterprise (difficult to predict, estimate and understand, especially with respect to its environment).

This avoids vagueness, over-simplification, misunderstandings, and over-generalization. We can be as clear, exact, and specific as is useful, at a given stage of planning.

For example:

Merck Core Purpose:

Scale: Estimated Better Days for defined [Life Form, default: Humans] as a direct result of defined [Product, Default: All].

Past [2014, Europe, Products = Tranquilizers] 100.

Goal [2024, USA, Products = All Merck Products and Services] 1,000.

Goal [2020, Worldwide, Products = All Merck Products and Services, If Merge Approved] 500.

Tolerable [2020, Europe, Products = Pharmaceuticals] 200±100? <- CEO Vision Statement.

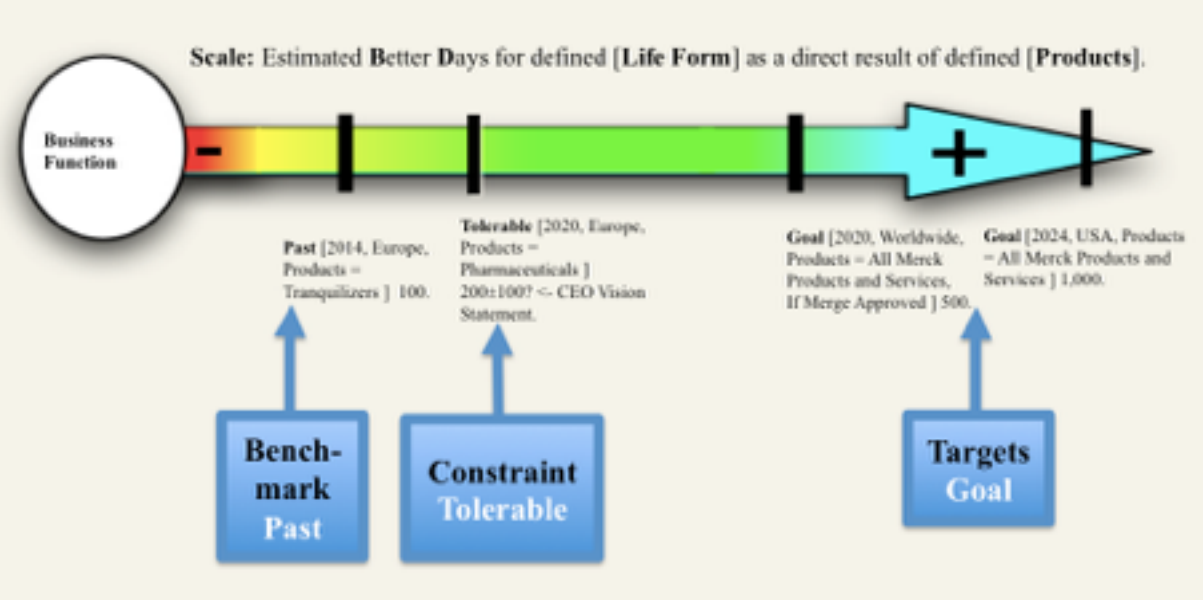


Figure 1 i Simple examples of defining interesting planning points along a Scale.

Any useful number of points on the business performance scale (in this case a 'Scale' for the 'top level performance' **core** purpose) can be defined, using any interesting set of the types of points (Past, Goal, Tolerable, etc.). And any set, or combination of, [qualifier conditions], a sort of 'adjectives', can be planned, *in addition to* any one of these scale points.

The consequence is that the *qualifier condition*s permit 'drilling down' into detail, of plans laid: *later*, and *at sub-lev*els.

- For example year by year, and country by country.

Notice that the Scale *definition* is being 'reused' (write once, use many times), by all these scale points.

This is why we take pains to be precise in defining a Scale (using 10 words, rather than one), and why we *parameterize* it ('defined [Life Form]').

Clarifying Objectives

It is central that objectives are perfectly understood, by all intended readers. All employees, investors, media etc.

Perfect *practical* clarity is a *nearly* attainable objective, using fairly simple means.

It is unacceptable, (a bad practice we have measured worldwide) for objectives to be so badly written, that employees and managers judge 30% to 70% of all specification words they read, to be either *ambiguous* or *unclear* (to at least *some* of the potential readership).

You will already have noticed some of the devices we use to reduce ambiguity. Scale + Goal for example, beats 'exceedingly'.

Many other devices in the Planning Language are not yet explained or mentioned. But they are available when you are ready.

Here are some *clarification practices* that have *already* appeared in the examples above:

1. *Consistent official definition* of key planning parameters (like Past, Goal, Scale). A formal Glossary exists [2,6]
2. Our drive to become *numeric* (beats nice words like 'substantially improved')
3. The use of qualifiers to define 'when', 'where' and other conditions. "[2020, UK, If Finance Approved]"

4. The consistent formal use of terms written with **C**apital **L**etters, indicating that the term is *formally defined*. Like: **Life Form**: {Human, Animal, Plant} in the initial S3 example above. And like "**P**ast, **S**cale, and **Merck Core Purpose:".**

5. There are dozens more devices, you can choose to improve clarity, too many to enumerate here [2].

Your practical 'organizational planning-improvement campaign', we believe, should be 'to reduce 'major defects' (avoid planning specification terms that might possibly cause misinterpretation, of serious consequence, by some reader) in planning.

The degree of improvement should be from a 'normal', but unacceptable, level of 30% or more defects, to a level of *less* than 1 defect per 300 words. A tough but do-able objective.[URL22]

Extending Understanding of the Objectives – Background.
In addition to the specification devices mentioned above ("Clarifying Objectives"), we have developed a large set of simple devices for adding *background* information to a fundamental objective.

It consists of a predefined set of 'parameters' (Scale and Past are 'parameters' too), and other Planguage [2] devices; as well as the ability to define any new additional parameters *you* find useful.

We already inserted some background information in the example above:

Tolerable [2020, Europe, Products = Pharmas] 700 ±200 ? <- CEO Vision Statement.

±200 means the tolerable *range* is 200 days (500 to 900 days)

? means even this is a *questionable* number or interpretation. Don't take it too seriously. Uncertainty.

<- is a 'Source' arrow, used to specify our *source* of the specification. In this case the 'CEO Vision Statement'. We could have also written:

Tolerable [2020, Europe, Products = Pharmas] 700 days.
Range: ±200 days.
Risk: incorrect interpretation of actual CEO slide 25.
Source: CEO Vision Statement, Jan 1 20xx Brussels.

Here is a small sample of some of the other available background statements (with illustrative text after the parameter):

Supports: Core Purpose.
Supported By: Top Long-Range Objectives.
Constrained By: Core Values.
Implementation Responsibility: CEO.
Plan Owner: Strategic Planning office.

With all the statements you might want to use, you can easily fill a page, or a slide, with 20 to 60 statements for a single defined objective.

It is up to you, to use or create, what you find valuable to add, as background to the core specification. The full specification, for a *single* objective, forms a small collection ('database') of 'everything' we need to know, in relation to that objective.

Of course, subsets, right down to one-liners, can be extracted for specific presentation purposes, while other statements can be drilled-down to, on demand.

Once when we were having a top management fight in London, about using this format (the Marketing Guys wanted to keep it simple and unintelligible) a seasoned director stopped the show by saying:

I have estimated that we spend on average £200,000 for each one of these objectives, and too frequently screw things up. If defining an objective in 40 lines instead of one line solves that problem, then that is a small price to pay, and a necessary investment in getting our business right" (Thanks BW).

If you use the simple principle of investing more effort, in management planning quality, *only if it pays off*, you should not end up with unnecessary bureaucracy.

I know we have too much meaningless low-quality verbiage in planning, everywhere, today. So do you, I expect.

My suggestion is, in fact, to write *less* in total, and to make it *'reusable'*; and to *raise the quality* of what we *do* write - by two orders of magnitude. Get rid of those too many major defects per page.

This 'specification quality improvement' is measurable using the methods immediately in Part 2 below.

Part 2. Quality Control of Objectives

Measuring the Quality of Management Planning Specifications.

Most management plans (Objectives, Policies, Core Values, Strategies) are never formally measured to any standard.

They may be discussed, edited and approved, but there is no formal measurement involved.

We have developed a simple measurement process for measuring the various qualities of management planning.

It is based on a software process known as 'Inspection' [7], and on a simplified variation we have developed, *Specification Quality Control* (SQC) [2, Chapter 8].

The method is completely general, and applies to any writings or specifications, both technical plans and management plans.

The SQC principle is simple and classical. We measure deviation from *standards*. Deviations from our 'planning rules' are called 'specification *defects', or 'planning defects' if you prefer*). We focus on a significant class of defects called '**major** defects'.

All major defects are by definition unacceptable. They create risk of damage to your results and economics.

We try to get a rough, but useful, estimate of the **major defect density** for a given plan.

For large documents we use 'representative samples' as sufficient basis for the defect density estimate, to keep costs down. An hour or so is usually sufficient for most purposes.

Once we can acceptably measure the 'defect level' (major defect density per 300-word page), we can judge whether the specification (of, say, corporate objectives) meets our 'reasonable standard' maximum level of defect density, for allowing plan release.

Zero defects would of course be nice. It might accidentally be achieved.

But as a practical matter we have to be satisfied with a radical improvement, to a level where it pays off to release the specification to other subsequent levels of planning, and deal with remaining imperfections afterwards.

This is called the 'Exit' Level of quality of the specification.

A *good* initial exit level is:

'*maximum estimated remaining'* **10 major defects** per 'page' (300 words).

A *more ambitious* level is maximum 1 major/page.

Notice that we do not necessarily release known defects. We calculate the possible level of defects, based on

- a sample of defects found, (about 30% are found in pages checked)
- and an estimate of defects probably not found yet (the other 70%).

For example, if we looked at a 10-page document, and our checking team found 10 major defects in a sample of a few pages (1 to 3 pages is fine).

We would estimate the total number of defects per page. Like this.

If we assume our team's defect-finding effectiveness is about 1/3 (they fail to find 2 out of 3 defects present – a normal result) then the estimate is that there are about 30 majors in the sample page – 10 found, 20 not found yet.

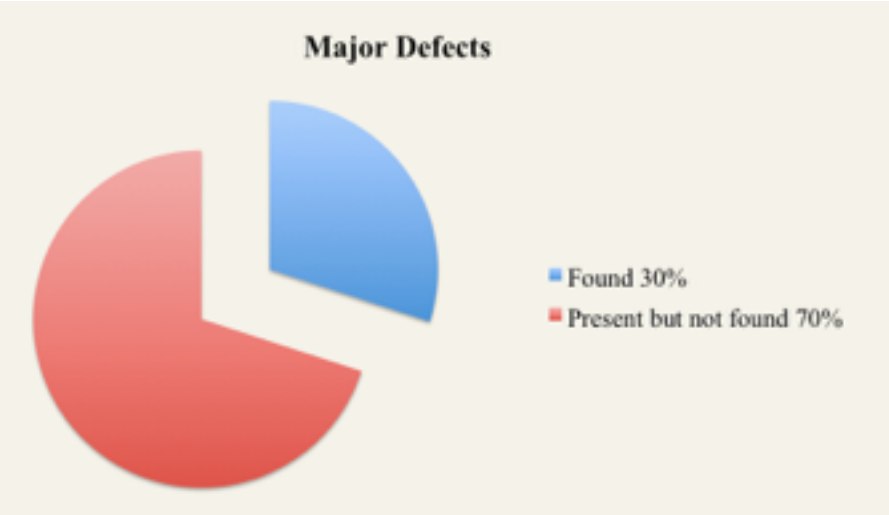


Figure 2 A. when reviewing a planning document using a few people, the team's effectiveness is limited to finding about 30% of the Major Defects actually on the page. But that means if they find 30 Majors there are about 100 Majors total. So fixing the 30 found is not much use. We have to deal with all 100 somehow.

And, assuming our sample pages are accepted as representative of the other pages there are about 10 (pages) x 30 majors = 300 major defects in the document.

This is usually considered an 'obviously unacceptable' result. Exit is therefore 'failed', and the whole document must be reworked to a higher adherence, to the formal written standard.

Re-checking after trying to improve the plan will determine if it can exit, using a different,or random, sample of the 10 pages.

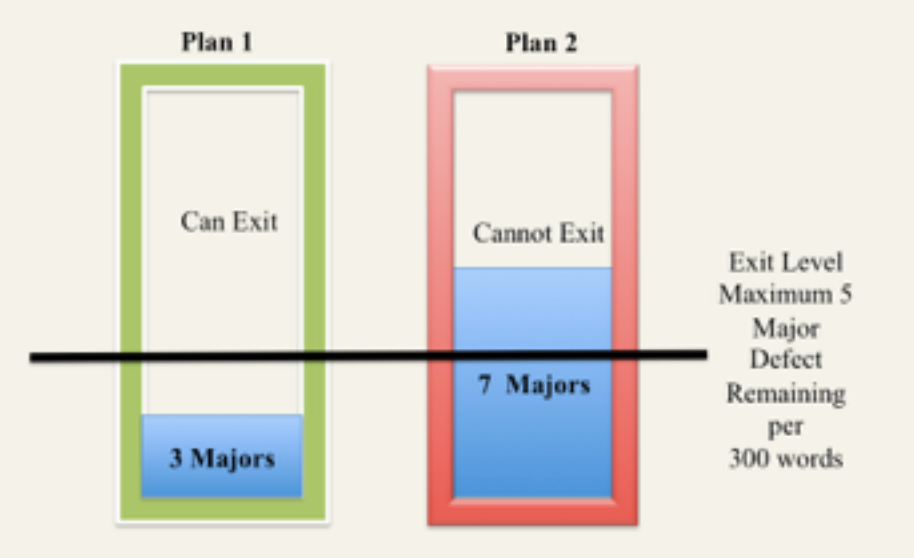


Figure 2 B. Planning documents can be measured for conformance to your standards. If even a sample of a plan says that defects exceed the Exit level, they cannot be allowed downstream. That would do more damage than good. This 'planning process Exit discipline' will motivate planners to learn, and follow, planning standards first time around.

Our experience, when initially checking any document (technical or managerial) that has *not* been subject to this discipline, is that the estimated defect density is about 100 major defects per page, or more; even by the simplest of standards (clear, unambiguous); as judged by senior managers or engineers.

This result is of course unacceptable; but since nobody measured before, nobody was conscious of the problem.

Mini Case 2 A: CEO Garbage Out

Once at a large US telecoms organization, my client decided to measure the major defects in the Corporate Quality Policy issued by the CEO, and hanging on the walls, as a single page declaration.

I forget the exact result, but it gave us all a good laugh, because the major defect count was well over 100.

That means the one-page policy could be misunderstood by employees on up to 100 different points.

Of course nobody dared tell the CEO, but a senior executive I had lunch with afterwards, acknowledged the problem was widespread.

What we have found, both on a large corporate scale, and in large scale engineering, is that if the Exit level standard is taken seriously by management, then the quality level produced, the first time around, will rapidly move towards that exit level. There are, as for as I know, no large scale studies of this for *top management*.

Mini Case 2 B: Clear Measurable Top Level Objectives: then 15 Years Success.

But, one 20,000 person corporation I worked with for several years, who adopted the clear measurable objectives methods at the board, CEO, and directors level; reversed over 5 years of losses, and went into profit for the next 15 years – unlike any competitor!

I like to think this was a function of much clearer management thinking, enforced by the CEO (RW). All of the credit belonged to the dynamic engineering-trained CEO.

But he was 100% behind my suggestion, and behind the consequent effort, to get him and all his direct reports to quantify their objectives.

The Corporate Planner, Peter Hall, and I trained the directors (half day), and they needed it. The Board was behind the CEO on this issue too.

People will quickly teach themselves how to follow the 'how to write good objectives' standard (more below).

The alternative, to reducing defects at this early planning level, is that each major defect will corrupt efforts downstream, and cost considerably more finally, than the effort to write things properly in the first place. A stitch in time saves nine.

The management-planning quality-control is logically divided into two phases.
Is it clear?
Is it right?

If a management planning document has, say, 100 ambiguous and unclear terms per page, then it is premature, to say the least, to attempt to evaluate whether it contains powerful, useful, aligned, relevant ideas, to support core values, or other higher levels of planning.

In simple terms, we must first measure that a plan is reasonably intelligible. Only when it is intelligible, can we proceed to evaluate whether it is 'the right stuff'. Here is some more detail on those issues.

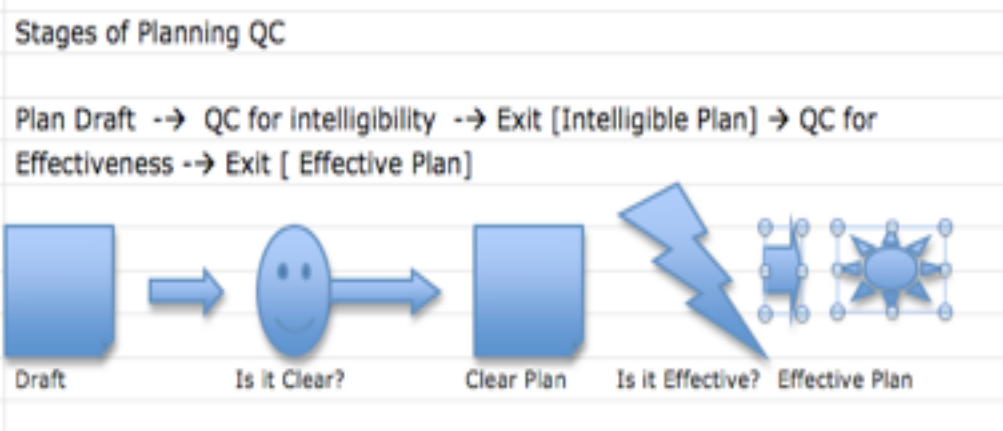


Figure 2 C. The planning 2 stages of quality control: 1. Clear?, 2. Right?

Testing Your Objectives for Clarity: The Simplified Spec Quality Control

Our first task when a non-trivial size draft plan (say core values, or corporate objectives) is ready, is to run a check on *intelligibility*.

Step 1: define 'Intended Readership' for the plan.
Make a list of all the various types of people (lawyers, engineers, Board Members, middle managers, consultants, new employees etc.) who must be able to reliably interpret the plan.

Step 2: Standards for Intelligibility.
Make a list of 'Rules' for writing, which if followed, would give you the level of intelligibility needed by your Intended Readership – *all* of them. Include the *weakest ones* in their *weakest moments* (stress, pressure, tired).

I have found the following Rules to be basic and sufficient for most beginning purposes:

C1: UNAMBIGUOUS: The Objectives will be *unambiguous* to the Intended Readership. *One interpretation: the right one only.*

C2: CLEAR: The Objectives will be *clear* enough for all of the Intended Readership, to correctly and completely, carry out relevant strategy planning, as intended, and to check for themselves, that it is a correct interpretation, *even under worst case conditions*.

C3: ENDS NOT MEANS: The objectives will be stated in terms of the ultimate *'end states'* desired (at that level of planning), and NOT in terms of strategies (means) perceived as leading to those end states. *How well, not HOW.*

Step 3: The Spec QC Checking Process

- Convene a small team of credible senior people (2 to 6).
- Present the rules to them.
- Ask if they agree in principle to these rules and understand them.
- Give them about a page (300 words) of text each (can be all same page, or some different pages).
- Give them about 30 minutes for the page.
- Ask them to mark any word of term that violates these rules.
- Ask them to categorize each rule violation as a 'defect'.

- Ask them to classify each defect as either 'Major' (really must be corrected, consequential), or minor (rule violation, but does not really matter much in practice – for example because it is not a decision driver, but is a footnote).

Step 4: Reporting, and Process Validation

- Each checker reports major defects they claim to have found. For example 10, 5, 20, 12 Majors.

Step 5: Calculation of Defect Density

- Estimate that the team has found about 40 unique major defects [2, 7 pp299-301]. Twice what the largest claim (20) was.
- Estimate the team has found about 1/3 of the actually present defects just now. And that there are therefore about 40 +80 = 120 Majors in the page.
- We cannot fully explain here exactly why we do this (hints are given!), so trust it, or read the references.
- We are trying to move the estimate nearer the truth, than the raw data alone.

Step 6: Exit Determination

- If the exit level (maximum estimated remaining Major defects per page) is, say 10, then we have a clear case of *failed exit* (even if we had skipped Step 5!).
- Revise the document, if Exit fails, according to the Rules.
- Otherwise release the document as *intelligible* (but not yet reviewed for being 'the right stuff'

Figure 2 D. The Plan Quality Control process. If the plan has a low violation of planning rules, it can be economically released 'downstream'. If not, it pays off to reject it, and get a version that can be safely released.

IF EXIT FAILS
If the level of unclear and unintelligible content is as high as this example above, then

- editing the plan, according to the Rules, is required,
- and a new Quality Control is required, until successful Exit is achieved.

If the specification is difficult to interpret, then it is premature to take the next step of deciding if it is a 'good' (effective for purpose) plan.

On the other hand, even a plan that is, at this stage, zero major defects, might be a bad plan for the real business!

Clarity is not the same as effectiveness. But it is a necessary first step.

Example:
Let us assume that the following terms were marked as unclear/ ambiguous:

S3: **Scale**: Estimated better days for living things as a direct result of our products.

Let us call that '3 Major defects'.

- After all,they make the Merck Core Purpose unintelligible!

We could attempt clarification by rewriting as:

S3: **Scale: Estimated Better Days for defined [Living Things] as a direct result of defined [Products].**
Better Days: days where the entity themselves, or other better judge, would judge their life to be better than without Our Product.
Living Things: {Human, Animal, Plant}
Products: {Patents, Drugs, Machines, Licenses, Services, Distribution, Education, Motivation, Others}.

Testing Objectives Against Core Ideology
If we can 'Exit' from the 'Clarity' quality control process, with a sufficiently intelligible specification of objectives, then we have the basis for deciding if those objectives are "the right stuff."

How do we evaluate to see if a management objective is *good enough*?

We can evaluate a management objective, for cost-effectiveness and risk, against the following Rules (as an example):

E1. Objective Effectiveness Estimate:

*The **set of effects**, we expect the objective to have, on our own higher-level objectives (for example our 'Core Purpose' Goals) **is estimated**, with reasonably credible evidence and sources?*

***E2. Costing**.*
*The **resources** required to deliver this objective, at specified times, places and conditions (the [qualifiers]) are credibily **estimated**?*

E3. Specification of conflicts.
*A: If anything with the objective, or planned strategies for it, in any way threaten to conflict **with our Core Values**, or with **any other constraints** (Policy, Law, Custom), we must **list these formally** under suitable Background headings, such as Threats, Issues, Risks, Assumptions, Dependencies, Conflicts.*

B. We shall not censor any potential conflict, however improbable. But notes, on our evaluation or mitigation, can be included.

Standard 2. Simple sample Rules for Checking Objectives against a Core Ideology.

In other words we need to evaluate effectiveness, costs and conflicts.

Plan Clarity	Plan Effectiveness
Clear	Main impact on Goals is estimated
Unambiguous	Side effects are estimated
Complete	Costs are estimated
Consistent	Constraint violations are specified
Variables quantified	Risks, threats, mitigations, assumptions are specified
	Issues not resolved are specified
	Potential conflicts are specified

Table 2 A. The two classes of standards for checking a plan. First it needs to be intelligible. If it passes that test, we are then 'enabled' to judge its effectiveness for purpose (for our objectives). Both classes of reviews here, result in an objective and quantitative evaluation of a plan's suitability for purpose. Very few businesses today have this rigor in their review process. Few seem aware that they could have such a process.

Using the rules to evaluate plan effectiveness for purpose
To do the first element (E1, above) we need to make an estimate of 'the degree to which this objective is expected to contribute to one or more higher objectives'.

Let us simplify by assuming that this higher objective is the Core Purpose itself, the primary dimension we want to improve. We defined one such in this paper above (**Merck Core Purpose**).

Let us now assume we have a lower level supporting objective called:
"**Top Long Range Objectives**"

Let us assume that we believed that the set of Top Long-Range Objectives [Year = 2020] would be capable of *exactly meeting* the **Merck Core Purpose, Goal [Year =2020].**

- Then we could assert that belief by estimating the impact as '100%'.

We could offer 'evidence' for our '100% effectiveness' belief by referring to the set of facts, reasoning and sub-estimates:

Evidence: *TLRO:MCP 2012 Estimation Basis****.***

We could indicate the degree of uncertainty of this estimate, by a statement like:

Uncertainty*: ±50%*

We are now doing a simple form of 'Impact Estimation' [Full detail Ref. 2, Chapter 9, Impact Estimation], and see below.

It would be a matter of planning policy and management discretion to accept these estimates, review their evidence, and decide if the uncertainty was acceptable; enough to approve and proceed.

But the main point here is that, if we take the trouble to quantify objectives, and specify conditions, for targets and constraints; then we have enabled ourselves to 'reason further in a quantitative way', about the impacts of plans on their objectives.

We are now more likely to see weaknesses in a plan, at an early stage, and take steps to remedy it early.

		.....Deliverables						
		Telephony	Modularity	Tools	User Experience	GUI & Graphics	Security	Enterprise
Business Objective								
Time to Market		10%	10%	15%	0%	0%	0%	5%
Product Range		0%	30%	5%	10%	5%	5%	0%
Platform Technology		10%	0%	0%	5%	0%	10%	5%
Units		15%	5%	5%	0%	0%	10%	10%
Operator Preference		10%	5%	5%	10%	10%	20%	10%
Commoditization		10%	-20%	15%	0%	0%	5%	5%
Duplication		10%	0%	0%	0%	0%	5%	5%
Competitiveness		15%	10%	10%	10%	20%	10%	10%
User Experience		0%	20%	0%	30%	10%	0%	0%
Downstream Cost Saving		5%	10%	0%	10%	0%	0%	5%
Other Country		5%	10%	0%	10%	5%	0%	0%
Total Contribution		90%	80%	55%	85%	50%	65%	55%
Cost (£M)		0.49	1.92	0.81	1.21	2.68	0.79	0.60
Contribution to Cost Ratio		**184**	42	68	70	19	82	92

Table 2 B. An example of a real (2005) top management Impact Estimation Table. The Objectives (name tags only) on the left were quantified with defined Scales, and Goal and Stretch levels of performance. This table satisfies rules E1 and E2 above, and below: effectiveness and cost estimation.

Evaluation of Resources.

E2. Costing*.*
The resources required to deliver this objective, at specified times, places and conditions (the [qualifiers]) are reasonably estimated ?

It is **not logically possible** to estimate, let alone know, the resources (time, effort, money) needed to achieve any level of performance, **if** you only know the required level of performance; but have *not yet* determined the *specific strategy* (means), to reach that level, under specified conditions.

Yet, if we do not know the resources needed, we cannot yet approve the objective, because we might not have, or get, the resources necessary.

We can only approve of the objective itself, as a necessary condition for reaching our even higher-level objectives, hoping we can later find suitable cost-effective strategies.

And we can approve the **next stage,** of determining one or more strategies, and costing them. Only on the basis of credible resource estimates, within our resource capabilities.

Then, we can give the final go ahead, do it. Do the strategy, and get us to the promised Goal.

Note that, even with the best well-known strategy, and the best available cost *estimates*; I am a skeptic about what the *real* costs (time, effort, money) will really be. The situation is usually complicated; unpredictable.

The simplest remedy I have, is that - even after conditional strategy approval, based on initial estimates - we *still* need to monitor *early and gradual r*esult improvement, quantitatively, and take action if a predicted result stream is not happening.

This feedback evaluation is much more practical when the results are quantified as suggested above [9].

In the impact estimation table above, you will see that the monetary cost of each strategy was estimated, and could then be compared to the estimated impacts of each strategy.

Conflict with Constraints

E3. Does the objective in any way threaten to conflict with our Core Values? Or with any other constraints (Policy, Law, Custom).

The objective, and in fact its implementation strategy, must both be reviewed initially, and potentially be reviewed in the light of partial feedback from partial implementation, continuously, for any emergent unacceptable conflict with any defined constraints.

These defined constraints can include Core Principles, Policies, Laws, Local Culture, and any other constraint we agree to respect.

Conflicts need to be documented in the plan, the magnitude estimated, and a decision as to whether the conflict, or potential risk of conflict is acceptable needs to be made.

Part 3. Strategies.

The term 'strategy' has many possible interpretations, but I will define the sense I use it here, too.

A dictionary definition says: "a plan of action or policy designed to achieve a major or overall aim".

I will stick to that essential idea, and clarify.

A strategy is *any means* directed towards an *end*.

It is what we plan, and do, or perhaps 'do *without* formal planning', *in order to achieve* one-or-more higher priority objectives (aims).

A strategy is **anything** that might, or will, serve our aims; and *certainly **not** restricted, as the dictionary definition indicates*, to 'plan of *action* or *policy'*.

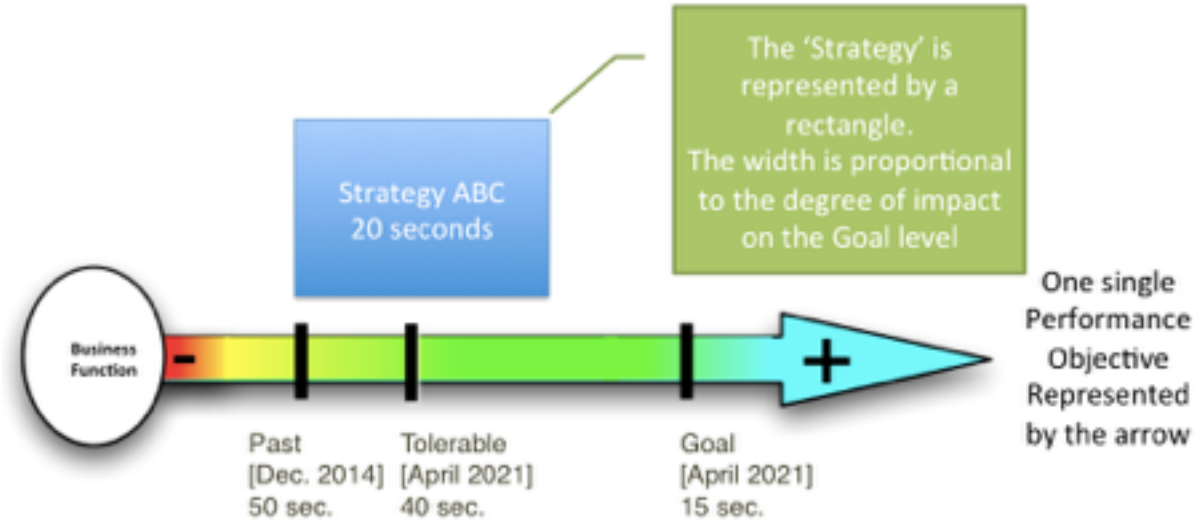


Figure 3 A. A 'strategy' is something we plan to do, with a hope that it will have a positive effect on at least one objective's Goal. A numeric estimate, of the future strategy impact, is called an 'Impact Estimation'. In this example the estimate is 20 seconds progress towards the Goal.

A given objective may be supported by *other and different supporting objectives*; themselves supported by related strategies (which may be classified as supporting objectives too!).

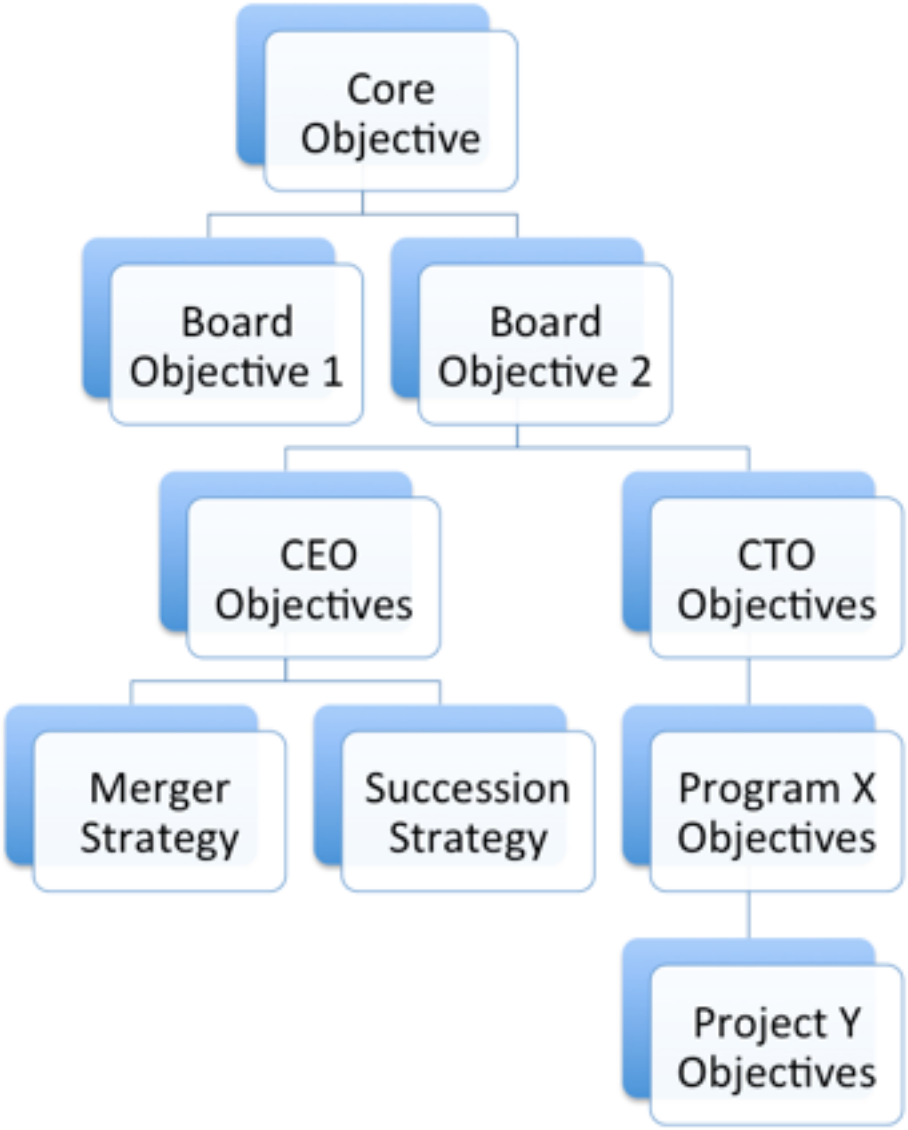


Figure 3 B. A simplified and symbolic hierarchy of objectives and supporting strategies. In our world, these relationships, the thin vertical lines, can be quantified, estimated and measured.

Strategic Objectives
Strategic objectives were defined by Keeney [5] as a level of objectives, that, at *our own* level of responsibility (or some defined level of responsibility) were the strategies *we* had chosen, in order to support the objectives 'one level up' (our bosses *own* objectives), which Keeney called 'Fundamental Objectives'.

So, for example, the Board might have a set of *strategic objectives* for supporting the Core Purpose.

Notice that this *set of ideas*, about the relationships of levels of objectives, can apply to *any level* of the organization. Fundamental – boss, strategic – me.
Fundamental objectives are not *my* business to change, approve, or be ultimately responsible for. But my *strategic* objectives had better *support* them.

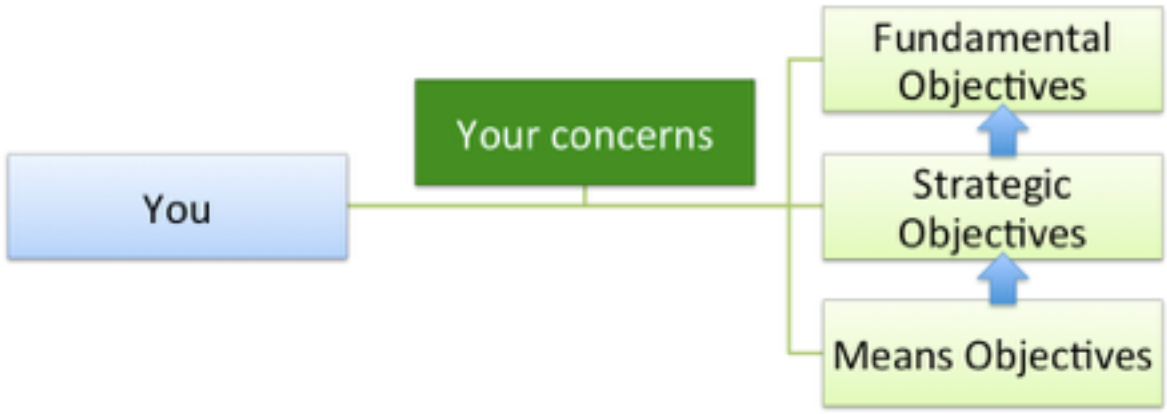


Figure 3 C Your primary responsibility is to meet your Strategic Objectives. But you need to make sure they actually satisfy your next level up 'Fundamental Objectives'. And, you need to worry that the Means Objectives, which are intended to satisfy your Strategic objectives, really do the job, on time, on budget.

Means Objectives

Keeney's [5] "Means Objectives" are the sets of objectives that support achievement of *my* strategic objectives. They do not have to come from 'my staff' or 'my subordinates'. They could be set, and delivered, from *any* partner, supplier, or organizational component that is capable of delivering results, which will support achievement of our *own* strategic objectives.

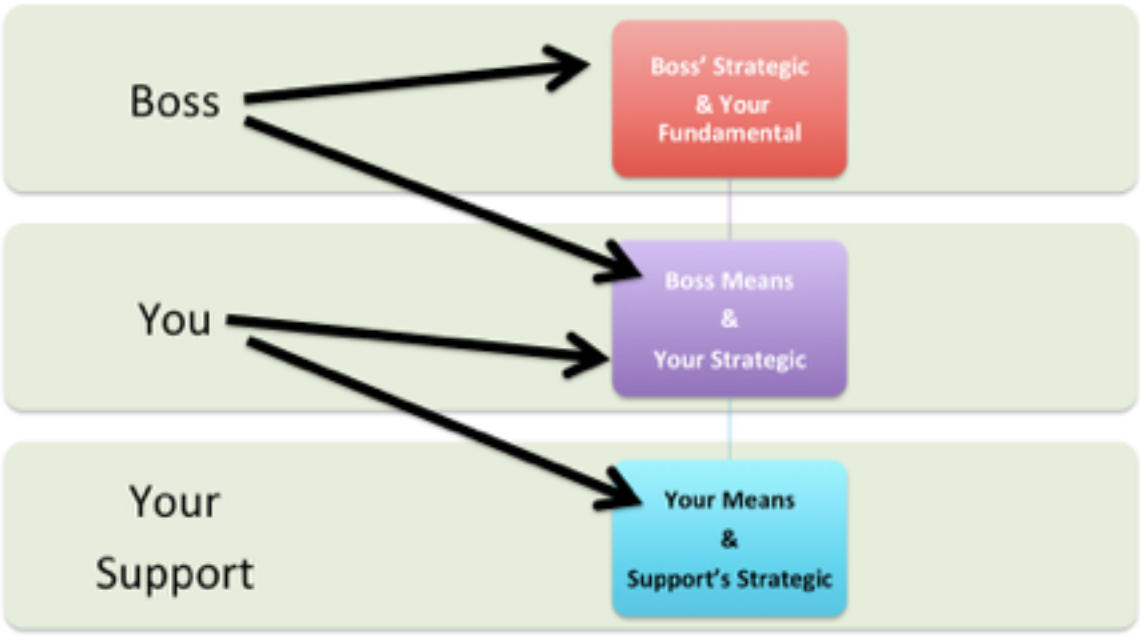


Figure 3 D. The perception of what is' ends' and what are the 'means' depends on your stakeholder level. 'One mans meat is another mans poison'.

Abstract and Concrete Strategies:
All objectives and strategies come in two basic different forms, *abstract* and *concrete*.

1. **Abstract Strategy**: a future result (Goal Level) is specified, but not a specific *means* for actually getting there. An 'abstract strategy' is really a 'variable result *objective' (an improvement statement)*. Sooner or later someone has to make some *concrete* 'means' decisions (*a specific decision, as to what to do in the real world to get that improvement*) as to how to get to the target levels. But, from the *point of view* of the 'next level above', the *abstract* strategy is still a 'strategy' (a means to *their* ends).

Examples of *abstract* strategies.

- *Increase productivity by 20%.*
- *Reduce Capital Expenditure by 10%*

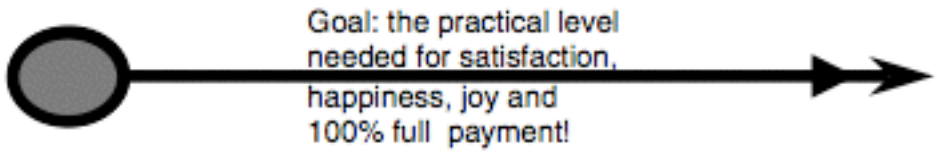


Figure 3 E. An 'abstract strategy' is expressed as a scalar objective, and it is designed to support a specific higher-level objective, with some required level of performance. It does not itself contain a particular specific 'concrete strategy', that might really deliver that level of performance. So, at some point we have to find real concrete strategies, to make it work, at intended levels of performance.

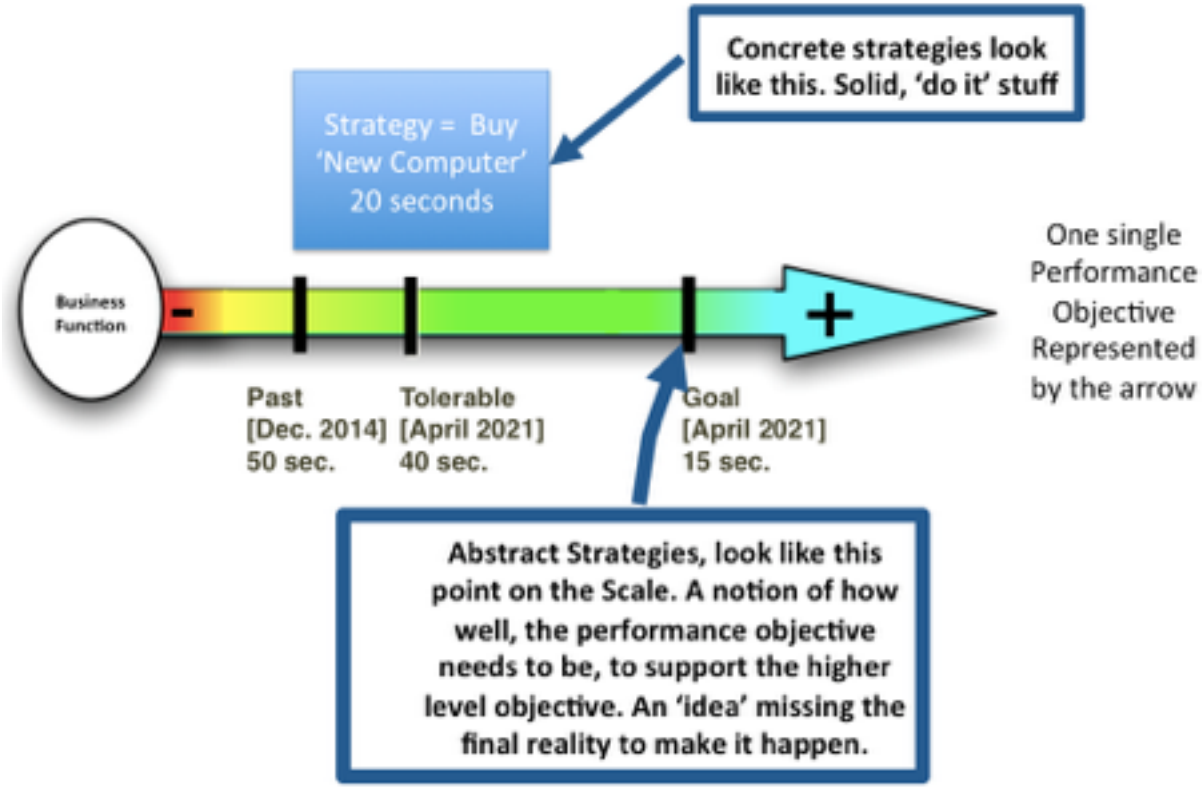


Figure 3 F. The difference between concrete strategies (just do it), and abstract strategies (be 'this good', (somehow)). That abstract strategy level will support the next level of objectives up. ***If*** *we can find suitable concrete strategies for reaching the Goal level.*

2. **Concrete Strategy**: a specific decision is made regarding the *real-world means* for supporting the objectives, on the level above it. A concrete strategy will give you 'whatever results it will give you', even if it does not in fact support the next level up, as intended.

So, committing to a concrete strategy, that *fails* to deliver our expectations, is not a good idea. It might be better to formulate it as an abstract strategy, and declare that, 'the means that *in fact satisfies* the targets', is the 'only real' right strategy.

Example of concrete strategies.

- *Layoff 50% of Temporary Workers*
- *Double the number of Permanent Hire Trainees.*
- *Use Supplier X*
- *Drop Product Line Y*

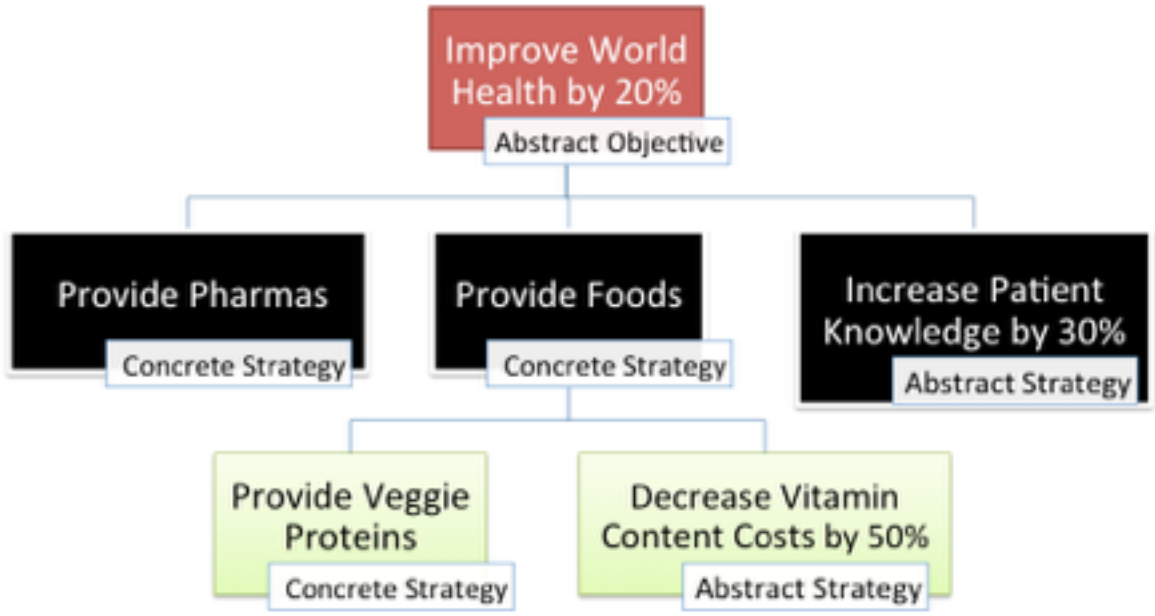


Figure 3 G. Strategies can be concrete (just do it) or abstract (let us achieve it), but the abstract ones, need, at some point, to have concrete ideas behind them.

Figure 3 H. Keeney's Levels of Responsibility and perception. All of these will have to operate by respecting a number of constraints. Constraints include budgets, deadline, and conformance to law, culture, policy and other agreements that apply.

In practice, the planner is always concerned with multiple objectives, and multiple constraints, that need to be dealt with 'simultaneously' in the planning process.

Simultaneously means that we want to achieve all performance target objectives, within all resource constraints, and by respecting all other known constraints. We will be looking for a set of strategies to achieve that balance.

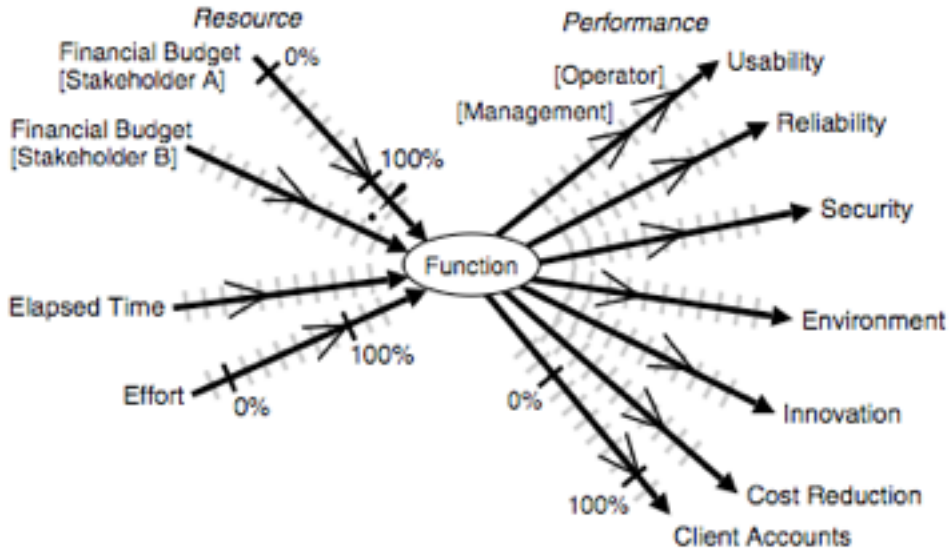


Figure 3 I: A model of an Abstract Strategy, with defined function, and ***multiple*** *performance and cost attributes. The arrowhead on the scale (--->---) arrow, represents a goal or a budget level.*

Strategy Specification Standards

We need to have something better than letting strategy planners just write down any set of words they feel like writing, and calling it a strategy.

The strategies have to be specified, and evaluated so that they credibly will meet the multiple performance objectives, within all constraints.

Here are some ideas of some basic standards, which we can use to encourage good practice when planning strategies.

Basic Rules for Strategies: General Principles

1. The **Detail** of the Strategy specification must be sufficient to **allow reasonable estimation** of all critical performance and cost attributes we can expect.
2. The **detail must be sufficient** to **correctly determine** if we have violated any applicable **constraints** (for example Core Values).
3. Estimation of the **expected range of impact** on the primary objective shall be made.
4. Estimation of the **side effects** on all **other objectives** shall be made.
5. The resources needed to **install, operate, and decommission** the strategy shall be estimated.
6. Analysis of whether the strategy alone, or *together with any other strategies* **violates any known applicable constraints** (starting with Core Values) needs to be made.

These considerations are intended to ensure that the specified strategies are intelligible enough to evaluate, with regard to their intended purpose (improving at least one objective), and with regard to all other side effects of interest (other objectives, resources, constraints, risks).

They help us answer the question: "Is the Strategy well-specified ? ".

They do not yet answer the question: 'Do we have the right strategy?" – but they lay the basis for asking and answering that question.

Deriving A Complete Set of Strategies

Let us imagine

- we had 10 Top Level Objectives, and 10 Top Level Strategies.

- Let us further imagine that each strategy was estimated to satisfy a different objective's target-levels on-time, exactly (±0% deviation from the estimate).
- Let us also assume that no objective or strategy seems to violate any constraint (starting with Core Values)

In this simple-world 'model' we could say that we **seem to have enough** strategies to satisfy all our objectives. There are no constraint violations. We are ready to proceed to actually delivering the strategies, and seeing the target performance of our organization emerge measurably.

In our real world we will need to manage strategy finding by:

- adding proposed strategies **incrementally** to the total set of strategies, until we seem to have **enough strategies**, even with uncertainty, to proceed. Until we have enough 'safety factor'.
- keeping our eyes open for each new proposed strategy, to see that it does **not violate** any constraints. Not least, resource constraints (time, effort, money).
- even when we seem to have a sufficient set of strategies, we can consider 'optimizing', for example to find **equally powerful strategies at much lower costs**, or with lower risks.
- keeping track of this strategy selection and evaluation process, **using a tool** – better than human intuition.
- We will suggest an **Impact Estimation Table** is a very good tool for this purpose. We discussed this to a limited degree above (Figure 2 A – example of Impact estimation).

Part 4. Quality Control of Strategies

Strategy QC for Clarity
We can perform a review (Specification Quality Control, SQC, [2, Chapter 8] of **the strategies, as a set** with regard to the rules we outlined above.

These rules and considerations amount to *making sure that we **have enough detail** to 'reasonably assert' that the strategies have a fighting chance to reach our objectives.*

If we pass this first test (fighting chance), then we are ready for the more-critical question:

"Are the strategies *really* going to be sufficient to reach our objectives, on time, within budget?"

This question still has two phases, for **Complicated** systems

(where there are no easy answers about what will happen when you mix various strategies in a cooking pot):

- **Phase one**: model and estimate the big picture on an Impact Estimation Table.
- **Phase Two** (a series of incremental cycles of testing and measuring the strategy implementation): add one single promising strategy to your current environment: see what really happens: measure all performance effects, and all costs.
    - This is essentially the same as a Deming 'Plan, Do, Study, Act' cycle.
    - But we have our own way of describing it.Below.

Figure 4 A. The Evolutionary Value Delivery Cycle. This encourages a step by step exploration of strategies and sub-strategies, measuring what really happens in practice, in a Complicated system. Locking in good results, and rejecting bad results quickly.

The *initial* IE Table is a set of estimates. It is a hypothesis. 'If we use these strategies, then we believe we can get these results".

But, in our real world of organizations, internationalization, and technology and new economics: nobody really knows. Nobody can estimate very accurately. Lots of things can turn out differently from our estimate, our hypothesis.

So, we have two options:

- **Big Bang**: build the new system using all the suggested strategies implemented, and committed at once. And pray.
    - People have been using this method widely for years, and the history is discouraging. We get too many really bad surprises.
- **Little Bangs**: we add small increments of our strategy hypothesis to the existing system, and see what happens. Like good management scientists we learn what is true and works well enough to keep. We learn what does *not* work. Dump it. Do not scale up. And continue the search for things that work better.
    - We gradually build up a series of strategies that really work.
    - Our 'worst failures', are small efforts, experiments, which 'succeed' in proving that we should not integrate them into our real system, and we should not scale up using them.
    - We feed the good results that we *choose* to keep *in place*, **into the Impact Estimation table**: now we are doing the 'results accounting'. We can compare to the 'Budgeting' of the original value and cost estimations.
    - We are gradually building up a picture of how good we are getting, and what it costs.
    - We can more easily see the gaps to our valued Goals, and see the remaining resources, budgets and deadlines; and decide intelligently what to do about them all.

'Little Bangs' (cute, I could not resist the name), we call it the **Evolutionary Value Delivery** method, or '**Evo'**, for short. Is nothing new. It has been good practice in science, engineering, management and society, not to mention nature, for a long time. It works.

But somewhere along the way, with business school academics, rather than practicing business titans (Jobs, Musk, Edison), we totally lost the plot.

One documentation of this is Hopper's excellent history, The Puritan Gift [2]. Deming, Out of the Crisis is another excellent source on the matter [9]. There are many, in our references.

Impact Estimation (IE) of Strategies: QC of Power.

Design Ideas ->	*Technology Investment*	*Business Practices*	*People*	*Empowerment*	*Principles of IMA Management*	*Business Process Re-engineering*	*Sum Requirements*
Customer Service ? <->0 Violation of agreement	50%	10%	5%	5%	5%	60%	185%
Availability 90% <-> 99.5% Up time	50%	5%	5–10%	0%	0%	200%	265%
Usability 200 <-> 60 Requests by Users	50%	5–10%	5–10%	50%	0%	10%	130%
Responsiveness 70% <-> ECP's on time	50%	10%	90%	25%	5%	50%	180%
Productivity	45%	60%	10%	35%	100%	53%	303%
3:1 Return on Investment	50%	5%	75%	45%	15%	61%	251%
Morale 72 <-> 60 per month on Sick Leave							
Data Integrity 88% <-> 97% Data Error %	42%	10%	25%	5%	70%	25%	177%
Technology Adaptability 75% Adapt Technology	5%	30%	5%	60%	0%	60%	160%
Requirement Adaptability ? <-> 2.6% Adapt to Change	80%	20%	60%	75%	20%	5%	260%
Resource Adaptability 2.1M <-> ? Resource Change	10%	80%	5%	50%	50%	75%	270%
Cost Reduction FADS <-> 30% Total Funding	50%	40%	10%	40%	50%	50%	240%
Sum of Performance	*482%*	*280%*	*305%*	*390%*	*315%*	*649%*	
Money % of total budget	15%	4%	3%	4%	6%	4%	36%
Time % total work months/year	15%	15%	20%	10%	20%	18%	98%
Sum of Costs	*30*	*19*	*23*	*14*	*26*	*22*	
Performance to Cost Ratio	*16:1*	*14:7*	*13:3*	*27:9*	*12:1*	*29:5*	

Table Part 4. Actual Impact Estimation table for US Army Persinscom System.

The above example shows the use of an IE Table to evaluate a set of management strategies (named in the top row) against a set of quantified objectives (referred to in the left hand column).

We can see the *rough relative power of each strategy* in the row 'Sum of Performances'.

We can see the rough *impact of all strategies, on each single objective* in the column 'Sum Requirements'.

We can *monitor cost and time resources* relative to a stipulated budget in the two rows 'Money' and 'Time'.

We can get some idea of each strategy's *value for money* by looking at the bottom row.

The % estimate is

- computed from a 'real scale' (like 'seconds') estimate for each objective (usually), in relation to a 0% baseline (like 'Past').

What is the expected impact, using the scale of measure defined for each objective.

The expected impact is:

- based on specified 'evidence' for the estimate
- based on specified 'sources' of the evidence

And each estimate includes:

- ± uncertainty or range of possible results ('65%±20%')
- a 'Credibility' category (itself based on evidence and source) of 0.0 (none) to 1.0 (Perfect).

This means we can *chose to be* quite *rigorous*.

Management can insist on rigor, on the part of those working out the tables, so they have a more-solid decision-making basis.

But it is not uncommon, initially, to be less rigorous, to get a quick feeling for how the planning is going, and the weaknesses that are glaring at us.

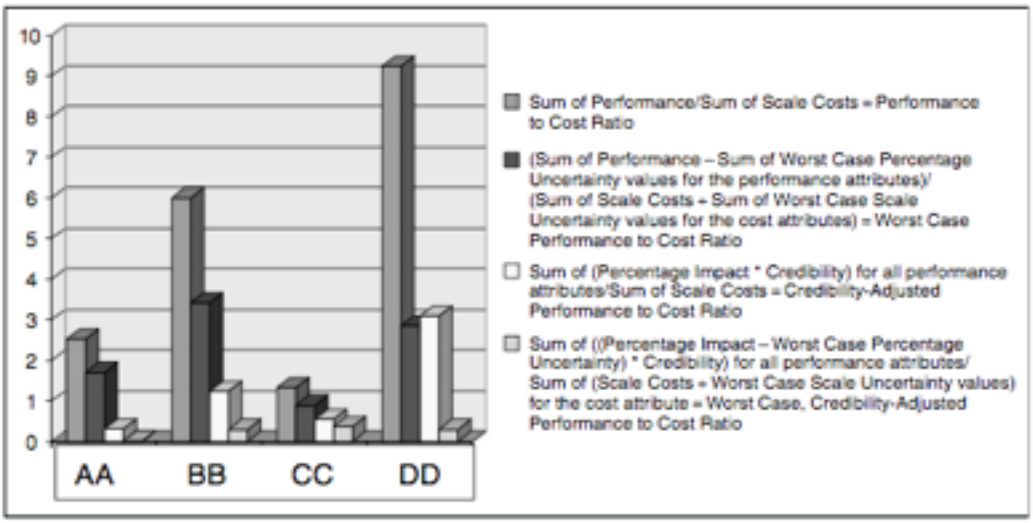


Figure 4 B: Management can look at the strategies from various risk level points of view. Source [2, p.287]. This is a top level summary from a tool built by Kai Gilb. Most of our clients build their own local tools using spreadsheets.

It is quite an amazing experience, to sit in planning meetings and presentations where this quantified environment, for objectives and strategies is present. This discussion is systematically focused on the weak points in the overall plan, which are numerically very visible.

The discussion is incredibly 'logical' and based on the ground rules:

- "Can you give that strategy such a high estimate, from such an incompetent source?"
- " Who cares if the strategy is so great for Security, we can see the damaging side effects for all other critical objectives, and the totally unknown, not-guaranteed cost and time aspects. Forget it! Find something better!"
- "Well, that is a great strategy, if you are an optimist, but we need to be sure it works, and there are far too many risky and dodgy circumstances surrounding it, documented in our planning. We have to go with the sure things that are also available in our plans".
- "We could only go for that strategy, if it can be *easily removed*, without financial and time loss, if it fails in smaller incremental trials; and if we get much more credible contract guarantees from the supplier, who is not very credible as far as our analysis goes."

Photo 4 Real London Bank, where there was a fantastic different logical planning discussion, based on quantified objectives, and quantification of the strategies.

Business meeting should always be like this, but very rarely are. Planguage is a balanced accounting system for management decision making, with the non-financial values being as-well-quantified as the financials.

Summary Part 4 Quality Control of Strategies:
The main idea, at this level of review, is to make sure that we have *probably* got a *reasonable* set of strategies, capable of helping us reach our target levels of objectives, on time.

We need, above all, to worry about obviously weak strategies and obviously weakly covered objectives.

Management likes to see a 'safety factor' of 2x to 4x. Especially if they are committing to an irreversible decision on this basis.

However, our ability to accurately estimate impacts and costs is inadequate. and cannot blindly be trusted for big irreversible decisions.

The IE Table is only one intermediary tool; hopefully better than what you are using at this stage today.

If we really want to make sure that our strategies will meet our targets, then we have to deploy them *incrementally,* and track the results, learning as we go. The 'Evolutionary' Strategy Implementation method. Part 5 below.

Part 5. Evolutionary Deployment of Strategies

Assuming we have a strategy set without obvious failings, as *yet*; we will normally insist that the strategies are implemented in small increments.

I have had good multi-decade experience using a guideline of 2% (of resources like budget or time to market) as a value-delivery step size.

The important thing is that the step size is small enough to

- limit risks of failure,
- to get something done in the short term,
- to keep up pace and motivation,
- and to allow us to see cause-and-effect more clearly.

I have seen that most top managers do not seem aware of how easy and possible it is to decompose *any large plan* into a series of smaller, incremental value delivery, plans.

Their subordinates are not helpful in suggesting this either.

There are however a large number of known tactics [2, page 314, fig. 10.6, "20 ways to decompose systems into small steps") [URL16] for decomposition. Most of them are based on simple imaginative common sense.

The key idea here is that top management adopt a planning policy that large plans *will* be implemented in small (2%) steps. No arguments! Most people can figure it out, under pressure to do so. The rest can be trained!

Some simple decomposition ideas are:

- Do one strategy at a time
- Do one sub-strategy at a time
- Do strategies in one location successfully, then scale up.
- Do not try to get the final target objective at once, increment your way towards it!

The Principles of Tao Teh Ching (500 BC)

"That which remains quiet, is easy to handle.
That which is not yet developed is easy to manage.
That which is weak is easy to control.
That which is still small is easy to direct.
Deal with little troubles before they become big.
Attend to little problems before they get out of hand.

For the largest tree was once a sprout, the tallest tower started with the first brick, and the longest journey started with the first step"

One of the most important opportunities you have, when you adopt this planning method, is that you can get very rapid and visible results.

You choose by planning, exactly which results to go for initially, and you can chose the ones with the best value for money, by looking for high numbers on an Impact Estimation table.

You can make sure your organization knows what it is doing, step by step, and if not, things are stopped, or corrected, before time and financial drain make you look like a foolish manager.

You can also delegate the power to get the job done, based on the objectives. Then you leave it to your subordinates to find out which strategies really work, and which do not.

A Simplified Evo Process

Background: A simplified version of the Evo process to use on small projects. It also serves to help understand the larger, full-scale Evo process.

Evo 1: Gather from all the key stakeholders the top few (5 to 20) most-critical goals that the project needs to deliver. Give each goal a reference name (a tag).

Evo 2: For each goal, define a scale of measure and a 'final' goal level. For example:

Reliable: **Scale**: Mean Time Before Failure. **Goal**: >1 month.

Evo 3: Define approximately 4 budgets for your most limited resources (for example, time, people, money, and equipment).

Evo 4: Write up these plans for the goals and budgets (try to ensure this is kept to only one page).

Evo 5: Negotiate with the key stakeholders to formally agree the goals and budgets.

Evo 6: Plan to deliver some benefit (that is, progress towards the goals) in weekly (or shorter) increments (Evo steps).

Evo 7: Implement the project in Evo steps. Report to project sponsors after each Evo step (weekly, or shorter) with your best available estimates or measures, for each performance goal and each

resource budget.

Evo 8: On a single page, summarize the progress to date, towards achieving the goals and the costs incurred.

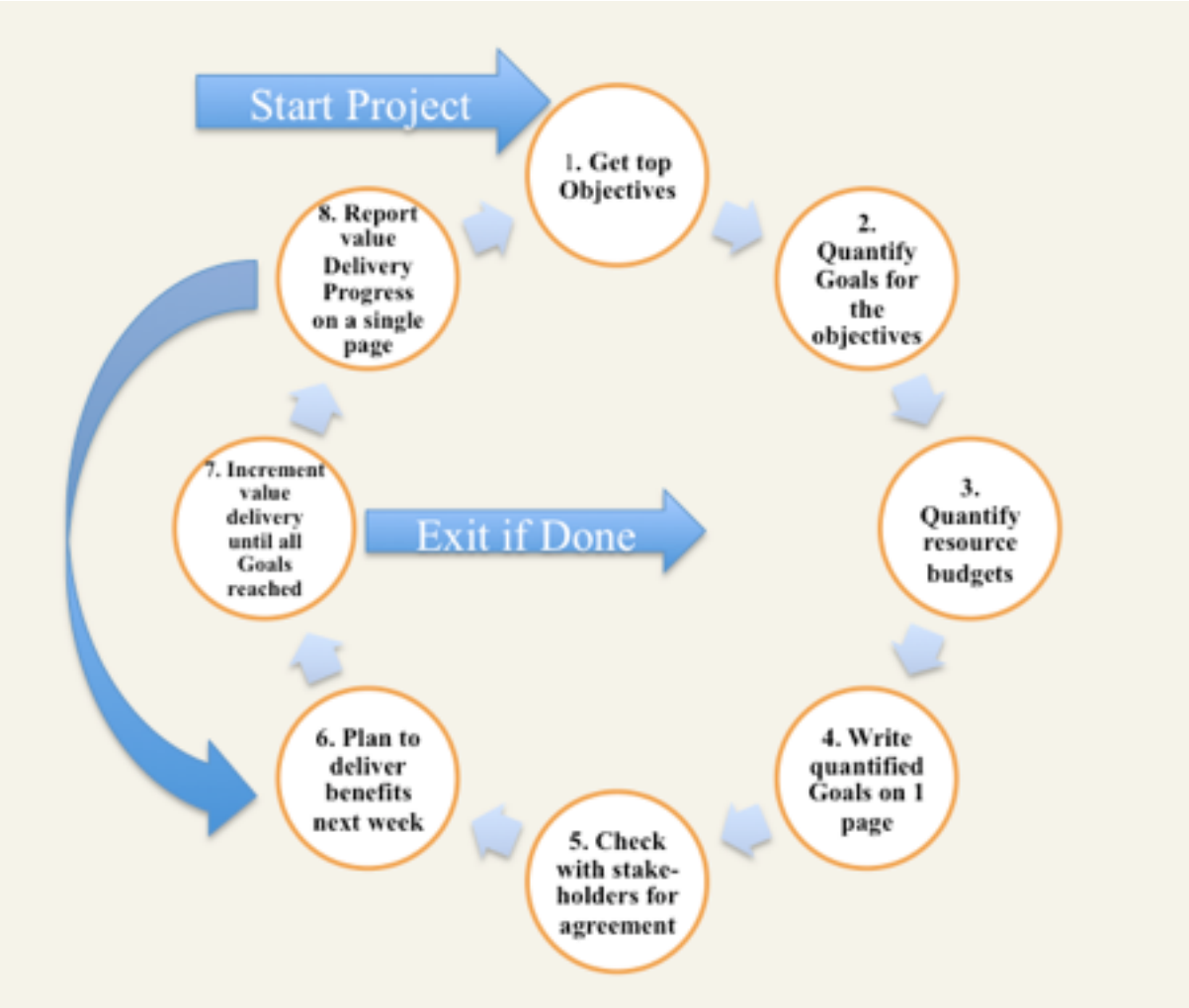


Figure 5 A. One view of the 'Evo', value delivery process.

'Evo' Project Management Policy

- **Efficiency:** The project manager and the project will be judged exclusively on the relationship of progress towards achieving the goals versus the amounts of the budgets used.
- **Creativity**: The project team will do anything legal and ethical to deliver the goal levels within the budgets.
- **Value Reward**: The team will be paid and rewarded for benefits delivered in relation to cost.
- **Freedom**: The team will find their own work process and their own design.
- **Feedback**: As experience dictates, the team will be free to suggest to the project sponsors (stakeholders) adjustments to 'more realistic levels' of the goals and budgets.

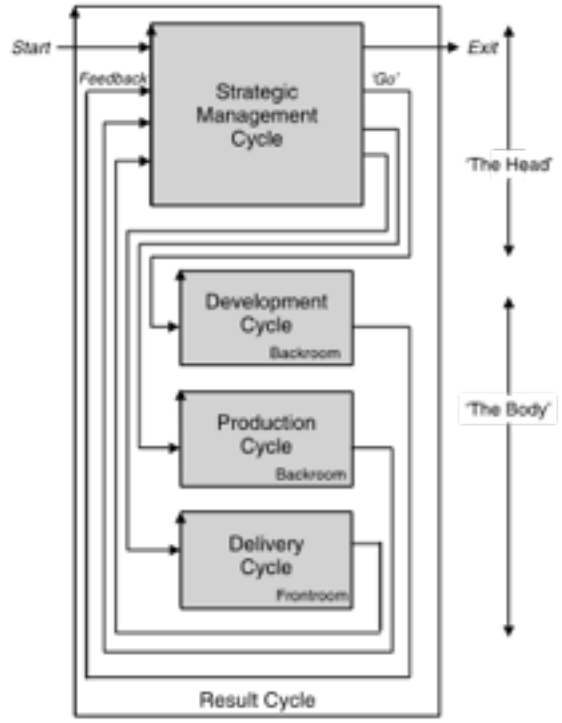


Figure 5 B. Another model of the 'Evo' process.

The Head manages the project.
1. T**he backroom development cycle** (like a kitchen preparing many desserts and soups, and roasting a lamb) can run in parallel and develop things that take more than one delivery cycle, to ready for integration into the system. See also Fig. 6.3 and Fig. 5.1 for Backroom visualization.

2. **The backroom production cycle** will get *everything* ready for implementation, in addition to the basic development of an implementable strategy (dishing the food onto plates in the kitchen).

3. **The delivery cycle**, in the front room (like waiters delivering food to guests) of delivery to stakeholders, will do whatever it takes to create real measurable value for stakeholders.

The backroom/front-room concept:
1. allows you to keep a steady rhythm of value deliveries, in spite of the fact that some strategies need more time to develop and ready, than a single cycle.
2. allows parallel development, with value delivery.

Summary: Value Planning at the Vision Engineering Level.

- Core Business Ideology can be translated into clarified and quantified statements.
- Strategies can be quantitatively related to quantified objectives.
- Strategies can be implemented in small incremental steps, to give early results, organizational learning and no-fail projects.

Local **REFERENCES: 'VISION ENGINEERING', Part 0 to 5.**
(there is a larger set of references at the end of the Appendix)

1. **Built To Last**: Successful Habits of Visionary Companies,
By James C. Collins and Jerry I. Porras. Harper Business Essentials, 1994
ISBN 0-06-051640-2, $10.10, paperback, amazon.com

2. Gilb, T., "**Competitive Engineering**: A Handbook for Systems Engineering, Requirements Engineering, and Software Engineering Using Planguage.", 2005, Elsevier, ISBN 0-7506-6507-6.

3. **Priority**
A. Gilb, T., et al **'Managing Priority'.**
http://www.gilb.com/tiki-download_file.php?fileId=60

B. **"Choice and Priority Using Planguage:**
A wide variety of specification devices and analytical tools."
http://www.gilb.com/dl48

4. Collins and Porras: HBR lead article, September/October 1996, Harvard Business Review. Chapter 11 in [1, pp. 219-239], **"Building The Vision"**

5. Keeney, Ralph L. 1992. **Value-focused Thinking: A Path to Creative Decision-making.** Cambridge, MA and London: Harvard University Press. ISBN 0-674-93197-1. http://www.fuqua.duke.edu/faculty/alpha/keeney.htm/.
See also Hammond, Keeney, Raiffa, "Smart Choices", Broadway Books, 1999, $14.95, ISBN 0-7679-0886-4

6. Gilb website for papers, slides and case studies: www.gilb.com

7. Tom Gilb and Dorothy Graham, **Software Inspection**, Addison Wesley, 1993, ISBN 0-201-63181-4. Japanese translation also published.

8. Nassim Nicholas Taleb, "**Fooled by Randomness: The Hidden Role of Chance in Life and in the Markets**.", Penguin, £9.99, ISBN 978-0-141-03148-4, 316 pages, Copyright 2004.

9. Gilb, Tom: **Estimation or Control?** Published in SQP, USA
http://www.gilb.com/tiki-download_file.php?fileId=460

Value Planning Appendix: Chapters 1 to 10

The appendix is suitable for detailed study of the Planning Language.

For experts, coaches, consultants, book and paper authors and even managers who want more examples, more references, more facts and deeper insights.

Each of the 100 Sections is headlined by a Principle which summarizes that section.

You can read this in sequence, or you can dip in anywhere that interests you.

The One-Page 'Value Planning' Book. 2

Introduction to the Book Structure and Intent. 3
Value Planning Appendix: Chapters 1 to 10 64
Chapter 1. Objectives. 65
Chapter 108
2. Strategies. 108
Chapter 3. Levels of Interest and Levels of Control. 166
CHAPTER 4. Value Delivery. 212
Chapter 5. Decomposition (by value, by responsibility) 256
6. Prioritization, Evaluation 296
Chapter 7. Risk Management 344
Chapter 8. Delegation, Outsourcing, Contracting 406
9. Communication 464
Chapter 10. Quality Management. 535

Chapter 1. Objectives.

'Objectives' are the plans we have for *what we want to achieve*, independently of which 'strategies' ('means') we might later select, to achieve them.

Section 1.1 Quantification of All Critical Performance Objectives, Especially Qualities..

Principle 1.1 Quantify Objectives

All critical performance objectives
***can* be quantified,**
and *must* be.

An 'objective' is a stakeholder-valued future performance level, that we plan to work towards achieving.

An 'objective' is the planning artifact representing the values that our stakeholders prioritize.

Objectives are usually a degree of *improvement* over the current state.

But an objective could also be about *'not getting much worse, too quickly'.* Which might be an 'improvement' over our fate, 'an improvement over a potential future state', if we did *not* plan otherwise.

An 'objective' is a future plan, *for a defined **set** of planned future states, in various times, places and circumstances ('ifs').*

***Sales*: Scale:** *€ Net Long Term Profit on sales*. **Goal** *[By 2020, Europe, Digital Watches, If New Eu Tax Law = Passed] €22 billion*.

Planning language example 1.1 A. A one-liner objective for 4 defined dimensions.

Notice that we collect the set of conditions ('dimensions'), that 'qualify' the objective, with [square brackets]. This 'multi-dimensional specification' is a useful and very-flexible planning convention.

[By 2020, Europe, Digital Watches, If New Eu Tax Law = Passed]

Planning language example 1.1 B. A Qualifier, telling us more exactly 'when, where and if' the numeric target (€22 Billion) is valid. This helps us 'model', represent when planning, our complex reality. We can specify any number or type of dimensions that are useful.

Objectives as a 'set'.
Objectives are not always a **single** target number, for example, not like *'The Efficiency Objective is 99.98%'*.

Objectives are *often* a **set** of numbers, planning different futures, in *space*, in *time*, and for *various other conditions*; for the same 'joint concept name' (a formal, consistent unique defined Capitalized 'Tag'), like 'Efficiency', 'Productivity', or 'Competitiveness'.

Figure 1.1 A. An objective with a set of several sub-objectives.

The term ’objective' is used to describe *any set* of future performance plans, that we find useful to ‘group together’, under a single objective’s ‘Name Tag’.

One useful attribute of this ‘Objective Set’ planning representation is that it *prompts you to thinking, asking, and deciding, cumulatively* about what all the various interesting combinations of ‘when’, ‘where’ and ‘if’, that might be useful to plan for.

The key idea is to distinguish between combinations with the following attributes:

- risk of failure
- value to stakeholders
- costs of implementation
- political visibility

so that we can select, and do earlier, those combinations that have *highest priority* for our political, economic and practical environment.

Articulate specification gives you better choices, to *prioritize* some of those high-value combinations for early value delivery.

The ‘Productivity’ objective, in the examples, can encompass any useful number of more-specific sub-objectives. It can also be constructed or viewed as a multi-level hierarchy of sub-objectives.

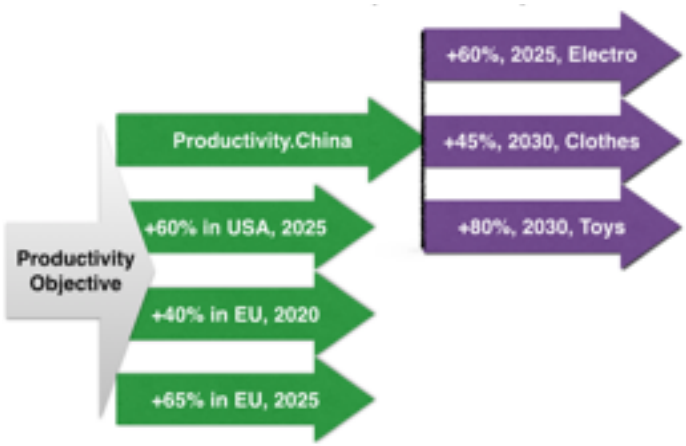


Figure 1.1 B. Multi-level hierarchy of sub-objectives.

Notice our specification convention of *hierarchical tagging* of planning objects, using the '.' between tags of the hierarchical levels to 'connect' any interesting levels of planning as in *'Productivity.China'.*

Notice that we can add in any interesting set of dimensions that we wish to plan for; in this case the product types ('Electro', 'Clothes', 'Toys').

Quantification of Objectives.

The fact that we can use words, like *'enhanced', 'improved', 'better'*, to describe our objectives, is evidence that these objectives are *variable in nature*, and that they *can* therefore be *represented using numbers*.

One simple 'objectives planning format', amongst many, for doing this is **'**OKR (Objectives Key Results)'. [URL73] as used at Intel, Google and elsewhere**.**

This book is going to present a range of enhancements to the simplest basic formats for specifying objectives. Such as OKR. These optional enhancements will serve to improve information about risks, priorities, and alignment with other objectives.

There are two primary initial steps to basic quantification of objectives, 'Scale + Levels'.

- First a '**Scale** of Measure' needs to be defined.
- Then interesting ***levels*** (like Past, Goal, Tolerable) of performance, of that 'quantified scale', need to be specified.

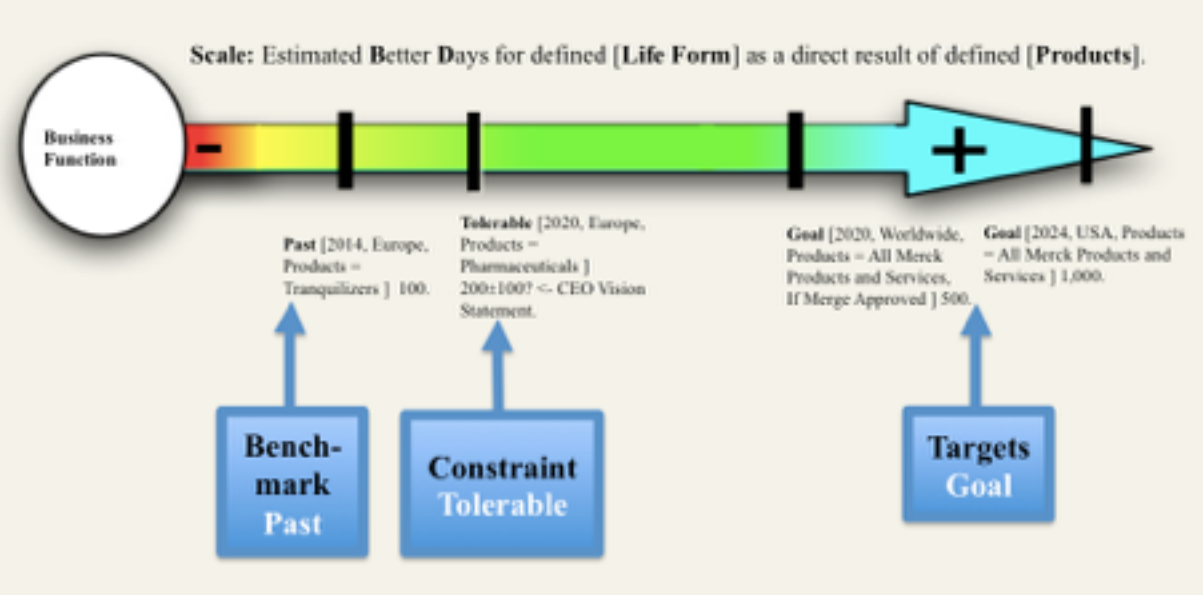


Figure 1.1 C . The arrow icon, ->, represents the 'Scale of Measure'. for a performance variable for a defined business function (the Oval Icon). The Scale variable 'defines' the objective's nature (top line). This figure shows three primary concepts of 'levels of quantification' along the scale: Benchmarks, Constraints, and Targets. 'Past' is a type of Benchmark level. 'Tolerable' is one type of Constraint level. 'Goal 'is one type of Target level. (Figure was used earlier, Fig. 1 i, Merck case).

We use three basic categories of 'performance objectives':

- **work capacity** (how fast, how much we can do),
- **financial objectives** (earnings, savings), and
- **qualities** (how well we function).

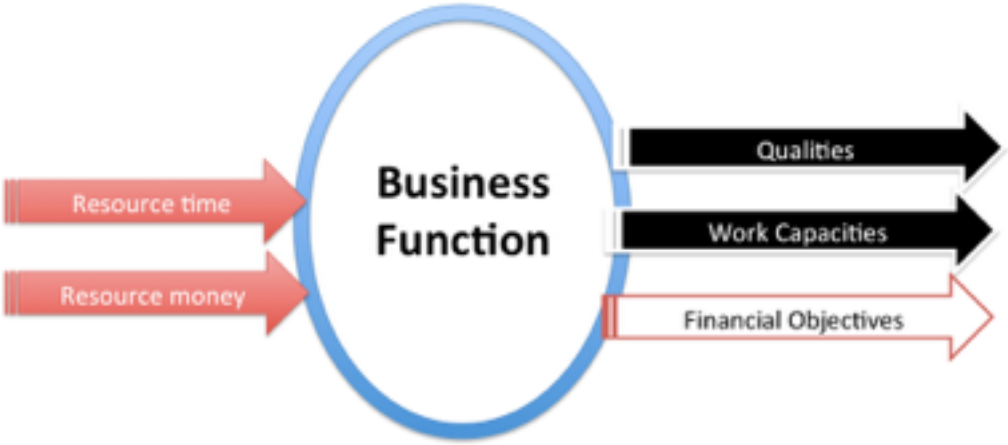


Figure 1.1 D. Three types of 'performance' objectives. The right-hand arrows. O->

The left-hand arrows, ->O, represent any limited resources (needing management) needed to develop and maintain the performance objectives. Budgets and deadlines for example. (Reuse of later Figure 10.6 E2).

Most Planners Need Help to Quantify the Quality Objectives.
Most managers need little instruction in quantified specification of the first two of these; 'financial' and 'capacities'. But managers usually need *considerable* help in dealing with the other side of the 'un'-balanced scorecard, **qualities**.

Qualities describe **'how well'** a system (organization, process, project, product, service) performs.

We have found no exceptions to the principle that; '***all critical performance improvement objectives can be quantified'***.

And that includes all **quality** objectives. All 'qualities' can be expressed numerically.

If you don't know how to quantify performance objectives, especially 'quality' performance objectives, or only *some* of them, we will suggest some methods below [URL9, TEDx].

Commonly stated 'word-only' objectives, *like 'world class quality', or 'enhanced responsiveness to market dynamics'* will be unclear, probably even to the originator themselves. 'Unclear' means that even *they* will struggle to make an acceptable, useful, and clear-to-others interpretation, if you ask 'what does that mean?'

But, worse, there will be quite different interpretations, for *every different person*, who reads or hears these 'word-only' objectives.

You only have to ask people to write out their personal (mis-) understanding of such vague phrases, to see this inevitable confusion. I call this process the 'Ambiguity Test'. And it is fun to do it in a serious meeting. People are usually 'blown away' at the lack of common understanding of critical objectives.

The 20-30 minutes used to make the 'lack of common understanding', of critical objectives, clear to all, in a meeting is a very good use of time. Because it might lead to a superior and powerful common understanding of your objectives. And might just save your project from failure and years of wasted time and money.

You do need your team 'aligned to the same music composition', don't you?

You might not initially arrive at the, with hindsight, 'right' objective: this might take time and feedback. But by getting a truly common-to-participants, and well-understood objective in the planning, you can more easily learn how well that specification works *early*, and modify it, as needed, later.

'Ambition Level' Phrases are not enough.
'Word-Only Objectives', like '*vastly superior market recognition'*, are a waste of time, except as a starting point for better definition, or as a popular summary [URL1].

We often use these vague phrases as a 'background component' of an objective specification, called an '**Ambition Level'** statement. We might even quote the top manager source of the Ambition statement. But we don't *ever* stop at such a 'rough' level of specification. The objective specification is always subject to further quantified definition (using Scale etc.). This is an unbendable standard, a Rule [1F], for the Planning Language.

Your top-level *critical* objectives deserve that small additional effort. And the clarification effort should have a huge payback.

Efficiency:
Type: Corporate Marketing Objective.

Ambition: 'Competitive Marketing Efficiency' <- CMO, Annual Report.

Scale: The **M**arketing **E**fficiency in defined [**Markets**] for defined [**Divisions**].

Marketing Efficiency: defined as the ratio of % Market Share in Gross Financial Sales / Related Marketing Costs as % Gross Expenses.

Goal [Within 3 Years, **Markets** = EU, **Divisions** = Electronics] 10% Sales / 20% Costs, or similar ratio: also expressed as 0.5 efficiency.

Planning Language example 1.1 C. An 'Efficiency' objective can mean many things, but here it is pinned down to a clearly defined objective. 'Marketing Efficiency' can also mean many different things. But it is clarified here.

The problem is that so much current planning practice stops, *before* the *necessary clarification* of objectives. Planning stops at this 'Ambition Level' containing 'nice sounding words' only.

The conventional planning process usually *never* clarifies the objective, or even gets the right *level* of objective. That is what we see in practice.

The nice-words-only objective, for example, is very likely to be only a presumed 'means' to your true ends. You might *get* what you ask for (a 'means'), not what you really want and need (true 'ends').

Methods for quantification: [URL1, URL9, URL10].

- **Method 1**: **Manual Thinking Process.**
  - Use your imagination and domain knowledge.
  - Decompose your objective into a set, or list, of sub-objectives, if possible. 'Love is a many-splendored thing'! (see Section 1.7 below)
  - Describe, for each sub-objective a quantification 'Scale' in 5 to 20 words: one word alone will never be unambiguous and clear enough.
  - Make each 'Scale' *relevant* to the *key value* your stakeholders are interested in.
  - Tailor the Scale in *their* language, to *their* environment.
  - Do NOT, yet, try to describe 'how to measure' it. That comes later (the 'Meter' statement). You are working at the conceptual level of the 'volt', the 'sales per customer', the 'Patents produced per year' level. Quantification.
  - Add useful '[scale parameters]' (in square brackets), like

    - **Scale:** average per day number of **[Customer Types]** buying **[Product Types].**

      And define these subsequently in the 'numeric level statements' like;

    - **Goal** [Deadline = next Year, **Customer Types** = New, **Product Types** = Luxury] 500 buys.

  - Don't worry if you don't get it 'perfect' on the first try.
  - Draft *something*, and then work on improving your draft.
  - Insights will come. Especially in a group. Add them in.
  - Try *using* the Scale with a sample of a Goal statement. Then you will see needs to improve the Scale, because of your experience of using the Scale for the Goal specification.

  - The Scale statement can be reused for many different specifications of levels, as discussed above in this Section 1.1. For examples, different Goals for different time horizons, or different markets.

- **Method 2**: **Handbook**.
  - look for a suitable scale in a **handbook** of 'scales of measure'.
  - There aren't many handbooks, but there are *some*, even some 'standards' (which are of course 'all different'.)
    - Your organization should develop its own handbook one day.
  - For example, here is my free Scales handbook for you, 'Scales of Measure' Chapter in Competitive Engineering [1 B, free download URL].
    - You will find the common '-ilities' there: Usability , Security, Maintainability. In depth.
  - Tailor the templates you find there, to *your* domain.
  - The ability to 'suggest Scales of measure' should be built into tools such as [URL73, Smith's Tool]. As it was in the earliest Planguage tools [5E, Posem, Aspect Engine by Krzanik]

- **Method 3: Google a Scale:**
  - Use *'your keyword'* + 'metrics'
  - Example : 'Office Productivity metrics'
  - What you need, or near enough, tends to be on the first internet page shown.
  - You should always check this 'internet search option', before you make a fool of yourself at a meeting, by saying 'XXX cannot be quantified'. Believe me, they all can!
    - *If they vary*, then they can by definition by made numeric. Even if you *don't* find ideas on the internet.
  - Your personal lack of knowledge, is not proof that nobody knows. People can be arrogant about this.

"In business, the idea of measuring what you are doing, picking the measurements that count like customer satisfaction and performance... you thrive on that."

Bill Gates (1955 -)

Practical Tip 1.1: Just do a Scale and a Goal.

Pick a currently critical objective.
*To specify its quantification, use the simple basic format, in this example: a '**Name Tag'**, **Scale**: , and **Goal**.*

Perhaps you can also add 'qualifier' conditions, as in the example below (like 'When'. 'People', 'Correctly', 'Respond') to define critical parameters about your objective.

For example:

Responsiveness:

Scale: Hours needed for defined **[People]** to **[Correctly]** **[Respond]** to defined **[Situations].**

Goal: within 24 Hours**:**
When = End Next Year**, People** = Director Level, **Correctly** = Legally & Without Complaint**, Respond** = Take Action resolving Situation**, Situation** = Emergency.

Do not worry yet about how to actually measure things, along that scale: that is the role of our 'Meter' specification. Meter definition will be dealt with later [Section 1.6].

But rest assured, when you quantify with the Scale statement, there seem to always be one or more useful Meter options to fit it. Meters are usually just common sense and domain knowledge. Again, you should be able to Google suggestions for Meters.

Someone has done this measurement before and published it on the internet! Or they have at least published something close enough, to inspire you to tailor it, to exactly what you need.

*You and your plans are 'unique'. but not **that** unique. You do not need to reinvent the wheel. Just adjust earlier experience to your current needs.*

Policy 1.1: 'Clear Quantified Objectives' Policy

- **All critical planning objectives will be expressed with defined Scales of Measure and Numeric Levels.**

- Why ?
  - Force ourselves to think deeply and clearly.
  - No management BS [URL1].
  - Taking responsibility for results.
  - Clarifying limits to responsibility.

Section 1.2: On Limiting Your Project to 10 Critical Objectives, Initially.

Principle 1.2: Prioritize Your Top Ten Objectives.

It is sufficient to initially focus on
up-to-ten
most-critical objectives,
at any given level-of-responsibility

There are usually far more than 10 planned-system attributes that we value, which *some* stakeholders would *like* to improve.

But if we try to *identify, and work on,* 100-or-more things at the same time, we will likely lose focus on the most-critical 10 things. The top 10 will not likely be achieved early, if ever.

We believe and practice, that any given level of responsibility (*project manager, CTO, IT Architect - for example*) should consciously limit themselves to a handful (*or 2 handfuls*) of the most-critical objectives, initially.

When these values are delivered, or at least safely delegated to others, or on their way to being reached initially, and then being scaled up: *then* it is time to turn to the *next* set of 'objectives' priorities (*priority number 11 to 20 if you like*).

Objectives Brainstorming Meeting.
One way we do this, is likely to be in a meeting of project and system related people: 'stakeholders'. Stakeholders who we need to 'get to agree what the important objectives' for a project are, or what they are for a given stakeholder responsibility level.

We ask the stakeholders to list the *names* of their most critical objectives, and to decide what the top 10 objectives, maximum, are. The rest, the not-top-10 can be collected, for future use; 'below the line', under the top 10 objectives.

This brainstorming session usually takes about 1 hour of meeting time, and it is not too difficult to get 'pretty good' agreement in that hour. But you do need to constrain the meeting to doing this in an hour. Some people can discuss objectives 'forever'.

This initial list of objectives is not 'final' or 'binding'. It just needs to be *sufficient for purpose.* The next-stage purpose is *refinement* and *clarification* of the selected objectives.

Naming and identifying the objectives is just the *first* stage of definition.

It is unreasonable, and illogical, to expect serious *commitment* to these (undefined) objective's 'names' until they have been more well-defined, and quantified, later, so that people know *exactly* what they have *agreed* to prioritize.

Make that point, about insufficient definition, to the meeting participants, and initial agreement, in an hour, will be even easier.

Tell the 'objectives brainstorming' participants that much better *prioritization* can be made, when *well-defined* objectives exist. Then, together with the corresponding strategies suggestions, and with some knowledge consequent *strategy-costs of reaching planned-levels*. We can make more intelligent choices of objectives.

We shall discuss this advanced prioritization process in detail, later [Sects. 1.9, 2.8, Chapter 6]

That initial 'objectives prioritization' process, with better definition of objectives can take the rest of the first day, of hard parallel work on defining objectives.

The definition of Scales, Goals etc. is needed, to be absolutely clear to all, just what is actually being 'prioritized', and selected for initial, short term, value delivery.

The multiple Stages of Prioritization of Objectives.
It might take another day or two of focussed work to find and evaluate the corresponding **strategies**. Only *then* can we analyze and understand whether we can 'possibly' afford the luxury of the target levels we wished for!

It is easy to dream about perfection of objectives (ever hear someone say '24/7' ?) but most people cannot afford the *cost of perfection*: **infinity**.

∞

This, 'strategy costing' process, is a third stage of prioritization of objectives.

The actual costs of delivering a value level are decided by the actual *strategies* selected to deliver the values. *Not* by the value levels themselves. For each value level there are several strategies, and some are far cheaper than others.

You cannot 'cost estimate' based on objectives alone. You need to take a look at the *strategies* you will use, to reach the desired levels, on time [URL11].

Here are six stages of objectives development,

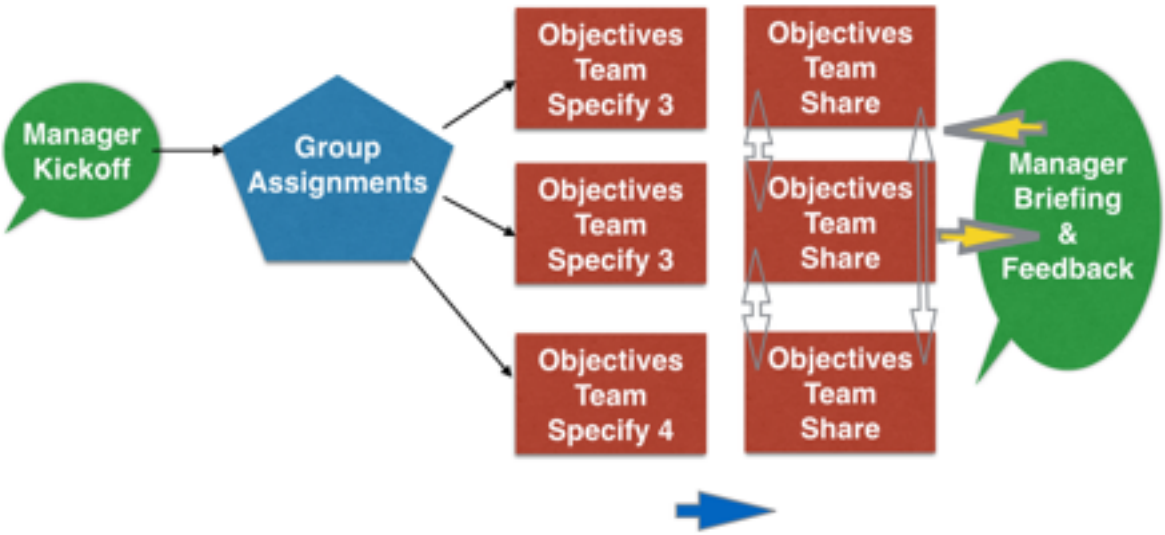
1. top ten nice-sounding names selection, by a stakeholder team
2. re-prioritize objectives based on clearer quantified value definitions
3. re-prioritize based on the 'costs of the strategies' needed to reach desired levels of performance.

Further methods of *prioritization of value delivery,* will be discussed in detail this book [Chapter 6]. But here is a short summary.

Some additional levels of the process of 'prioritization of objectives' can be characterized as:

4. balancing the demands of *other* high-priority objectives against any one objective in particular. This is the 'competition for limited resources' to reach desired levels. If you actually had unlimited resources, *and you do not have them of course*, you could then have 'all values', at 'perfect' levels.
5. feedback from pilots or trial installations of chosen strategies, and consequent re-prioritization of objectives, when 'real costs' and 'real levels of effect' on objectives are actually measured.
6. changes in priorities due to emergence of either 'new objectives', or of 'dramatically new cost-effective strategies'. Before, and even well after, initial deployment of the system planned.

Figure 1.2 A. Stages of refinement of the definition and prioritization of objectives. See above for detail. This is not a fixed set of stages of objectives development. Many variations are possible, But the main point is that objectives need to get developed in stages. There is no one simple process for developing them. There is a logical process to these particular stages.



You can perhaps see that 'prioritization of objectives' is not a simple one-step process, and you cannot simply assign 'subjective weights' to objectives.

Weights are static and subjective. We need 'dynamic' and 'objective' methods to survive our complex reality.

Subjective weights (like 50% weight, ***, High), as a prioritization tool, as in Balanced Scorecard, and many other planning methods, are dangerous planning tools! [URL33 (Prioritization), URL62 (BSC), URL35 (QFD)]

Back to the Team Process: the first day of finding the top ten objectives.

Three people might be working, most of the first day, on three of the 10 objectives for example. Defining them better with Scales, Past levels, Tolerable levels, Goals, and multidimensional conditions.

The collective stakeholder team would aim to mutually present, collate, and discuss their more-detailed objectives to each other, during the last remaining hour, before an 'end of day presentation' to the Boss (10 to 30 minutes), or to another major stakeholder.

Teams work in parallel, most of the day, to get enough time in that single day, to handle all 10 objectives reasonably well.

Figure 1.2 C. The first day of the project startup week. Focus on objectives.

We are, in those last two sessions of the first day, looking for feedback and buy in, as well as 'learning the planning language': by *reading* it, and by *having discussions* within the stakeholder team.

There is a need for *someone* to act as coach and trainer, if the planning language, and this process of developing objectives, is new to all other participants.

This process works well, and we use it as 'Day 1' of our 'Evo' Startup Planning Week, process [References to the SPW standard, and to the cases slides are URL3, URL4, URL5].

Top 10 Objectives Quantified On 1 Page in 1 Day.

We normally finish, in 1 day, with the 'top ten objectives quantified', *on a single page.* We might have to edit just the essentials from the team's more-detailed specifications to get things down to a single page.

This 'all quantified on 1 page' achievement is fairly dramatic for those who experience it for the first time.

People who have never seen well-articulated objectives at all, let alone reasonably well-quantified objectives, are impressed. Powerful and simple.

One-page quantified objectives is done predictably, by design, in a single day, and presented on a single page or projector image.

We sometimes enjoy telling people in advance that we are going to quantify all critical objectives on a page, in a day. Most people think we are exaggerating.

But they themselves always manage to do it. It is a matter of constrained prioritized discipline.

Some corporate projects, at some of our clients, drag on with expensive management meetings through months of trying to agree on objectives, without achieving such clarity and decision.

This quantified, well-defined set of of critical objectives is the essential 'core' of any project or any plan. Sharp and simple.

The initially drafted *single* objectives with their supporting background *detail* (see Fig. 2.2 B) can take up a half page or more, each.

A small team may have labored over that one objective for 1 to 3 hours that day. A lot of related thoughts get documented. Good, but it takes some space.

Operational-Control:

Scale: % of trades per day, where the calculated economic difference between OUR CO. and Marketplace/Clients, is less than "1 Yen"(or equivalent).

Past [April 20xx] 10%.

Goal [Dec. 20xy] 100%.

Planguage 1.2.1 Real London Bank Example of 1 objective done on 1st day. OUR CO, and xx are artificial masks.

But no matter what the initial single-objective detail turns out to be, each objective can be condensed to essentials for a one-tenth page presentation.

Here is a real example of a one-page 13-Objective Summary, slightly masked for client detail (Planguage Case 1.2.2).

Planguage Case 1.2.2 An actual 1 page presentation edited from 1 day of planning by a London Bank team.

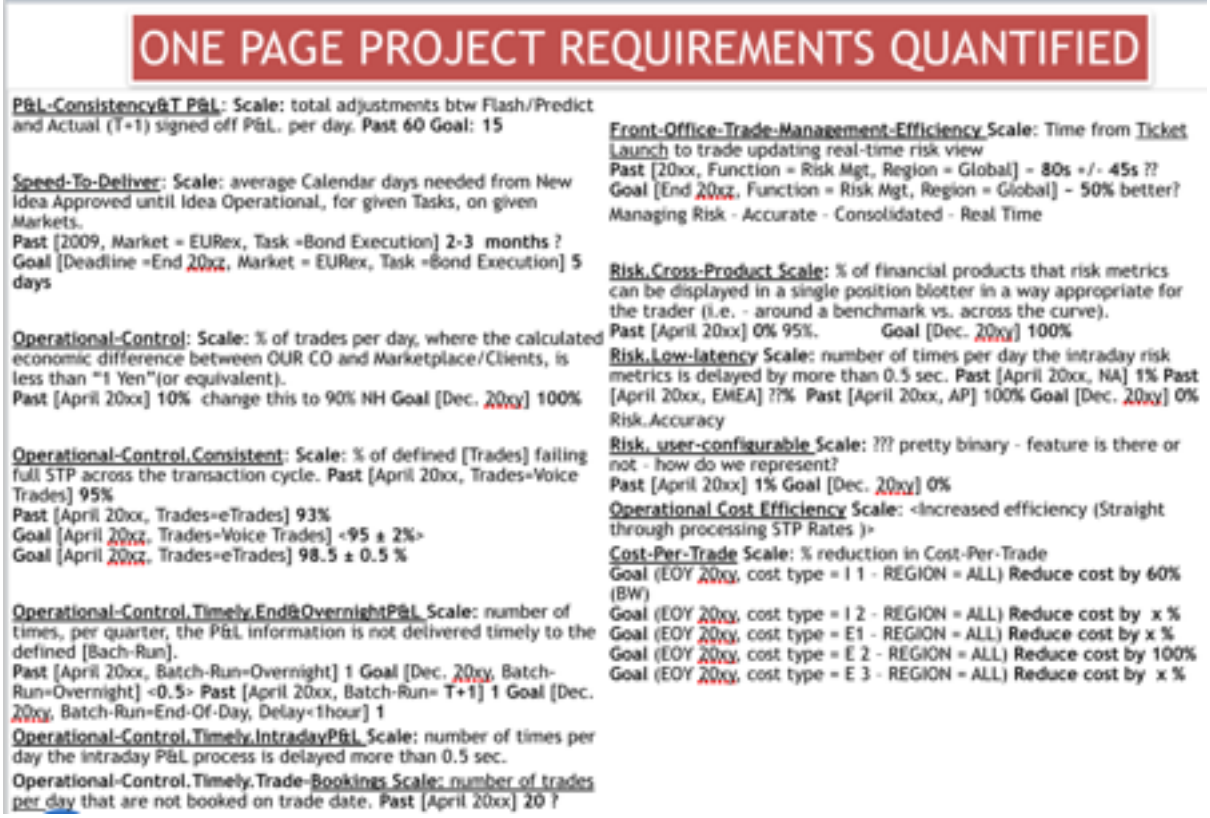


Figure 1.2 B, below, is a simplified example of another client's 'one page of objectives', which might be good enough for certain presentations and purposes. But not detailed enough for other purposes.

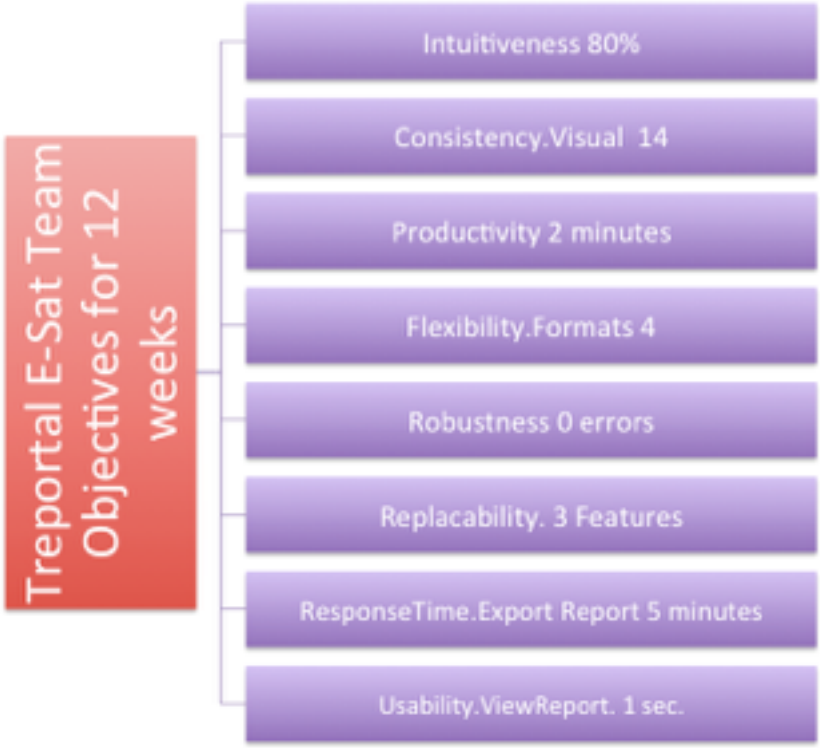


Figure 1.2 B : very simplified presentation of top 10 quantified objectives for a real client project, for their coming quarter of a year.

The necessary 'Scales of Measure' (see Section 1.1 above) are not included here, *but were properly defined at the client site. Otherwise the numbers would have no meaning. The 'scale' words we added, for the Figure's presentation intelligibility (sec., minutes, Features). is not a 'proper' scale. We need, for ultimate purpose, much better definition like 'from when', 'to when', and 'for doing what'. They did that, of course.*

Case 1.2: Manager Insists on Quantified Objectives

Bill, a banking VP from New York, asked his boss in London, Barney, about Barney's main objective for their project's Startup Planning Week [URL5].

Barney replied: "I'd be really overjoyed, if for the first time in this Bank's history, we managed to *quantify*, and thus *clarify*, the primary objectives of our large IT projects, at all." (Barney, leadership!).

Bill (being a smart politician), privately, decided to spend an extra day, with Kai Gilb, making sure the quantified objectives were top notch, for his boss.

Bill and Kai did this *second* day on objectives, while others on the team, and I, started work on the top 10 *strategies*, in parallel, on Tuesday.

Refinement of objectives, endlessly, is always possible, but not always necessary for immediate purpose.

Our immediate purpose, the second day, was to brainstorm the top ten *strategies,* to meet our top ten objectives. Refinement is not necessary, beyond initial 'Scale' and 'Goal' specs for the purpose of brainstorming your strategies.

Practical Tip 1.2

- *Take a look at your current project.*
- *Select a maximum of ten critical objectives, from your point of view.*
- *What do you guess would happen, if you managed to achieve **all** of those, halfway through your project budget, or halfway to your deadline?*
    - *Because that is a real possibility, according to our experience, if you focus on few well-defined objectives, and prioritize high-value deliveries early.*

POLICY 1.2 Top-Ten Critical Objectives Policy

- **The first day of any project or major effort,**
    - **we will decide, for the moment,**
    - **on the top-10 most-critical objectives: and**
    - **quantify them, on a single page,**
    - **for responsible management approval.**

- *Why?*
  - *Because all consequent effort (strategies, estimates) is logically impossible, without this clear-objectives basis to work from.*
  - *Failures at this 'objectives' stage threaten to become failures at all consequent planning stages.*
  - *We need to start focussing on 'value', early in the planning, and this 'top-ten prioritization' is a good, simple first step.*

- *What if we need to* change *the top ten?*
  - ***Do that.*** *Change the objectives, when and as you need to.*
  - *But do not use inevitable changes to objectives, and new insights about them, as an excuse for having initially fuzzy objectives; just names or nice-words.*
  - *Do your best today. Evolve better tomorrow.*
  - *It is surprising how often we experience that the top ten objectives are remarkably stable, even unchanged for the duration of a project [URL82, Richard Smith].*

Section 1.3 Clearly connecting the level of responsibility with objectives.

Principle 1.3 Your Objectives Support The Level Above.

Your level of objectives
must clearly support
the level above you,
your 'boss'.

Ralph Keeney [3] proposed an excellent practical idea to sort out your responsibility, from your bosses responsibility, and your support team's responsibilities.

- Your objectives (called '**strategic'**) must clearly support the achievement of the *next level of objectives above you* (your 'bosses' objectives, called '**fundamental'**).
- Any objectives that presume to support *your* strategic objectives, i. e. the objectives of your subordinates, or support teams, are called '**means**' objectives. The means to reach your strategic objectives.

This scheme for organizing objectives helps in many ways:

- We know what we must *prioritize*, and what we can *change.*
  - *Priority 1: Fundamental Objectives*
    - *you might tactfully hint at the need to change these*
  - *Priority 2: Your Strategic Objectives*
    - *you are in full control of change here*
  - *Priority 3: Means Objectives, serving your Strategic Objectives.*
    - *others can change these*
    - *you can influence them if you need to*
- We do not undertake responsibilities which we do not 'own'.
- Our colleagues have a clearer idea of what *our* role and responsibility is.

The beauty of Keeney's concept, is that it is flexible enough to fit any size and complexity of organization.

The implication of his concept, for our planning specification, is that we need to add some 'background' information (Figure 2.2) to all objectives, making the relative role of those objectives clearer to all participants in planning.

There might not really be a simple relationship, like impacts up, is impacted from below, between a given objective and other planning elements,. This complexity [URL87] is because a single objective can be serving multiple interests simultaneously. These many interest can be very different levels of the organization, or outside of it.

But that does not stop us, in Planguage, mapping as many relationships as we want to, if they are useful to consider.

Here is an example of that information:

Contract Flexibility:
Type: Project level Critical Objective.
Owner: Project Manager.
Supports: CTO Objectives, especially Technical Adaptability.
Scale: The Speed which a **C**ontract can be **C**hanged at minimum cost of loss to reflect **C**ircumstances.
Goal: < 1 month.

Contract: All IT Services and IT Products.
Changed: Deleted or modified.
Circumstances: changed economics, or failure to live up to expectations.

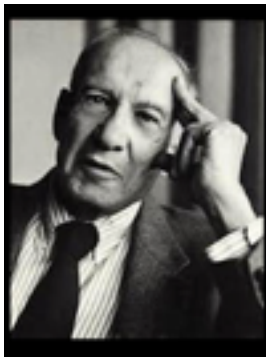
Deadline: This Year.

Supporting Strategies:
FlexiCon: www.FlexibleContracts.com.
Supporting Objectives:
Legal Dept: % of Flexible Contracts in Force.

Planguage Example 1.3 The mapping of many relationships for a single objective.

"To be sure, the fundamental task of management remains the same: to make people capable of joint performance through common goals, common values, the right structure, and the training and development they need to perform and to respond to change."

— Peter F. Drucker, *(1909-2005)* ***The Essential Drucker***



Practical Tip 1.3 Map Objective Relationships

Try to draw a graphical map, a sort of mind map, from a single critical objective, to all related stakeholders, and other related planning elements such as objectives it supports, and its means objectives or strategies.

Why?

- *to make it clear that there are many relationships*
- *and to help you see that mapping them by a series of Planguage statements might be a good way of keeping track of them.*
- *note that automated Planguage tools can be used for the most fundamental relationships [URL73]*
- *if you refer to a defined capitalized Planguage tag, then keeping track of relationships is logically simple, and it can be automated in many ways.*
- *one client, Confirmit, used 'wiki' names (one name with 2 words, the second Capitalized, as MyName, without a space) on all relationship names. This meant that access to all tags was a click on the tag itself, as if it were a URL. It was easy to check up on the meaning of a relationship tag.*
- *one method of keeping track of relationships at any 2 levels is the Impact Estimation Table (Chapter 6, URL21, URL75).*
    - *the strength is in ability to do many to many relationships*
    - *and a second strength is that the degree of relationship is quantified.*

rsmith@rsbatechnology.co.uk

Figure 1.3 An Impact Estimation Table using Richard Smith's Tool [URL75]. It shows a numeric, 'degree of impact,' relationship between any Strategic objectives (left-hand column) and their Means Objectives (designs to impact your strategic objectives, top row of 'designs'). The automated tool helps keep track of the relationships, and is good at modelling multiple strategic objectives to multiple means objectives. The IE Table, and this tool, can handle multiple related levels of objectives.

Policy 1.3: Responsibility Clarification

- **Written specification, immediately tied to objectives,**
  - **shall clarify the *level* of responsibility**
    - (for formulation, changes and result delivery),
  - **and will clarify which objective (s) it supports**
    - (explicitly defined).
- Why?
  - Nobody should be in doubt about their responsibility and its limitations
  - People should not confuse ends (priority) with means (far less priority)

1.4 Making the degree of supporting planning for your objectives visible

Principle 1.4 Loyal Subordination:

All your subordinate's objectives
must clearly support
your objectives

We typically have many, and varied, sources of support for reaching our own objectives. Direct subordinates, contractors, consultants etc. Let us call any instance that helps you to reach your own objectives, your 'support team'.

You will agree that *clarity of responsibility*, about how they support your objectives, is necessary.

This has some implications.

- If they do not know *exactly* what your objectives are, they cannot support you very effectively.

- If you change, even a few details of your objectives: they should be *informed*, so they can change their support correspondingly.

- If you choose to hide your objectives, or to formulate them unclearly: then *you* are responsible for your support team's lack of ability to serve your interests.

- If you choose to tell them ('dictate') 'what' to do (the means to your objective), rather than your smarter option, of simply telling them 'how well to do it' (in terms of your objectives); you bear responsibility for that choice, so be conscious of it. Normally, let them figure out 'how'!

However, once you have made your objective excruciatingly clear, your support team can and should be held accountable, in various ways:

- They should agree, or clearly disagree, that they will support reaching *some* of your goals, to *some* degree.

- They should be able to show a credible (numeric, experiential, guaranteed) relationship between *their* activity and plans, and their hope of helping you reach *your* strategic objectives.

- They should be able to show measurable numeric progress, at least using leading indicators, that their plans are working in practice. Early and frequently.

- They should expect credibility and rewards, based, not on what they have done – with good intent – but what they have *delivered* of *your* values.

- Outside contractors should be prepared to put their money where their mouth is, and base payment on your results, not just their effort. [URL6, No Cure, No Pay].

Product Design Productivity:
Type: CTO Level Objectives.
Scale: Average number of New Product Variations per Lab Employee, per Year.

***G1*: Goal** [in 3 years] 5. <- Estimated by Jens Jensen.

Result Responsibility [G1]: CTO.

Impacts [G1] Planned Product Volume [G2020]: by 50% <- Est. JJ.

Planguage Example 1.4. The 'Impacts' statement, for the Goal 'G1' claims that if it is fulfilled (5) in 3 years time, it will contribute about 50% of the planned improvement towards the Goal 'G2020' for the objective 'Planned Product Volume'. This is a means of expressing the assumed relationship between two levels of objectives. It is 'Impact Estimation,' without the Table format.

The Four Keys of Great Managers:

1. "When selecting someone, they select for talent ... not simply experience, intelligence or determination."

2. "When setting expectations, they define the right outcomes ... not the right steps."

3. "When motivating someone, they focus on strengths ... not on weaknesses."

4. "When developing someone, they help him find the right fit ... not simply the next rung on the ladder."

— Marcus Buckingham

Practical Tip 1.4. Clarify your knock-on effect

- *Ask yourself, and ask others: What are we expecting, of results at the level above us, if we achieve our goals?*
- *Why haven't we written that claim explicitly?*
- *Have we documented who made the claim?*
- *Have we documented who is 'result responsible' for getting to our Goal there?*
- *See Example 1.4 above.*

Policy 1.4: Relevant Support Policy

- **Any element of support for your objectives, should**
  - **directly show an estimated relationship to your specific numeric objectives.**
  - **Be prepared to adjust when your objectives are adjusted.**
  - **Be evaluated on cost-effectiveness and timeliness in helping you reach your objectives.**
- Why?
  - So we know what to expect, and who is responsible.

Section 1.5 Alignment. Every plan, including your own, has to be clear about what *other* objectives it is supporting. The 12 Tough Questions tool. The 13th Question the 'Record' statement.

Principle 1.5 Clarity of Alignment

All other people's plans,
if they claim to help you,
whatever they are called,
must clearly support achievement
of *your* numeric goals,
on time.

Here is a list of 'all other people's plans', whatever they are called, that might claim to be in your project's or organization's interests:

- plans for subcontractors and consultants, that are paid from your budget
- contracts, and agreements
- sub-projects and their plans
- strategic plans
- meetings, training
- recruitment, and downsizing in your sphere
- etc.

We should be able to ask: *'what is the expected impact on our objectives and our budgets, of any other supporting plan?'.*

Of course we should lead the way with extensive documentation in our plans, as to which other plans *we* are supporting. Use the statements 'Supports', 'Impacts', or Impact Estimation Tables.

If other plans do claim impact on our plans, then we need to:

- test their claim (12 Tough Questions, below)
- document the tough questions answers in our own planning
- document the claimed relationship; '**Impacted By:** Strategy X.'
- thank them, and tell them we are expecting them to come through for us, *'so please let us know if things change, and we can help out'*.

If the answer is *'no impact claim' (towards our own objectives) 'but we need to do it anyway'*, then identify the real reason for effort (legal, compliance, image, corporate policy) – and possibly accept it. But do not let a 'necessary evil' sway you from your critical-value delivery path.

The objective of the tough questions, is to make both parties think, about what they are expecting, and if it is realistic, or risky.

> ***My 12 Tough Questions*** *[URL15, for detail], summarized.*
> *Try them out next time you are presented with 'good ideas' or 'good plans'!*

1. NUMBERS
Why isn't the improvement quantified?

2. RISK
What's the risk or uncertainty and why?

3. DOUBT
Are you sure? If not, Why not?

4. SOURCE
Where did you get that from? How can I check it out?

5. IMPACT
How does your idea affect my goals?

6. ALL CRITICAL FACTORS
Did we forget anything critical?

7. EVIDENCE
How do you know it works that way?

8. ENOUGH
Have we got a complete solution?

9. PROFITABILITY FIRST
Are we going to do the profitable things first?

10. COMMITMENT
Who's Responsible?

11. PROOF
How can we be sure the plan is working?

12. NO CURE
Is it no cure, no pay?

Introducing the 'Record' Level: a background planning statement to make you 'get real', and perhaps to challenge your team. The 13th Tough Question.

The 'Record' statement is a note , a 'background' specification, integrated in the objective. It gives information about some 'real extreme achieved' in the area of this particular value.

Record statements should include information about when, where, who and how, and a *source* for more information. It also has to be expressed, or converted to the same 'Scale of measure' used in the objective's specification, for direct numeric comparability.

Initially this Record specification, and there can be more than one of them if useful, is designed to achieve the following:

- force us to do some research in the area of this value
- force us to recognize how good the competition is, and might get better than, by the time we reach our goals.
- help us to interpret the intent of any Goal and Tolerable level specifications better:
    - Is the goal set near the Record?
    - Is the Tolerable or OK constraint level set above the Record?
    - Why?
- Do we know the strategies, the technologies, that were used to achieve the Record? Are they available to us? Have we considered them? Can we afford them?
- What does it mean if we do *not* try to beat the Record? On the Market. For our image.
- If our targets are planning to beat the Record, what does this mean for risks of failure? For budget and deadline uncertainty?
- Does our team really include real experts in this area? **Real experts** know the record levels, and how they were achieved.

A Record Example

<u>Organizational Agility</u>:
Type: Organizational Objective.
Ambition: radical competitive improvement in our ability to change services.

Scale: Average Hours for a defined [**Employee**] to Change a defined [**Service Type**].

Past [Last Year, Employee = Branch Manager, Service Type = Open Account] **5 to 15** hours. <- HR Director.

Record [Last 5 Years, Employee = Branch Manager, Service Type = Open Account, Internet Bank Competitor] **0.3** Hours <- Banking-Records.com.

Goal [Next Year, Employee = Branch Manager, Service Type = Open Account] **0.1** hours <- our HR Director.
Justification: we need to be seen to be superior. <- CMO.

Planguage Example 1.5. By including a 'Record' specification, which is some relevant 'extreme' , we are in a better position to discuss how realistic it is to aim for a certain Goal level.

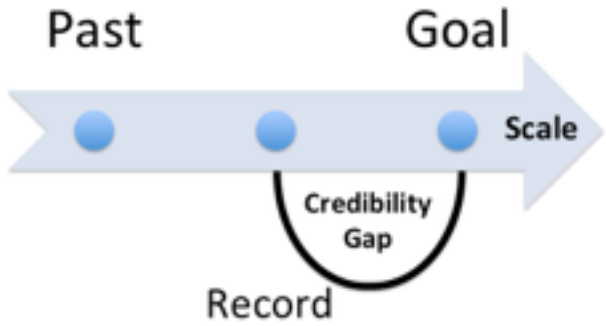


Figure 1.5. The 'Record' statement, is a benchmark, that forces us to 'get real' with our ambitions. Experts 'know' what the records are. Think sports, and you have got it. Nothing wrong trying to beat a record: but you had better be aware that it 'exists' , first. If a Goal is set to a level much better than the Record, that is 'audacious', and we need to make our ambitions credible by identifying effective world-class strategies, and resources to back up the effort.

Case 1.5: The Purpose of the Record Statement

I was in an aircraft hanger at North American **Rockwell** (now Boeing), and right beside where I was holding a training course, was a full-scale wooden mockup of the Space Shuttle (like the photo below).

I taught them that the purpose of the 'Record' statement was to advise us that we should normally **not** try to exceed it. It is then risky, costly.

The next break, a manager advised me that people were disturbed by my message. *'Tom, that's what they do here: break Records!'*.

After the break I changed my tone, and added, that the Record statement, for 'those who went where no man had gone before them', was a 'challenge', when the Goal was set higher than the Record level..

Peace secured, that day. And a lesson learned for me. Organizations are 'different'.

Photo 1.5 Space Shuttle mockup, at Seattle Space Museum

"It is a good thing to learn caution from the misfortunes of others." Publius Syrus (46 BC to 29 BC), Roman Author.

I really like this advice, and think I have practiced it my whole life, without being 'taught' the principle. No pain: Gain! Why wait for your personal pain?
and Santaya: "he who does not learn from history, is doomed to repeat it."

Practical Tip 1.5 Applying the 12 Tough Questions.

- *Keep a copy of the 12 Tough Questions in your smartphone notes, so they are ready for use.*
    - *Eventually, memorize them, for rapid application.*
    - *at the very least learn 'have you got some numbers to back that up?'*
- *Print the 12 Tough Questions on the back of your business card, or any card - put a plaque on the wall! (© T. Gilb.com, with my permission, right here)*
    - *Share this skeptical spirit with others. Use the card.*
- *Exploit the '12 Tough Questions' at your next meeting, or the next sales presentation you attend.*
- *Or even use them silently to yourself when reading proposals, plans, presentation slides.*
- Take the 'helpful' attitude, *'I suppose you know what you are talking about. It sounds interesting to us, initially. So I would like, your help, to get some more information, to help sell your suggestion to my company, and to my project'.*
- *This is a great BS filter.*
    - *I have seen directors demoted, and told-off by CEOs, in front of me, because of the Tough Questions immediate power of exposure.*
- *The 'secret' is that **you** do not have to be the expert, at what they are presenting: but if they are not expert enough (in relating to your needs), these questions will make that painfully clear to all present.*
    - *and if you want to avoid 'embarrassing people' publicly; hold your tongue in public, and send these questions later, more privately; email or private meeting.*
- *Avoid pushing it to the 'embarrassing' point. Pull back when they get the idea you are not to be messed with. They had better come better prepared next time, if they want your support.*
- *Let suppliers and corporate staff functions know you would like to do business with credible suppliers, who will stand by their claims, and make their claims clear.*

Policy 1.5: Confront Assumptions Policy.

- **We will use clear simple, confrontationally polite, questions, to find out which activities are *really* supporting our objectives seriously.**

- Why?
    - To send a message that you are serious about your objectives.
    - To motivate your support team to think better, and think more purposefully.
    - To provide a better set of facts and assumptions to support a contracting process.

Section 1.6 Measuring delivery of your objectives, using a planned 'Meter'.

Principle 1.6 Measure Reality.

All objectives,
can and must have sufficient measurement methods,
based on their defined Scale,
to give knowledge of current levels.

Principle 1.6. We need early and frequent feedback on progress towards our Goals. Measurement processes can be several different ones, for the same objective, and the same scale of measure. Sometimes quick and dirty measurement is more useful than academic levels of measurement, and 'too late' measurement data (after the horse has bolted). Sometimes the measurement process has to work in a contractual framework.

"In physical science the first essential step in the direction of learning any subject is to find principles of numerical reckoning and practicable methods for measuring some quality connected with it.

I often say that when you can measure what you are speaking about, and express it in numbers, you know something about it;
but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind; it may be the beginning of knowledge, but you have scarcely in your thoughts advanced to the state of Science, whatever the matter may be."

Lord Kelvin, 1893, Lecture to the Institution of Civil Engineers, 3 May 1883

Personal note*: this Kelvin quote changed my professional life. I hope it might have a similar influence upon some reader. This book is all about metrics, if you think about it.*

Many people mix up, or combine, the concepts of 'quantification' and 'measurement'.

They typically use the lazy excuse, that 'perfect measurement' is too difficult, in order for them to avoid doing 'quantification'. Illogical!

There is of course a clear enough distinction between a budget and accounting, between a volt and a voltmeter.

But people consistently mix up the concepts, to their disadvantage.

So did I, for a while, before a more-enlightened Swedish professor from Gothenburg pointed out the 'obvious' distinction.

I just repeat the 'mantra': '*volt, voltmeter'*, or '*speed, speedometer'*, and I've got the difference.

Notice Kelvin, in the quote above (which determined the direction of my professional work since about 1965).

In a single sentence, Lord Kelvin distinguishes between *quantification* and *measurement* twice, and three times in the quotation! This is not by accident.

'**Quantification'** *alone,* has great merit, even if you never actually carry out any measurement!

A *budgeting* process, for example, makes you think about what can and might happen: even though the later accounting results might be very different from your budgets.

Beam Me Up Scottie

If I tell you *'I want to travel to the moon and back, in 1 second'*. I have quantified.

My idea is very clear. In fact, so clear, that you dismiss it, as unrealistic.

So neither of us is concerned with how we would actually measure how fast I did my moon trip.

Meter not required.

A budgeting process gives you some constraints (budgets) that you later will probably have to respect, when real measurement (accounting reports) threatens to exceed those budgets.

The same distinction holds for forming a scientific or engineering hypothesis (quantification), and consequent experimentation to determine if it is proven or not (scientific experiment).

Quantification is, above all, a useful tool in communication between people.

Numbers *clarify*, what words *hide* and *confuse*.

We recognize that **quantification** (in practice, 'defining a *scale* of measure', and 'some interesting *points* on that scale') **alone** is useful.

We also know that it is usually also useful, sooner or later, to actually **observe reality** numerically: to measure in practice. But not 'always'.

Measurement gives essential contact with the real world.

If measurement is early and frequent, then we can usually adjust our plans, to be in better contact with reality, and in contact with our own objectives and constraints.

Measurement does not have to be 'perfect'. In fact it cannot be literally perfect, as engineers and scientists clearly acknowledge. Kelvin was not fanatic, as you can read, above.

So the 'measurement' questions and opportunities are:

- What exactly *is* sufficient measurement *quality* (accuracy, precision, credibility) for our purposes, and what is the *lowest-cost* measurement process, that has *satisfactory* quality, for our purposes.

- At different stages in the system development process, for different purposes, we can decide to have quite *different* measurement processes. Which measurement processes do we need to plan for, budget for and use? 'Horses for courses' as the British say.

- The choice of measurement process, since it depends on many scalar attributes (precision, accuracy, setup cost, cost to measure, credibility, repeatability), is really an 'engineering design' decision. Can we 'design' our measurement processes?

- That means it is a matter of finding the measurement process that best fits our objectives, within our constraints. We cannot just take *any* measurement process that comes to mind.

*Figure 1.6. The Scale is an abstract concept, to help us think about good or worse attributes of anything. The '**Meter'** is a tool or test process to help us get some idea of where on the Scale a real system is. How 'good' the system is.*

Here is a simple example showing the distinction, and the choice, of more than one measuring tool, for a single Scale of measure specification.

Team Cooperation Capability:
Type: CTO Level Organizational Improvement Objective
Ambition: much better and consistent cooperation between team members and between teams in technical projects.

Scale: average % of **P**roject **H**ours spent with **C**ooperative **C**ontent between **T**eam **C**omponents.

Meter [Early Stages of a Project] samples of logged hours, by Project Manager, monthly, 1 hour of work.

Meter [Analysis of Completed Projects] Database analysis using student trainees, presenting reports and conclusions.

Goal [within 2 years] at least 20%-40%.

Project Hours: as logged in project logs, and charged against a project.
Cooperative Content: writing or oral activity directed to others, with purpose of sharing and/or getting feedback.
Team Components: Any people within a Team communicating with each other. Any part of a team communicating outside the project team, with the purpose of learning or sharing.

Planguage Example 1.6. Two different Meter specifications for the same scale. One for early project feedback. The other for project completion testing. The Meter specification is usually short (20 words or so), at this stage of specification. Meter statements do try to contain words that are critical for accuracy and cost (for example 'Student Trainees', 'samples'). But the detailed design of the Meter process is for specialists, such as test planners.

"If your result needs a statistician
then you should design a better experiment"

Ernest Rutherford, ***Nobel Prize in 1908. 1871-1937***

Practical tip 1.6 Keep Measurement As Simple As Possible.

- *Use the cheapest simplest measuring process that will give you adequate feedback for purpose.*
- *It is more important to get some early and frequent feedback, week by week, on your critical objectives' value delivery, than it is to have accuracy greater than ±20%.*
- *This is business, not Nobel Prize Science.*
- *All choices of measurement process need to 'pay off', to give value exceeding their cost.*

Policy 1.6: Plan Measurement Formally, and integrated in planning of objectives.

- **Formal written plans, to measure value delivery in practice, will be integrated with the specification of objectives.**
- **Why?**
  - It makes us consider at which points in the development and operational processes we want to measure,
  - and makes us consider *different* levels of measurement capability, and their costs.
  - It will help avoid excessive measurement.
- **How?**
  - The 'Meter' parameter can be used for specification of different types of measuring processes.

Section 1.7 Some Objectives are 'Compound': they consist of a 'set' of scalar variables.

Principle 1.7 Compound Objectives Need Multiple Scales

Some objectives are *compound*,
they have multiple dimensions,
with different nature
so a good way to manage them
is to have multiple scales defining them.

"Love is a many-splendored thing", the old song says. But height and weight have but one single dimension. [URL10, Quality Quantification]

One problem you will have encountered in trying to clarify, or to quantify, objectives, is that there might be no *one* satisfactory dimension of measurement.

You cannot conclude, from your failure to find one single scale of measure, that there are *no* scales of measure.

The correct answer is very likely to be that there is a satisfactory 'set' of Scales, which *together* best describe the objective in practice. This is very common, and normal.

When there are in fact several potential definitions of an objective that you can think of. Which one is the right one? It is tempting to ask.

All of them and more might be the right answer!

An old electrical handbook recommended dividing up concepts 'until quantification became obvious'. (EDIT note maybe Ireson, Reliability (not hytta at home) Handbook , check it, or Juran (cannot find it there 18.9)

Rene Descartes (1596-1650) recommended the same approach ('Discourses on the Methods'). Quote 1.7 below.

Case 1.7 A IBM Usability

Once the CEO at IBM decided that Usability was the wave of the future, for the new Personal Computers, and Tom was asked to help out by IBM. Tom suggested *quantification* of Usability, but it took months before we realized that this was **many dimensions**, not one. The many dimensional model was adopted by IBM. [Ref URL10D, CE Book Ch. 5 for detailed examples]

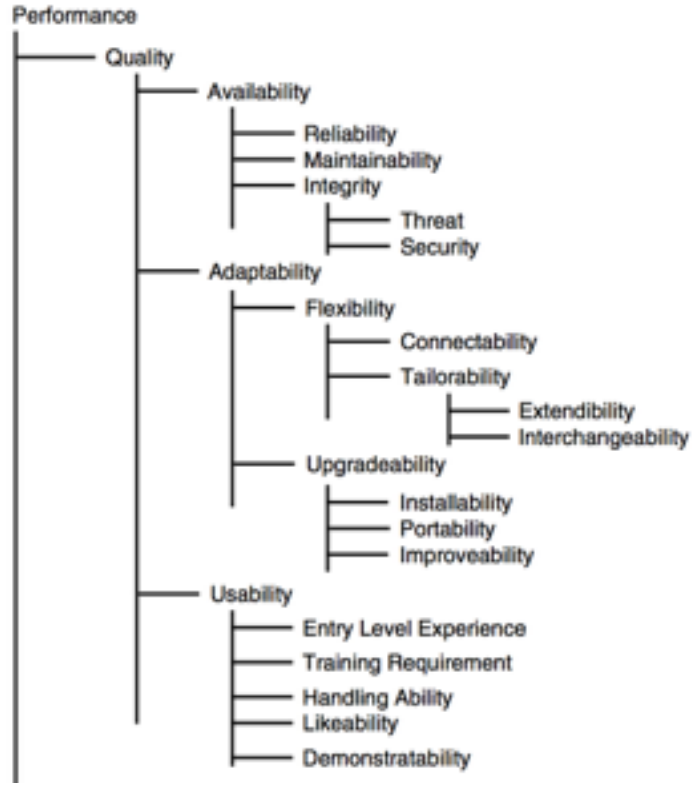


Figure 1.7 A . Some compound quality attributes. Including the one we worked out at IBM in the case above. (Source [1] CE Book, Figure 5.3, this free download [1B] contains details of Scales and of more usability sub-attributes)

Case 1.7 B. Large Project Failure Caused by Poor Top Management Objectives.

A client of ours, asked us to analyze a large failed project (8 years old, $160 million wasted thus far, 90 project team members currently).

I was consulting to the 5th project manager. He did not want to fail.

The CEO had initiated the project, to radically improve the 'robustness' of a major product.

The product was failing too often, and for too long a time for major customers. A failure with a customer could end up costing them roughly $1 mill, more or less. Big.

Their original CEO-approved requirement, which existed without being taken seriously, for 8 years, was:

"Rock Solid Robustness" (official specification headline)

There was some further specification about not breaking down too often (2 weeks), and being fixed quickly (10 minutes). Combined with a long list of strategies for achieving this.

Tom's suggestion looked more like this:

Rock Solid Robustness:

Type: Complex Product Quality Requirement.

Includes:
{Software Downtime,
Restore Speed,
Testability,
Fault Prevention Capability,
Fault Isolation Capability,
Fault Analysis Capability,
Hardware Debugging Capability}.

Planguage Example 1.7 Decomposition of a Compound Objective into Elementary objectives. (see Case 1.7B) above.

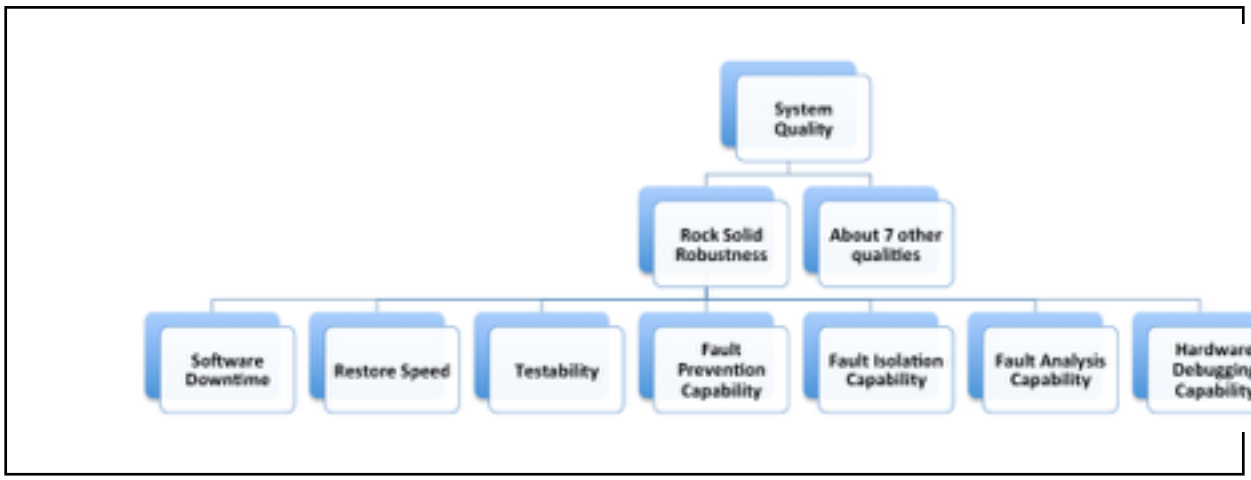


Figure 1.7 B. Decomposition of the Compound Objective "Rock Solid Robustness", into 7 elementary sub-objectives.

Case 1.7 B Continued.

The first 3 sub-attributes Tom then quantified in less than an hour. Over a beer, as it turned out.

They should have all been quantified, and used to drive the project, 8 years earlier. On the first day of the project [URL8A].

Gradual value improvements should have been delivered towards the (not) quantified Goals in the first months of the project.

They ultimately took this advice and succeeded. But it took 2 more years to turn the ship around.

For more detail on this case, including the 3 quantifications over a beer see
[URL7, Robustness decomposition slides], End Case 1.7B

To accept nothing as true which is not clearly recognized to be so: that is to say, carefully to avoid precipitation and prejudice in judgments, and to accept in them nothing more than what was presented to my mind so clearly and distinctly that I could have not have no occasion to doubt it.

To divide up each of the difficulties which I examined into as many parts as possible, and as seemed requisite in order that it might be resolved in the best manner possible.

To carry on my reflections in due order, commencing with objects that were the most simple and easy to understand, in order to rise little by little, or by degrees, to knowledge of the most complex, assuming an order, even if a fictitious one, among those which do not follow a natural sequence relatively to one another.

In all cases to make enumerations so completely and reviews so general that I should be certain of having omitted nothing.

Rene Descartes. Quotation 1.7.

Practical Tip 1.7

- *Assume that each of your top level objectives are compound (a set of Scales define the objective), not elementary (1 Scale only) until proven otherwise.*
- *Make a list of possible sub-attributes. Brainstorm a list with colleagues.*
- *This is often more realistic and useful, though it is of course initially more planning work.*
    - *(You can simplify all planning, by doing none!)*

Policy 1.7: Decompose Compound Objectives

- **Critical top-level objectives shall be**
    - **decomposed into their elementary quantified components,**
    - **when this will give better management of the top-level objective.**

- **Why ?**
    - Because this gives more realistic understanding and consequent treatment of essential aspects of the problem.
    - It forces people to think more deeply
    - It eases the path to quantified manageable objectives
        - If we try to oversimplify, and force the objective into a single scale of measure, then the 'reality' of those other dimensions of the objective will likely cause us to find the management of the project to be '**complicated'**. Things will pop up, related to those other dimensions, you excluded, and you will wonder what is going on.

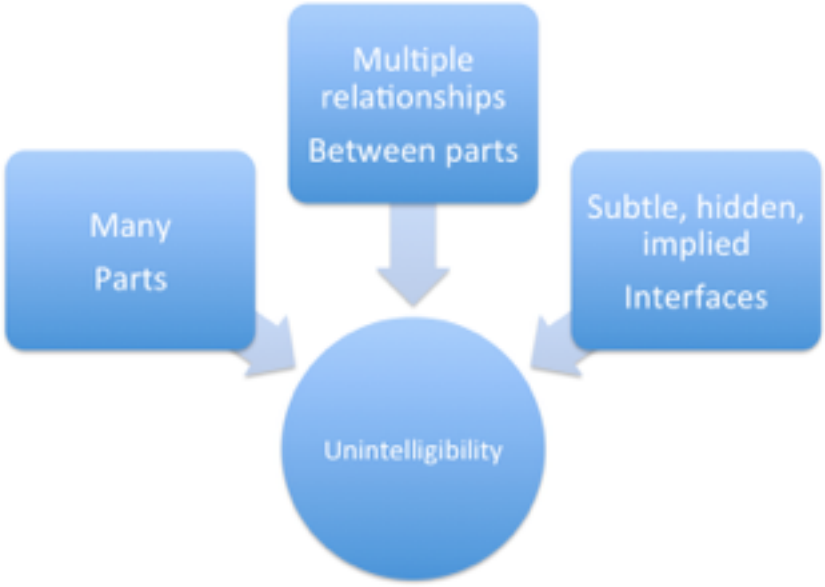


Figure 1.7 C. Systems can seem complicated, unintelligible past, present and future, for a variety of reasons. Compound parts, unrecognized and unspecified, is one possible reason, amongst several others.

Section 1.8 Object-Oriented Planning: Extracting Presentations from the Plan database

Principle 1.8 Written, and approved, does not mean 'static'.

Reflect Reality Rapidly:

**Changing the specification of objectives,
and other plan objects,
is a natural and necessary response to
insights, feedback, competition, and politics.**

Just because an objective is *written*, or it is *quantified*, does not mean it is 'chiseled in stone'.

In fact, one reason for writing things down, is to *clearly* see any changes later.

A reason for *quantification* is to more-clearly realize that a numeric *change has been made*, however small.

Our policy must be that *changes will be clearly communicated*, to all parties who are potentially impacted, or interested. The consequences of changes (like cost or time increases) should be intelligible to all concerned parties.

Even the smallest changes can have *large* consequences.

It is therefore important to be able to *sense* changes, and take appropriate action quickly.

In one published case study (AT&T, 5ESS system, Communications of ACM) the primary factor was a change in telephone switching system availability from 99.90% to 99.98%.

Only 00.08% change in one factor.

But the project cost was 8 years time, with between 2 to 3,000 people (personal communication to author) were involved.

So, imagine the consequences if you are not numeric ("highest availability") or do not have the 4th digit? (99.9%)

Mini Case 1.8 AT&T Sensitivity for a single digit.

We believe in a number of tactics when planning, to make *discrete change* clearer.

We believe that changes should not primarily be reflected by issuing a new document, or a new plan, or a new set of slides.

Small but significant changes can easily get lost, or be overlooked. This would just create a bureaucracy without concerned and responsible owners of the ideas.

We believe that changes should be tracked, documented and managed at the 'statement' level. (example, Goal, Scale, Expert).

We would specify, interested parties, in a given 'plan object' using some of the ideas below:

Market Adaptability:
Type: Marketing Director Critical Objective.
Stakeholders: Marketing Director, CTO, Product Director, Sales Director.
Owner: Chief Marketing Planner.
Expert: Supply Chain Manager.
Version: 17 July 20xx, 12:31.
Quality Control: last approved 10 June 20xx.
Scale: …..
Goal: …..

Planguage Example 1.8. Some of the Planguage specifications we can make, to enable us to change the specifications rapidly and safely later. For example, knowing who the players are, at the level of each objective, helps us see who to consult with, about changes.

The Stakeholder list makes us aware of the main players concerned with any changes to a given plan object. They can be informed (even automatically), and they can review and approve, if appropriate.

The 'Owner', in the above example, is the *specification* owner for this objective, for a given plan object. Nobody else can make an official change, and the Owner is responsible for doing it responsibly; for example by informing stakeholders, and doing required quality control.

The Owner might also consult with the domain Expert, before publishing a change. The owner would be responsible for adhering to any rules and processes that the organization determines are legally, economically, and culturally necessary regarding a given change.

Version control (example above), for example using a date, time stamp or version number helps sensitize readers to changes in the specification. And a QC date and status reminds us that QC is, or is not, done.

There is a concept emerging here. **The Planning Database.**

I do not see a plan as a 'document', or even a set of documents.

I see a plan as a set of specifications that relate to one another: 'strategies to meet a project's objectives', for example.

I see each individual 'objective' or 'strategy' as a planning 'object'.

That 'object' alone is a sort of 'mini database' and it can contain, an accumulation, for the *life* of the system, of all manner of *potentially* interesting data elements, such as Parameter Specifications (like Scale, or Expert) about a given objective, strategy, or other object type of interest, to planning. No limits.

Then we can, at any time, for any purpose, select a 'Plan Extract'. And we can edit it in any way we like: graphics, coupling to related detail in other plan objects (the potential strategies for 2 objectives, for example).

I suggest we call this 'Object Oriented Planning'.

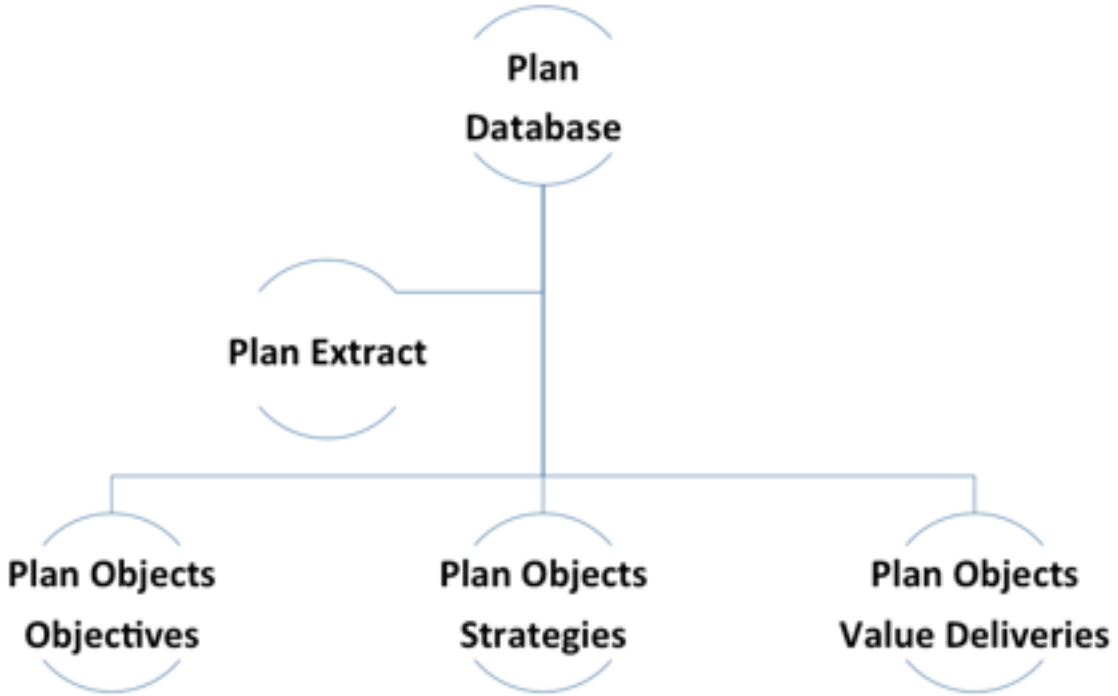


Figure 1.8 A. The concept of a Plan database, composed of plan objects, of any useful detail, and constantly improving detail. This includes feedback, as delivery evolves, on how things went with the plan object (the strategy or the objective, or the Value delivery).

Extracting the detail you need, when you need it.
Not all of this plan object detail has to surface every time you quote or use an objective's specification. But all detail belongs in the planning database, if your plans are to be taken seriously; and if you need to reflect change, quickly.

If all detail is not systematically available, this could cause confusion, in the absence of easy access to all the human 'players' for that plan object.

We should be able to focus on *just the right amount* of planning detail for a given purpose, and then to be easily able drill down, or seek sideways, for potentially interesting relationships, as need dictates.

On the complexity of a complex dynamically changing plan.
No individual can understand the consequences of the totality of a plan.

The most we can expect is for domain experts to keep track of *their* plan object, and its relations to other plan objects. We need fairly automated tools to help us

- sense relationships that need to be explored
- sense result consequences, impact on objectives, costs and risk levels.

We need tools like Impact Estimation tables, coupled with automated tools to keep reasonable track of the consequences of the many inevitable changes as a plan cumulates in detail, and as the plan is partially implemented. Humans reading plans cannot do that job effectively [URL87].

The Master Plan: a single instantaneous version and source.
There is one subtle, implied, rule here. There is only one single valid, current 'master' version of anything in the database.

Multiple concurrent *copies* of the 'same thing' are not allowed. This eliminates a potential consistency problem, because an object cannot be inconsistent with itself.

For each 'Master Plan Object Specification, there should be a single Specification '**Owner'**, the only instance empowered to update the Master.

Another benefit, of a Master Plan Database, is that, even when you simplify presentation, by for example showing an *'Ambition: To get the highest productivity in our market'* statement, there is no doubt that the unambiguous detail is tightly coupled to that Ambition statements, in the Scale and Goal statements. And you can easily access that detail.

The nearest thing to a plan *copy*, would be a collection of 'Older versions' of any changed element of the plan. Not older versions of the plan; older versions of any *detail* that has changed. We should be able to automatically keep a good history of change: who did it, what was it, and perhaps why?. Perhaps even 'un-planning', undoing a bad plan.

Automated tools for doing this pretty well with Planguage exist, such as Kai's Tool [URL68]. Which we have used for several years. And Smith's Tool [URL73].

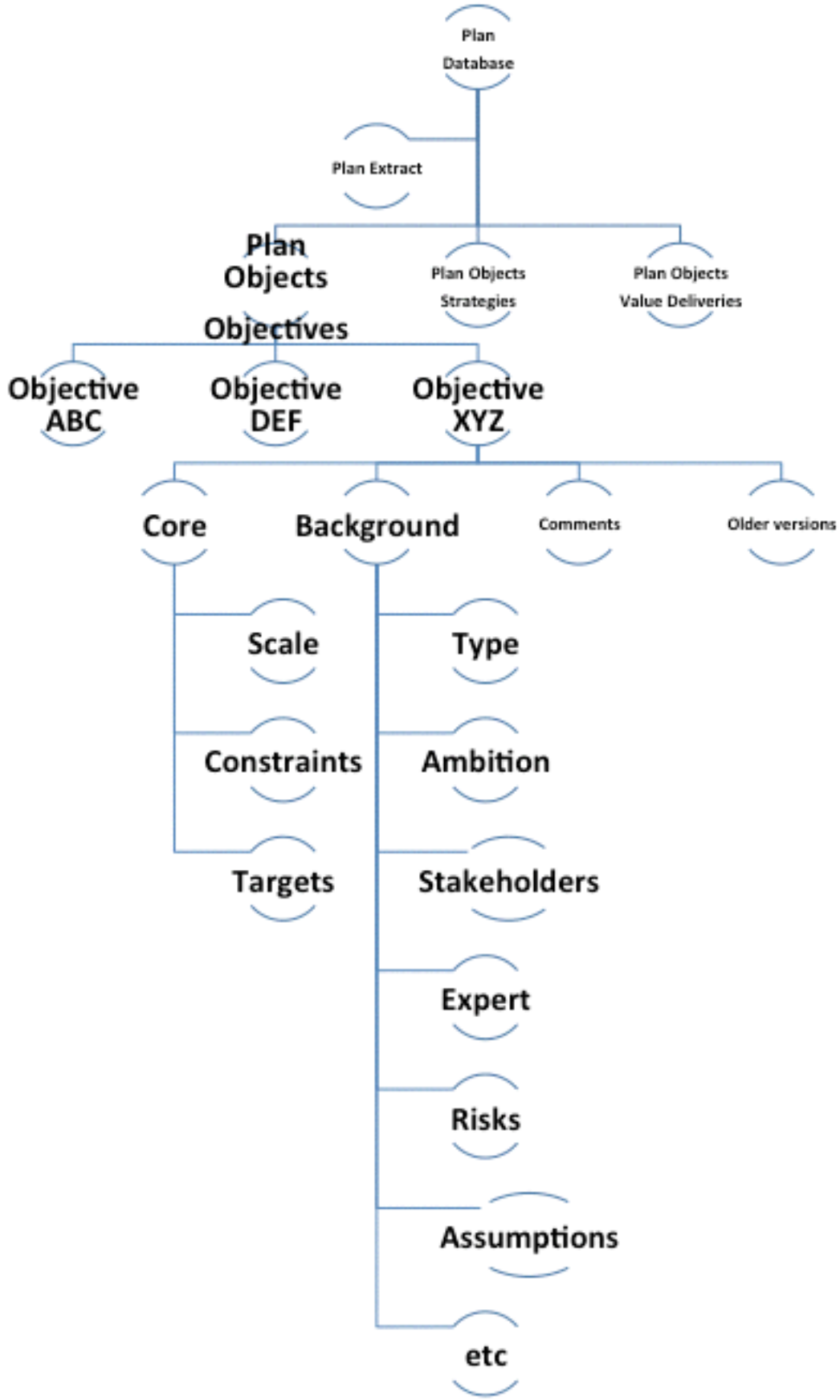


Figure 1.8 B. The plan element hierarchy. Each plan object is composed of a set of Planguage statements.

"Very few managers know how to effectively tap the biggest source of performance improvement available to them: namely, the creativity and knowledge of the people who work for them."

— Alan G. Robinson, The Idea-Driven Organization: Unlocking the Power in Bottom-Up Ideas, http://alanrobinson.com

Practical tip 1.8 Appointments Energize

- *When you take the trouble to 'appoint' stakeholders representatives, experts, and owners for a plan element, with their permission, you are also accessing their ideas, their experience, their future knowledge, and their sense of ownership and responsibility.*
- *Appoint people who 'care'. And they will.*
- *Their name is on it, they are visible, they are recognized.*

Policy 1.8. Change plans quickly, but responsibly

- **Serious planning objectives**
  - **will include information**
  - **allowing us to change details rapidly,**
  - **but safely;**
  - **so that *all affected parties***
  - **are made aware of the changes.**
- **Why?**
  - Anything less that this and you will continue with obsolete plans, and out-of-sync decision-making; not highly competitive.

Section 1.9 Multidimensional Targets: 'decomposition' and 'prioritization' tools using 'qualifiers'. Down-sizing plans for experiments and short high-value delivery cycles.

Principle 1.9 Represent Rich Reality.

Any number of
objective-levels
can be specified
in any number of
useful dimensions
of time, space, and events.

It can be dangerous to have a single number to represent your objective.

If you do a single number, then it is, unfortunately, logically necessary, to specify the biggest target level, covering all your needs, forever under all circumstances. And you have abdicated the useful 'worst case' constraint specification possibility.

That is a bad idea! That extreme target level will be far more than necessary for most purposes, and will probably be delivered far too late for most needs, at perhaps ten times less cost-effectiveness.

*We practice **differentiation** (like 'market segmentation') of targets (what we are aiming to achieve), and also differentiation of our 'scalar constraints' (worst acceptable objective levels).*

The simple reason for this differentiation is that we can plan more competitively by clearly separating high-value short-term situations from the 'other 95%', and delivering that priority value, quickly.

For example, instead of just specifying

Goal: 20%.

we could set different goals for specific segments of time, environment (who, where), and events.

[Qualifiers] the adjectives of the planning language.

*We call these 'differentiation' planning-specification concepts **'qualifiers'** of any type (Past, OK, Goal etc.) of level on a Scale.*

*I like to write that term as **[Qualifiers]** to give a strong reminder, with the [square brackets] of how we use them.*

We start using them as Scale [qualifiers] in a 'Scale' statement.

Scale [Average time for [**People**] to complete a [**Task**] Successfully].

A basic Scale specification with a qualifier using 2 variable Scale parameters, [People] and [Task].

We continue using them in [qualifier statements], just below the Scale.

Scale [Average time for [**People**] to complete a [**Task**] Successfully].

Past [Last Year, **People** = Full Time Employees, **Task** = Days Work.] 7 hours ± 2.

A systems analysis 'Past' statement. No 'future' levels-to-reach, are specified, yet.

By implication, Planning Language grammar, specifications, like 'Past' in the example above, are 'attached' to the Scale above, together with any useful types of Scale level specification [Targets, Constraints, and Benchmarks}.

We use exactly the same-spelling qualifier words in the Scale, and in specs below. This is to make the connection clear. They are all 'Capitalized' to indicate they are formally-defined concepts, for which the Qualifier term ('People') is just a 'Tag'.

We can also use the [qualifier] in any other planning specification, where it might be useful.

For example:

Name Tag [Children].

Supports: Efficiency [Financial Processes].

Includes: Productivity [Plant, Office, Salesrooms].

[Qualifiers] can be used as an 'adjective, in any specification they might be useful for.

Here is an example of a normal use of the [qualifier] in a target statement:

Goal [Deadline = 1st Release, Market = China, Consumer = Golfer, Assumption = Tax Free Import] 20%.

Which can also be written, for better legibility, at the cost of more lines in the statement:

Goal: 20%.
Deadline = 1st Release.
Market = China.
Consumer = Golfer
Assumption = Tax Free Import

The 4 qualifier parameters are edited as a vertical list, for legibility.

The statement below:

> [Deadline = 1st Release, Market = China, Consumer = Golfer, Assumption = Tax Free Import]

is a 'qualifier'.
It contains three *types* of qualifier. **When, What, and 'If'**.

There can be any useful number of qualifiers. In particular the 'What' dimension often has 3 to 6 more-detailed dimensions in practice.

The qualifier parameters do not have to be pre-defined in the Scale statement itself. But if they are defined there, without a default, then we need to define them in every scale-level statement (Past, OK, Goal etc.) otherwise we would have an 'undefined' situation. Maybe we could assume that ' = All' is intended. But it is better to be explicit, and leave no doubt to the reader of your plan. This is a pervasive Planning Language rule: be explicit. Leave no doubts.

The 'If dimension' is less frequently used.

But it is almost illogical NOT to use the 'When' dimension: since that would allow 'fulfilling the objective in infinite time'. I.e. never while you live. So we normally just add a time or deadline dimension as the first qualifier parameter. '[Deadline = 1st Release..]', in the example above.

Deadlines Determine Strategy Type.
Time constraints have a powerful influence on our choice of 'means', our 'strategies', for satisfying our objectives.

Immediate delivery can require very different, and very much more costly strategies, than if we can put off delivery for a few years.

Or, on the other hand, being able to put off the delivery for a long time might allow much cheaper new emerging technology to be used.

Figure 1.9 A. The 3 basic statement qualifiers. Time (When), events (If), and space (where). These are the basis for helping us to avoid 'over general' plans which apply to everything in the universe, since no exception has been made. Competitive 'Value Planning' needs to differentiate to compete.

The Qualifier statement is a set of 'conditions', and the planned level statement, or benchmark statement, is only valid, when all qualifier conditions ('if', 'when' and 'where') are true, or valid.

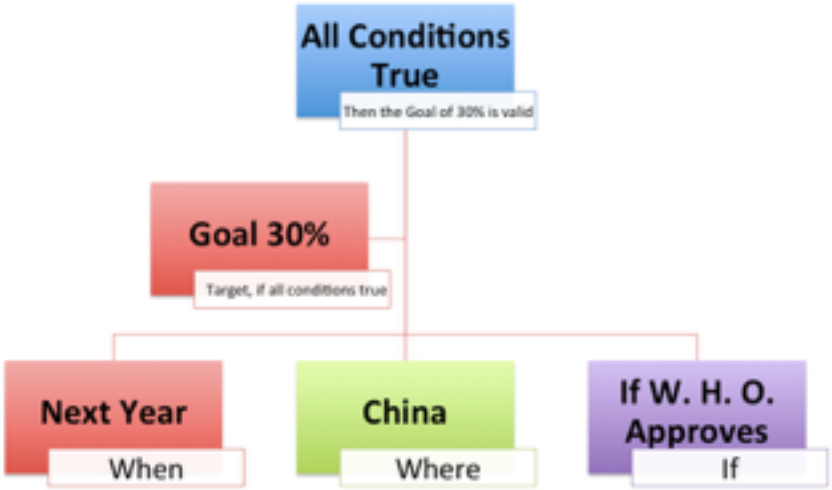


Figure 1.9 B. A planned Goal which is totally contingent on all 3 conditions being 'true' (valid, applicable) at once.

Conditional Plans.
One result of 'all qualifier conditions must be valid', is that we can plan for events that *might never happen*, but when they do happen, a valid plan, or obligation to plan at that moment, kicks in automatically.

Let us call that 'conditional planning'. It is a useful tool for 'risk' planning. Planning for risks of conditions where they are rare but unknown frequency. At least there is a plan in place, and it is cheaper than to try to find out if the event with happen.

Of course you can, at any time in the planning process, have as many different Goal statements, or any other Scale level statements, with as many different qualifiers, as you need. You can plan for any set of long-range, medium-range and short-range levels for the objective, as you need, and *when* you feel the need.

Figure 1.9 C. You can use any useful quantity of qualifier conditions to express a target segment for your plan. This allows you to be more competitive, by focusing your plans on the most profitable, or critical, segments to deliver value to.

Emergent Planning
Your planning detail can *emerge* in parallel with your real value-delivery process of your project. Detailed statements can emerge, as you get feedback from real-life delivery; learning about new markets and stakeholders, that are worth catering for.

We usually as discussed above, *predetermine* some of the qualifier parameters, but not necessarily all of them, in the 'Scale' definition. The predetermined Scale parameters are conditions we will *normally* want to specify in the scalar level statements below.

So we could for example have used the Scale specification, with 'Scale Parameters': like

Scale: average annual % of our Corporate Unit Sales for a defined **[Market],** and a defined **[Consumer].**

We can also embed a default option in the Scale itself.

Scale: average annual % of our Corporate Unit Sales for a defined **[Market: default = All],** and a defined **[Consumer: default = Internet Buyer].**

Default Scale Parameters mean that we do not have to define these in planning level statements later, as they are the 'default type'. This default specification is automatically re-used, or 'assumed'.

This Scale statement suggests that we can define any useful combination of 'Market' and 'Consumer' in the target (example 'Goal') and constraint (example 'Tolerable') specifications; as well as in Benchmark (example 'Past') specifications for comparison.

An optional, 'background' 'Benchmark' statement (Past, Record, Trend, Ideal) enables us to see how much improvement we are planning. Goal minus Past = Improvement.

Figure 1.9 D. One qualifier, like 'People', can have a well-defined, but expandable, set of qualifier instances (like 'Married'). Each of these can have a formal written definition, when that is useful. Sometimes the term, like 'Pensioners' is obvious enough for the planning purpose. But you might like to define 'VIPs'

Planning Thoroughness

Once a qualifier condition category, like [People] is established, it becomes logically tempting to define, as far as possible initially, the potentially interesting contents of that category (Children, … VIPs). This can be extended and changed any time, as needs and knowledge emerge.

Each of these categories *can* be well-defined:

Children: defined as: any resident or citizen from Conception until end of 12th year of age.

Formal definition of a concept tagged 'Children'. This can be local to a planning element, such as a single objective, or Global to a larger plan set, like for a project.

This *category specification* also leads to recognizing, that you might have failed to have any plan whatsoever, for *certain* categories. A 'completeness test' is possible. Oops, we forgot 'Widows'.

Or you may see that you have failed to plan for certain categories, in *combination with* certain other Qualifiers, like 'Country, Area, Experience'.

These [Qualifier Parameters] are one set of tools for developing your understanding of your **Stakeholder** set.

We can also express a group, or set, of these 'at once':

Goal [People = {Teenager, Married, VIPs}] 50%.

Goal [People = {Teenager & Married & VIPs}] 50%.

The {…} is a 'set' parenthesis symbol ('keyed icon') for a list, group, or set of things. The '{set parenthesis}'.

The comma separation (Teenager**,** Married**,** VIPs) means that the target applies to *any one* of the types of people ('or').

The '&' means that the target applies only when people have all 3 properties at the same time ('and').

Qualifiers is a decomposition Tool.
This whole method of using qualifiers in a plan is, in other words, a method for '**decomposing**' the plan into *smaller parts*.

Prioritizing Qualifier Combinations Early
One advantage of decomposing a plan, is that we can select the qualifier sets with the highest value, and prioritize delivering their performance levels early. This is competitive value-planning in practice.

We not only can sequence the more-valuable sets of stakeholders early; but because we have *decomposed* things into much smaller (example 1/50 of all possibilities) units, the cost and time to do something early, is much less.

Decomposition to Experimental Size
Another attribute of this decomposition, is that the smaller units of value delivery, are better suited for 'experiments' and 'pilots' in value delivery.

If we fail totally in the 'pilot', we have the option of not repeating the failed strategies for other qualifier conditions (the 49/50 rest of the combinations).

And whether we get success, problems, or total failure, for a value delivery experiment, we will have learned from valuable experiences, which can be applied as we move forward, to attack other qualifier combinations.

In fact we might have even learned *which* qualifier combinations are smart or dumb to do, next.

"There is no such thing as a failed experiment, only experiments with unexpected outcomes"

Richard Buckminster Fuller (US engineer and architect, 1895-1983)

TIME Magazine Cover: ***R. Buckminster Fuller*** *–*
Jan. 10, 1964. Cover Credit: Boris Artzybasheff

Practical Tip 1.9 Smart [qualifiers]

- *Select a set of qualifiers which gives low risk for failure, and high value of early delivery.*
- *implement it early.*

Policy 1.9. The *Smart Differentiation* Policy.

- **Specify objectives,**
  - **by detailing clear ideas of 'when', 'who', 'what' and 'If',**
  - **so as to maximize our short-term value delivered**
  - **and longer-range competitiveness.**

- Why?
  - To avoid delays to urgent selected stakeholders
    - Delays caused by working on less valuable stakeholders, before delivery to the more-valuable stakeholders.
  - To slice up into do-able short-term action
    - So that low risk experiments are possible.
  - To force ourselves to think more clearly, and in more detail about our important stakeholders, and what they really need, and how *fast* it is worth 'getting the value delivered' to them.

Section 1.10 Benchmarks: warning signals for planners.

Principle 1.10 Butterfly.

**The slightest numeric or other change in an objective,
can trigger surprisingly large consequences,
in necessary strategies and their costs.
So, be careful what you ask for;
you might not need it,
or be able to afford it.**

If the state of the art for up-times of a (software, telephone) system is 99.998 %, what will it cost you to demand *the best*, a competitive *edge*, say 99.999% ? Just 00.001% better ?

First, nobody knows! There is only one way to find out. Do it.

As all real engineers know, full time 100%, perfection, is not possible, for a known cost. The cost is infinity. Engineers never *seriously* plan for perfection.

"To know what you know and what you do not know, that is true knowledge."

Confucius

Precisely in highly-competitive situations, you are *pushing the border*, the *record*. And *nobody* knows, until it is done.

As we pointed out in the above (AT&T case, 1.8), the answer can be shocking. 24,000 work years of effort to move the level from 99.90% to 99.98% availability. And this can only be satisfactory, for extremely deep pockets. Management cannot simply, seriously demand '24/7'.

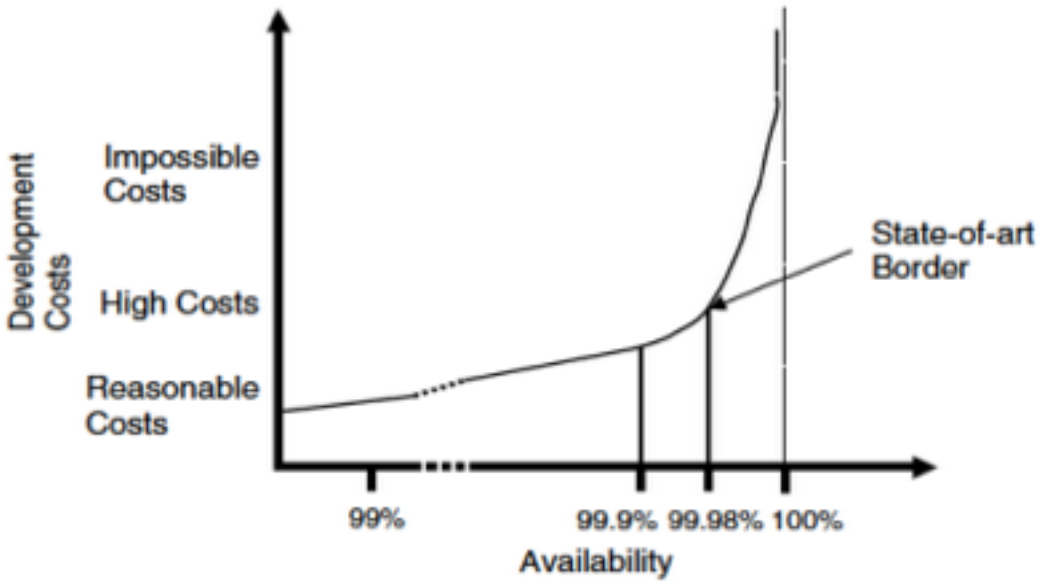


Figure 1.10 A. The AT&T Example. As we approach perfection (100% availability in this case) the costs tend towards infinity.

Perfection Costs More Than your Budget
The cost-estimation problem [URL11] is complicated since there is little or no experience at the highest levels, especially beyond the 'Record' (State of the Art), to guide us.

Notice, in addition, that a very small change in the Goal, can cause a disproportionate change to the cost, which will probably not 'pay off'.

So, in our Planning Language, we have ways of giving ourselves *warnings*, and of understanding *why* we have chosen *particular* levels of an objective, as our Goals or Tolerable levels. We call these warning devices 'Benchmarks'.

Planguage Example 1.10.

<u>Customer Service Availability</u>:
Scale: % of 24/7 a customer gets a qualified answer without waiting or failing with the qualified answer.

Past [Last Year, Our Main Service System] 95% <- Service Report.

Record [Last Year, Our best competitor] 98% <- Their PR.
Record [Worldwide, Last 10 years, Similar Customer Service. Systems to Ours] 99.98% <- Industry Surveys.

Trend [by Next Year, Based on Last 5 years, Our Main Service System] 93% ?
Trend [Next Year, Our best competitor] 99% ??

Example of 3 types of Benchmarks.

Based on these benchmarks, above – what is a reasonable plan? How about this? Below.

Tolerable [by Next Year, Our Main Service System] 99% ? <- Mkt Dir.

Goal [by Next Year, Our Main Service System] 99.5% <- CTO.

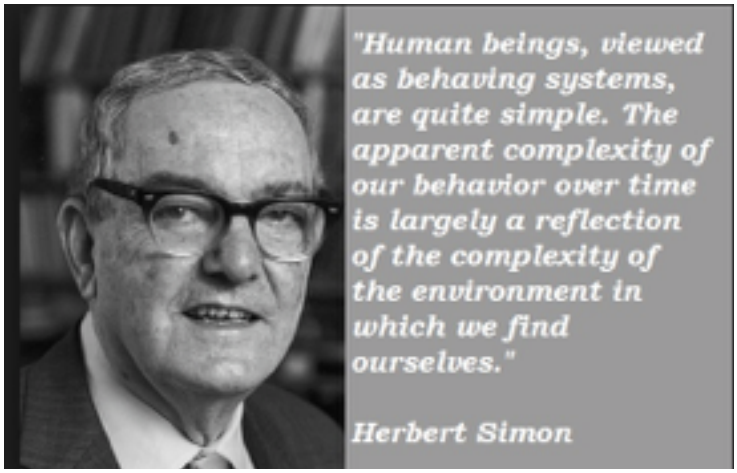
Examples of a constraint and a target, in response to the benchmarks above.

Do the benchmarks (Past, Record, Trend) explain our choice (99%, 99.5%), and drive us to plan for the levels of the objectives we have suggested?

How does **Goal : 96%** look to you now?
Assuming you want to be a winning competitor.

> ***"One finds limits by pushing them."***

Herbert Simon (1916-2001) Nobel Memorial Prize in Economic Sciences



Practical Tip 1.10

- *Research the relevant benchmarks, like Record, Past and Trend, before having a serious discussion about your targets and constraints.*

Policy 1.10 Get Realistic

- **Base your plans on realistic information about state of the art, and the state of your competition.**
- **Specify that information so that it is *integrated* into your objectives, and preferably updated (e.g. use Past, trend, Record)**

Why?

- So you can derive realistic and competitive plans
- So you can explain and justify your objectives to buy-in and approval instances.
- So you can prevent unanchored (to 'reality') managers from demanding more than you really need to do, or afford to do, or is possible to do.

Chapter 2. Strategies.

'Strategy' is the 'management' word I use for the *concept* of the 'means' to our 'ends' (i.e. here, 'ends' = 'objectives').

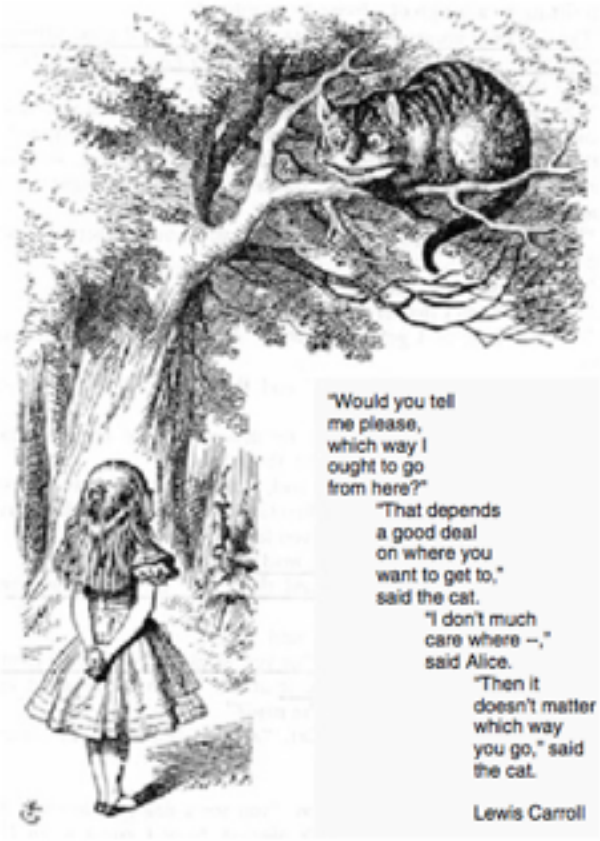


Figure 2.10
Alice and the Cheshire Cat. Illustration by John Tenniel, wood-engraving by Thomas Dalziel. From Chapter 6, Alice in Wonderland by Lewis Carroll.

Figure 2.0 Alice and the Cat. If your objectives are unclear, then any random strategy will be enough to keep you 'Busy doing nothing, working the whole day through. Trying not to find lots of things not to do' (nice song). [1, Figure 2.10]

I consistently find that planners have unclear understanding of the ‘Strategy' concept

I also find that planners have almost no ability to manage (estimate, measure, design towards) the multiple attributes of performance and cost that strategies contain.

Section 2.1 How to know if a strategy will really work

The Satisfaction Principle, 2.1.

**Strategies are
only as good as
their ability to satisfy objectives.**

The Hollywood equivalent is 'you are only as good as your last film'

A strategy, whatever its initial intent or hope, that does not, in fact, help us towards *any* of our objectives, is of no use to us whatsoever.

Strategies should *not* be done for their own sake, or because they are conventional, popular or 'requested by a powerful instance' (I'm referring to YOU, boss).

The proposed-strategy *purpose*, in terms of meeting our objectives, within our constraints, must be known, and published. What's it good for?

Figure 2.1 A. Clear objectives are an absolute logical necessity for any meaningful process of finding, and evaluating, strategies. Unfortunately most planners skip step 1, the 'clearly' part.

Strategies should be a direct function, of the need to satisfy the critical, required (constraint and target), *performance levels*, for the various types of *concurrent objectives*, and the many *simultaneous constraints*.

Some people find 'strategic planning', *complicated* to understand: it is.

But the initial problem they are dealing with, is *not* the 'complicatedness of *reality'*. Which is bad enough.

They are unwittingly dealing with the *chaotic* ***planning*** *environment*, which they themselves have *chosen* to use.

The chaotic planning environment is characterized by:

- Lack of quantified performance objectives
- Lack of full and clear definition of constraints
- Lack of initial estimation of the impacts on all of these, of any candidate strategy
- Lack of detailed-enough definition of a strategy, to actually understand its performance impacts and costs.
- Lack of detailed and unambiguous-enough *definition* of strategies; to ensure that strategy implementers will actually (and legally) implement the strategy that was in the minds of the planners.
- Lack of quality-control inspections and reviews, to make sure the planning meets the (usually missing) standards or rules for good planning
- Lack of small-scale pilot experiments with strategies, before scaling up, and before irrevocably committing to their use.

This book is completely devoted to, specific, practical, methods for *removing* the chaotic planning environment.

Adoption of our planning ideas, will never stop 'reality' from being chaotic, but at least we have eliminated one *avoidable reason* for falsely claiming that systems are 'complicated'.

We need to make sure it is the *real systems,* we are complaining about, not *our own poor planning practices*. The bad practices that are pervasive everywhere today: no thanks to Business Schools at all.

It is illogical to discuss any strategies, without ensuring a clear logical connection, to the objectives they will support, and also to the constraints they must respect.

If the strategy, for any reason, fails to deliver, in theory and later in practice, progress towards our goals, in relation to the assumed and acceptable costs of doing it; then it must be quickly dropped, in favour of better strategies.

It is dangerous for managers, themselves, to *suggest* a strategy (or *worse*, insist on it), however well-intended, to support a project. The manager cannot know for sure it will succeed. And the manager cannot guarantee that it will.

The *manager* should focus on:

- Making sure that there are *very clear critical objectives* at the base of the strategy planning process
- And then, see to it that there are a reliable, and through, set of methods, for finally deciding, which strategies will survive the tests of a complicated market, or complicated working environment

The exception, to 'demanding' a strategy, might be that the manager insists that their *suggested* strategy be measured, and dumped if it is weaker than others; or if its efficiency is poor, compared to estimates. But that is the same as the 'proper processes' mentioned above, and suggested in detail in this book.

All strategies, however successful elsewhere, may fail in whole or part, in your particular case. This is a well-understood engineering paradigm. It is even a well understood financial investment paradigm.

The situation is complicated.

You really don't know what the results will be until you can measure them in practice. Even then, the next small strategy increment, or system environment change, can destroy any good result you might temporarily have gained. You sure as heck cannot *simply accept* a strategy idea from anybody, however credible they may seem.

The strategy must be proven in practice, in the long term. Anybody who does not understand that, is not a trustworthy advisor.

All parties need to be aware of this, and ready to abandon failed strategies as soon as possible.

Figure 2.1 B. The process of identifying and validating that strategies really will, and actually do 'deliver the goods'.

The strategy process must logically start with clear objectives 'to be met'.

All candidate strategies must face 'systematic quantified analysis':

1. Do we believe they are powerful enough to reach our objectives, and within resource budgets
2. Can we estimate their power and costs
3. Can we measure, in small-scale value-delivery experiments, that they deliver what we estimated?

Figure 2.1 C. ***The Evil Circle of Complicated Systems****.*

"What could possibly go wrong?', with a strategy.?

A lot. And we planners have to deal with it all, to succeed.

Putting a strategy into a system is complicated business.

This chapter, and other chapters, will go into depth about how to deal with the uncertainties of finding *good enough* strategies for *complicated*, *compound* and often *chaotic* systems.

Case 2.1 The $100 Million Project that never delivered value in 8 years.

In one client project, mentioned earlier (1.7), there were 8 major objectives (all vague, like 'Robust', as noted above).

But for each one of them, a large number of technical strategies were specified. About 10 strategies per objective. Guess, what?

The Technical teams spent 8 years, 90 people on the project team, and between $100 million to $160 million outlay, buying and developing the 80 or so technical strategies. The boys had great fun with the latest toys, and the suppliers raked in the money.

But….. absolutely no improved value was delivered at all in the 8 years! No 'robustness' for their customers, was improved.

Now, who is responsible? The Project team, the project manager, the suppliers, the CIO, or the CEO ?

My conclusion is that the CEO was very directly responsible (for insisting they 'did not waste time clarifying objectives, as the CIO suggested'), and a weak CIO (who was unwilling to confront the CEO on this) was partly responsible.

Everybody else was just 'getting on with the project', full steam ahead, damned be the cost. Just doing as the CEO dictated.

If the CEO had taken a different approach, things could have been different. He should have said (and had as *policy*).

"This is urgent, that means it is urgent to deliver a vastly improved 'robustness' and the other 7 objectives to our customers."

"I don't care HOW (strategies) you do it. But I do care that we make rapid, early, measurable progress, towards agreed relevant customer objectives, each and every month this year, and until we reach our objectives. "

"We will NOT have a 'revolutionary' project, to replace the current customer system (which 'might take 8 years, $160 mill., and fail').

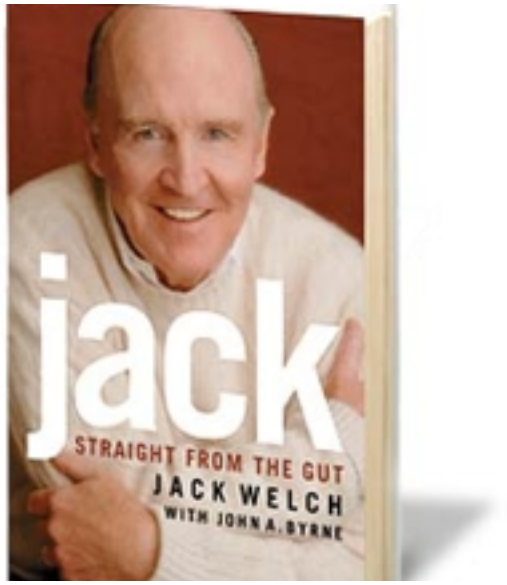
But we will achieve revolutionary quality improvements to the old current system." (says the CEO, in theory: but he did not)

In terms of planning, we need to make the distinction between 'ends' and their 'means' very clear. And we need to make sure that the people working on delivery of the strategies are painfully aware, that their job is primarily to move the objectives towards their Goal levels on time. Using whatever strategies actually work *well*. [URL11, shows this delegation of power].

Quotation 2.1 ***Aligning Rewards with Measurements***

"You have to get this one right. … a universal problem: What you measure is what you get – what you reward is what you get. Static measurements get stale. Market conditions change, new businesses develop, new competitors show up. I always pounded home the question 'Are we measuring and rewarding the specific behavior we want?' "

Jack Welch, former CEO General Electric [4, page 387]



We need to specify exactly what we think we are planning to *achieve*. We need to be clear about these objectives' relationship to specific named strategies, up front. Here is the starter kit

New Business Development:

Type: Chief Marketing Officer Critical Objectives.

Supported By: New Markets, New Products, New Sales Channels.

More details here… for the Objective.

Scale: ….

Goal: ….

Planguage Example 2.1. Specifying which strategies we think primarily contribute to reaching this objective's Goal (New Markets etc.), using the 'Supported By' statement.

Here is what is going on here:

- The objectives and the strategies all get a unique, and long term 'tag', for all related planning and delivery processes.
- Tags are in **A**ll **C**apitals, as are all other formally-defined planning ideas.
- Tags are things that can be used to reference any planning object (as we did here in 'Supported By').
- The 'Type', statement declares what we *think* the **New Business Development** objective is ('Critical Objective')
    - We prefer to underline *and* **bold** the tag of the master definition, to help the eye find it.
- We are also specifying which 'level' or 'stakeholder' it belongs to (Chief Marketing Officer)
- **Supported By**, makes it clear that for the moment we believe in, and are betting on 3 strategies.
    - But their exact definition (below *their* **Tag**, *elsewhere* in the plans) may change, as we learn and *experiment*, to see what *really* satisfies the Goals in the objective **New Business Development**

Practical Tip 2.1. Ask how much effect the strategy has.

- *Every time, well - at least one time this week, when someone suggests a strategy at a meeting or presentation ,*
- *Ask an 'innocent' question:*
    - *"Sounds interesting. Can you estimate the degree to which it will meet our Goal, on time?" [URL15]*

Policy 2.1. Make ends and means distinct.

- Make it explicitly clear what the *real* objectives are, and what the supporting strategies for them are 'believed' to be.
    - Don't *ever* let people think they can just execute the strategy, with no responsibility for the intended objectives.

Why?

- So you do not end up with the strategy alone, and fail to get the results you *really* want..

Section 2.2 Why we have to worry about strategy side-effects, and *how* to worry. Responsibility for the whole effect picture.

Principle 2.2 Don't Be A One-Trick Pony.

**Some strategies
are more effective
than others
for one *single* objective,
but that is not good enough.**

Some strategies are extremely effective for *one single* objective, one of your full set of many objectives. That 'fitness for the one objective' is not enough to decide to choose that particular strategy!

But a lot of planners act as if 'fitness for one objective' is enough. They ignore the other objectives totally in their planning. They are a 'one-trick pony'.

You have to consider *several* aspects, of each and every strategy.

Figure 2.2 A. A symbolic set of multiple objectives.

You cannot focus on one of your many critical objectives, like 'Security' alone.

Because, like it or not, single strategies can affect all your critical objectives, in significant ways.

Too-powerful security strategies, and you are going to hurt the '*ease of use'* – and vice versa, for example.

Maybe '*all the other strategies',* impact '*all the other objectives',* too. They do, no 'maybe' about it.

Narrow one-dimensional analysis is as irresponsible as a chess player only considering one aspect of a proposed move.

I am amazed to see planners planning with simple lists of strategies for a single objective, like 'Security' without *any consideration* of any other aspects than Security.

Security Insanity. An example of 'narrow' strategy analysis.

If you have ever encountered a security password like

xnx4n88mmvmsl2?MM888f6G

to protect a local wifi?

Like in a café?'

The Pentagon is not that well protected! (Ok inner local security, like encrypted at Smartphone level, is 'better')

Keying in 'xnx4…….6G' in from a smartphone is a nightmare.

That is 'security design gone crazy'

Sometimes I get such a password right on 3rd try. Mostly I now give up in advance.

Don't do this to fellow citizens! But perhaps it is OK to confuse our enemies this way?

I love passwords like: 'enjoy'.
And perhaps a new one, another day: 'drinkresponsibly'.

In fact in some situations I am bowled over by the *fearless* 'NO PASSWORD. Enjoy the free Wi-Fi with your coffee', sign.

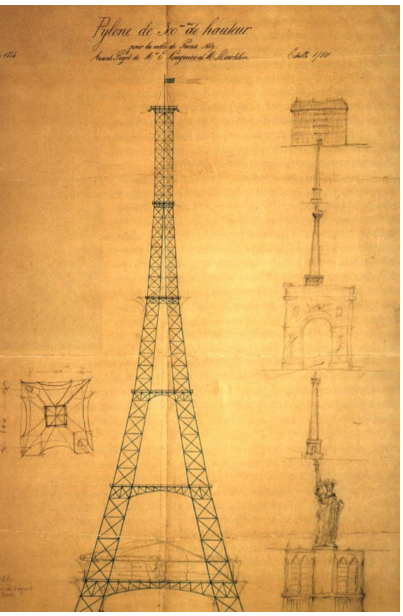
The strategy specifications, I see, in some instances, might be 'designed' by 'security experts', or (worse) 'security *architects*', who take *no responsibility for anything else,* except security [URL66], and hardly that.

They usually specify the usual copy-and-paste collection of buzzwords, and little more. *'Encryption, passwords'*.

Making Good Strategy Decisions.
Good decisions on strategies, will do more than to 'ask how effective they are in a single dimension'. Just as, when considering a new recruit, we do not just ask about just 'one skill' or their 'salary needs' alone.

How to make good strategy decisions.
'Good' strategy decisions, at least the *analytical* basis of decisions is defined by the process just below. We will consider the following, about a strategy (and we will deal with these aspects in more detail, later):

- how *effective* will the strategy be, in reaching our *main objective* (the one it is picked primarily to serve)
- how *risky* is the strategy, compared to more-conservative, well-known strategies?
- What are the *short-term costs, the long-term costs*, and the *cost uncertainty*?
- How does the strategy impact, all these considerations above, for *all other critical objectives,* at this same level of responsibility? Side-effects analysis.
- How does this strategy act, on any-and-all *constraints*, such as budget, deadline, legality, *politics*, stakeholder support, … ?



o The Eiffel Tower design contest brief, required French technology ('politics'). For example the elevators. But finally the American 'Otis' got the contract. The French much later 'rectified' this with replacement French technology. So strategy constraints are not 'holy' but they must be considered [URL83].

Picture 2.2 A. Original first plan of the Eiffel Tower, by Maurice Keochline. One of Eiffel's staff engineers. Eiffel did not like the idea at first, but later he had an apartment in the Tower!

- What are the interesting Background specifications, that will help us evaluate and manage the strategy (see Figure 2.2 B)
    - For example, who is the 'Sponsor' of the strategy?
        - oh, it was suggested by a supplier! Hmmm.

In short, we need to have a complete, relevant, *top level* strategy evaluation before committing, even for trial use, to a particular strategy.

But I would still do 'strategy identification, specification and evaluation' in a matter of *days*, not weeks. For example using our Project Startup Week process. [URL3, URL4].

Anything less, is not smart management, because it will end up costing far more in delays and disappointments, than any extra energy used for smarter planning.

Come on! 3 quick days of work, and it is done. Quantified objectives, Quantified Strategies. Is that really asking too much for your 'important' project? Try. Once. Then make up your mind.
Do it alone in your weekend, if you are really shy. Then make up your mind!

A first round of analysis of a strategy, can be drafted in an hour. That means we might make about 10 rough estimates of how a strategy is expected to affect our top-ten objectives. A bad decision will waste weeks, months, and years.

Early strategy analysis insights like:
- nobody knows the impact on *that* critical objective
- we estimate the range of impact is from 10% to 95% depending on …
- we do not know if it is legal in all our projected markets
- the long-term life-cycle maintenance cost is unknown and potentially scary

may lead to a different strategy entirely, or at least more-solid research on *this* one, *before* we adopt it.

At NASA (Fig. 2.2 A) , when they were spending 5% to 10% of total project costs *more (i.e. spending a total of project costs of 15% to 25%)* on 'managing project objectives', they consequently avoided 100% to 180% cost overruns, that some impatient colleagues experienced. These cost overruns were reserved for the projects who only devoted 5% to requirements. Very bad management, short-sighted economics.

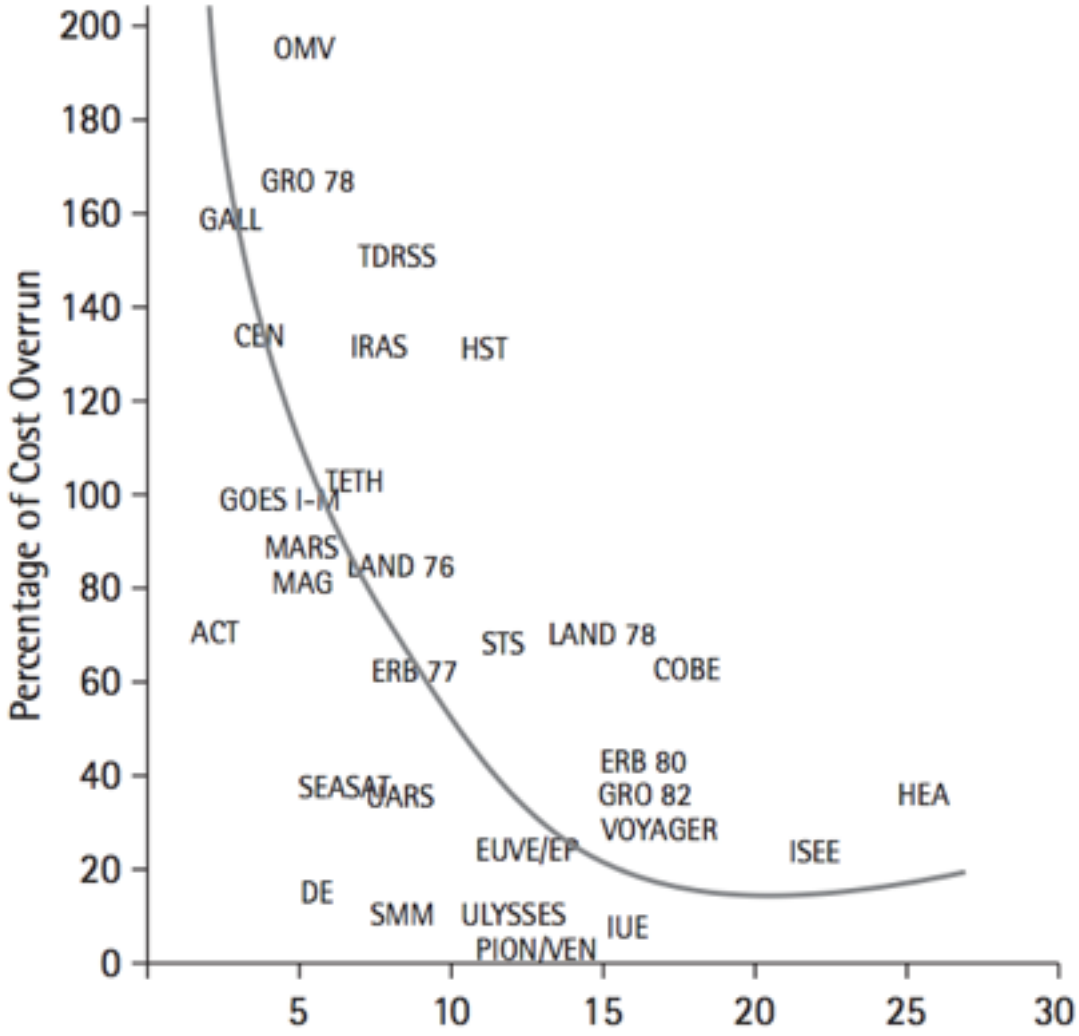


Figure 2.2 A NASA ***% of total project costs spent on 'objectives'***

Source: Hooks, I. 1994. Guide for managing & writing requirements. Boerne, TX: Compliance Automation Inc. [URL11, Estimation]

This suggested 'strategy analysis process', detailed in 'How to make good strategy decisions' (above), raises a long list of questions about any, and all, strategies. We will be treating many of these strategy evaluations, and richer specifications, in even more detail later.

So for the moment, let us indicate how we could 'make notes' in the planning about what **we know, and what we clearly do *not* know** much about.

At the very least, we need to have a culture that *explicitly documents its unknowns, risks and uncertainties*: with **pride** in 'knowing they don't know'. The 'fools' pretend they know, and are afraid someone might find out about their ignorance.

> ***"To know what you know and what you do not know, that is true knowledge."***

Confucius again.

I once won a contest for the wisest 'course slogan' for G M Weinberg's 'Problem-Solving Leadership Workshop' (www.geraldmweinberg.com). I have a round yellow pin badge, where my winning slogan is printed, to prove it. It says **"I know I don't know"**.

This seemed to me, to be enabling the potential 'wearer of the badge' (all graduated course participants) to seem wiser, than if the badge just said

"I don't know". I did not realize at the time that Confucius agreed with me.

I'd like to think I am the re-incarnation of a wise person (would't you?). But, I don't know that for sure.

PS, while we are on the subject. In my next life, if there is one (and I don't know yet), I hope to read a copy of this book, early in life! It would save me a lot of wasted effort. I'll read it and say: 'hey, this guy thinks like I do'. Or perhaps, with useful optimism, 'this is too much effort. There must be a simpler way!'

Responsibility for a Plan. Responsibility for it all.

Reverse Forensics! Name the guilty *before* they commit the crime.

'**Responsible Planners'** can 'raise questions', for example:

Issues [Goal ABC]: xxxx.
Problems: nobody in-house can identify a much cheaper strategy.
Risks: This strategy could bankrupt a supplier.
Value Responsible: ?.
Spec QC: none yet. Planned for 24 December.
Defect Level: 33/page.
Supplier: <Space X>.

Some simple ways of raising potential problems in Planguage 2.2A.

Laser Engraved Acrylic name plates showing all companies involved with the fabrication of parts, including Rivercity Industries for their metal sheets.

Image showing the placement of all name plates together at the entrance of the of the structure.

The full picture of name plates being displayed at the entrance of the gourd.

Having documented potential problems in the planning object, not in 'meeting notes', where they will get 'lost', we have an opportunity to work towards 'acceptable answers'. And we have 'reminders' if we fail to do so.

Ah! What a term, '*Responsible* Planners'. Just 'invented it'.

Reminds me of the tradition of Architects and Builders, carving their name 'in stone' of their architected front entrances. As I recall from Egyptian times at least? (Imhotep, Senenmut ?) We sure know who they were today. Will your plans be remembered that way?

Pictures 2.2 B, Names of participating companies engraved on their building. Texas. http://lasercuttinglab.com/portfolio/laser-engraved-name-plates-overland-architects/

How would *you* feel if you were told that *your* name will be prominently displayed on the product or service you are planning? *Steve Jobs engraved early Mac designers names inside the plastic case!* Or at least, your name might be be put into the planning object specifications you had worked with (see example below).

My experience of 'naming people', voluntarily (nice tactic) or *not* (if they are getting paid for it, they cannot object?), in the planning objects *(Technical Expert, Domain Expert, Cost Estimator, Suggested By, Reviewed By, Approved By)*, is that they immediately wake up 'in the meeting', and take the whole thing a *lot* more seriously.

A very high value, low cost tool, I'd say!

*"Hey, wait a minute. Before you put **my** name on that, I'll need a chance to research it more deeply!"*

('You mean up to this point you could not care less about what we are discussing and planning?', I think to myself.)

Amazing the practical power of 'my name is on it' (my honor, my life, my job!, my career, etc.).

Picture 2.2 C This is the guarantee by Raigad Fort's Chief Architect / Engineer "Hiroji Indalkar" engraved on the wall about the 'strength' of the Raigad Fort.
Wow, one quality chiseled in stone! Guaranteed by the 'planner'. His name is chiseled in stone also, elsewhere. (see URL85)
http://raigad-fort.blogspot.no/2010/10/guaranty-by-hiroji-indulkar-raigad-fort.html
Permission requested from website. 12th to 15th Century AD.
PS Hindi readers who can send me a translation? Thanks. tom@gilb.com Aug 2015.

(Edit note, notice different spelling indulkar (URL) and Indalkar (site text).)

Here is a simple example of beginning a 'process of analysis' of a strategy decision.

Security Strategy:
Type: Top level security strategy.

Responsible Strategy Planner: Tom Gilb, Chief Planner, Aug 14 2015.
Security Architect: Steven Shad, of Yale Inc. Santa Monica.
Side Effects Analyst: Wendy Bartlett, Business Analyst.

Description: xxxxxxxx (the basis for analyzing impacts).

Supports: Security Objective.
Impacts: Usability, Training Costs, Terrorist Laws.
Risks: Unknown operational cost, Unknown Maintenance Cost.
Issues: not thoroughly defined, no contract guarantees, effects not proven on similar projects.

Planguage Example 2.2 B. Some ways of documenting various aspects of a strategy. Most everything is 'background' information about the strategy. Notice the 3 explicitly named responsibilities.

'**Supports'** tell us the main objective this strategy is here for. '**Impacts'** gives us a list of other objectives or constraints it might affect, for better or worse. '**Risks'** warns us for some problematic dimensions. '**Issues'** lists questions needing answers, which might warn us of more problems with the strategy.

Figure 2.2 B. Some examples of potentially worthwhile strategy 'relationships' to document. These are 'Background' Specification items. They supplement the core Strategy Description.

Keep a systematic written map of known strategy relationships

A single strategy:
- in addition to potentially, and very likely,
- *impacting* most all of our objectives and constraints,
- at the same level, and at different levels,
- has a large number of potentially interesting relationships,
- to other types of planning objects.

This graphical list (Figure 2.2 B) is just a starter kit. The real world of the strategy is 'Complex' (Compound in structure; consisting of interconnected parts), and *that* compound-ness can lead to it being complicated to understand.

We cannot *model* all potential interactions when we plan, nor can we easily model the degree and timing of such interactions.

But we can *try to keep track of* interactions and relationships;
- try to keep relationships in mind while we plan and re-plan,
- make it easy to access strategy relationships
- try to avoid forgetting the relationships, by *writing them down* in the planning-object specification, objectives and strategies, primarily.

There are just too many related things to a strategy, for even the best of minds to know, and to keep in mind.

Make it easier for yourselves.

Write relationships down, accumulate them systematically, from the many knowledgeable minds, and keep this background specification around (in your Plan Database, in each plan object, each objective and each strategy).

Once the relationships are keyed in, in 5-10 seconds, and directly associated with the planning object, they are near zero cost to keep, in case they are useful. The moment you separate such relationships from the planning database object: they are gone, not available. Little practical use. Rather too many relationships than 1 too few.

You can never be sure when it will be critically useful to know about a certain relationship. A plan relationship unknown and undocumented could hide a trap, just waiting for you to stumble over it.

Individuals cannot be expected to remember all relationships, but as a planning group, over time, they can contribute to collecting the relationship information in writing, in the plan database, in the related planning objects.

Personally I capture relationships in meetings, immediately they are mentioned or implied. Helps me keep interest in the meeting.

Anyone on the team should be able to easily update, or suggest an update to such background information. Err on the side of caution.

That gives us a fighting chance of making use of the relationship information when we need it.

"At CARE, a leading humanitarian organization, we recognize people live their lives in a holistic manner. Issues such as health care, education and economic empowerment cannot be addressed in a vacuum. Thus, effective programs need to tackle the multiple root causes of poverty."

Helene D. Gayle
M.D., President and CEO (2006-2014+) of CARE USA

Practical Tip 2.2 Ask about other connections

- *When it becomes obvious that there is too much exclusive focus for a strategy, on a single dimension; ask -*
  - *"Are there any **other** performance objectives, or costs, that this **might** affect, in interesting ways?"*
- *If you get no coherent suggestions, those people (that includes if you are 'talking with yourself') are not to be trusted, as knowledgeable, expert, or having realistic experience.*
- *Get somebody who can answer the side effect and cost questions from experience: an 'expert'.*
- *Nobody has to possess knowledge and experience for everything in advance. But you do need to recognize when it is responsible and wise to get more facts. Them's smart cookies!*

2.2 Policy: Thorough multi-dimensional strategy analysis.

- **We will try to estimate the impacts of all strategies, on all our critical objectives.**
- Why?
  - Just *naming* a strategy, as 'suitable for one objective', does not *at all* qualify it for investment. Think totality, get real.
- How?
  - All major strategies need to be systematically considered, with respect to *all* objectives, constraints, and uncertainties: at least for an hour, at least 10 lines of documenting your ignorance about them.
  - Ultimately, an Impact Estimation Table does this job in a thorough and visible manner.

Section 2.3 Looking at Multiple Impacts of A Strategy; And doing so 'after the battle has begun': Dynamic strategy selection.

Principle 2.3 Decathalon Champion.

Some strategies are more effective than others,
for *overall* objectives;
and even this fact,
might only emerge gradually,
as time goes on.

We have noticed, when making numeric estimates for strategies, that they seem to significantly (10% to 40% of the way to the Goal level) have impacts on many (4 to 7) of the *other* top ten objectives.

Figure 2.3 A. In this graphical representation of the impacts of 3 strategies, on 4 Objectives. We can see they all have some impact, but that Strategy 3 has a ***dominating overall impact.*** *If I had to choose one strategy to start exploring first, it would be Strategy 3, for that reason. 100% in this case means 100% of reaching the Goal on time.*

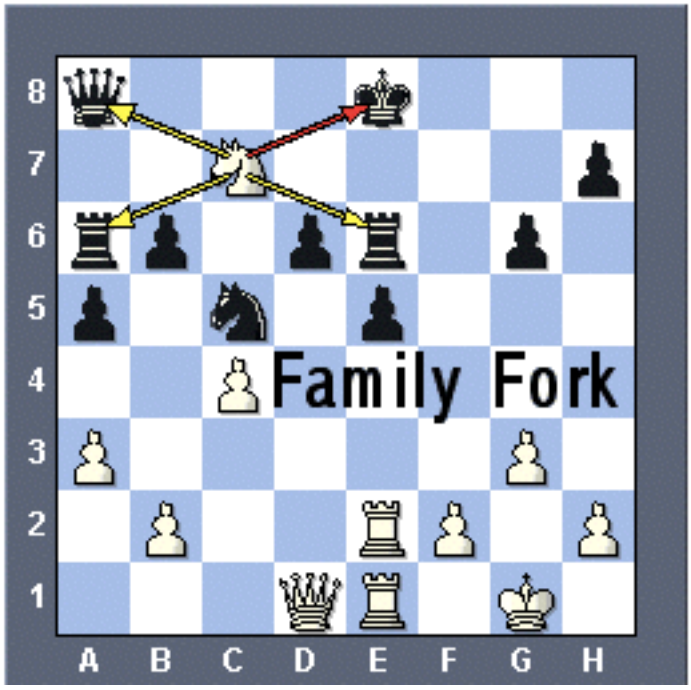


Illustration 2.3. A Chess position (a strategy choice) commanding many 'objectives' at once.

This 'multiplicity of impacts' is not unlike what you find for chess moves, and in many other aspects of real life (like employees, employment, and marriages).

There seems to be some natural 'law' that everything comes with interesting impacts on quite a lot of things.

This is 'complex': many things with many connections. The real emergent performance characteristics of these connections is 'complicated' to *predict*. But, it is much easier to simply *observe*, if strategies are implemented in a series of small value-delivery reversible steps, as we normally recommend (more later). Observe and see the increment of the values over time.

All strategies have *many* interesting impacts. *Interesting* impacts, includes the well-known engineering notion of *negative side effects*, on some objectives and constraints.

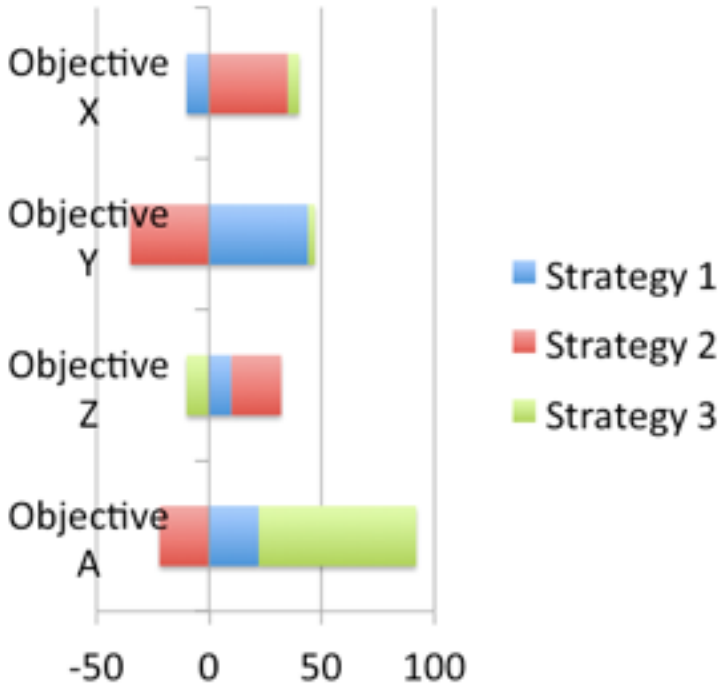


Figure 2.3 B Side effects, and negative side effects.

Strategy 2, above, has *as many negative* side-effects, as it does positive effects, its net effect is zero. The other two have almost the same net effect (66, 68) so they are a toss up. But only if their costs are the same. Otherwise the 'most efficient one' can be our first choice.

Assuming this 'diversity of impacts' largely holds true, it makes sense to

- pick strategies that have the largest *total net positive effect*, on all of your critical objectives,
- at acceptable risk levels, and
- with acceptable short-term and long-term costs, for scarce resources (time, money).
- Or, simply make sure strategies give 'value**s** for cost**s**'.

Very much like picking a series of generally smart, overall, chess moves. Even in chess, there are 2 clear 'costs', the 'move' and 'time'.

Quote 2.3 A. Strategy Evolves Reacting to Feedback.

"The Prussian general staff, under the elder Von Moltke, perfected these concepts in practice. They did not expect a plan of operations to survive beyond the first contact with the enemy. They set only the broadest of objectives and emphasized seizing unforeseen opportunities as they arose... Strategy was not a lengthy action plan.
It was the evolution of a central idea through continually changing circumstances." *[4, p.448]*

And Tom would like to add

"Moltke's strategy is remarkable for its judicious economy of force, for he was wise enough never to attempt more than was practicable with the means at his disposal." *[URL69]*

Photograph 2.3 Helmuth Karl Bernhard Graf von Moltke (1800-1891)

We wanted to insert that quote above, (Source: via Jack Welch, GE), to emphasize that there is no such thing as the 'right strategy', 'known in advance'.

There are strategies, which will, in fact, really get you to your objectives. And these *might have to be discovered gradually*, one by one.

However, we can generate a fairly good, promising, set of strategies, to try out, before selecting some to start with, and before scaling up.

Figure 2.3 C. A hypothetical case of the estimation of multiple impacts of 3 strategies on 4 objectives. 100 represents the Goal level planned.

In this hypothetical case (Fig. 2.3 C), it looks like Strategy 1 is slightly better than Strategy 2 in overall effect. A toss up really.

But, it may be *too early to decide* which one of these to start with. Because there is a big gap of about 50% in reaching all the objectives (100). So we need to see if we can find one or more additional strategies to fill the gaps.

Even then, if a 'Strategy 4' actually filled all the gaps; we still have to consider the **costs** of the strategies. If 'Strategy 4' costed ten times more than any one of the others, then it would not be the *most efficient* strategy.

Maybe we would prefer to start actually *implementing* a more cost-effective (efficient) strategy, like Strategy 1 or Strategy 2, first, and see what happens.

We might also consider finding a much cheaper, but similarly effective Strategy 4 (Design to Cost [23, Miles] method)

The main point is that, with some impact estimates on the table, we can discuss and reason about our options. We can do so in spite of the complexity of the strategy/objectives relationship [URL87]. And hopefully we can make somewhat better decisions, on average, over time.

We will later, in this book, get quite-formal, in estimating impacts of strategies (Section 6.5). But consider the simple idea that there is a great deal of difference between a 1% impact on reaching my Goal level, and a 99% impact. It is in fact, 100x or 'two orders of magnitude'.

The difference is worth considering: and the difference is sometimes the difference between failure and success. Between guaranteed failure, and a possibility of extracting success on time, by tuning the ideas to reach your Goal levels..

Ideally, we could estimate the 'exact future impact' of our strategies. But that ideal is **not** available in our compound, complex, competitive, dynamically changing real world – the complicated world.

However, that annoying fact does not prevent us from having some *'numeric opinion'*, and also 'trying to get the 'order of magnitude' roughly right'.

We don't have to guarantee the estimated future. But we could generate enough good 'guesstimates' to *increase the probability* that we were likely to be using, *potentially useful*, perhaps-with-some-tuning, strategies.

And, if nothing else, we might generate a numeric estimate that a colleague strongly *disagreed* with, which might be then *worth discussing*, before committing to a strategy.

So just imagine, until we suggest, hopefully, even-better methods (to be discussed later), that, rather than just indicate *there are some* impacts for a strategy:

- imagine that we *force ourselves* to make a numeric estimate of the strategy impact,
- in brainstorming mode,
- conscious that the numbers are not commitments.
- These estimates are just *clearer communication of individual knowledge or prejudices.*

Here is a simple example of trying to do that. Just for the sake of getting *something* better than *nothing*.

Hoping to distinguish between 0.6%, 6%, 60%, or 600% effects, if we are lucky. Trying to develop at least a basis, for deciding what looks most promising to try out, and measure, in practice.

Security Strategy:
Type: Top level security strategy.
Description: xxxxxxxx.

Supports: Security objective: 80% of the way to the Goal level.

Impacts: Usability 20%, Training Costs 30%, Terrorist Laws: Negatively

Risks: Unknown operational cost (10x?), Unknown Maintenance Cost (10x implementation cost??).

Planguage Example 2.3. A simple attempt to say something about the degree of values and costs for a strategy, numerically. Using words like 'substantial', 'a little', and 'extremely good', does not make it more right. But the use of these fuzzy words makes it more difficult to verify and discuss with respect to observed facts.

Quote 2.3 B On Fostering Quality.

Quality is never an accident; it is always the result of high intention, sincere effort, intelligent direction and skillful execution; it represents the wise choice of many alternatives.

William A. Foster *Perhaps not a real person! [F16].*

Practical Tip 2.3. Fools use mere words, Angels use numbers.

- ***Whenever you read, or hear, nice sounding words,*** *indicating a degree of effectiveness of a strategy:*
  - Like, *'top notch security', 'state of the art user-friendliness', 'best in class service', etc.*
  - ***Ask if these can be expressed with numbers***
  - *Ask if the person saying or writing the nice words, is incapable of expressing the ideas with numbers, meaning they are 'ignorant of the facts', but they are trying to convince you to 'go with their ideas'.*
  - *Ask if you, or others, can define scales of measure, and estimate the degree of impact, of a strategy on that scale.*
  - *If you cannot even do that, you have no hope, and are at the mercy of the wolves. There is some instruction in this book on the subject of quantification of objectives (Section 1.1 and much else).*
  - *In any case: even if both parties (strategy proponents, and you the strategy evaluator) fail to express the ideas of strategy goodness, quantitatively, you will at least realize that the 'blind are leading the blind'. Your risk of failure is very high.*

Policy 2.3: Strategies will be evaluated quantitatively,

We will **estimate, and measure, in all *critical* dimensions: Words are not enough.**

- Why?
- If we are to hope to start a process of getting a clear idea of the strategies' values and costs, then we have to cross the 'great divide' between *words* and *numbers*.

- **How?**
- **Start the estimation process, when you first identify a strategy.**
  - **Refine it when useful.**
  - **And finally measure** the selected strategies to 'confirm or deny' the initial estimates.
  - Then you can begin to understand, at least, the current facts;

- **Do this** before '*change'* and '*scaling up'* make a mockery, of even *these* temporary estimates!

Section 2.4. Evaluating the multiple cost aspects of strategies, in relation to project resource budgets. The simple 0 to 9 method.

Principle 2.4. Value For Money

Some strategies are more 'efficient' (cost-effective) than others.

It is *not* enough to understand how *effective* a strategy is for the top level critical objectives. We have to understand the *corresponding costs* of the strategies. Both initial investments, operation costs and maintenance costs, and decommissioning (like cancelling a long term contract, or replacing with very different strategies).

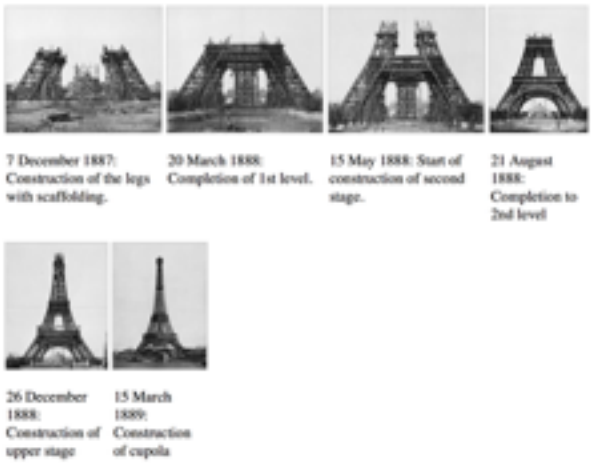
ET Case: The Eiffel Tower was, by contract, designed to be **easy to disassemble**, within 20 years after the Exhibition.

In 1967 it almost was disassembled for a Montreal Exhibition.

But, an emergent property, ability to communicate radio signals from that height (324 m±15 cm), motivated the French to keep it up.
[URL83].

Case 2.4 Ref: 'Decommissioning Costs', above, The Eiffel Tower was designed for easy disassembly after 20 years. Maintenance costs are huge, paint alone 50-60 Tonnes of paint every 7 years for rust protection. New elevators etc.

Picture 2.4
The Eiffel Tower assembly.
Imagine it being disassembled in reverse order



In short, we cannot focus on value alone. We need to look at 'value for limited resources': time and money.

This is not 'simple'. It is complicated to understand, and predict, and there are no simple solutions.

But we have some planning methods that might help you avoid the fate of many projects, that are at least 3x over budget and deadline [URL11].

And our methods could help those projects and systems that have lost control of their long term operational and maintenance costs so badly, that giving them up for new systems seem the cheapest alternative: as long as we continue to mismanage qualities and costs.

Again, I am suggesting that 'a stitch in time saves nine'. You have to spend an hour or day, to avoid wasting a week or a year. You have to be mature enough to try to manage resources intelligently.

We get the distinct impression that people do not care, that management is not mature enough, it is not *their* money after all, just corporate or taxpayers: so who cares. Sad, but this book is for those who might like to take more responsibility, and would like some ideas on how to do so.

Our ideas about cost-effective strategies are:

- We need to look at the efficiency (value/cost) of strategies, not mere effectiveness.
- The argument, 'it is effective' is not as good as the argument 'it is efficient'
- The argument 'it is effective' is not as interesting as the argument of how efficient a strategy is in relation to *your defined* Goal level, and your defined deadlines, and your defined budgets!
- Of course, the effect**s** are *multiple*, and the cost**s** are *multiple*. You *cannot* simplify this dominant reality. Deal with it.

- You will be able to choose a candidate strategy to try out, using multiple estimates; but you cannot *settle on keeping* a strategy, and scaling it up, until you have corresponding measurements as to both effects and costs.

Here is a simple 'efficiency estimation' technique we have used, to simplify, but dramatize, the efficiency of strategies.

It works well at meetings to get decisions made, and discussions going.

But it is largely 'subjective consensus'.

We normally personally prefer more 'objective' fact-based evaluations [URL21]. But this simple technique can still be useful. Especially when you have little time, and need to get a group of people on board.

The 0 to 9 Method (V/C) for Strategy Evaluation
The effectiveness scale is simple; 0 (nothing) to 9 (highest, 'more than we need').

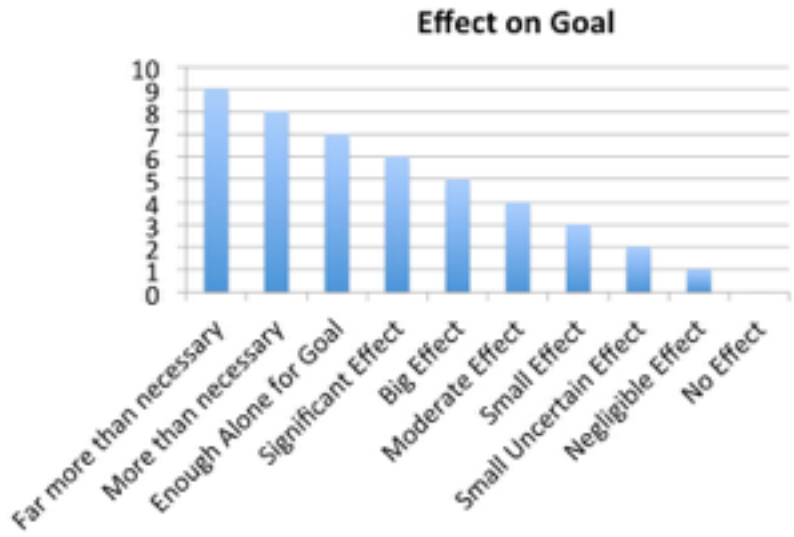


Figure 2.4 A. The subjective rating of the 'effect of a strategy' in relation to an objective of no particular definition, or, if we are lucky, reaching a defined Goal, or better

The cost scale is also 0 (practically no cost) to 9 (high and unpredictable).

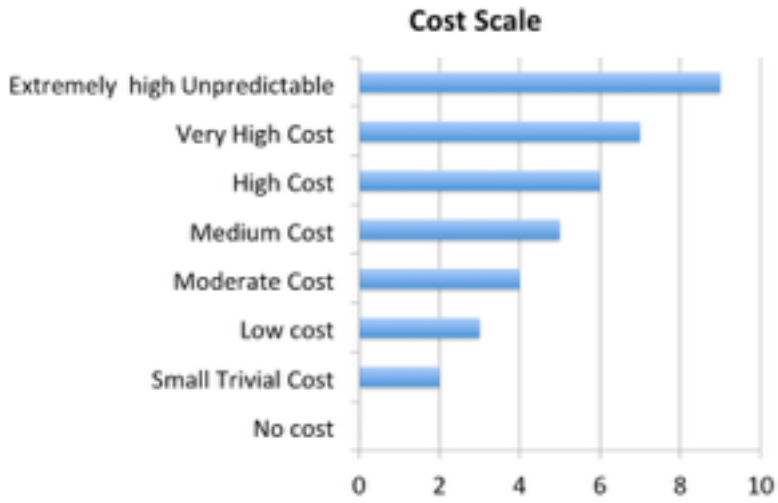


Figure 2.4 B. The subjective rating scale for the overall cost of the strategy. Which costs and which budgets are not at all defined. This is a simplified rating, an opinion.

People are asked to, without discussing or consulting, write down their *opinion* for each strategy. An average or median is computed from all the individual results. One could have discussion about any 'extreme' *deviant from the group* values, if people wanted to share why, or how, they arrived at these numeric opinions.

The Result could end up something like:

Strategy A: 9/1
Strategy B: 7/7
Strategy C: 3/6

And that makes Strategy A look like a winner, or something to try out initially, and measure. Then we can see if it is as good as people think it is. Hopefully before large-scale commitment.

We are *not* trying to *predict* or estimate the effect and the cost. We *are* trying to a get a small group, to rapidly arrive at an agreed consensus, about which strategies should be tried out earliest.

So the subjective numbers can be dead wrong, but we move ahead with trying out strategies. And hopefully, for the most part, pretty good strategies, in relation to their costs, will come out ahead.

And if not, well, they hopefully get tested, maybe measured, and we learn something. And if even that does not come to pass, the group made the bad decision, and has to live with it.

And if all of this is unsatisfactory, we suggest that you use the more objective methods elsewhere in this book (IET). You are then saying; it is worth somewhat more effort to raise the chances that we are doing smart strategies early.

This method has its weaknesses, but it is better than a shouting match.

We ourselves prefer the more objective methods, like the Impact Estimation method [URL21] which we will present later. But we present this just to get the basic 'efficiency rating' concept across.

Quote 2.4 Hidden multiple attributes need accounting for

The reality is gas prices should be much more expensive than they are because we're not incorporating the true damage to the environment and the hidden costs of mining oil and transporting it to the U.S. Whenever you have an unpriced externality, you have a bit of a market failure, to the degree that eternality remains unpriced.
Elon Musk (1971-)

Practical Tip 2.4 Think About Your Strategy Costs

- *List your short-term resources, like time-to-deadline and capital expenditure*
- *List your long-term costs, like maintenance and insurance*
- *Estimate the costs for each strategy.*
- *Now look at the 'net-value over the cost' picture; illuminating!*
- *It is not easy, but the unplanned-cost surprises are much harder to live with.*

Policy 2.4 Strategy Efficiency

All strategies need to be evaluated (estimated, then measured) based on their numeric effectiveness in relation to costs.

- This can be done as objectively or subjectively, as you have time and reason to do.
- But don't *forget* the *cost aspects* of your decisions!
- Be aware of the weaknesses of any method you choose. How subjective or objective are they?
- And always know, you must *verify any estimates* with *real measurements* of the strategies, before *committing* to them on a larger scale.
- And, just to complicate matters,

  - any new strategy, added to some successful ones,
  - can destroy the previous effects and costs,
  - and invalidate the earlier decisions to use those strategies.
  - Live with it! More importantly, expect it, and know how to deal with it.
  - Shall we suggest a term for all this? 'Strategy agility' .

- Why?
  - to reduce risks of unpleasant surprises later, at much higher costs to deal with then.

SECTION 2.5 Strategy choice depends on residual resources. Dynamic Design to Cost.

Choosing the inexpensive options.
You might choose a strategy, from several sufficient-effect options, based on whether it costs money or time, or other resources to implement.

Not only that, but you might be forced to choose an effective-enough strategy option mainly because it uses resources you have enough of, and does not use resources you are over budget with. It depends upon which one of those resources you currently have *available* in our budget.

A variety of Resources can suddenly disappear
Think about nearing the end of the project, and one or more key budgeted resources is used up, like 'money'. Or you have lots of money left, but only a month to some final deadline, where 'the law goes into effect' in the EU. Or your key project staff is already committed to another high priority project from next week: no people. Or the pilot site will become unavailable a week before you need it.

The consequence of this is that you cannot efficiently choose a strategy until you can look at the levels of 'resources remaining' in your resource budgets, and make a choice that does not run over deadline, exceed other resources or financial budget.

Principle 2.5 Complex Changing Resources.

Some strategies are more efficient
than others,
depending on which resources
you actually have available.

Why we need to consider the resource consumption of strategies dynamically.
Strategies might, on occasion, be fairly equal in overall efficiency, with regard to both their effects on multiple Goals, and their effects on consuming resources generally.

But, imagine a situation where you are one week from a deadline, you have only used 10% of your capital budget. There are two strategies with about the same overall efficiency. But strategy X can be implemented in a single day, and strategy Y will take at least a month to get some results from.

In this particular case, Strategy X wins, because it is the only one that can react to the deadline constraint, in a single day. It would win even if Y had better Goal effectiveness (unfortunately 3 weeks too late).

A Single Resource might be decisive, even if an alternative strategy is generally better.
The point is, that in specific value-delivery cycles, a *single factor* might be more decisive, at that moment, than the general principle of generally good values, for generally few resources.

More importantly, this situation *might not be clear in advance*. Earlier in the project.

The last value delivery cycle could unexpectedly eat up the rest of your money, or the rest of your time. Or key people can get sick, leave the project or company. Murphy's Law applies [URL55].

You cannot commit to using a given strategy in advance: you must decide on the spot
The point is that we cannot usually foresee such situations, because they are dependent on actual-to-date resource-consumption, from earlier implemented strategies.

The *resource consumption* of Strategy X might not even be *discussed* yet, when you initially considered Strategy X. So we cannot decide in advance to use Strategy X at delivery cycle 23; or even to use it at ever.

We cannot decide about Strategy X, if we are ignorant of its resource consumption, and ignorant of the remaining resource at the time Strategy X is implemented.

That would be an illogical design act. Though people do it all the time, they decide without considering resources.

This is one reason why we overrun budgets and deadlines.

It is also one reason why systems *seem* 'complicated' (unpredictable), while the truth is we never actually *considered* the resources, and their relationship to other strategies, at all.

We have to make our strategy selection decisions dynamically; like most things in life.

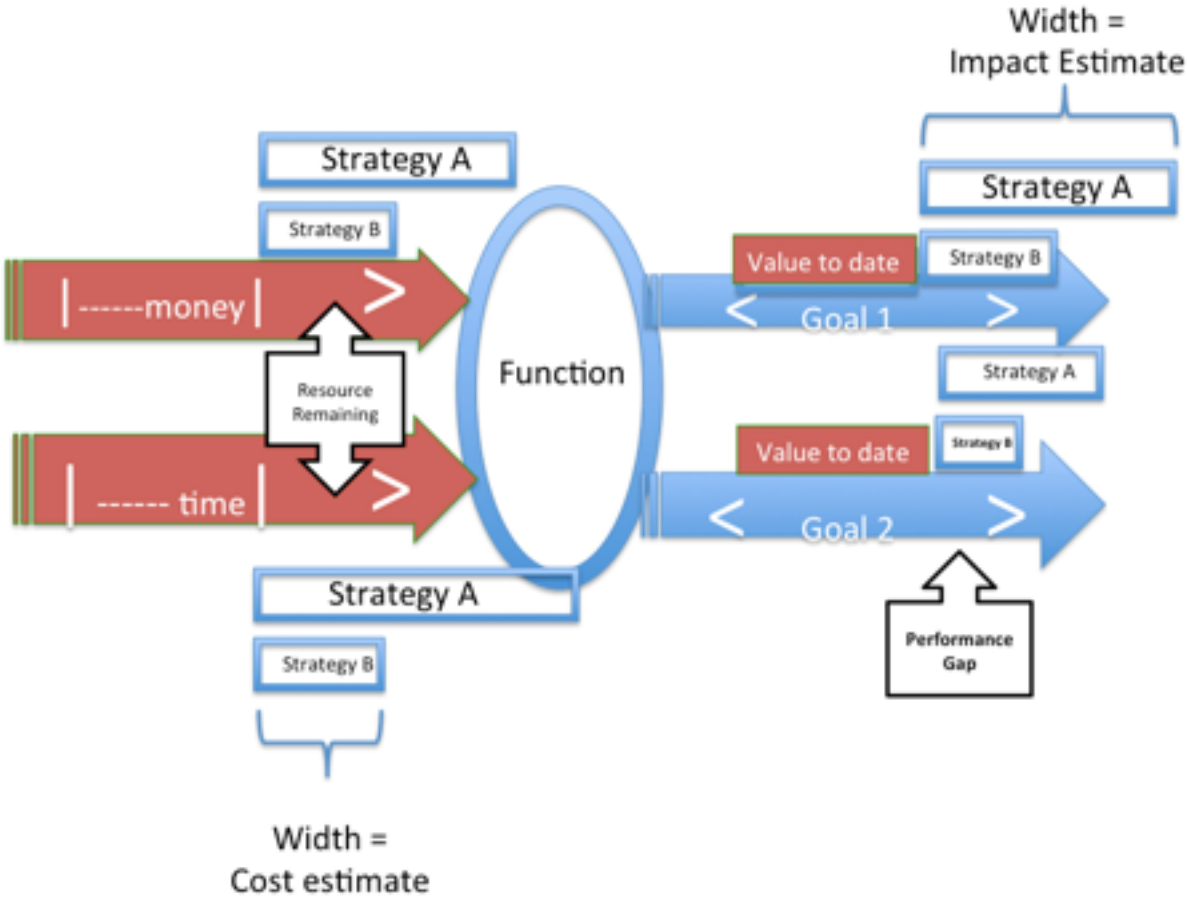


Figure 2.5. Evaluation of 2 strategies with respect to 2 resources. The width of the Strategy rectangle is the extent of the cost or effect.

Comment on Figure 2.5: Evaluation of Values and Costs

- Before we evaluate Strategy A and B, about half the money and time budgets have been used, to deliver other strategies, giving 2/3 of the values planned, for Goal 1 & 2. With about 50% Resources Remaining.
- There is a performance Gap of about 1/3 of the way to the Goal.
- We look at Strategy A, which is tempting, because it easily fills the performance Gaps, and more.
- But, Strategy A far exceeds both time and money budgets; so it is *disqualified* (unless there was no other option, then we might beg for more resource to do it, or reduce the Goals to fit our resources)
- Strategy B is estimated to get us to *both* Performance Goals, and it is within the remaining resources. So we will go with Strategy B.

Case 2.5 : IBM Federal System Division Cleanroom Practice.

Photo 2.5 The IBM FSD Cleanroom Method was used for Space Shuttle Ground Systems.

Case 2.5 IBM Dynamic Design to Cost.

A very successful method, for large military and space projects, was the 'Cleanroom' method [URL41]. It was very 'dynamic', in strategy selection, with 2% of project duration (monthly) used for value delivery cycles.

Robert Quinnan describes the process control loop, used by the IBM Federal Systems Division Cleanroom method, to ensure that cost targets are met. And they were under budget, or near it for 10 years, many projects.

"Cost management. . . yields valid cost plans linked to technical performance.
Our practice carries cost management farther by introducing design-to-cost guidance.
Design, development, and managerial practices are applied in an integrated way to ensure that software technical management is consistent with cost management.
The method consists of developing a design, estimating its cost, and ensuring that the design is cost-effective"

[URL13] (p. 473) Quinnan, IBM extracted from IBM SJ URL41

He goes on to describe a design iteration process trying to meet cost targets, by either redesign, or by sacrificing 'planned capability.' When a satisfactory design, at cost target, is achieved for a single increment, the *'development of each increment can proceed concurrently with the program design of the others.'*

"Design is an iterative process in which each design level is a refinement of the previous level"

[URL13, p. 474] Quinnan, IBM

It is clear from this that they avoid the big-bang cost estimation approach. Not only do they iterate in seeking the appropriate balance between cost and design for a single increment, but they iterate through a series of increments, thus *reducing the complexity* [URL87] of the task, and *increasing the probability of learning from experience.* Experience is won as each increment develops, and as the *true cost of the increment* becomes a fact.

"When the development and test of an increment are complete, an estimate to complete the remaining increments is computed."

[URL13, p. 474] Quinnan, IBM. *(end of Case 2.5)*

In strategy notation terms, we can track this learning (estimate/actual):

<u>Strategy A</u>:
Estimated Cost [by Architect] 5% of budget.
Actual Cost [Final Prototype] 25% of budget.

Planguage Example 2.5 Notation for Tracking Strategy estimates and actual costs.

What if the first 5, of 50 roughly-planned strategy increments, cost 5 times more than initially estimated? What is your hypothesis? What can you do about it?

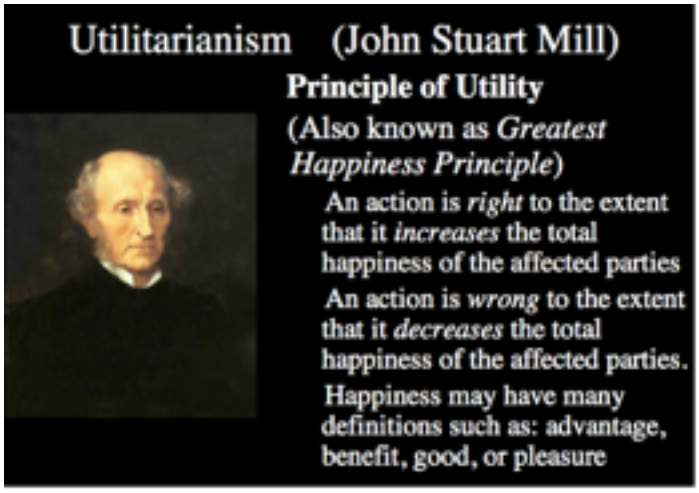
Quotes 2.5 Mill: Choose your path, or be a copy cat.

"He who lets the world, or his own portion of it, choose his plan of life for him, has no need of any other faculty than the ape-like one of imitation.

He who chooses his plan for himself, employs all his faculties.

He must use observation to see, reasoning and judgment to foresee, activity to gather materials for decision, discrimination to decide, and when he has decided, firmness and self-control to hold to his deliberate decision."

John Stuart Mill, On Liberty, 1859
English economist & philosopher (1806 - 1873)



This can be paraphrased for our purposes:
Strategies are good when they increase the total happiness objectives of the affected stakeholders.

Practical Tip 2.5 Estimate Costs.

- *Make estimates for all resources needed, for all strategies*
- *Keep track of resources used to date for all strategies implemented*
- *Then you can better judge which strategies you can afford to use on the next value delivery step.*
- *If you are not willing to do this, the system will seem complicated, and you will get 'surprises'; so don't be surprised by them.*

Policy 2.5 Learn Real Costs Early and Often: Adjust to Cheaper Strategies.

- **We will assume that all initial cost estimates are wrong, and possibly very wrong.**
- **We will estimate all impacts on Goal levels, for each increment of strategy,**
- **and then measure and record observed costs,**
- **together with the Strategy specification.[Planguage Example 2.5]**
- **We will learn fast. Consider redesign of the strategy (find a cheaper one!) and try again if necessary. Like IBM.**

- Why?
  - So that you can deliver your projects to the highest planned values, always on time and under budget, like IBM FSD managed to do for years on end!

SECTION 2.6 You cannot use a strategy that violates serious constraints.

Principle 2.6. Valid Strategies.

All strategies must avoid violating stated constraints

Any single constraint on the project, can cause a strategy to be totally invalid.
Constraints can pop up at any inconvenient time during your project: new laws, new corporate policies, budget cuts, competitive announcements.

This means that the strategic planner needs to have access to a current list of valid constraints. It means that strategies need to be validated just before investment and deployment, against this constraint list.

It implies risks of: suddenly having to find a different strategy, of unknown costs, and of unknown effects and side effects.

One way of integrating some constraint ideas into the objectives themselves, is to use 'Qualifiers' in a requirement level statement.

For example:

Product Portability:
Type: Marketing Director New Product Objective.
Ambition: much faster product conversion for new national markets than any competitor. <- Marketing Director.
Scale: Average Weeks from Market Go Decision, to Product Ready for Market.
Goal: < 3 weeks.

Market Type: Asian Languages, Euro Languages.
Commitment: Adequate Budget.
Partner Status: Agent Contract Signed.

Planguage Example 2.6 *Example of stating some constraints (last 3 statements+ 2 in Scale), in an objective, that all supporting strategies must know about and respect.*

In this example, there are 5 documented constraints, which must all be 'true', from '**Market Go Decision' to 'Partner Status:** Agent Contract Signed.' If any of these is NOT in force and true, at any moment in the project, or afterwards, then there is no 'less than 3 weeks' Goal' in force.

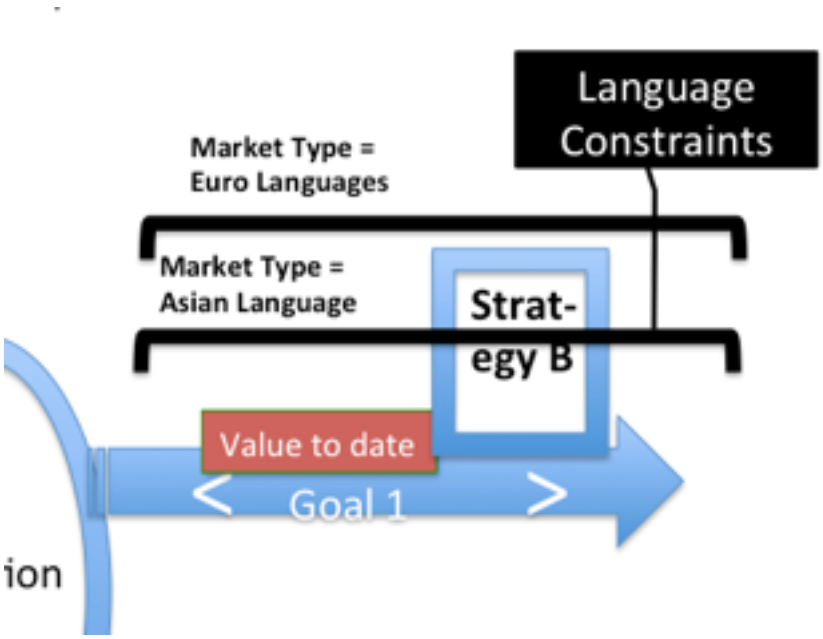


Figure 2.6. Illustrating a Strategy in Violation of one constraint.

Comment on the Figure 2.6:
Violation of a constraint, disqualifies the strategy.
Strategy B satisfied the cost constraints (Figure 2.5). And B gives 'effect enough' to hope to reach our Goal level.

However. There are some other constraints (Planguage Example 2.6). Strategy B satisfies the Constraint 'Euro Languages'. Graphically it does *not* violate the black line symbolizing the constraint.

It does however violate the 'Asian Language' constraint. Shown graphically by the Strategy B rectangle icon exceeding the constraint bar; not being within the constraint.

"If we define optimism broadly as the tendency to maintain a positive outlook, then realistic optimism is the tendency to maintain a positive outlook within the constraints of the available measurable phenomena situated in the physical and social world"

[DeGrandpere, 2000, p. 733]

and

"I think frugality drives innovation, just like other constraints do. One of the only ways to get out of a tight box is to invent your way out." [Jeff Bezos].

Practical Tip 2.6. Deal with constraints up front.

- *Ask yourself if you really have a good list of all constraints, relating to both strategies, and objectives. Probably not.*
- *If you do not deal with them now in planning, they will return the haunt you badly later.*

Policy 2.6 Know your limits.

- **We will not waste time, by planning strategies that violate known constraints.**
- **So, we will know our constraints, and invent ways around them; or try to remove them – which is *also* invention.**

- Why?
  - It saves someone a lot of problems.

SECTION 2.7. Divide big ideas into much smaller value-delivery increments.

Principle 2.7. Varied Value:

Complex strategies
can be decomposed by value.

[URL16]

Strategies can almost always be decomposed into independently implementable sub-strategies. If you know how. If you really want to.

Figure 2.7 A. Almost any 'strategy' can potentially be decomposed into independently implementable sub-strategies.

One decomposition method, of particular interest here, is to decompose the strategies by *value*. This means that every sub-strategy delivers *some value* to *some stakeholders*; and some strategies deliver *far more value* for their cost than others.

This means that we can pick the famous 'low-hanging fruit'.

We can schedule high-value sub-strategies for early delivery. We can even at some point, choke off the supply of sub-strategies, when the estimated values do not outweigh the estimated costs. We can deliver sub-strategies as long as it is profitable to do so. And we can put other strategies 'on hold', until such time as they might 'become' profitable.

This is a very simple and logical concept. And surely, all of us practice it at some level of our life, intuitively, as when eating a meal.

But our observations, of a large number of organizations and of management planners, tell us that:

- Managers and planners are unfortunately not trained (in business school) to decompose by value (they are taught work-breakdown structures, something else)
- Managers and planners do not practice *decomposition by value* in conscious observable ways, with corresponding documentation for their process and thinking (there is no policy to do so).
- Managers and planners make excuses, for why it is *not possible* to divide strategies by value (but the excuses are *invalid*).
- The question is not, if we are going to have a value stream flow: the question is if we know *how* to decompose strategies of all kinds, to actually get the value flow, *early*, *frequently*, and in *small* increments (like weekly).

Quote 2.7 A.
Descartes on decomposition:

"To divide up each of the difficulties
which I examined
into as many parts as possible,
and as seemed requisite
in order that it might be resolved
in the best manner possible."

[Rene Descartes (1596-1650) Discourses on the Method].

20 PRINCIPLES OF HOW TO DECOMPOSE STRATEGIES

In our writings we have summarized 20 teachable principles of decomposition (see Figure 5.6 A), derived from practical application experience. [URL16 A, and [1, URL18]]

The main principles are '***persist'***; and '*use your intelligence and common sense'.*

The other 19 principles follow, by observation, of what we actually do to decompose when we 'persist'.

Most people break through, and find useful decompositions, when they are not allowed to *give up.* They find these decomposition ideas within minutes. It helps if they are *motivated* (by their managers and peers) to find decompositions by value.

Here is a way of *expressing* the planned decompositions:

Strategy A. [Country, City, Product Line, Service Level]

A1 [Country = UK, City = London, Product Line = Magic, Service Level = None] 50% of Planned revenue.

A2 [Country = USA, City = Los Angeles, Product Line = Magic Version 2, Service Level = 24 Hour Help] 30% of Planned revenue.

A3 [Country = Norway, City = Oslo, Product Line = Magic Version 2, Service Level = {24 Hour Help, Norwegian Language}] 15% of Planned revenue.

Planguage Example 2.7. Decomposing objectives using [qualifiers] such as location, product line and service level.

The first line, Strategy A, lists the aspects of decomposition (a **Generic Qualifier**]. The 4 qualifiers (Country….. Service Level) is a first level of decomposition, in itself.

Each qualifier can have many attributes. All useful combinations, similar to A1, A2, and A3 above are potentially useful decompositions.

A simple question helps to understand if we are on the right track. 'Can we do this combination in a week?'

The following 3 examples of decomposition (A1, A2, and A3), select the decomposed sub-strategies, and document them. Together with an estimate of 1 dimension of effect (Planned revenue) each is expected to contribute.

The subtle difference between decomposition to build, and decomposition to delivery value.

We are extremely interested in *these* things, listed below, with each instance of decomposition:

- The ability to actually deliver some real measurable value, however small, in a single value delivery cycle (we think: weekly)
    - This implies that the stakeholders, who are by definition going to receive the value, are somehow actually available to do so.
    - If my suggestion of weekly value delivery cycles sounds a tight squeeze for you; consider Lean Startup [URL29] that reports experimenting with value changes 60 times a day or more! Or consider the 3-day Gulf War I (half a week).
- So that means we have to be able to acquire, construct, test, integrate, and field trial, all within a single (weekly) cycle.
    - If this sounds strange or difficult, we have clients who do this regularly and continuously for many years on end, so we *know* it is practical.
- If you decide to decompose to these small value delivery cycles. If you have management clearly behind you in want it to happen. If you learn the methods in this book to help you. If you have imagination and intelligence. *We have found that it can always be done.*

We are not very interested in decomposition like *this*:

- decomposing to small steps of construction or acquisition, *without* the immediate consequent ability, to deploy to stakeholders, and to measure the benefits delivered, or not delivered.

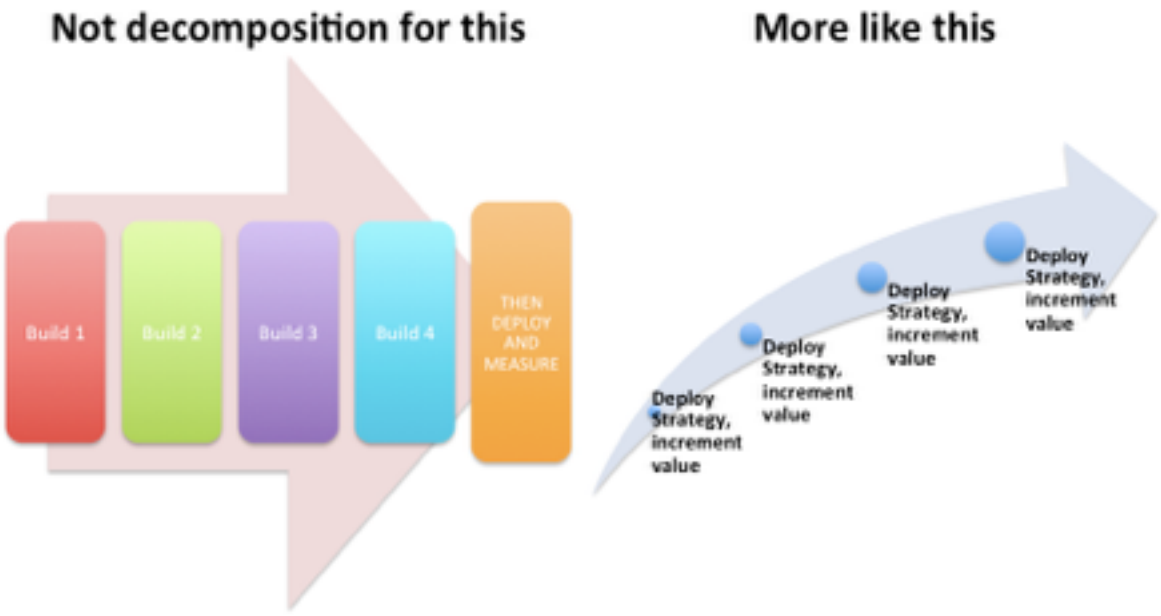


Figure 2.7 B. Two attitudes towards strategy decomposition. 1. Build and see what happens later. 2. Deliver value as you build change: measure progress, learn faster.

Quote 2.7 B. Big Plans into Action Steps.

Have a bias toward action - let's see something happen now. You can break that big plan into small steps and take the first step right away.

Indira Gandhi *(1917-1984) Former Prime Minister of India*

Practical Tip 2.7 WHAT ABOUT DOING SOME THING USEFUL?

- *Just ask yourself and your team: 'What can we do by end of next week, to deliver some value to real stakeholders?"*
  - *And if you get a good answer, why not?*
  - *Repeat this, until nothing profitable to do is left.*

Policy 2.7 Do 1/50th at a go.

Strategies which would cost more than about 2% of critical resources, shall be decomposed into sub-strategies, which are independently implementable, and deliver better-value-than-cost.

Note: the 2% (of time to deadline or of a budget) is a useful guideline, and always possible choice, I have found; even in the early stages of a project (2nd week) [URL4, Startup Week]

Why?

- because it is generally possible
- and you get faster results
- and you can get rid of bad ideas before they damage your project
- that is what ibm FSD used in cleanroom (monthly over 4 years = 2%)
-

SECTION 2.8 Delivering A Stream of Value Improvements to your 'bad old system', is likely to be more successful than building a big new system.

Principle 2.8. PARTIAL DELIVERY.

Early delivery of high-value, and low-cost strategies is normally possible, and normally more successful.

Tom Peters has a lot to say about delivering value in small steps. He has observed 'business success' by doing so. The theme pervades his many books.

Quote 2.8 A. Cumulate Excellence Daily

"The "Excellence Standard" is not about Grand Outcomes.

In Zen-like terms, all we have is today.

If the day's work cannot be assessed as Excellent, then the oceanic overall goal of Excellence has not been advanced. Period."

— Tom Peters (1942-) , The Little Big Things: 163 Ways to Pursue EXCELLENCE, tompeters.com/

PS I could not resist the 'Leaders focus on intangibles' part. Quote 2.8 A

It has amazed us for years, how often we see managers accepting a plan, to solve a problem, a plan which involves *long-term waiting, for years and inevitable delays*, all for '*urgently needed' results*.

The usual scenario is:

- The old system is 'said' to be beyond repair.
- It '*must'* be replaced by a new modern system
- Then, 'everything' will get better
- PS there is often a *major supplier* in the background who will earn a lot of money, by supplying the wonderful new system, and earn even more, if it is 3x over budget, and years late.

We believe that, instead of bad promises and delayed results:

- it will generally be possible to start getting improvements to critical values for critical stakeholders *immediately* (next week and every week)
- That if an older system really *does* need replacement with a 'new' system, then that replacement construction needs to be done *in parallel*, not *instead* of, getting results. In what we call the 'backroom'.
- That the best, fastest, and safest place to start getting, real, and fast, improvements is with the *old system*: *because it is there*.
    - And because real improvements can be sensed *immediately*, as a measurable difference: no excuses about why improvements are not there, or what the improvements are
- That most *investments* in improvement, on the old system, can *generally be transferred* to the 'new system' platform, when it is ready.
- That if management decides to do things this way, they *<u>can</u>*
    - But they might need to *motivate* people
    - They might need to *train* people
    - They might need to bring in *outside* 'imagination' when their own team lacks skills, and imagination, and motivation for doing things 'evolutionarily'
    - It might help if the managers in charge, had *leadership* from their directors, to do things this way!

Strategy Decomposition leads to Sub-strategy Prioritization
We discussed here earlier, the ideas of decomposition of strategies (2.7).

Once they are decomposed, the sub-strategies need to be *evaluated*, regarding their impacts on our objectives, and their costs.

Then, we need to adopt a policy, regarding the incremental sequencing, of delivery of the sub-strategies.

There does not have to be a *single* strategy sequencing policy, and there can be a series of *different* policies; appropriate for the current stages of the project, and for the current external environment. What is important is that managers manage, by adopting clear guidelines, for choice of sub-strategies, to be put into play.

The sub-strategy 'selection policies' should be based on:

- The quantified value-improvements, that can be met, in the direction of planned Goals
- The costs for doing so
- The environment around us, which might not be adequately reflected in objectives, and in formally stated constraints (politics, corporate change, competition, disasters.

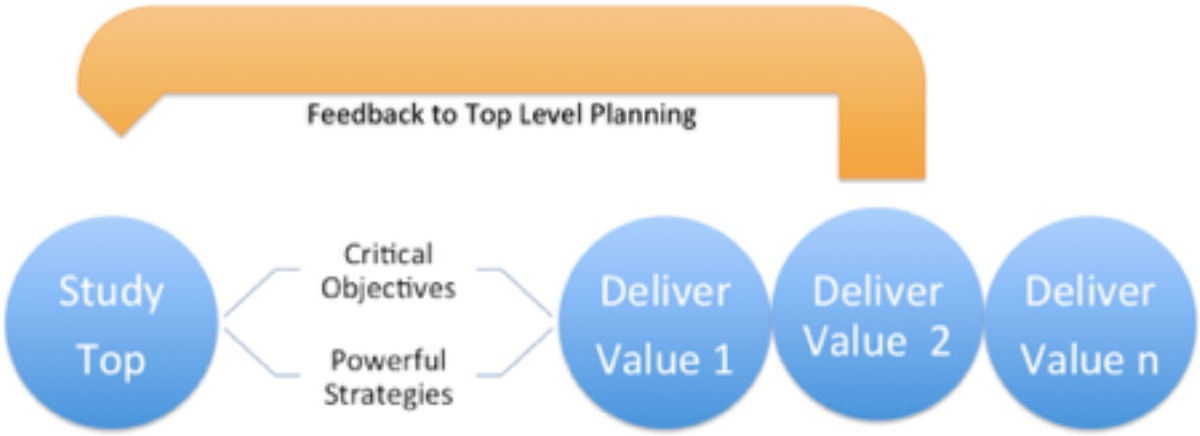
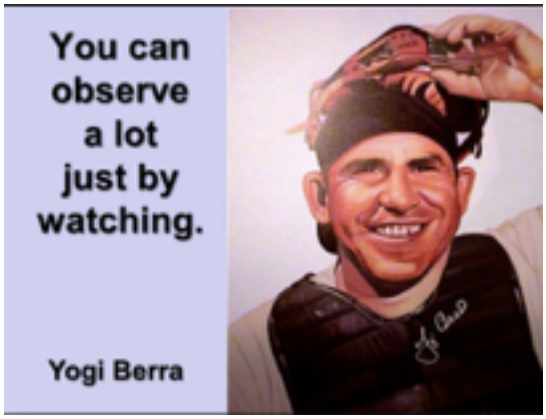


Figure 2.8. After a very brief (1 week [URL5]) determination of top level objectives and strategies, we need to start the value-delivery cycles. But we can learn from these experiences, and feed back facts, to top level planning.

Quote 2.8 B. Act Now, A wait may be too late.

"It gets late early out there"

Yogi Berra, Baseball Philosopher, (1925- (89 in 2014))



Comment: we need to orient our projects to get early results, instead of waiting months and years building systems, that in any case ultimately fail.

Practical Tip 2.8 Just Improve It!

- *What is the most impressive value delivery you can imagine?*
  - *Start trying to deliver some of it in the next few weeks, a week at a time.*
  - *Call it 'pilot studies', but lock-in good results.*

POLICY 2.8 'PLAN, DO, STUDY, ACT ' (method taught by W. E. Deming, 1900-1993)

- We will focus on finding and implementing early and frequent sub-strategies (excuses are *not* acceptable!):
  - To test the strategies
  - To learn what works, what it costs, and how much
  - To change fast, to better ideas
  - To improve team morale and credibility
  - To improve the project economics (self-financing projects?)
  - To train our support teams in total systems thinking, necessary to get real results from real organizations

Why?

To get control over compound systems and complex relationships before they get control over you.

SECTION 2.9 *Strategy* Risks: what can go wrong with those great ideas?

Principle 2.9 Each Strategy has Risks.

There is a set of risks
with each strategy,
that you cannot ignore
safely,
but you can *plan* for them,
for safety.

We are using Risk in this subchapter with the ordinary interpretation 'bad stuff can happen'.

More formally:

'Risk' is the **possibility** of Damage occurring.

Concept 2.9 Risk, slightly more formal definition.

No strategy is risk free. It *might* work as advertised or expected, with luck. But, 'are you feeling lucky today, punk?'.

You cannot *know* it will. You have to assume it *might* not. And *prepare to act* when it does not.

So what preparation can a planner do, and what can management encourage?

What actions can be taken, when bad things happen, or when risks are identified?

Case 2.9 A. Ericsson's Quality Policy.

Ericsson of Sweden had a Quality Policy, when we worked with them for about 10 years (1990s).

It said that "**risk management was the responsibility of every engineer**." [URL39, URL20]

That means that 'risk management' applied to **all planners** in Ericsson.

Even 'managers' are engineers in Ericsson.

We liked that, and believe it is a good general principle.

Risk management is not safely delegable to a 'specialist in risks'.

The reason that risk is not 'delegable' is that risk is not simple, specialized, or limited.

Risk – of negative outcomes, in relation to our objectives – is pervasive, complex and involves mixed disciplines and specialities.

Risk is everywhere, in every little detail the planner might encounter. Literally in every word, every number, every communication.

Maybe we need to be 'risk psychopathic'? Not 'risk adverse'.

Case 2.9 B Teaching Game at Ericsson.

We had a little game we played with the Ericsson engineers, who Tom and Kai were training in planning.

We put up a one-page, 20-element detailed 'objective' or 'strategy' on a screen, and asked them to identify *anything* related to risks, that they could see there.

Maybe you guessed our game ?

Every tag, every parameter, every value, every background specification, in Planguage, was *somehow* related to risk.

Everything, if missing or corrupted, could arguably lead to something going wrong.

Everything in Planguage was there for a good reason: to defend us against risks.

We do not believe you can eliminate risk entirely. In fact, in a competitive environment, it is *necessary* to take risks, to win [Quote 2.9 B, below].

We do believe in being *conscious* of risks, and *managing* them, so they do not become nightmares, or nightmares too late.

Here are some planning structures that help you think about risks: there is much more, later, in this book:

Strategy A. [Country, City, Target, Product Line, Service Level]

A1 [Country = UK, City = London, Product Line = Magic, Service Level = None] 50% of Planned revenue.

A2 [Country = USA, City = Los Angeles, Product Line = Magic Version 2, Service Level = 24 Hour Help] 30% of Planned revenue.

A3 [Country = Norway, City = Oslo, Product Line = Magic Version 2, Service Level = {24 Hour Help, Norwegian Language}] 15%±5% ?? of Planned revenue. **<- German Sales Planner**

Planguage Example 2.9. Some ways of expressing different risk environments. Can you spot the risk management components? See below for some answers.

Comments on the example.

- The *decomposition of Strategy A* into A1, A2, A3 helps define a realistic and useful definition, or 'subsets' of Strategy A. We do not risk understanding that any *other* options are included, or planned, yet.
- These could have been intentionally decomposed into high risk and lower risk sets. A3 shows signs of being a bit 'special'.
- The set of Generic Qualifiers (**Country, City** etc.) *limit* the scope of consideration, clearly. But also permit us to ask which valid combinations we have *not planned* at all. Yet. Omission risk.
- The **15%±5%** (A3 statement) reduces the chance that anyone will expect, or assume, 15% exactly. We also announce that there is a risk the reality will be in the area 10% to 20%.
- **<- German Sales Planner** (last phrase). Informs us that the Norway plan was estimated by a German, and a Sales Planner. This is a warning that there may be a risk of irrelevant competence
- **The ??** is a clear warning that the estimate is not to be taken seriously. There is a risk it is very wrong.
- **'N**orwegian **L**anguage**':** the capital letters '**N'** '**L'** are a signal that this is a 'formally defined' term, somewhere. If it is not in fact formally and properly defined, there is a risk that the specification will be misunderstood. *Hint, there are at least 4 'official' Norwegian languages (Bokmål, Nynorsk, Sami, Kven (never heard of it either!))*
- The Statement **Tags** (Strategy A & A1 & A2 & A3) permit us to have one single tagged 'master' planning element, independent of updates, avoiding the confusion of multiple versions, in multiple plans and presentations. All plans must refer to these tags, rather than, dangerously cutting and pasting. Updating the master plan element, updates all references to it simultaneously.
    - We find it amazing how little formal 'tagging' conventional planners do, in their plans; and how little co-ordination there is of various versions of the 'truth'.
    - They are doomed to be misunderstood.

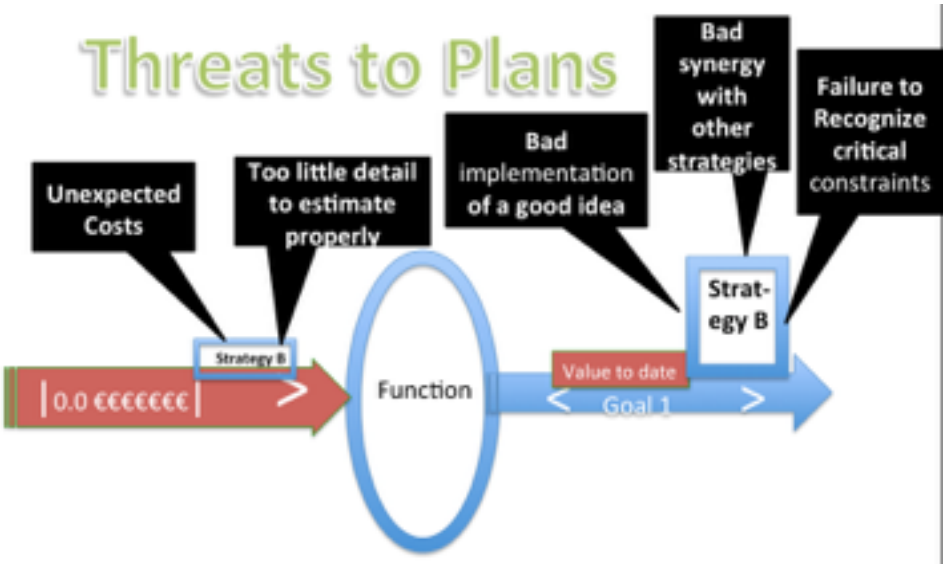


Figure 2.9. Some few, of many, threats to the values and costs, estimated and expected, for a strategy.

The Risk picture: a structured view: Threats, Attacks, Damage.
'Threats', are specific things that can happen *potentially*, and when they really do occur, they 'attack' something, and cause 'damage' [Section 7.2 for more].

The attacks in planning, are a threat to our *expectations* of the benefits and costs of using a particular strategy. The real experienced benefits can be much worse than expectations, and so can the real costs. The 'damage' is 'lower benefits' and 'higher costs'.

Consequently, in order to improve our ability to make plans that 'deliver as expected':

- We need to identify threats to a reasonably correct *evaluation* (estimate) of a strategy's impacts
    - *Example*: by having systematic data about such strategies, from experience, the internet or any credible sources
- We need to reduce the potential for those threats to become effective attacks (like giving us a bad estimate, from seemingly reasonable data)
    - *Example*: by investing in specifying strategies, so we give enough detail, so that estimates are based on better descriptions
- We need to consider how we can reduce the damage done, when such attacks succeed (we got a bad estimate, anyway)
    - *Example*: by getting feedback early from reality.
        - so we can adjust or replace a costly strategy

Quote 2.9 B. Via Peter Drucker.

"One of the most rigorous theorems of economics (Eugen von Boehm-Bawerk's Law, 1851-1914) proves that the existing means of production will yield greater economic performance only through greater uncertainty, that is, through greater risk.

While it is futile to try to eliminate risk, and questionable to try to minimize it, it is essential that the risks taken be the right risks.

We must be able to choose rationally among risk-taking courses of action, rather than plunge into uncertainty on the basis of hunch, hearsay, or incomplete experience, no matter how meticulously quantified."

Eugen von Boehm-Bawerk, Austrian economist (1851-1914) [F17]
This is Peter Drucker, interpreting the meaning of Boehm-Bawerk's theorems.

Practical Tip 2.9 Risk Finding Competition

- *For each strategy, arrange a little 'competition', amongst co-planners and managers. Like this one:*
- *Independently, in 5 minutes, let each individual write down at list of **at least 5 risks with the strategy**.*
- *Tell them that the 'more risks they find', the 'deeper insight' or 'knowledge' it shows that they have, about the strategy.*
- *Later, edit these ideas into the Background specification for that strategy. Credit the individual who found the risks, in the specification. (Risk: XXX <- John and Alexia)*
    - *Create a permanent 'memory' of the risks.*
- *Next game: suggest a mitigation for each risk.*

"If you don't actively attack the risks, they will attack you".

(T. Gilb 1988 [5] p. 73. A Principle)

POLICY 2.9: Fight Risk Intelligently And Constantly

- Risks of not reaching objectives, within constraints, are many and complex.
- **But all planners need to identify risks, prevent risks, discover risks early, and mitigate risks.**
- This is a constant battle, and a main purpose of competitive value planning.

Why?

- avoid late-in-process, and specialized-risk-people risk analysis.
- deal with the very many small elements of planning, that have risk elements, immediately.

SECTION 2.10 Ends and Means distinction.

Principle 2.10. Be careful what you ask for.

**Don't mix up your objectives with your strategies,
you might end up getting *what you asked for*,
not what you *really want*.**

Ends and Means Defined:
'Ends' is the general name for any future thing we want to occur.

'Means' is the general name for any method we might use, to help us get all or part of the way, to our 'ends'.

In planning, we have many names for '**ends'**:

- Objectives
- Goals
- Requirements
- Targets
- Design Constraints
- and many more.

And we have many names for different types of '**means'** to get to those ends, like:

- Strategies
- Tactics
- Supporting Objectives
- Methods
- Designs
- Architectures
- Solutions
- Resources
- and many more.

Your 'objectives' are your *main focus*. Your corresponding strategies for reaching these objectives, are NOT your main need or desire or focus.

The strategies can be exchanged for better strategies, without changing your objectives.

You should not fall in the common trap of 'fixating yourself, on any currently-adopted strategies'. You should not *mislead* your supporting team into believing that those *strategies* are what you *really* want.

If the strategies are not provably working well; dump them. If you identify much smarter strategies, dump the old ones. If you do not dump inferior strategies this consistently, you cannot imagine you are serious about being competitive. Think like a top athlete!

Keep your 'flexibility to dump strategies rapidly', 'without undue pain or costs'. That means, don't write contacts with high costs of cancellation!

That 'dump' flexibility is something you need to plan consciously.

Don't lock yourself in by contract. Don't let outsiders take over your core competencies, competencies which cannot easily be replaced by other suppliers.

Keep reminding your support team that the strategies are NOT the point, the *objectives* are, their results are!

Avoid putting your *strategies* under any plan 'Heading', like 'Main Objectives'. That can mislead people into focussing on the strategies *themselves*.

One Man's Meat is Another Man's Poison

One point we shall discuss more in depth later, is the fact that all this is from '*your* point of view' only. Or more generally, it is defined from the point of view of 'a defined stakeholder' (example, you).

Your 'objectives' are regarded as *their* strategy by your 'boss'.

Your 'strategies' are certainly someone else's 'ends'. For example your supplier's.

Main point:

Ends and means are defined,
only with relation to a stakeholder,
who wants
***those* ends,**
and plans those means
to get to their ends.

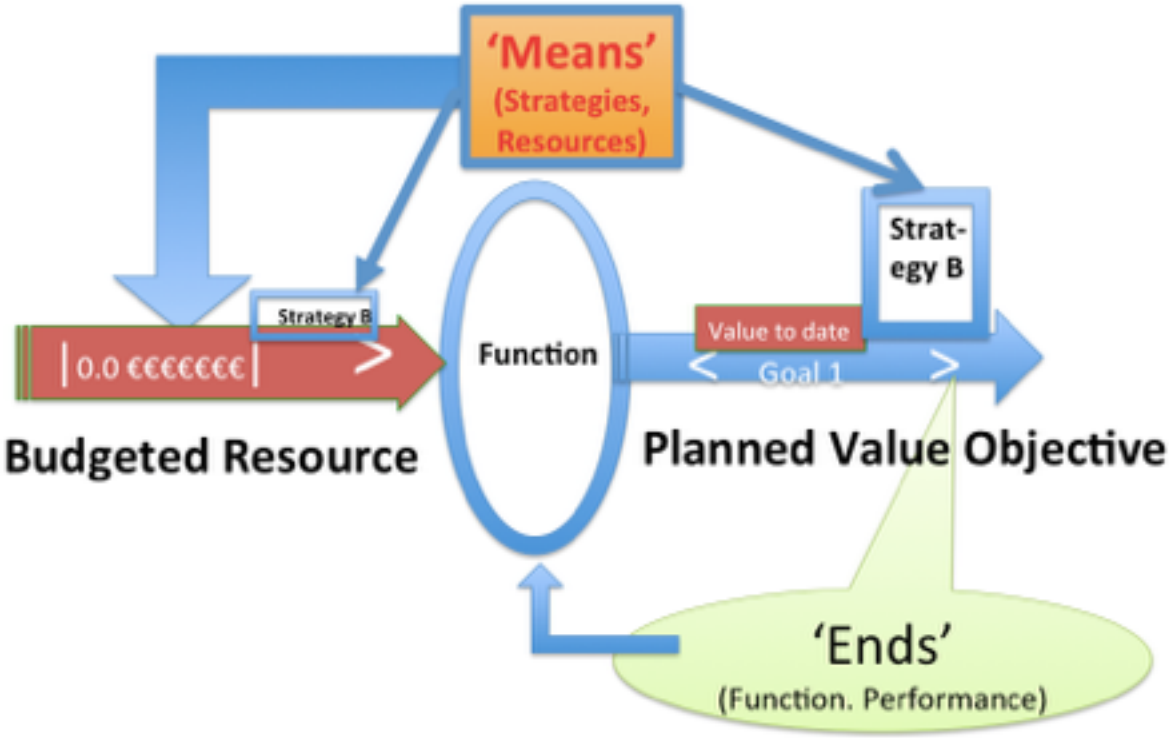


Figure 2.10. A graphical difference, between ends and means. Using Planguage Icons, Ends are *what you want*. Means are *ways to get what you want, at some costs*. There are two distinct types of ends. Functions, and Performance attributes.

Avoid mixing strategies in statements of objectives; like:

"We are going to do Strategy A, ***in order to*** *enhance Strategy B,* ***so we improve*** *Objective X."*

Link Words Are Strong Hints about Mixing Up

The phrases 'in order to', 'so as to', 'for' , 'so we improve'; are 'Link Words'.

Link words are a sure sign that you are mixing up your own ends and means.

Of course they are *potentially* related. But, if you put them in the same sentence, like that, they become 'too intimate'. Prematurely committed.

They will *seem* like the 'real objective', when in fact they are NOT.

People will ultimately deliver the ***strategy*** to you, but will probably ***not*** deliver your 'real objective'. That then, is *your own* 'planning' fault.

Case 2.10. Real Bank Project Top-Level Objectives; measured and rewritten in an hour.

Here is a real top-level set of objectives, for a major project (about £30 million). This was what the responsible director gave us as his top-level objectives. How many strategies can you find, using link words as a clue?

• "Rationalize into a smaller number of core processing platforms. This cuts technology spend on duplicate platforms, and creates the opportunity for operational saves. Expected 60%-80% reduction in processing cost to Fixed Income Business levies.
• International Securities on one platform, Fixed Income and Equities (Institutional and PB).
• Global Processing consistency with single Operations In-Tray and associated workflow.
• Consistent financial processing on one Accounting engine, feeding a single sub-ledger across products.
• First step towards evolution of "Big Ideas" for Securities.
• Improved development environment, leading to increased capacity to enhance functionality in future.
• Removes duplicative spend on two back office platforms in support of mandatory message changes, etc."

Example 2.10. Real Example of Bank System Objectives.

The Programme Director himself identified about 6 strategies, and 9 unintelligible ideas, in about 15 minutes, when we asked him to do so. He also volunteered to us, in writing, that this was an 'unacceptable' practice. But, we all knew it was 'the culture' everywhere in his organization.

He challenged us, to show how *any of this* could be *expressed more clearly*, and how objectives could be *separated from* the strategies.

"Your choice", we replied.
And he picked "*Improved development environment".*

So, in about the next 20 minutes, rounding out our first hour's meeting, we drafted *this:* (the 'XX' notation is used to disguise identity):

Development Capacity:
Version: 3 Sept 2009 16:26
Type: Main Objective for a project.
Ambition Level: radically increase the capacity for developers to do defined tasks. <- TsG

Scale: the Calendar Time for defined [**Developers**] to Successfully carry out defined [**Tasks**].

Owner: Tim Fxxx
Calendar Time: defined as: full working days within the start to delivery time frame.
Past [2009, {Bxx, Lxx, Gxx}, If QA Approved Processes used, Developer = Architect, Task = Draft Architecture] 15 days ±4 ?? <- Rob

Goal [2011, { Bxx, Lxx, Gxx }, If QA Approved Processes used, Developer = Architect, Task = Draft Architecture] 1.5 days ± 0.4 ?? <- Rob

Justification: Really good architects are very scarce so we need to optimize their use.

Risks: we might use effort that should be directed to really high volume or even more critical areas (like Main Objective).

Planguage Example 2.10. Restatement of "Improved Development Environment".

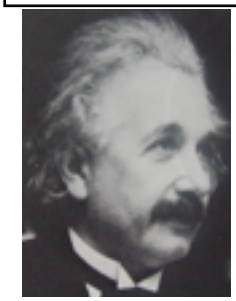
The initially-included 'strategies' are *gone*, for now, and the focus is back on the *real objectives* of the director. He helped us work this out.

We do not have to 'throw away' the suggested strategies. We can keep them in a 'list of ideas to be evaluated'. When we have much clearer objectives.

But there is nothing, no clear Goal, to evaluate strategies *against*, as long as the objectives *themselves* are still fuzzy thinking.

Quote 2.10. This ends/means confusion seems to be an old problem.

"Perfection of means,
and
confusion of ends,
seems to characterize our age"



Albert Einstein (1879-1955).

Practical Tip 2.10: Planning Jujitsu

- *When people suggest, as an 'objective', something you suspect is really a 'strategy' or a 'means'; ask them 'Why?'*
- *Their answer is probably closer to the real objective. Document that. Get their agreement that it **is** their real objective (or ask another 'why?')*
- *Then put the 'strategy' in a list entitled, 'Suggested Strategies'.*
- *Thank them for their strategy suggestions, and for their insights into the real objectives.*
- *And move on.*

Policy 2.10. Don't let the 'means' become 'ends'

- **We will clearly *separate* our level of objectives, and the *always-temporary* strategies. We will *never* let a strategy get misunderstood as our *real* objective.**
    - Strategies mixed, or hidden, in objectives is a 'major planning defect'.
- **Why?**
    - If you do *not* separate ends and means, then you will get unexpectedly bad values and unexpectedly bad costs; and you will falsely blame this on the 'Complicated' system.
    - The reality is that your bad 'planning specifications' made it impossible to estimate the result. You got the result that happened, from the strategy; and from any misunderstanding of even what the strategy really was.
    - If you want 99.00 % of a defined value. Say so, and persist in trying different strategies until you get 99.00%. This is 'Dynamic Design To Cost' [URL12, URL11]: and it is far more 'predictable, and not complicated at all.

Chapter 3. Levels of Interest and Levels of Control.

Keeney… URL19 and [3]

Your plan is certainly related to other levels of planning:

- Serving higher interests directly (your boss)
- Serving higher interests indirectly and ultimately, through your boss
- As part of parallel efforts to serve higher level interests at the same time
- Your plan is at level that one or more other planning efforts are trying to serve: your suppliers and subordinates.

This chapter has ideas about how to be a good team player with all of these related planning and operational interests.

it is about ‘alignment' of planning.

Section 3.1 The Plan Relationships: your links to others.

Principle 3.1 Lone Link.

**You are only one level,
in a team hierarchy.**

You are *one link* in a chain of people and projects, that you are serving, and below you, there is a link your own support team.

Everybody is in the same position; they are but one link in the chain. And the number of related links to your single link is probably greater than 2.

It is useful to be clear about exactly *where* you stand (the standpoint being, any project or activity you chose to focus on), and who your nearest relations are (up and down links). Being *you* is probably challenging enough alone; what you need to *avoid* is:

- Taking over your bosses responsibilities
- Interfering with your support team (subordinates)
- Having someone else think they own *your* responsibilities

So, you need to create clarity, hopefully agreement, with the other links. Who is *responsible* for what. At least you yourself should not be confused, and that is all too easy to do.

When specifying a plan, we offer a number of devices available to help make these distinctions.

For example:

Market Flexibility:
Type: Marketing Director Objective.
Supports: Corporate Profitability.
Stakeholders: Marketing, Production, Distribution.
Potential Strategies: Greater Supplier Flexibility, Greater Distributor Flexibility.
Spec Owner: VP Marketing.
Expert: Marketing Consultant Jane X.
Constraints: EU legal considerations.
Ambition: 10 x faster market adjustment **<- CEO.**

Scale: The average calendar time needed to make a defined [**Adjustment**] in a defined [**Market**], for a defined [**Product**].

Past [General average] 4 weeks **<- Expert.**

Goal [By = End Next Year, Adjustment = **Additional Distributors**, Market = Asia, Product = Toys] less than 1 week. **<- Mkt Dir.**

Risk: EU and specific Asian import countries regulations for safety, might be vastly different. **<- Legal Department.**

Planguage Example 3.1. Examples of specification devices for documenting relationships to one single objective. How many relationship specifications can you count in this example? I find about 20.

Every statement here gives some insight into the players you are serving, who are on your core team (*your* link), and supporting you.

In Planguage you can specify *any useful* relationship information, at any time in the planning process. It costs very little, but it clarifies relationships for yourself, and for others. This can activate the relevant people, and keep you from treading on their toes.

Figure 3.1 A. Some basic classes of specification in planning. 'Relationship Background' is a large set of links to other people and things.

Most of the information in the Example 3.1 above is not the 'core objective' itself. Most specification statements are 'background' information: or, if you like, 'linking' information. Background specification in a plan costs little, but its absence can easily end up costing very much more.

Figure 3.1 B. Some common relationship parameters in a specification.

So, we believe it is good insurance to invest in capturing this knowledge in writing, rather than leaving it as tacit or implicit knowledge.

One person, like yourself, might know most all of this relationship information: probably not ALL of it. So a cooperative effort is needed to build it up, maintain it, and make it available to all useful participants.

Quote 3.1 ***On Responsibility for your Place in the Chain***

> ***"What we try to say is that it does not matter (what political party you vote for). You are equally responsible for your place in the culture, and you must make a contribution, and you must accept responsibility, for what goes down on your watch."***

(Martin Sheen, Actor, AKA President Bartlett)

Practical Tip 3.1 Develop a Stakeholder List

- *Ask all planners to brainstorm a list of at least 10 stakeholders for a given objective, or strategy.*
- ***Stakeholder**: any person, group, or thing (law, policy, standard) that might be impacted by this planning object.*
- *Combine the efforts to one list.*
- *Specify the top 5 as Major Stakeholders*
- *And the others as Minor Stakeholders*
- *This will increase awareness of many considerations, risks, problems with the planning object.*
- *The Spec 'Owner' (empowered to change the spec) should follow up. Develop assumptions, stakeholder needs, risks, issues.*
- *Some of these stakeholders can usefully be highlighted as a separate category of relationship (Owner, Expert, Contract), not just a generic 'Stakeholder'.*

Policy 3.1 Map Important Relationships

- **We will map**
  - **all potentially useful and critical relationships between**
  - **a planning specification (objectives, strategies, action plans)**
  - **and the related stakeholders;**
  - **including inanimate stakeholders like laws and policies.**
- Why?
  - Because it is cheaper and faster to deal with potential problems early, than after you have made major irrevocable investments.
  - If you fail to map relationships, and consequently to deal with their needs, you will consider your system to be 'Complicated' because you will get so many unpredicted results, and emergent-property surprises.
  - But perhaps much of your misled interpretation of 'complicatedness', is simply because you have not taken the trouble to map the many relationships in advance, and planned to deal with them.
  - This is called 'bad planning', and it can make a system seem more 'complicated' than it really is.

Section 3.2 'Who' decides , or 'how' do we decide, on which strategies to use?

Principle 3.2 Stick To Your Knitting.

Leaders and managers
should not suggest or decide strategies.
They should keep focus on objectives:
and leave strategy to their staff,
and to measured proof, of strategy effectiveness.

We know that you, as a manager, are just trying to be 'helpful' by suggesting strategies. And maybe use your hard-earned wisdom. And maybe hint that you are up to date on new ideas. And maybe to be be a 'team player'. And maybe to get out of your management silo.

But, stick to your knitting (to 'result leadership'). Don't suggest specific strategies, unless asked. Do not misuse your authority, and then maybe have to 'take the blame' for 'your' ideas. Take credit for allowing other people to come up with ideas that deliver your results. Take credit for delivering results to your boss, without *that* person's interference. Golden rule?

Case 3.2 Brüel & Kjær

Once when working for a large successful multinational Danish electronics company, the founders of which had worked personally with Hewlett and Packard in the earliest day in the US, I told my client that I was sure I could get Mr. Kjær, the CEO to formally improve my strategy suggestions.

I had my meeting, but I did not get the approval I was so certain of. Mr. Kjær politely dismissed me with a smile, saying "We'll see you all at the product launch".

I was at first disappointed and confused. But this was in fact the same deep company culture that had made them successful.

And the same as at HP.

I should have taken my cue from the photo on his wall of the two B&K founders riding in an open car, in the 1930s with Bill and Dave (H & P).

The CEO refused to take part in our technical decisions. He trusted us, and gave us the responsibility for figuring out what was necessary and what worked successfully, 'by the product launch'.

Why ?

- People are strongly motivated by their *own* ideas [6]
- People resent unasked-for advice
- They might be tempted to use your idea, and make you responsible for failure
- Hopefully you are using experts who know better than you, or have more time to figure out what works best.
- This is a distraction from the results you really want to get. Keep focus on those results, do not stray from your path!

Quote 3.2 A. Keep at it!

Leonardo da Vinci (1452-1519) on Persistence:

"I do not depart from my furrow.
Obstacles do not bend me.
Every obstacle is destroyed through rigor" [7., p.79]

Lean Production:
Ambition: near zero goods in warehouse. **<- CTO**
Result Responsibility: Production Director.
Strategy Support: The Lean Manufacturing Corp. Consultants.

Planguage Example 3.2 A. Responsibility Specification.

Specifications, like the example above, make it clear *who* is responsible for the results, and *who* for the strategy.

Here is an example of how to direct subordinates gently, when absolutely necessary.

Available Domain Knowledge Growth:
Type: CTO Level Corporate Objective.
Scale: Annual frequency of access to our Domain Database % improvement over previous year..
Goal [Annually] 10% or more, than previous year.

Tolerable [Annually] 5% or more, than previous year.

Constraints:
C1: access must be for real internal needs.
C2: domain knowledge must be produced by our corporation, or competitors.
C3: all outside knowledge must be legally acquired, and usable.

Justification: Corporate Transparency Policy.
Source: Legal Dept.

Planguage Example 3.2 B. Gentle guidance using specified constraints, rather than explicit choice of strategy. Notice 'Tolerable' and 'Justification' are also a form of constraint: meaning 'restrictions on choice of strategy'.

The above example shows how to specify some constraints on the selection of a strategy, without directly determining what that strategy will be.

It 'frames' or 'fences' the solution. It would be useful if all constraints had a corresponding Justification, and specified source (responsibility, audit-ability, review-ability).

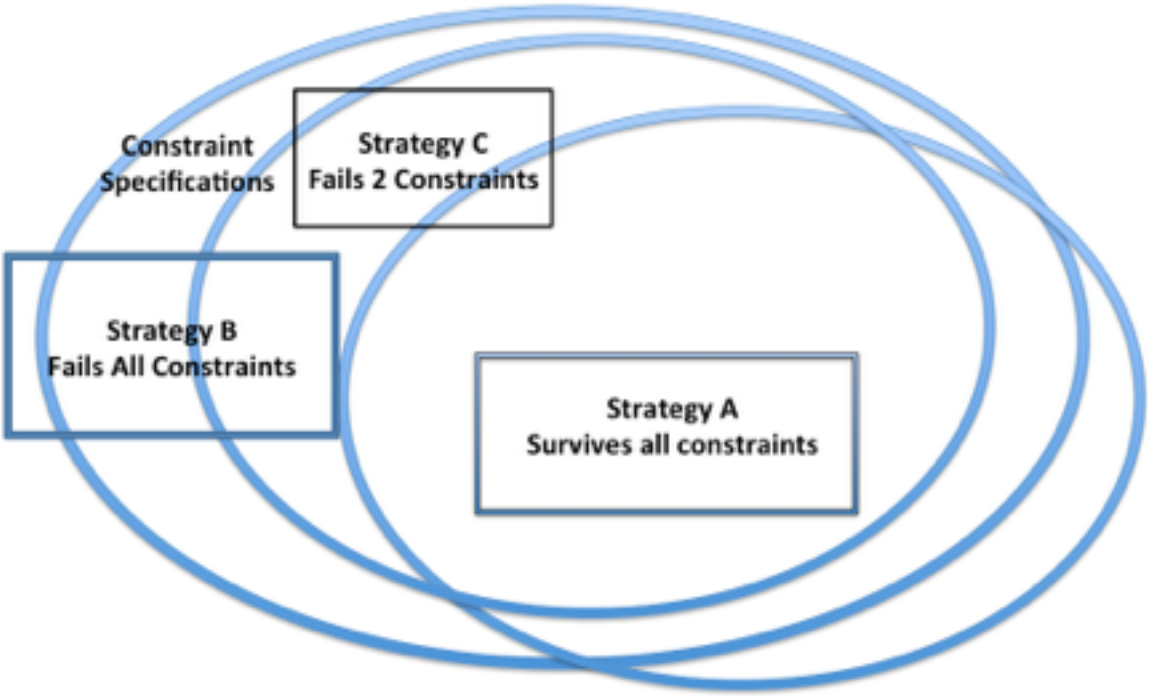


Figure 3.2. Strategies must not fall foul of specified constraints. And important constraints need to be formally specified. Formally specified constraints are cited here, as a tool for managers, to nudge strategy choice, in relevant directions, without actually dictating the exact strategy.

Quote 3.2 B. Constraints Free You.

"The more constraints one imposes, the more one frees one's self.

And the arbitrariness of the constraint serves only to obtain precision of execution."

Igor Stravinsky. *Composer, Russian, (1882-1971)*

Practical Tip 3.2: The Mafia Offer (Money or Life).

- If you do catch anyone, manager or not, sneaking their strategy suggestion into the 'objectives': try the following:

1. Ask politely 'which objective the strategy supports?' (maybe it is already stated with a link term: 'Do S **to get** O')

2, ask then: 'If you could obtain the Objective, *without* using that strategy: is that OK, or would you prefer to do the strategy, *even if* you do not achieve the objective?

- This will clarify what is really important, and perhaps they will 'back off' of insisting on the strategy.

Policy 3.2 Let strategy designers make *their own* decisions.

- **The design and selection of a strategy is, at all times, primarily the responsibility of the group assigned to finding and implementing the strategy.**
- **They cannot be unnecessarily limited, or directed, in their choice of strategy, by the instances that need the results.**
  - The exception is when the result-level management defines a specific constraint on the choice of strategy, as one of *their* requirements, and takes clear formal written responsibility for that decision.

Why?

- to allow people nearer the coal face to make the strategy decisions
- based on feedback early and frequent, about strategy impacts

Section 3.3 Focus on Improvements.

Principle 3.3 Results Speak Better Than Intentions.

Judge others
on how well they support
***your* objectives.**

Keep your focus, for your supporting teams, on the *measurable* results **they** actually produce.

- Avoid discussions on *their* direct strategies (your 'means objectives'): that is *their* business
- Ask if there is anything, at all, that keeps them from meeting your Goal level on time? Remove their barriers.
- Ask them to *document* any constraints they experience, on their ability to produce results, for future discussion. The value of a constraint may be less than the value they can produce without it.
- Challenge them to use their imagination, creativity, wisdom, intelligence, and all possible sources to find the smartest and most efficient strategies.
- Ask yourself if all manner of rewards and recognition, for your support team, are aligned with *achievement* of the objectives. Even simple *recognition* is a powerful motivation [6]

Case 3.3 A European Aircraft Control System.

We examined a sample of real plans for real projects, using Spec QC [URL23]. The number of major defects found by them, using our process, was, as expected, far too high. So we also looked at how to re-specify in Planguage to make it more intelligible.

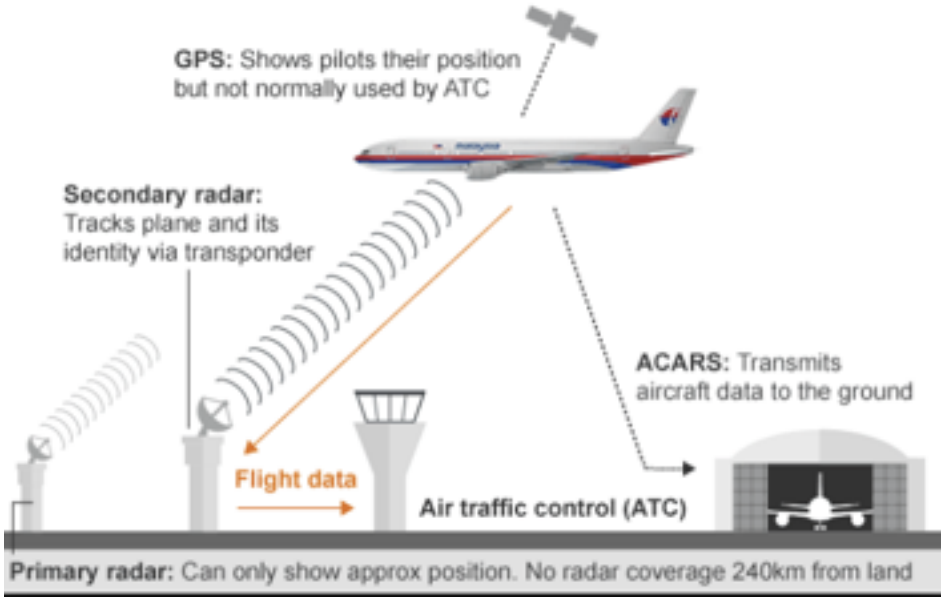


Figure 3.3 A. for Case 3.3 Flight Control Elements. (Source: BBC, MH370 Mystery)

Here is the initial statement or objectives, which we identified (underlined text) as containing too many major defects:

"The system SHALL also be able to filter unwanted transmission types (for example Mode A/C or certain downlink formats reports)

to prevent

target output rates becoming excessive for Data Fusion system or other Data Links"

Notice the first half "system Shall ….reports)" is a *strategy*. One clue is that the *link term* **"to prevent".** So they already lost focus on their objective. It seems as though the *strategy* is 'required'.

On the second part *"Target output rates becoming excessive…..Links",* there is no definition of 'target output rates' (no Scale), and there is no definition of 'excessive' (a Tolerable level).

So while the objective is 'identified', it certainly is not 'clarified'. In fact I suspect that the strategy itself, is the *only* 'clarification', but even that, is quite unclear: *"be able to filter unwanted..". How much filtering? What is 'unwanted'.* Not defined.

The scary thing here is that your 'life', while on an airplane, might depend on this type of specification, and such 'fuzzy' specification is this bad, in the *entire project (we took a random sample)*, the entire organization (we looked at a variety of projects), and all other such organizations we have seen in Europe (2 others actually, one Euro wide).

That is just the way things are; our very poor planning culture. I am not even sure there is a serious will to improve things at the management level. Life goes on as before. No planning quality leadership. No academic training in these things.

Here is a beginning of a rewrite, of the above specification, in Planguage:

Output Rates:
Type: Quality Requirement.

Ambition Level: "prevent target output rates becoming excessive for Data Fusion system or other Data Links" <- 2. . 2 Project nr A0213XXX Bxxxxx.

Scale: < Data Messages of defined [Transmission Type] per Second ? > for defined sets of [Targets] for defined [Data Links: default = Data Fusion System].

Past [Bxxxx Airport, 2012, Transmission Type = Target Position, Target = Airline, Data Link = Data Fusion System] XX Mess/Sec.

Tolerable [Bxxxx Airport, 2012, 2015, Transmission Type = Target Position, Target = Airline, Data Link = Data Fusion System] XX/2 ?? Mess/Sec.

Planguage Example 3.3, for Case 3.3.

The expression in fuzzy brackets (‘< Data …… Second?>’) means we were just drafting a better ‘Scale’ definition, but that we did not yet have enough time to pin down a definition that all parties agreed on.

But we are getting a draft of the format, and people can see there is a much better way to plan life critical systems.

------ End of *Case* 3.3 ----------

Focus on your level of objectives
Keep your focus, regarding *your own* strategic objectives, on whether they are still well-tuned, and appropriate for delivering results, to your next level, to your stakeholders.

It isn’t just about your ‘boss’: it is about the critical stakeholders – and there can be many.

- Are you providing an early frequent stream of results to your stakeholder’s fundamental objectives?
- What are your excuses? Remove the roadblocks, don’t make excuses. Find ways.
- Point out to your objective-related stakeholders, your perceived constraints for delivery of their objectives; suggesting that if they can supply more resources, remove unnecessary constraints, or low value constraints; you can deliver their values better.

Quote 3.3 A

“You can't grow long-term if you can't eat short-term.

Anybody can manage short. Anybody can manage long.

Balancing those two things is what management is.”

Jack Welch CEO GE (1935-)

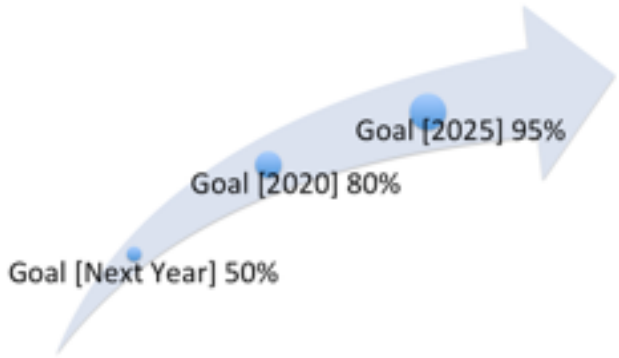


Figure 3.3 B. Any one objective can have any useful number of Goals, in any time period.

Here is an example of specifying *one* objective, *both short-term and long-term.*

Legal Creativity:
Type: CTO major objective.
Ambition: maximize useful patents, and inventions.

Scale: the annual rate of patent pending or patented technology, in defined Areas.

Past [Area = All, to **2013**] ~200.

Tolerable [Area = All, **By End 2020**] 250.

Goal [Area = Human Interfaces, Wearable, **End 2025**] 150.

Planguage Example 3.3 Setting both short term and long term goals: 'Balance'.

In the specification example above, we can see an attempt to 'balance' the short-term and the long-term objectives.

Quote 3.3 B. Focus on real results, not seductive ideas.

"However beautiful the strategy, you should occasionally look at the results.

Sir Winston Churchill (1874 - 1965)

Churchill, recommending at least 2 strategies. None of them 'Surrender'.

Practical Tip 3.3 How to spot a false strategy.

- If the so-called objective does not answer the question: 'How much better do we plan to be', then you know it is not a *real* objective, at least not a *usefully clear* objective.

Policy 3.3 Focus on the Numbers.

- **We will keep our focus on delivered-to-stakeholder numeric results,**
  - or at least early leading indicators of progress,
  - not 'work done' !
  - But, *improvements* made.
- We want to see the *measurement data* for improvements.
- We would like to help people to *remove barriers* to their progress.

- Why?
  - So, we can get bragging rights to our boss, and praise and reward our staff and suppliers for doing good work.
  - So, we can go home on weekends and not worry about work.

Section 3.4 Aligning Levels of Responsibility Numerically for Values and Costs.

Principle 3.4 Numbers Nurture.

You *can* both
***estimate* and *measure*,**
your effect on your own value delivery level,
and your support team's effect
on your *own* goals.

What you expect from *others*, regarding measurement and facts, should be the same measurement, and fact sharing effort, you are committed to do for your own responsibilities.

"Do unto others, as you would have them do unto you."

Bible, Luke 6.31

Figure 3.4 A. We can classify all related planning objectives, in relation to our own level of objectives. The arrows flow 'downstream'.

Fundamental Objectives
The objectives above us, that we are directly, or indirectly, working to support, with our Strategic Objectives', we call 'Fundamental Objectives'.

They are the fundamental basis for *our* planning. They are basic inputs to our planning. We plan to help meet *their* Goals, by meeting our own Goals..

We need to try to set our ('our own level personal') strategic objectives, so that our objectives will help meet the 'downstream' fundamental objectives.

We cannot set objectives subjectively to our taste or whim. Our objectives need to be judged, and reviewed, in relation to the fundamental objectives we take, or should take, *some* responsibility for.

We are quite probably only one of several contributors to the downstream result (Fig. 3.4B). But we need to analyze and clarify the degree of value which *we* are expected to contribute.

If we are ambitious, we will bid to contribute as much value as possible, and *then* we will meet or exceed those value-delivery expectations.

If the Fundamental (Downstream, Corporate, your 'boss') Objectives change, then it follows that we need to consider changing our corresponding supporting objectives. It is also evident that our goals, and any changes need to be *quantified*, so that we can sense changes, downstream, that affect us.

Non quantified objectives are non sense.

<- Tom Gilb
I like this statement so much that I had to be sure of getting credit for it!
You do get the double joke (sense)?

Means Objectives: Our 'upstream' support team.
Any objectives set by any stakeholder supporting us, with the intent, or at least 'obligation' to help us reach our Strategic Objectives, are classed as **'Means Objectives'.** They are the 'means' to *our* 'ends'.

From *their* point of view, upstream from us, *our* Objectives, are *their* 'Fundamental Objectives'. *They* are supporting *us*.

Consequently, **we** need to communicate our objectives, and our constraints to our upstream support team, clearly and quantitatively.

If **we** change *anything* that might affect their ability to support us, like a Goal level, or a Goal deadline, then it should be obvious we must *communicate those changes* to all affected support teams, and perhaps insist on a reply. 'Do **they** plan to make any changes in response?'

Keeping track of which stakeholders who need to be informed of changes is part of the role of specifying relationships in the objective, or strategy specification.

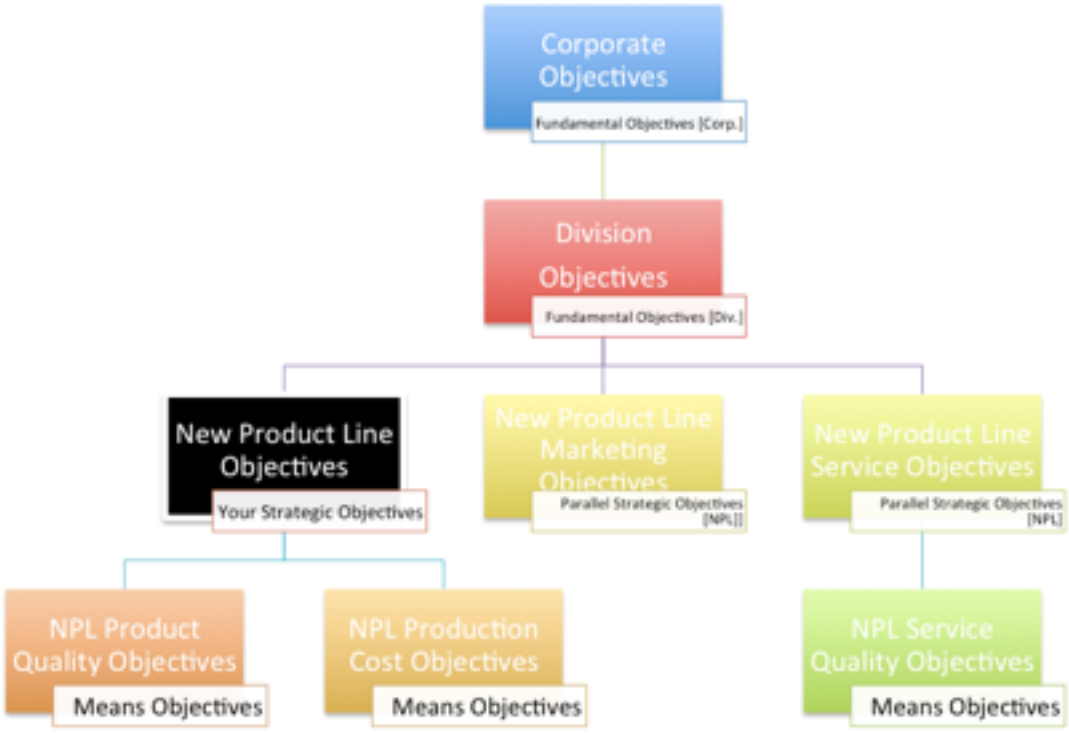


Figure 3.4 B. Multiple levels vertically and horizontally, in relation to your Strategic Level.

Multiple Related Levels
Notice that there can be any number of levels of *concern*, both above and below *your* level. There can be any number of different contributing levels of support, *below* your level. There can be any number of *parallel efforts* to your own strategic plans, also supporting *your* Fundamental Objectives.

Complex but not complicated.
The relationships are not simple in structure or in understanding. They are compound (several elements), and complex (up down and sideways relationships)

But they need not be unduly complicated (difficult to understand, track) to the planner **if**:

- we take the trouble to 'map' them, for example with Planguage statements, so they are transparent and remembered, by the plan database, long-term, and thus available to all current and future planners, near and remote. The relationship 'memory' has to be digital, and it has to be tightly integrated with the related plan objects.
- we take the trouble to quantify (estimate and publish) mutual expectations of value delivery and costs (example using Impact Estimation Tables)
- we take the trouble to measure impacts, early and frequently; so as to better see cause and effect (using IE Tables to track)
- we recognize mutual responsibilities: to support and to communicate *change*. These are visible on IE Tables, and especially on multi-level IE Tables [URL36, Bring case]

A comment on the relationship of objectives, to stakeholders.

> ***"Are you sure it's the right thing to do?"***
>
> ***(i.e. sell bad stocks to good customers)***
>
> ***"For whom?"***

Kevin Spacy replies in 'Margin Call' Film about an event in the early stages of the Financial Crisis 2008. Quote 3.4 A.

It gets 'better'. A reference to downstream stakeholders and value.

Sam Rogers (Kevin Spacey) : the real question is : who are we selling this to?

...

Sam Rogers: And you're selling something that you 'know' has no value.

Source: IMDb, Margin Call, 2011, Quotes.

Making Sure Your Work is Aligned
Make sure that your gradual improvement in achieving your own strategic objectives, is *really* connected to your downstream Fundamental Objectives. Those objectives you try to *serve* with your Strategic objectives.

That means, your planning and efforts need to be clearly connected to your 'bosses' objectives, and other stakeholders you are indirectly serving. And perhaps connected to parallel efforts also serving your downstream stakeholders.

There is rarely anything as simple as 'my bosses' objectives. You are usually serving a large set of different stakeholders. You 'boss' might just be a useful symbol for that set of stakeholders. Your boss might indeed be concerned and responsible for many or all of them! Stakeholders *rule!* And some stakeholders, like a 'government', have more power than your boss, over your planning. So, it is particularly important that we are explicit about all critical stakeholder relationships, as dealt with elsewhere [over 295 references to that word in this book, but for example 3.6, 3.7, 3.9 below].

Can you measure the effects of your support teams' *means* objectives, on *your* (strategic) objectives?

Can your stakeholders, at least, perhaps, hopefully, experience, at a gut-feel level, everyday level, that things are *changing for the better*?

Are things speeding up, people saving time, costs going down, or customer service getting better?

If you are doing effective value delivery, on critical factors for stakeholders, then *measurement* might be 'unnecessary' (but useful) because the changes might be *so obvious* to people, and such a *relief* from the old problems, that everyone knows 'something good' is happening.

A Basic Assumption of Evolution of Results.
There is an assumption we make here, which we need to remind the reader of, now and later ['Evo', Part 5, & Chapter 4].

We have abandoned the whole idea of building a 'great new system', and 'everything will be better'.

Experience, for decades, tells us that this 'big bang approach' is not likely to succeed. And even if 'good things' happen at the end of a initially 3-year project, delayed by another 4 years (3+4 = 7): it is too late, and the pain has been suffered too long.

If you are so new to the real world, that you have not experienced your own failed project (avoid *that*, learn from others' failures, they are everywhere), just try to Google 'failed projects': they abound in every business, yours is no exception. Failed projects are perhaps the majority of all projects, depending on how you define failure.

One often-cited Survey (Chaos) defined *failure* (in part) as costing *more* than twice the initial budget! 99% over budget was not any sort of failure. What if your company suddenly paid you less than half your normal monthly salary? Would *you* call that a failure?

Our big assumption is that we are going to ***evolve*** *the 'delivery of value* to stakeholders', immediately (for us, second week of a project) and frequently and incrementally (every week until we are done).

That means we are going to estimate, every week, how much value we think we can deliver, for a week's work. Then, we are going to try to *measure* how much value we actually achieved *that week*. [URL4, URL5]

Now, *that* will keep us focussed on delivering *our* strategic objectives.

If we, or our suppliers, are incompetent, clueless: and unable to ever deliver *even small increments* of value for real – then, as a manager you know what to do. Find someone who *can* deliver. Hopefully they are standing in line waiting for you to call upon them.

The Knock-On Effect

But here is the tricky part. You have achieved your *own* objectives. But can you actually measure the knock-on effect with your stakeholders.

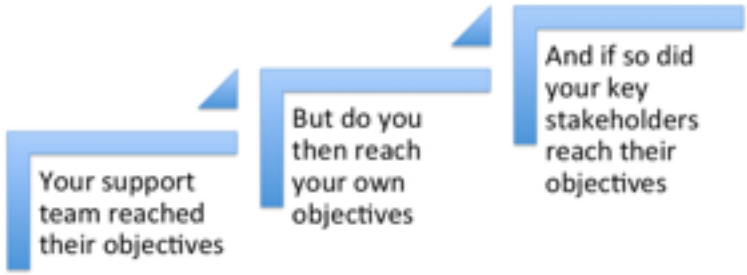


Figure 3.4 C The chain of knock-on effects

If you save time, does it *really* save money? If you make a product more user-friendly, does it *really* sell more? If you increase the market share, does profit *actually* go up?

You need to participate in measurement at the stakeholder level, and learn as early as possible, maybe using leading indicators, if there is a *valid connection* between your strategic objectives, and the downstream stakeholders' (there are many stakeholders) fundamental objectives.

Figure 3.4 D. Example of levels of related objectives, with knock-on effects, upwards.

There could well be a missing link, like motivation, training, enough time, that needs to be part of the overall system, in order for your contribution to actually deliver knock-on value properly.

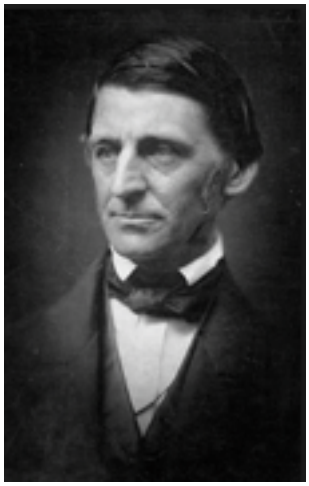
You cannot afford to leave that consideration to your stakeholders. It needs to be part of *your* plan and *your* project!

We call this 'systems thinking': all aspects of the system that matter, not just your speciality.

Quote 3.4 B. Create your fate.

> ***"Shallow men believe in luck or in circumstance. Strong men believe in cause and effect."***

Ralph Waldo Emerson (1803-1882) Photo ca. 1857



Practical Tip 3.4 Sanity Test: Are we reasonably connected?

- *List the top 3 fundamental objectives you are supporting*
  - *Ask your 'boss' (next level downstream stakeholders) if they agree, that you* are *expected to help them achieve these objectives.*
  - *Are your objectives going to impact those 'fundamental' objectives?*
- *Have you communicated your strategic objectives quantitatively 'upstream' to your suppliers?*
  - *If not, how can they support you?*

Policy 3.4. Measure at two levels: yours and theirs.

- **Keep close track of 'Means Objectives' progress, upstream**
- **Weekly measurement, at *least*, is a useful practical policy.**

- Why?
  - To make sure they are aligned with *your* current strategic objectives.

Section 3.5 How many strategies do you need to reach your objectives?

Principle 3.5 Takes Two To Tango.

Your objectives
might need to be reached
using _multiple_ strategies.

It normally,not always, takes more than one strategy to reach an *ambitious*, competitive Goal. That is what makes it the Goal level competitive. It is not so easy!

One strategy might get you part way to your numeric goal. But it will then usually need to be supplemented by other strategies, to get you all the way to your goal.

You can hope, but never know, how many, or which, strategies you will need, until you actually measure, how well they are doing in practice.

	Strategy 1	Strategy 2		Total for Strategies	Shortfall
Goal A	0%	80%		80%	20%
Goal B	85%	20%		105%	none
Sum A+B	85%	100%			

Planguage Example 3.5 A. A very simple Impact Estimation Table.

The Meaning of the % estimations.
The % numbers are estimates of the % of the distance to the Goal (on time), that we *believe* each strategy will measurably result in.

Adding the numbers up for a single Goal, as we do horizontally, is a simplification. We hope it will give us a reasonable picture of the sum effects of the strategies; at least the magnitude. See 'Total for Strategies' column above (PL 3.5 A)

If they add up to a small number like 16% you have almost no chance they will really get you to your Goal. If they add up to a number around 100%, you might get there with luck. To play safe, we often go for a safety margin of 2x, and the numbers need to add up to about 200%.

Management, given a choice, will ask for safety factor 2x.

Discussion of how we might interpret the table. (Planguage Example 3.5 A)
Goal A is not impacted at all (0% increase above the benchmark) with Strategy 1, and only 80% (of the way to the Goal level on time) contribution from Strategy 2: so we are 'short' at least 20% of strategy impact, needed to reach Goal A.

More strategies are necessary, or a reformulation of the strategies so that they have more effect.

Goal B is, in theory, impacted enough, by Strategy 1+2 in total to reach Goal B.

We need, however, to really do both strategies, in a pilot, or scaled down, and see what total effect we reach in practice, to be reasonably sure.

This is an example of 'Impact Estimation Tables': a way of visualizing the effects of multiple strategies on multiple goals, which we will return to later. [1, URL21]

Specifying Impacts as a background statement in a strategy specification.
Here (below, PL 3.5 B) is another way of specifying our estimate, of 'how good' the strategy is expected to be.

This is not a table, and it can only handle the impact of *one* strategy, on any number of Goals.

But it does have the convenience of something we can use to capture information during the development of a strategy specification.

When you have several *strategies*, you can choose to copy the estimates to a table. In fact in our automated tools [URL68, URL73], this is automatic.

Strategy 1:
Type: *Major strategy specification.*
Description: *(<sounds compelling, and it is latest fad>)*

Estimated Impact ***[Goal A]*** *0%,* ***[Goal B]*** *85%*

Planguage Example 3.5 B. A simple numeric estimate on 2 different Goals.

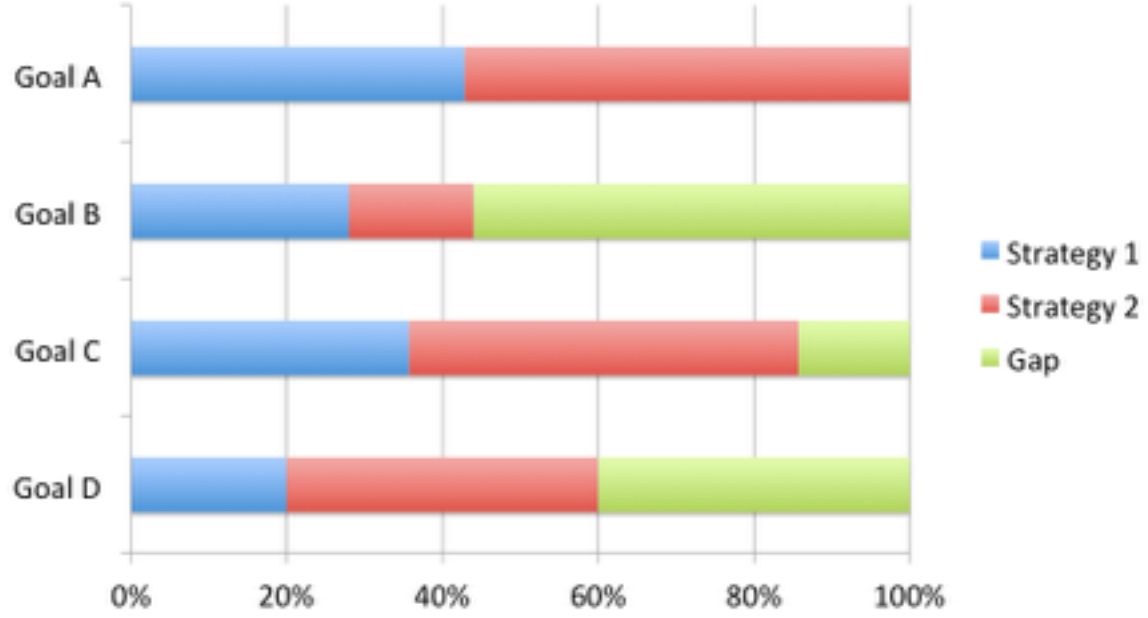


Figure 3.4. A graphical representation of the impacts of 2 strategies on 4 different Goals.

Reasoning about about the situation, after we have 2 strategies on the table. Do we have enough yet?

- Goal A is *theoretically* satisfied by the 2 strategies.
- Goal B is only about 44% satisfied by the 2 strategies and needs considerably more, or better, strategies.
- Goal C is estimated to be 85% of the way, and who knows, with luck might get to the Goal. But it would be safer to think of a third strategy. However we can wait until we see what really happens with the first 2 strategies. It *might* not be necessary.
- Goal D is so weakly supported (60%) that we should probably immediately look for one or more additional strategies, which can affect that particular Goal.

The main point is that this Impact Estimation process gives us a basic tool for looking at the effects of several strategies, on several Goals.

We have a clear 'logic' which allows us to decide what we should do regarding strategy planning.

You need to have a plan for getting to the Goal in 'one leap'. Do not wait for failure before you see the need to plan better set of strategies.

"The most dangerous strategy is to jump a chasm in two leaps."

Benjamin Disraeli (1804 - 1881).

Practical Tip 3.5 Do at least one rough estimate immediately

- *Estimate roughly, using % of way to Goal, each new strategy suggestion, for at least one major objective, which it is supposed to be useful for.*
- *The resulting thinking, and discussion, will tell you and others a lot about:*
  - *how useful the strategy might be,*
  - *how much people really know about the strategy,*
  - *how well-formulated the strategy description is (vague, general, or very specific variant).*

Policy 3.5 Assume nothing until proven.

- **We will never assume that a given strategy will get us to our goal levels on time.**
  - **We might need more strategies, or might need to alter a current strategy, so that it is more effective.**
- **We will make a numeric estimate of the assumed effect of any seriously-contemplated strategy.**
- **Nothing is proven and accepted until, we measure the real effect of the strategy in place.**

- *Why?*
  - *Nobody really knows in advance for certain, there are too many uncontrolled variables deciding the result*

Section 3.6 Alignment and Re-alignment of your Strategic Objectives.

Principle 3.6 Continuous Alignment.

Your strategic objectives
depend on
your many stakeholders' changing realities:
assume you _might_ have to re-align them.

Your 'boss', the one who decides what you should do, is, effectively, 'your critical stakeholders'.

Stakeholders will hopefully reveal their needs and wants, and these will be turned into your prioritized Strategic Objectives. If you get these right, and deliver them, hopefully your critical stakeholders should be satisfied. Your 'boss' should be happy with that.

Your formal quantified objectives *might be wrong* statements of need and wants. We might also be wrong in our *interpretation* of these needs and wants. We can be wrong about objectives, with the best of intentions.

Objectives are not about right or wrong, but about balance between stakeholders, and balance with available resources.

In fact, things are so complicated, that it is *safest to assume the objectives are wrong*, to some degree.

But we should also assume, and make sure it is true, that we will *quickly learn what is wrong*, and *try to adjust* the strategies quickly

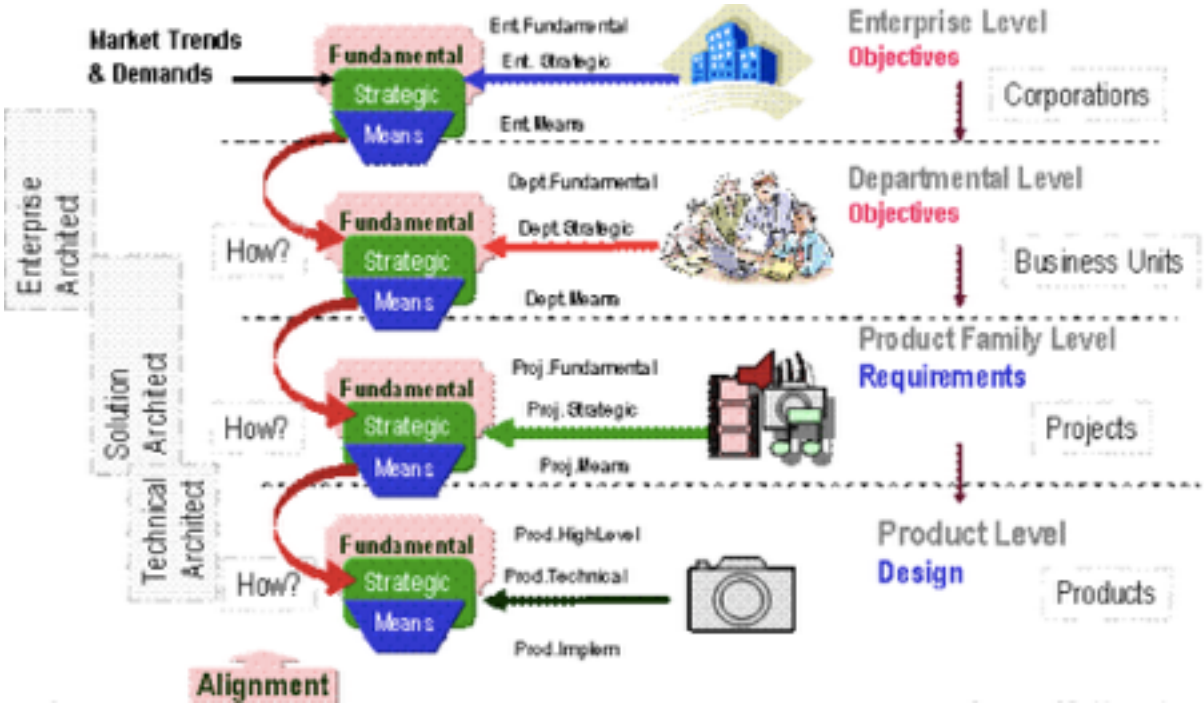


Figure 3.6 A. An example of how different levels of Objectives and Strategies can relate to each other.

So, should we challenge our stakeholders' choices of objectives?
Not directly. Let us assume that they will take responsibility for their choices, and they know they have to learn what is really right for them, and to adjust as soon as possible.

When they adjust, it is our turn to adjust and align. We need to be responsive to their changes. And certainly not 'complain' that they can't seem to make up their minds. A sensitive 'planning language' is a useful tool here. Hopefully the planning language is backed up with automated tools, to integrate the changes more easily [URL73].

We need to focus on our role of 'loyally delivering the results' desired by the stakeholder.

But we need to be very interested in what really happens at the levels of our downstream Fundamental Objectives. We need to be sure that our satisfied Strategic Objectives actually *do have the desired knock -on effects* on our downstream stakeholders.

This is for 2 reasons:

- *Our* own objectives might need tuning, so as to actually achieve the results expected from them.
- Our stakeholders' definition of their objectives might have to be changed by them, to react to changing conditions, and *we* need to react to that, by better-defined objectives, and possibly better strategies.

Here is a sample report, you might get, with an example of *actual* results, in relation to the ones you *estimated* you would get.

	Strategy 1	Strategy 2	Sum S1+S2	Safety Margin
Estimate for Goal A	0 %	80 %	80 %	
Estimate for Goal B	85 %	20 %	105 %	
Actual Goal A	**5 %**	**120 %**	**125 %**	+25%
Actual Goal B	**15 %**	**20 %**	**35 %**	-65 %
Strategy % 'to 100% 'Missing	**-180 %**	**-80 %**	**-65 %**	

Planguage Example 3.6. IE Table with both estimates and actual results fed back later. -180% means that Strategy 1 in fact gave only 15%+5% of the ideal 100% (Goal A) +100% (Goal B) we need to satisfy. Strategy 2 is better, alone.But the combination S1+S2 is only lacking 65% of what we need in total. So an S3 that could move us further towards our goals would be useful.

What would you as a manager do, if you got the above results from your 2 strategies?

Hint: you need some Goal B-relevant strategies! And it does not look like Strategy 1 or 2 is at the right level of effectiveness, alone or together. You need strategy '3'. Get creative.

The *actual* measurements bring us back to reality. We were overoptimistic, people generally are, in our theoretical estimates. So it is important that we get actual feedback measurements early, before we have invested irreversibly in a failing strategy.

Figure 3.6 B. We need to adjust our Strategic Objectives Goal levels (-->) so that the knock-on effect is really enough to help reach the level of Fundamental Objectives, that we are responsible for contributing to.

"Companies and their brands need to reach out and speak directly to consumers, to honor their values, and to form meaningful relationships with them. They must become architects of community, consistently demonstrating the values that their customer community expects in exchange for their loyalty and purchases."

Simon Mainwaring, ***author of 'We First: How Brands and Consumers Use Social Media to Renew Capitalism and Build a Better World'.***

Practical Tip 3.6. Eternal Change to Better Direction.

- *Never assume that your objectives can be 'frozen'.*
- *Assume that they were quite possibly wrong from the start: test them.*
- *Assume that they need to be changed after a while, for fast-changing conditions. Monitor your stakeholders critical conditions.*

Policy 3.6 Stakeholder Loyalty

- **We will be loyal to our stakeholders needs, yet be ready to help, by adapting quickly, when stakeholders decide they need to change their objectives.**

Why?

- Because continuous alignment with their real needs is the only realistic way to help them succeed.

Section 3.7 The Problem Of Your Stakeholders' Unclear Communication About Their Objectives.

Principle 3.7 Don't Solve Undefined Problems.

If you do *not* know exactly
what your stakeholders' current objectives are,
you *cannot* decide
what your *own* Strategic objectives should be,
to support your stakeholders.

Your 'boss' (your critical stakeholders, the ones you want to serve) will normally be unclear, misleading, and unsystematic in formulating their objectives. Forgive them, they are just human, and they are the victim of poor education and training. And, maybe your own Business Analysts need some better skills here?

But, you have to assume that stakeholders will display *serious defectiveness* in their crude attempts to articulate and communicate what they really want.

We regularly measure about 100 'major defects' (a defect, simplified = unintelligible word) per 300 words in initially stated objectives. Rendering the other 200 words useless. What does "Very ????? Reliable" mean, or "????? Competitive Production". [URL1] ('????' is a symbol for a term that is ambiguous and or unclear)

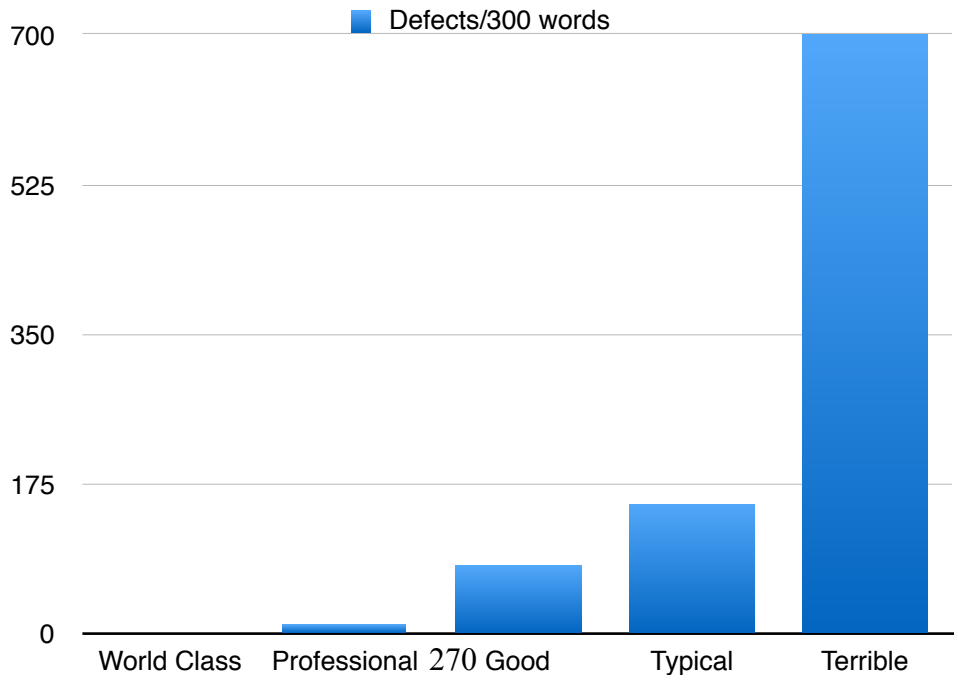


Figure 3.7 ***Planning Clarity Quantified.*** *The expected range of defects in a planning document, such as objectives.*

Comment on Figure 3.7. This is a measure of bad communication in your planning specifications. The 'World Class' level is attained by NASA, Boeing and Intel, and is about 0.1 to 0.2 defects/page before release to next process. And they *intend to catch those few defects in the later processes. See next Section 3.8, Part 3 and 10.2 for more on this subject.*

You probably wonder how we can get more defects than words. Many ways to do this, but the simplest credible explanation is that 'missing' but required specifications are also defective. The second simple reason is that a single word can simultaneously violate several standards (Rules). For example: Rule 1. it shall be in English. Rule 2. It shall be unambiguous. for 'Tres'.

I have measured the 'Terrible' level (700 defects/page) about twice in my career. Normally, sight unseen, I can be pretty sure people produce 'garbage' planning at a level of 150±70 defects/page. Unless they have already taken steps to reduce this 'natural' pollution level. The pollution level depends highly on the standards that are applied. There are no 'illegal' things, if there are 'no laws'.

But in the main I am initially measuring 'clarity' and 'unambiguousness'. because if those are bad, then it does not matter what else is written to a bad standard; you cannot even understand enough to decide if the other, more advanced Rules are broken. Most organizations have such poor clarity, that you cannot yet determine if they are planning the 'right thing', yet, at all.

Help your Stakeholders Express their Needs Clearly

The best thing you can do, to help yourself, is to **help stakeholders get clearer**.

Of course some stakeholders will never, or can not, directly articulate to you what their objectives are, clearly and correctly enough: so we need to be skilled in interpreting what their objectives *really* are, in spite of that.

There are many approaches. Here are some ways to help stakeholders with objectives:

- " We want very much to understand exactly, the result you want: preferably in measurable terms. If we do *not* understand, then we fear that you will *not* get what you want, or it will eventually arrive late, costly, and corrupted."
- "You might have a very clear idea or gut feel for what you want. But, for your needs to be really taken into account, to really get what you want, they need to be written down in a fool-proof way; so we *cannot possibly misunderstand*. You cannot assume that we know, all that *you* know. That would be a mistake on your part. Your implicit and tacit knowledge, needs to *be made explicit*. Please don't overestimate our ability, to *guess* what you really want. We will be wrong 99% of the time."
- "We will be very happy to help you, by pointing out exactly what can potentially be misunderstood, by us and our colleagues. And we will also suggest clarifications, that make your wishes more intelligible, at our level."
- "This is the only way for you to be sure that you will get, what you really want. And if finally you supply us with 'zero defect' objectives (crystal clear), we can probably give you fast, and error-free delivery of results. Win Win."
- We are really sorry to take up your time just now with this. But the other option is to take up even more of your time later, when we unintentionally fail to deliver what you really want. Let's get this 'right' now.

What you might choose **not** to say *directly to your* managers/ stakeholders/clients is:

- "You probably do not really know *how* to specify your objectives clearly, because you have no such culture and training." It is not really their fault. Almost everybody has this lack of training and culture problem.

- "If you let us help you, and observe the process and language we use, you will hopefully acquire a valuable communication skill, that you can use in dealing with other people, the rest of your career: 'much clearer ability to communicate'. It will even help you deal with your boss and critical stakeholders. This skill will help you get what you want."

If they are as smart as they think they are, they will hopefully figure this out themselves!

Here are the basic concepts for specifying 'exactly' what a *value objective* is (see rest of book for detail, like Chapter 1):

For **objectives**, including 'targets' (what you want to get, Goals), and scalar constraints (like Tolerable levels, the worst acceptable cases), you **must** define

- A Scale of measure
- At least one future desired point, or level on that scale (Goal, Tolerable, etc.)

Case 3.7 For example (real, 2014): CEO Objectives 'Improved' (radically).

Benefit Management:
Type: Top level CEO Objective.

Scale: The Productivity, in Value Goals achieved by **Deadline** by defined **Planners**, in defined planning **Processes**, for defined **Value** sets.

Goal: > 95% ? <- Guess Tsg
Deadline = End 2017.
Planners = Result Responsible Planning Teams.
Process = Systems Architecture.
Value = Long Term Adaptability (managing Technical Debt).

Tolerable: > 70% ? <- Guess Tsg.
Deadline = End 2018.
Planners = CEO.
Process = Top Level Board Objectives Planning.
Value = Economic Factors, Image factors.

Planguage Example 3.7. Example of defining a CEO Objective.

Case 3.7 The original formulation was:

> **"Ambition Level:** the Responsible Executive will be able to prioritize his efforts in a periodic and consistent manner."

This formulation *mentioned* some perceived-as-useful processes ('prioritize', 'periodic', 'consistent'), but *completely forgot* to mention the *main point* of 'value delivery'.

It was also preceded, when we tried to establish the main objective, with the following pseudo-objectives (*seem* to be, but are *not really objectives*): which we quickly agreed were just potential *strategies*.

- Strategy = Immediate Visibility of Initiatives
- Strategy = Visual Tool
- Strategy = Early Warning .. so I can intervene

The key 'trick' in identifying strategies is to ask 'why?', and if the answer is a higher level (higher priority) objective, then, by definition; you had initially identified a mere potential *strategy* (*not* your *real* higher level objective).

The second 'trick' is to give people a Mafia Offer:

- "Do you want the 'Visual Tool', even if you do not achieve 'Benefit Management'? "
- Or, "Do you want improved 'Benefit management', even if you do not have a 'Visual Tool' ?"

The 'Mafia offer' clarifies, for you *and them*, what their **real** objective is, and, at the same time, it clarifies what is merely a means/solution/ strategy, which they imagined, *prematurely* (before a clear objective is agreed) *was* their 'objective.

But these are again, just some potential ideas of how to get to the real objective.

And we cannot even *begin* a process of deciding whether the 'potential strategies' (like 'Visual Tool' in the case above) will, or will not be *valid* strategies. Because we did not, at that stage of formulating the objective, have any clear (Scale + Goal) articulation of the real goal. Let alone an official objective, *agreed* by the powers that be (our critically prioritized stakeholders).

For other classes of specification, primarily Condition Constraints, like 'It must be legal in the EU', or 'Strategies must not conflict with Government Procurement Policy': we need formulations, so clear and unambiguous, that they can easily be tested, or audited, on a simple 'Pass, Fail', level.

Condition constraint formulations like 'Use Best practices', or 'Use Winning Competitive ideas', would *not* be acceptable. They are just too vague to have any value. but that does not actually stop people today from including them in their plans!

"If I had one hour to save the world, I would spend 55 minutes defining the problem, and only 5 minutes finding the solution"

No sure source! But nice statement! No, it is not provably by Albert Einstein.
PS, the quotation author failed to allocate time to implement the solution in practice, before the end of the world. How about 50 + 5 + 5 minutes to implement.

Practical Tip 3.7. Quantify Agreed Objectives.

- *Rewrite all/some/at least one of your critical objectives using a Scale of measure, and a Goal level. Get the source, if human, to comment, perhaps agree that your draft is better, or modify it.*
- *if this tastes good, repeat for all critical objectives.*

Policy 3.7 Trust not initial inputs: Clarify and verify.

- **Assume that all initial statements of our stakeholders' objectives, can be misunderstood, and very probably will be.**
- **Clarify the initial statements, so they are measurable, or at least 'testable' (exist, not exist).**
- ***Then* translate this information into *your* strategic objectives. To help your stakeholders get their true, not misunderstood, values.**
- ***That* is our economic and logical process for fast delivery of the 'right' value.**

Why?

- So that stakeholders do not assume you're a fool later, for 'misunderstanding' them now.

Section 3.8 Getting 100 times clearer objectives.

Principle 3.8 Share Cear Goals.

If you do not *share*
your own *clear, complete and relevant* objectives
with your upstream Support Team,
you cannot expect them to make relevant plans
to support these objectives.

Principle 3.8. Same as Alice and the Cat [Fig. 2.0].

You cannot keep your real objectives 'secret' from the people who you need to help you reach them. The competitive business winner does not have to have *secret* objectives: just *superior* quality objectives, and *quicker* value-delivery capability. Compared to the competition.

The *best* protection against public Patent infringement is better new inventions, my father, Tyrell T. Gilb, would have advised. But he still held over 100 patents, just in case. And always had plenty of inventions in the pipeline all life long. In fact I once saw a plaque at his home from colleagues, congratulating him on having the audacity to patent, or try to patent, the oval shape.

Your necessity is to supply your Support Teams, who are *some* of your upstream stakeholders, with your own *intelligible* objectives and constraints.

Figure 3.8 A Sharing your objectives with your support team, so they can know what kind of support they 'need to align to'. Upstream stakeholders are not your primary result delivery stakeholders. But they can require certain qualities and values from you anyway. For example they might require, co-operation, payment, result crediting, or quality in communication

If you give your Support Team 'Garbage In', you almost force them to give you 'Garbage Out'. And, that is *your own* fault.

Rules for Rigorously Writing Objectives

One simple device for increasing quality of objectives, and other plans, is the concept of adopting a clear, simple, written *standard* for the objectives.

We call such a standard a 'Rule' set. Rules are **'rules for writing'**.

Surprisingly, we find that it is normal for planners to have no particular formal rules at all.

And, when we do check objectives against some of the most fundamental rules imaginable, basic intelligibility, we find that the rules are violated with a density of every 3rd word written.

All words are unintelligible, but some words are far more unintelligible than others. Intelligibility is not in the *word* alone, but in the context of the ideas being expressed. Words give us a rough idea, context supplies the intended meaning.

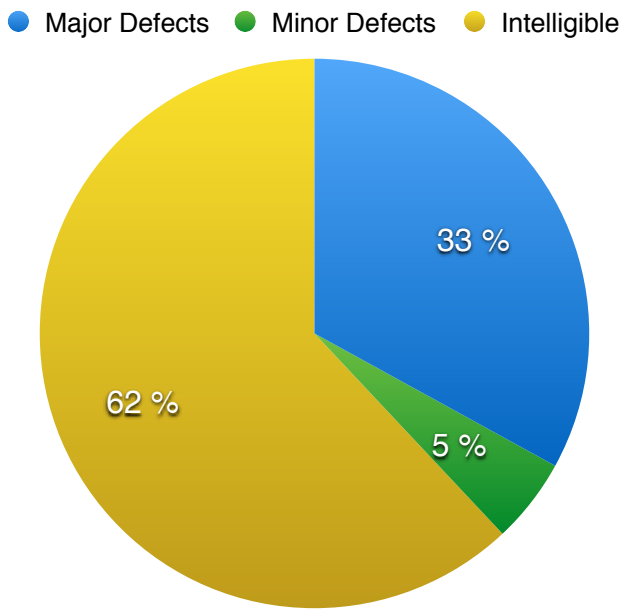


Chart 3.8. Far too much of our planning specification content is so unintelligible that it threatens the entire project with certain failure. But most planners do not measure, do not have standards, and do not manage the Exit of the plans. Something is very rotten in the State of Pen Mark. We create some interpretation in our personal minds. So, it seems OK. But it is not the right interpretation.
62% intelligible does not mean that 62% of the whole is intelligible, because the defects are spread evenly throughout all sentences. Hardly any statement can be intelligible. It is really 100% unintelligible. See Quote 3.8 Below.

If we do not have good enough context information, we will find that the the words appear unclear and ambiguous. Planguage supplies a lot of context. We call that 'Background' specification.

'Every 3rd word a defect' means: 100 defects, for example unintelligible words, in relation to the rules, per 300 words.

Common sense, or academic training, or local planning-culture, will never itself guarantee a high-quality statement of objectives.

You have to state specification Rules clearly, and you have to check that the Rules are in fact followed. If they are not followed, the plans have to be **'refused Exit'** to the other downstream stakeholders.

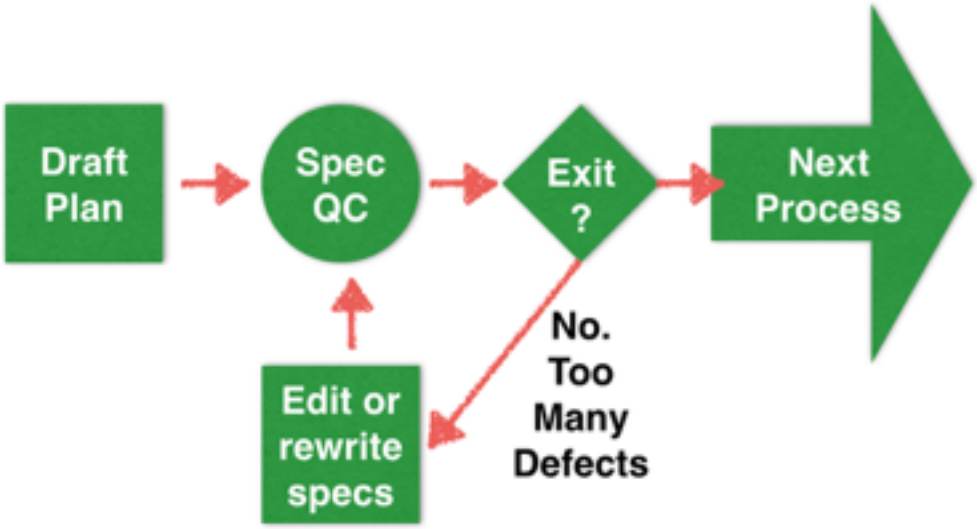


Figure 3.8 B (copy of Fig. 10.2 A) Planning specs that violate your organization's necessary rules of good quality specification, need to be refused 'Exit' to the next planning process. This is how we avoid 'garbage out.' This method supports the 'lean' process concept of 'getting it right' upstream. Avoiding costly problems downstream.

Here is an **example of Rules** that might reasonably be applied to 'management planning' objectives. A larger collection of rules is published in Competitive Engineering [1, and 1F is a free download of Rules]

R1. Objectives will be unambiguous to the intended readership.

R2. Objectives will be clear enough to measure, or test, their delivery.

R3. The objectives will NOT* themselves *be a 'means', or be* mixed up with *any ideas of the 'possible strategies' for reaching the real change objectives. They will be 'pure' results, values, qualities, or improvements.

R4. All 'improvement' ('Performance') objectives will defined by at least one Scale of measure, and at least one Target level (Goal, Stretch) or Scalar Constraint level (OK, Tolerable, Catastrophe, Fail) will be specified on, each defined scale. Preferably both a Target and a Constraint, at least

Planguage Example 3.8 Basic Planning Specification Rules. A simple summary might be: "Objectives will be measurable future result targets and constraints."

If you were to check your own current objectives against these rules, you would find massive violation. Mainly because nobody has ever seriously decided to enforce such a 'common sense' standard.

Case 3.8 : Here is a published case study from Intel, using our planning methods [URL22].

Rev.	# of Defects	# of Pages	Defects/ Page (DPP)	% Change in DPP
0.3	312	31	10.06	
0.5	209	44	4.75	-53%
0.6	247	60	4.12	-13%
0.7	114	33	3.45	-16%
0.8	45	38	1.18	-66%
1.0	10	45	0.22	-81%
Overall % change in DPP revision 0.3 to 1.0:				-98%

Table 3.8 for Case 3.8. Stages of 'product objectives' quality increase, from 'pretty good' to World Class release standard. Each new stage is triggered by an Exit failure. By not following standards well enough. See comment below.

Case 3.8 Continued.

In this case the Intel objectives, written in Planguage, to a clear standard, with proper training, managed to *initially* deliver, 'no more than 10 defects' (rule violations) per *page* (at Intel, '*page'* is defined as 600 words). Revision 0.3.

Over 20,000 Intel engineers have been trained in Planguage by Erik Simmons and staff, since 2000.

That 10 defects/page level is 'pretty good'. Without clear standards, training, and Planguage, the result would probably be 10x worse initially. About 100 defects per page.

But in this case the Intel *final* objectives process 'Exit standard' is very much higher. 40 times higher.

No more than 0.25 defects per page, approximately. One defect per 4 pages. So the 'objectives specification' is rejected, repeatedly, until they achieve this better level, at revision 1.0.

The reported payback for this discipline [URL22, URL23] was

- "Scope delivered at the Alpha milestone increased 300%,
- released scope up 233%
- Logic defects reduced by about 50%
- Defects that did occur, were resolved in far less time on average"

Feedback from Forums and other sources confirmed that the numbers were meaningful.

The most important thing here, is *not* that a poor quality specification is rewritten until it is good enough. We do *not* focus on *correcting* defects found!

The point is that the individuals involved in planning, and also in their own quality control, are 'forced' to learn how to achieve that Exit level *ideally* on the first draft, and can repeat very low defect-levels in the future.

We call this the *personal learning curve* [Diagram 10.2 (Ericsson), Case 7.7 A (Boeing)]. Individuals reduce their defects by about 50%, per cycle of feedback. They need a series of feedback cycles to get good enough to Exit. But they need fewer cycles, even for initial Exit of a new plan, if they 'learn' how to follow the Rules earlier.

The learning curve carries over to their future work.

If management does not *make* this happen, then one-hundred-times worse than necessary, defects-in-objectives quality, is the probable result – *forever*.

Quote 3.8 Zero Communication Defects?

"One should aim not at being possible to understand, but at being impossible to misunderstand."

Quintilian, 35-100 AD, Roman rhetorician from Hispania.

Practical Tip 3.8 A Quick and Dirty Quality Check.

- *Take a page of 'very important objectives' and count all words that* **you** *are not '100% sure' that you know the* ***exact*** *meaning of,* ***as intended by*** *the writer. Not your personal guess as to 'some interpretation', but the planner's intent. As they would confirm to you.*
- *What chance does this specification have of being correctly understood by other people, like an outsourced supplier, far away?*

Policy 3.8 We will have clear high standards, and measurement of compliance to the Rules, and planning process Exit level standards, and 'garbage intolerance'.

- We will raise standards for writing plans, by having
  - **Simple, clear, but powerful, rules**,
  - **measurement** that rules are in fact followed,
  - and a **numeric exit level.**

Why?

- to prevent 'garbage out', upstream
- to motivate people to do, and learn to do, much better quality planning.
- And to force us to *do it right* early (a 'lean' principle)

Section 3.9 Dealing with your many stakeholders

Principle 3.9 Mutual Interdependency Web.

Maybe you are not actually responding
to a simple hierarchical organization:
maybe you are
just in the middle of a lot of interdependent stakeholders

The Boss and Sub-ordinate relationship, is simply *one* possible planning environment. It might not be *your* environment at all.

Your planning activity can, potentially, be useful to a large number of various stakeholders, inside and outside your organization (which may be you alone, part time).

But the reality is bound to be that you have *many* potential stakeholders.

They all, by definition, have *some* objectives worth delivering by you, in order to do your job properly, as an employee, or consultant, or discipline manager.

Your 'formal' boss might be able to hire and fire you. You client can certainly employ your services or terminate. But your real usefulness to these hire/fire powers may really depend on your ability to plan effectively for a larger range of stakeholders, and to plan for beneficial-to-deal with, stakeholder needs.

Your job as a planner is to be able to *identify* those critical and useful stakeholders, and their needs-worth-you-dealing with (their objectives worth taking seriously).

In the planning specification example below, we consider 3 stakeholders, and 2 strategy options.

Leasing Agreements:
Type: Marketing Director Objectives.
Ambition: Solid leasing profit increase next few years.

Scale: number of annual average long-term value agreements in defined Markets for defined Products.

Past [2009-2015, Markets = EU, Products = Electronic] 5,000 agreements.

Goal [2016-2019, Markets = EU, Products = Electronic] 25,000±10,000 agreements ?.
Note: Gross Annual Value: £1 million.

S1: Strategy Possibility: Distribution Agreement with Thorne.
Strategy S1 Annual Cost: £0.3 ?
Note: Net Annual Benefit S1: £0.7 Million ?

S2: Strategy Possibility: Distribution Agreement with Thorne & Philips.

Strategy S1 Annual Cost: £0.4 ? <- CMO. Depends on contract agreement.

Net Annual Benefit S1: £0.8 Million ?
Issue: Would either distributor claim exclusive rights?

Planguage Example 3.9 Several stakeholders and strategies.

"Synergies are something that the CEO basically has to force to happen, because organizations are, generally like bodies in motion that tend to stay in motion. It's very hard to get big organizations to change. And it really takes a powerful mandate to force things to happen!"

- *John C. Malone (1941-) Billionaire American businessman and philanthropist. Ex CEO Tele-communications, Inc., Chairman of Liberty Media. Largest landowner in US.*

Practical Tip 3.9 Simple cross check.

- *Identify at least your top 10 stakeholders*
- *Specify them explicitly in your objectives*
- *Ask: are there any stakeholders who are not mentioned in your objectives?*

Policy 3.9 Thorough Stakeholder Analysis.

- **We will thoroughly identify all potentially interesting and critical stakeholders.**
- **We will systematically analyse their primary and critical needs.**
- **We will select those stakeholder needs which are critical and valuable for us, to include in our objectives and constraints.**
  - **(Objectives (to reach) + Constraints (do, or do not) = 'requirements').**
- **We will explicitly identify, and document, issues, risks, assumptions, and dependencies; that we discover during this stakeholder analysis process.**
  - **They will be integrated into the corresponding planning specification: right there where we can best be reminded of it.**

Why?

- To prevent projects from failing because we did not take a few days to thoroughly analyse stakeholders and their needs.

Section 3.10 It is not just about primary stakeholders, we need to include stakeholders with less power, but valid needs.

Principle 3.10 Stakeholder Value Efficiency.

**If you can potentially supply value,
cost-effectively,
to any other party,
they are a *potentially* interesting stakeholder
for you.**

A 'Critical stakeholder' is any 'person, position, group, or thing' that can make or break your 'project' or planned 'endeavour'. There may be very many of these, even for a medium-scale project (like about 50, think large £30 million banking system) and far more for really large and complex projects (think National Health System, think 500+)

They have their needs, and necessities, and some could not care less if they will destroy your project: if you cross them, or fail to satisfy their needs.

This is a far cry from the 'what does the customer want?' oversimplification.

You cannot offer *perfect customer satisfaction*, with an *illegal product* or service. *Not a valid option*, if you are constrained to being legal!

Here is an example of 'being explicit' about stakeholders.

Confidentiality:
Type: Marketing Product Line Objective.
Ambition: Safest place for personal data on the internet.

Scale: % Integrity of defined Data for defined Purposes for defined People by defined Attacker.

Integrity: defined as: data not being ever used in unintended, unauthorized or negative ways. Not stolen, shared, exposed, destroyed, corrupted, correlated or anything else regarded as negative by the data supplier.

Goal: > 99.998%.
[Product = Friend-Book 2.0**, Integrity =** All Types**, Data =** Personal Data Submitted or Observed Behaviou**r, Purposes =** Marketing**, People =** Paying Users**, Attacker =** Our Corp. & Any External Instance**].**

Constraints:
C1: Euro Privacy Laws.
C2: National Country Privacy Laws.

Issues:
I1: Marketing Partners as weak Links?

Assumptions:
A1: we (Product Development) are willing to invest in extreme data protection technology. No holds barred.

Risks:
R1: Legal pressure from National Security Agencies to share data. Mitigation [R1]: the data or access keys are not under our control, only user control and local storage ??

Planguage Example 3.10. How many stakeholders are specified in this?

The example shows identification, and specified concerns, for a large number of stakeholders (about 11). The presumed or approximate cost-effectiveness of satisfying, or *not* satisfying, any one of these stakeholders can be estimated. In some cases it will not pay off.

"A corporation is a living organism; it has to continue to shed its skin. Methods have to change. Focus has to change. Values have to change. The sum total of those changes is transformation."

Andy Grove (1936-), Retired CEO, Founder, Intel

Practical Tip 3.10 Quick Stakeholder List.

- *Get 2 people to brainstorm a list of at least 50 stakeholders within 30 minutes.*
- *So how many of these are you going to analyse the needs of ?*

Policy 3.10 Rich Stakeholder-Web Planning.

- **We will document, in our planning specifications, the relationship of what we are planning, to the related critical stakeholders.**

Why ?

- We want a rich understanding of the stakeholder interests, and stakeholder 'touching points', so we can plan to make stakeholders happy, and avoid unnecessary problems and conflicts.

Chapter 4. Value Delivery.

Planning *the value you want to deliver is a smart step towards actually delivering value to stakeholders.*

But smart value planning is not enough: the value plans have to ***actually work, measurably.***

And if they fail we need to find out ***quickly****, before we have invested too much in failure.*

Section 4.1 Value is Subjective.

Principle 4.1 Subjectivity:

**Values are determined
by the *subjective* choices
of many critical stakeholders.**

'Value' is far more than an economic concept.
Most people assume immediately that 'value' is mainly something to do with *economic* value: profit, earnings, income etc.

Multiple Values.
But, we are all well aware that we value many things in a non-economic way too. There is a website with a long list of human values [URL24].
A small sample: *Accomplishment, Compassion, Confidence, Decisiveness, Justice, Modesty, Perseverance, Practicality, Recognition, Responsibility, Service, Skill, and Wisdom.*

Value is usually very *multidimensional*, and economics is not usually a *major* part of the value dimensions. Not even in business and organizations.

As we can easily observe, different people, and different groups have different values; subjective values. They 'live' for these, and sometimes choose to die for some of them. We cannot stop them having a particular form of values, and we would not want them to stop us.

Sadly not everyone is willing to allow others the freedom to choose their value set. Deep and deadly conflict can result.

In business, the managers in current power, adopt to some degree, the organization's values, and in addition, managers have their *own* values- which if successful – will lead to organizational success, and lead to improving that manager's reputation.

The same holds true with sports coaches, and private enterprise managers. The 'coaches' have their own subjective values.

So, in order to get along with the rest of the world, at least with our critical stakeholders, whom we should be most concerned with, we need to learn other stakeholder's values: and *respect* them, and try to *help satisfy* them.

Changing their values is *not* a reasonable option. You can try, but it might be easier to acknowledge and respect other's values, in order to reach your own objectives.

One problem with values, is that we normally, in daily perception, describe the values with words ('honest'). Even the intensity or degree is described with words ('perfectly' honest).

We are satisfied with a rough perception, understanding, of others values, and of our own. *"I do not doubt her sincerity" . "You are very reliable."*

That rough value specification works pretty well at the person-to-person level, and when it doesn't, we get into conflict, and adjust our understanding accordingly!

In a larger than interpersonal context, meaning '*many people'* and '*up to millions of people'*, this simple way of relating to values and to other individuals is not a good idea. There will be too many misunderstandings; and not very good ways to reconcile them.

There are smarter ways to state values, and if we do not use them, we lose out to competitors who *do* clarify values.

What we need to do when there are many stakeholders; not just 'you and me'.

In simple terms, these are the adjustments we have to make, as we scale up from 'interpersonal relations' to 'mass relations':

- We need to be *systematic in identifying* all critical values of all critical stakeholders, not just whenever we happen to 'bump into' them.
- We need to identify values *early*, not wait until they are a problem.
- We need to *use numbers* to describe the intensity of the values; rather than just use nice-sounding adjectives, like 'superior'.
- We need to actually *measure* how well our strategies are working in practice: *early*, and well-before big problems, or failures are experienced.

- We need to systematically deal with a *large number of values*, and a corresponding *large number of strategies* to cope with them: at the same time. The individual human does this to a small degree, but the real organizational, and popular social demands for values are many more than the individual is naturally equipped to deal with.
- We need to recognize, and exploit the fact, that many values themselves consist of a set of sub-values. They are not simple and linear (like height, or weight) they are complex like *love* (URL10 A.), and like *'great employee'*, or '*impressive manager'*.

Managerial Competence:
Type: Complex HR Hiring Attributes.

Includes: {Initiative, Coping, ***Communication***, Intelligence, Compassion, Respect, Persistence, Winning Ability, Results Orientation, Delegation Ability, …}.

Planguage Example 4.1 A. Specifying a subjective set of the multiple dimensions of 'Managerial Competence'.

In the specification example above, we recognize that 'Managerial Competence' consists (subjectively) of many dimensions. And certainly there are quite different dimensions, for different organizational environments.

And even the first level decomposition above, might not be enough. So:

Managerial Competence**.Communication**

Type: Complex **Managerial Competenc**e Sub-Attribute.

Includes: {Presentation Ability, Summarization Ability, Listening Ability, Written Absorption Capability, Mass Inspiration Capability, …}

Planguage Example 4.1 B. A second-level decomposition, for 'Communication'.

Diagram 4.1. A subjective hierarchy of values regarding manager capabilities.

And this is before we take a crack at *defining* any one of these sub-capabilities, and before *putting a number on* our objectives. Like:

Presentation Ability:[Managerial Competence.Communication].

Scale: % of main points immediately understood by target audience.

Goal [Next Year] > 95%.

Planguage Example 4.1 C. Defining 'Presentation Ability' quantitatively. Defining the numeric variable with a Scale statement. Then defining the required future level with a Goal statement.

"Subjectivity as a governing law of the human sphere offers us the talent to cope with many of life's great stressors. It allows us to balance some of the most complex of social equations, like belief systems and spiritual predispositions. It also, and most importantly, shelters us from the independent indifference of objective reality, which in turn, threatens to challenge our egotistical, self perceived image as the centre of the universe and of all experience."

— Peyton Dracco [URL25]

I am not sure I understand this as the author intends, nor that I agree. But it sounds good! Subjectivity is a tool for helping us cope with complexity! TG [URL87].

Practical Tip 4.1 Agree on critical quantified values.

- *Allow individuals on your team or project to brainstorm their own professional critical value dimensions with regard to a single project or plan. 5 to 30 minutes*
- *Then charge them, as a responsible team, to come up with a list of maximum 10 dimensions that they can **agree on**, for the purpose of your project . One hour.*
- *Point out, that until each dimension is defined, for example with a Scale of Measure, they cannot be sure as to what they are agreeing to. 30 minutes each of 10.*
- *They should not 'agree' at the 'tag' level, but at the level of clear definitions, of the name of the concept.*

Policy 4.1 Respect Subjective Values.

- **We will note, and respect, the subjective values of our critical stakeholders, while trying to understand their values as objectively and completely as possible.**
    - Why?
        - So that we can deliver value to our stakeholders and satisfy their needs, so as to further our own agenda.

Section 4.2 Tailoring Value Planning For Better Alignment With The Stakeholder And Strategy Environment.

Principle 4.2. Value Tailoring

We need to capture
enough supporting detail
about stakeholder values,
and strategy attributes,
to cost-effectively insure that
we deliver those values,
as quickly as possible.

There are, as pointed out above, Section 4.1, a great many possible value ideas, that our critical stakeholders can subjectively 'invent'. Possible value ideas are not at all limited to a 'standard' set of values.

There may be few archetypes ('primitive models') of objectives, such as 'faster', 'better', 'more', and 'usability', 'effectiveness', 'security'. But we need to know, and consider *far more detail* about the 'stakeholder reality'. We need more than stakeholders *themselves* are even *consciously* aware of. Hidden threats and opportunities for example.

This requires adding detail about relationships, risks, issues, justifications, assumptions and many other such things that the planning language enables us to do easily.

They are as many and different interesting combinations of these real detailed stakeholder values, and their value parameters and descriptors, as the fertile human mind can conceive and imagine.

Think of *real* values as being *infinite* in variety. Archetype patterns are not enough. Not if we are going to understand and respect our stakeholders.

For this reason, our planning language must be able to *reflect and model* the wide variety of specific values that can pop up from our real stakeholder group. Most planning languages do not seem to allow for this principle *at all* [URL26].

Balanced Scorecard: an unbalanced way to plan values.

Maybe that, lack of definition, is why the 'Scorecard' needed 'balancing' in the first place. Balanced Scorecard when dealing with the non-financials, does a poor job of *adequately modelling* the non-financial values.

BSC does not *even* understand the need to *quantify all values (this is 'Core' Planguage specification)*, which the *financial* side of the Scorecard *does* understand. So, effectively, the Balanced Scorecard remains very 'unbalanced' to this day [URL62]. Quantification (of financials) wins every time, against non-quantified non-financial values.

But BSC does not give instruction on defining the non-financials, or even the financials, in a *rich* way so as to properly connect planning objects with the rest of the real world. For example by defining relationships, justifications, issue, risks, etc. (all this is 'Background' specification).

Some examples of tools to model a complex stakeholder reality.
The planning language, has a large number of devices for both tailoring and modelling a complicated reality. Planguage is open-ended for inventions or improvements that you might find necessary or useful.

Here are *some* of the specific planning language tools that assist in ***tailoring*** *[URL26 A] the definition and additional specification of a stakeholder value.*

You could add them to Balanced Scorecard if you wanted to. We have done so with BSC [URL62 B, Bank Case].

You could add them to any other planning methods that need these specification and planning tools like Quality Function Deployment (QFD) [URL35].

- **Definition of a Scale of measure.**

> **Scale:** % of the market.

- **Allowing compound sets of scales of measure.**

> **Good:**
> **Type:** Compound Objective**.**
>
> **Honest:**
> **Scale:** % of lines/all statements.
> **Kind:**
> **Scale**: Number of kind acts per day.

- **Allowing Scale Parameters. Generic Scale definitions.**

Scale: number of kind [**Acts**] for [**People**] in [**Places**].

- **Allowing and encouraging Terms in a Scale, and anywhere else, to be defined separately and thoroughly.**

Scale: number of **K**ind [Acts] for [People] in [Places].

Kind: *defined as*: showing love, compassion, helpfulness, tolerance, or helpfulness to others.

- **Allowing definition (= xyz) of a Scale Parameter in any type of scale level definition,** such as Past, Trend, Tolerable, Goal.

Trend [Next 5 Years, Acts **= Praise**, People **= Children,** Places **= Home]** 1.0/day.

- **Allowing any number of useful parameters in the Qualifier statement,** even if they are not *previously* specified in the Scale statement. ('By' and 'Method' are added in example)

Goal [Acts = Praise, People = Children, Places = Home, **By =** Parent, **Method =** Oral] 10/day.

- **Allowing a large number of 'background' statements to improve understanding of relationships, sources, and credibility of a single statement, or a larger Plan Object..**

Goal [Acts = Praise, People = Children, Places = Home] 10/day <- SGSG.
Justification: frequent exercise makes a habit.
Impacts: Reciprocity.
Auditor: Self, Phone Log.

- **Allowing specification data to cumulate gradually, as we need more detail, and as we get feedback from attempts at value delivery**.
  - Each specification of an objective or strategy is a 'mini database', of 'everything we need to know' about it. A semi-independent object.
  - We call this 'cumulation of information with core statements, and hopefully also background statements, and Commentary, ' a '**Plan Object**'.
  - 'Plan Objects' are the primary unit of planning: they include objectives, strategies and implementation steps.
  - All plans consist of a set of *related* Plan Objects

- There is only one single Master version of a Plan Object. All other references must use the Plan Object Tag to find the current Master.
- The Master version of a Plan Object is always consistent with itself. All others must reference the Plan Object name Tag.
- There is no 'final' plan specification. Just new improved versions of a set of Plan Objects.

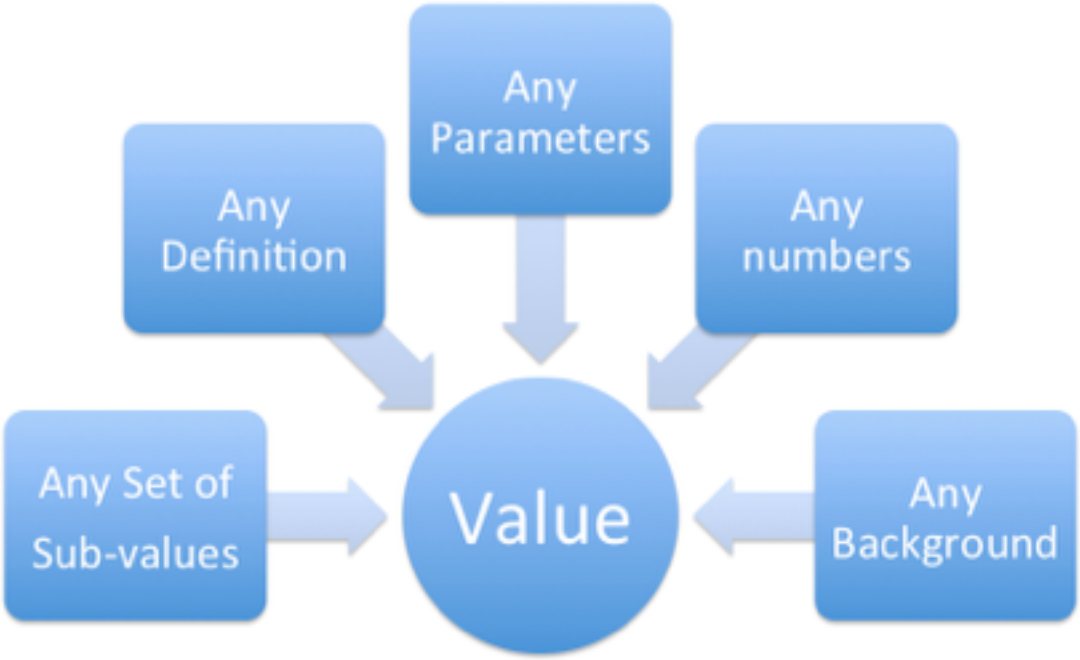


Diagram 4.2. A stakeholder value can be specified by a very wide variety of tools, so that it accurately corresponds to local reality. General specifications are generally a bad idea. Be specific.

"If you are going to achieve excellence in big things, you develop the habit in little matters. Excellence is not an exception, it is a prevailing attitude."

Colin Powell

(1937-) Former US Secretary of State.

Practical Tip 4.2 Capture Ideas in flight.

- *During meetings, where specifications are presented, and being discussed, make sure someone is assigned to pick up the fleeting remarks, that suggest we need to add some information, like a better definition of a term. "What does XX mean?" is trigger enough. If **anyone** has to ask, it is not 'excellent' enough.*
- *Maybe everybody participating on a team can be encouraged to take notes, and suggest specification improvements, in writing, perhaps as a quick meeting remark, to the Plan Object Owner.*
- *Be in a mode of constantly cumulating a planning object specification, with a view to making it more complete and more intelligible.*
- *The attitude should be spelled out as 'rather ten too many than one too few.'*
- *Do not let answers be just 'oral', at the meeting: write answers into the specification itself, so the question is answered for other people, and for everybody 'later'. This is low-cost insurance against misunderstandings. Talk is just talk; write is right!*

Policy 4.2 Plan Accurately

- **We are committed to:**
  - **invest cost-effectively in our planning specifications;**
  - **especially with the detail and realistic tailoring,**
  - **which helps us understand stakeholders correctly,**
  - **and do the right things for our stakeholders.**

- Why?
  - Because it is generally a very low-cost investment, that can pay off handsomely when you get it right early, and are seen by your stakeholders, to tune in on their reality.
  - And you can communicate this 'deeper understanding' to all parties concerned, over time, and in many places.

- **On the cost and volume of plan cumulation.**
  - The cumulated detail of a Plan Object, such as an 'Objective', costs almost nothing to maintain. It is digital.
  - The volume of detail can be ignored when not needed or wanted. It is 'buried' behind Parameter names and other symbols like quotes ('xxx'), Notes, and Background Parameter names [Figure 3.1 B].

- It may be consciously, automatically [URL73], or by default, hidden from normal view, until it is desired. Until we need to drill down.
- We can always extract the core elements, and any additions that we want to have, for particular presentations.
- If we are focussing on the top-ten objectives, for example, it does not matter if each plan object is a 'line' or 'ten pages' each. It is still low volume. But high criticality.
- Every little supporting detail is potentially of use, and of interest to some planner or analyst, some time. And it may well make a critical difference. It is low-cost insurance.
- let me make it plain: I 'hate' superfluous bureaucracy. But I love to know what might help me avoid problems, and might help me deliver real value.
- So '**I'd like to make things as simple as possible; but not so simple that I fail'.** No. Einstein did **not** say that. But he *sympathized* with it![24] You can quote ME on it however!

Section 4.3 Conflicts of Interest.

Principle 4.3 Viewpoint Conflict.

**Positive value
for one stakeholder
might well be viewed as
negative value
for another.**

Stakeholders might '*apparently'* agree on some objectives, at the level of the value 'name', such as 'Security'.

But they will probably need to plan for different *performance* levels, of differently defined aspects (like Scale, and Scale Parameters) of the same Name Tagged objective, like 'Productivity' or 'Service Capability'.

This is one reason it is important to clearly differentiate these *really different* requirements, contained in the same objective 'area'.

Apparent conflict of objectives, might well disappear, when the objectives are detailed, for each stakeholders different interests. Or, on the other hand, potential conflict might first appear when things are detailed.

Everybody is 'for' *good, healthy food*. But as individuals, even in the same family, we have quite individual opinions about what is good, and what is healthy.

A 'good compromise' is not when everybody agrees to the same food, but when the food provider makes plans to please all the family members *as a set of individuals,* fairly well.

Planning systematically with a good set of planning tools.
We need to plan value delivery by:

- identifying our distinctly-different critical stakeholders (Identify).
- Then identifying their *needs*, that make them distinct (Clarify).
- Then figuring out which of these needs we need to serve earliest (Prioritize). For practical, risk, low cost, perceived effectiveness and political reasons.
- Then we need to prioritize with respect to the value for resources of the needs (Efficiency).

and to plan all that, and to articulate our plan, we need some 'differentiation' tools.

Diagram 4.3. Some examples of various Planguage specification tools, so that a 'same name' (like 'Productivity') specification can be simultaneously tailored to several stakeholders, in a realistic and harmonious way: no over-generalization of a requirement. Many 'sizes' to fit many stakeholders, and to fit varied needs.

Here is a realistic example of using these tools to supplement the core specification, with background information, including 'administrative' Background.

Usability.**Intuitiveness**:
Type: Product Line Critical Quality Level.
Version: 28 July 2014. 15:02

Scale: Time needed to Correctly do a defined [**Task**] by a defined [**User**] given defined [**Experience**] and defined [**Knowledge Access**] for a defined [**Product**].

Past [Date = January 2014, Task = Assemble + First Use, User = Average, Experience = None, Knowledge Access = On Product Help + Manual, Product = Product Line X] < 10 minutes.

Record [Date = June 2014, Task = Assemble, User = Professional, Experience = Years, Knowledge Access = On Product Help + Manual + Internet + Tele Help Line , Product = X 3.0] 1.5 minutes.

Trend [Date = End 2016, Task = Assemble + First Use, User = Novice, Experience = None, Knowledge Access = On Product Help + Manual, Product = X 5.0] 15 minutes. "Getting worse as product gets complex"

Goal [Date = End 2018, Task = Assemble, User = Professional, Experience = Years, Knowledge Access = Online Help + Manual + Internet + Tele Help Line, Product = X 3.0] 1.0 minutes "would be new record".

Goal [Date = Mid 2018, Task = Assemble + First Use, User = Novice, Experience = None, Knowledge Access = On Product Help, Product = X 5.0] 1.5 minutes

Planguage Example 4.3 Useful background specifications in an objective. Different Goals for different stakeholders. The + indicates members of a set of things, or a list of things.

In this example, the Past, Record, and Trend are all 'Benchmarks' which is one type of Background information about the plan objective. They are *not the core objective itself (*not our *future* targets or constraints).

But these Benchmarks are useful *background* information. Background specifications help us with many planning purposes. Among these purposes are:

- determine if we have set reasonable and interesting, targets and constraints, with respect to benchmarks.
- determine priorities: what to do to satisfy a powerful stakeholder; what to do to avoid unnecessary risks; which strategies to deploy at a given stage of value delivery; and more.
- assess the degree of riskiness of an objective: is it far beyond a Record level?
- distinguish between the backgrounds of different stakeholders, so that individual stakeholder's targets and constraints are specified, in relation to this more-tailored information.

The 2 Goal statements, have 2 different users, with 2 different Knowledge Access situations: and 2 different Goal levels. Goals are an expression of a 'very satisfactory' and 'successful' level; for those users, under those other specified conditions, at that specified time.

Notice that the 2 different Goals, effectively *decompose* the problem, so that we can go for the second one by 'Mid 2018', and then focus on the first one by 'End 2018'. We can choose to deliver one value to one stakeholder first; and then turn our resources to the other stakeholder.

"The value systems of those with access to power and of those far removed from such access cannot be the same. The viewpoint of the privileged is unlike that of the underprivileged."

Aung San Suu Kyi. Nobel Peace Prize Winner. Burma. (1945-)

Practical Tips 4.3 Distinguish your stakeholders needs.

- *Make sure you always have one or more useful benchmarks (Past, Record, Trend) in every objective. Specify benchmarks before you specify targets and constraints.*
- *Make sure you clearly distinguish between different stakeholders, deadlines, and other critically different conditions (where, who) in 2 or more targets and constraints.*
- *If you combine stakeholders (and other conditions such as where, doing what), and thus you 'over-generalize', you have lost the power to differentiate, and to give early high value to stakeholders.*

Policy 4.3 Useful Distinctions

We will take the time to ferret out the critical distinctions in the needs of various types of our critical stakeholders.

- Why?
  - Because it is likely to result in a better more profitable product.
  - And the cost is probably small (1 day to 1 week is a good start) compared to the potential benefits.

Section 4.4 Estimations of Strategy Impact: Uncertainty.

Principle 4.4. Differences between estimates and reality.

***Expected* values** **will probably differ** **from *really delivered* values.**

We tend to be optimistic, or, to be fed expectations by optimistic inexperienced believers of a strategy.

Dealing with optimistic estimates.
There are systematic ways to compensate for this over-optimism, in the planning language.

We can ask for a likely range of uncertainty (for example 40%±30%, versus 40%±2% which seems 'less risky', but 'less promising') and we can ask for the **source** and **evidence** of these estimates.

The *evidence* for an estimate, and the *source* of that evidence help determine the **credibility** of the estimate (we put **credibility** on a scale of 0.0 (none) to 1.0 (Perfect) [Table 9.5].

The best tactic of all, is to quickly test and measure the asserted estimate *for real*, 'next week'.

Practice usually beats theory here. This is an 'experiment'. Win or fail; we learn fast.

Making the best of a less-than-ideal reality about estimating.
However 'bad' our initial and optimistic estimates *might* be, they are better than nothing.

Bad estimates are better than nice-sounding words (like 'a sure winner', and 'the new paradigm': they sure sound nice!).

What we *can* do, with estimates of the values to be delivered, is to try to determine *fairly good choices in the short term*, from the large collection of possibilities. Our choice does not always have to really be 'the *best* choice', as long as it is usually 'up there' in the *top-few good* choices.

There is no point in debating for a month, about doing something that takes only a week, to verify how well it really works. Action beats discussion. But quick quantitative planning for an hour or a day [URL5] beats a week of wasted effort.

We don't always have to be 'right' about the impact: as long as we *quickly measure,* to see *how wrong we were*.

It is not about perfect optimization. It is about having a good betting strategy in the face of high uncertainty.

Accurate Estimates is not the point: Saving decision making time, with reasonable estimates is the point.
It is not about making right estimates, it is about generally saving time, by making pretty good decisions *quickly*, on each of your fifty '2% value delivery steps' of your projects.

Evolving our value delivery and our feedback process: one step at a time.
Our major assumption is that we are going to deliver our *big change* as a series of smaller measured, cumulative changes.

We will betting no more than about 2% of our gambling 'wad' (capital, budget) at a time.

And we will be learning fast, about 'value actually delivered', 'real costs', and 'who are reliable' sources of good ideas and good estimates.

The probability of reasonable reliable estimates improves when:

- The strategy is **described in fool-proof detail** [Quote 4.4 A]**.**
  - Just enough detail to to ensure that what gets implemented is exactly what the estimator *thought* they were estimating.
  - Obviously, if too much is left to the imagination, and to the discretion of the implementer, the final measurement of the different, resulting, implementation, will differ from the expected results, even if the estimations were 'perfect'.
- The **planned measuring process** (the Meter specification, Section 1.6) is appropriate, reliable enough, and has a clear relation to the estimates being made.
- There is some **evidence** from past experience, what the result will be.
- The **source** of information, about the estimated results of a given strategy in the past are **reliable**. They know 'how to measure', and they know what 'credible results' are.

Objectives Example (simplified)
Planguage Example 4.4 A. An objective to start the search for a strategy.

New Methods Adoption:
Type: CTO Objective.

Ambition: to increase the rate of successful, and long-term, adoption of methods, which give improved benefits or costs.

Scale: The % of [**Groups**] that adopt certain [**Methods**] Successfully for over 1 year.

Meter [Groups = Systems Engineering, Methods = Estimation Methods] Self measurement and reporting, using a CTO-supplied template

G1: Goal [By = End Next Year, Groups = Systems Engineering + Software Engineering, Methods = Estimation + Measurement] > 50%

Corresponding Strategy Example (simplified, at 'Gist' level)
Planguage Example 4.4 B. A strategy that we hope will help us meet our Goal

Full Support:
Type: Strategy for change processes.

Description:
D1. Free (sponsored by CTO budget) deep (3 to 5 days) training for employees.
D2. Free expert coaching until Success (CTO Budget).
D3. Public Visibility of Team Results, both New method, and Knock-on effects (D4).
D4. Measurement of Knock-on effects of the better method on related factors such as Project Value Delivery, Value/Cost ratio.
D5. Cash Bonus to Group for successful measures.

Estimation Example (simplified)

Planguage Example 4.4 C. A first estimate of how well we think we will meet our goal with the strategy suggested. The '->' icon means 'impacts' . Worst-Worst means 'most pessimistic'.

Estimated impact: See[URL21]

Full Support -> Goal G1: 60% will be met.

Certainty: ± 20% of the full Goal level (i.e. 40% to 80% adoption).

Evidence: Last 5 years in our Corporation, successful adoption rate 30-60%.

Source: CTO statistics.

Credibility Level (Evidence + Source): 0.7.

Worst-Worst Case estimate: 40% x 0.7 = 28% Successful Adoption.

In the simple examples above we have tried to show how a strategy impact, for an objective, can be estimated, based on evidence from a known source.

This results in a conservative estimate of about half (28% of way to Goal) of the initial optimistic estimate. This is the part of the Planning Language called Impact Estimation [URL21].

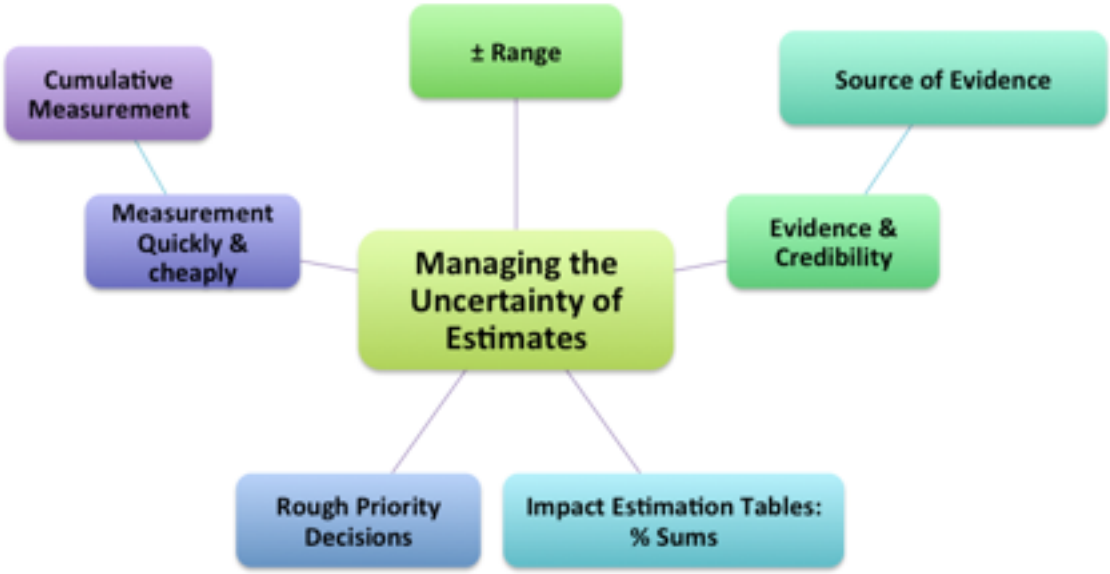


Diagram 4.4. Some factors that help us manage the problem of understanding and correcting, our estimates of the effects of strategies, on our objectives and costs.

Quote 4.4 A

"A common mistake people make when trying to design something completely fool proof is to underestimate the ingenuity of complete fools."

Douglas Adams (1952-2001) Mostly Harmless

Quote 4.4B.

Practical Tip 4.4 Estimate strategy costs and benefits.

- ***Costs?*** *When people suggest a strategy, ask them to estimate the rough magnitude of the long-term total cost to build and maintain it.*
  - *This makes them think, and recognize how little that they have thought about it, up to now.*
- ***Impact?*** *When people suggest a strategy, ask them to estimate what % of 'our most critical objective' it is expected to contribute to.*
  - *If they do not know what 'our most critical objective' is,*
  - *or do not know 'exactly how ambitious it is',*
  - *then politely remind them that 'meeting the objectives' is the **only** valid reason for looking at any strategy.*
  - *Ask them to come back when they can make an estimate, for their strategy and defend it. Ask politely.*

Policy 4.4 We will let Estimates *Guide us*, but we need to let *Measurements* Confirm validity.

- **We will normally make reasonable estimates, for the impacts of suggested strategies, on our goals and constraints, and use these estimates to decide, 'what to do first'.**
- **But we will never bet *heavily* on any *estimates*.**
- **We will believe, and commit, based on the measured results, of our own actual implementation.**

Why?

- *Estimates can rarely be trusted to be anything but 'wrong', or 'very wrong'.*
- *Numeric estimates are better than words ('very', 'substantial') but even **wrong** numbers have a lower potential for misunderstanding; so that it is easier to detect that they are 'wrong' numbers.*
- *Even actual early measurement can mislead us, as to the long term result: but it is more 'real' !*

Section 4.5. Dynamic Strategy Planning, to Fit Costs.

Summary:
- Costs Are Managed Step-by-Step using 'Strategies' as ultimate cost bearers.
- Objectives 'drive' costs only indirectly.
- We do not believe in conventional 'estimation'.
- 'Dynamic design to Cost' is a proven superior cost-management method.
- 'Plan within Costs'.

Principle: 4.5. Incremental Plan-For-Cost.

You can meet tight deadlines and budgets better,
if you select cost-effective strategies,
and get incremental feedback,
on their real costs and values;
keeping what succeeds, dumping what fails,
as you evolve to your goals.

Short version of Principle 4.5: 'Manage costs evolutionarily'.
'Evolution' = iterative + incremental + measurement + learning + change. See diagram 9.8 Value Management Process.

Cost Estimation is Difficult.
You cannot logically estimate 'the cost of reaching defined *objectives'* by analyzing those objectives *alone*. The *real* costs are embedded in the *strategies* you pick, to satisfy the objectives.

And if your specified objectives are *unclear (a too frequent situation)*, there is *no hope at all* of understanding costs they might require.

You do not have enough cost-driving parameters specified (COCOMO estimation method uses about 60 cost drivers, URL11), to successfully pick a cost-effective strategy. And you do not have enough cost-consequence data on the parameters you *do* specify. You will easily be wrong by factor 10 or worse.

If your management, customer, client, or you, asks, "What is it going to cost to reach these objectives?". The answer is, "nobody can know, until we have a correct set of strategies for getting there on time."

And even then, it is difficult. Difficult to know your strategies are a 'correct' set. A set that meets all Goals, on time, within budget. Difficult to know what they will cost. Let alone their effects on Goals.

Foolish Estimates.
There will be fools who will come up with an 'estimate' based on the objectives alone, without considering the necessary strategies. Probably using unclear objectives, since that is their culture. Anybody can spout a number, based on a false unclear premise and call it an 'estimate'.

'Just calling something by a name, does not make it so."

Quote 4.5 A. (G. M Weinberg, 1933-)
Jerry has a lot of wisdom, and many books (one of which we wrote together, Humanized Input,1976). www.geraldmweinberg.com
Compare this quote with Richard Feynman on 'naming': [URL79] Quote 10.7 A.

But that is why these 'estimators' are fools, and why you would be a fool to act on their estimates.

The good news is, that if the fools try to give an 'estimate' based on *faulty objectives*: you have then been duly warned about them. You do not really want such people to determine your career success or failure.

Re-training them might be an option. Foolishness is usually due to bad culture, not to low intelligence.

Objectives *Drive* Costs - But Strategies *Determine* Costs.
The target value levels, and delivery deadlines, of objectives '*drive'* the costs.

The more ambitious the objective, in degree; and the nearer the deadline for reaching that level, the more-likely that costs will increase.

If you do not care about the value levels, or which decade you get them, then the cost can tend towards zero.

But, at an extreme, even some ambitious objectives can, if you get lucky, actually allow you to find zero-cost strategies to reach the objectives, on time. *If* you get lucky, punk.

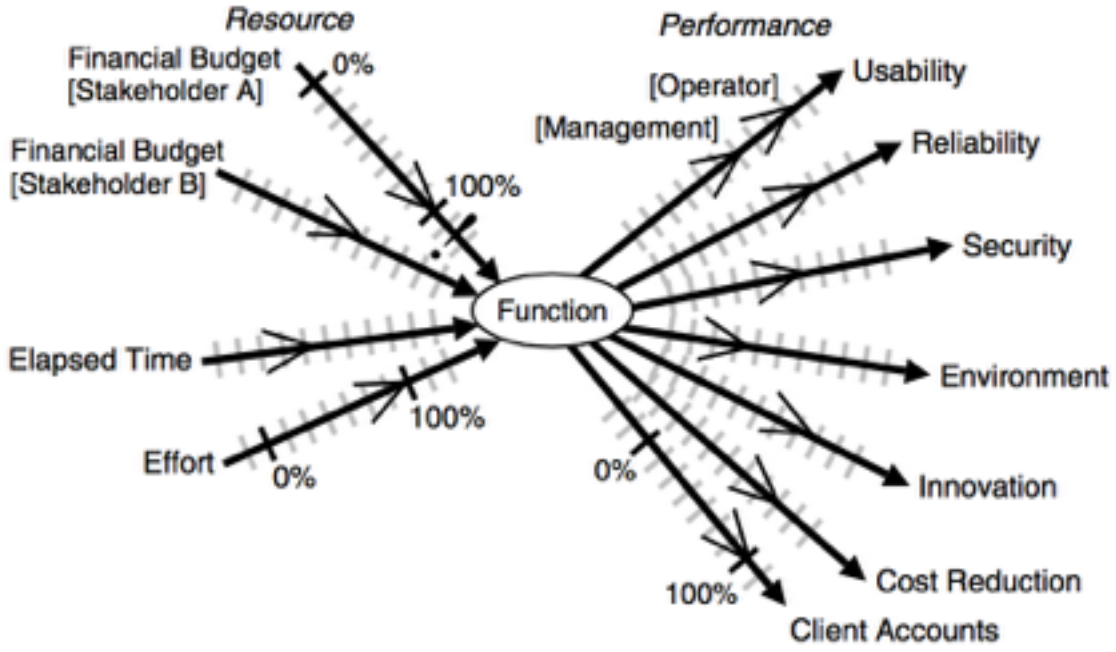


Figure 4.5 A. ***Th'e inevitable set of attributes for any strategy.***

All strategies have three major sets of attributes. Functions - what they 'do'. Costs - the resources they consume. And Performance attributes - degrees of goodness (values), including all qualities.

No matter what initial reason you selected a strategy for, for example for 'high security', you will get the ***rest*** *of these attributes, like it or not.*

So, the 'prudent planner' (nice concept!) has to become aware of these 'side effects' (side effects are ***not*** *the main reason we chose the strategy) and decide if we can tolerate the side-effects.*

Impact Estimation tables are a tool to help us map the side effects.
Source [1, Fig. 5.5]

It is the *strategies that determine the costs*.
You can, logically, only 'reasonably hope' to estimate the costs of a set of ***well-defined*** *strategies.*

Though we, unfortunately, are normally not good enough at defining strategies very well [Chapter 2, Part 3, Part 4].

And even then, with a 'reasonable cost estimate', there are problems.

Only a rough estimate is possible: there are so many unknown future conditions for the strategies. Unknowns, and usually unknowables.

Unknowns in strategy deployment, the host environments, operation, under stress, cost impacts over a lifetime of change, and new stakeholders, and new stakeholder needs.

You are good, and lucky, if you get 'order of magnitude' cost estimate, *before* you measure the strategy in practice.

A suitable strategy can be 'free of costs', or 'paid for already', or 'already deployed in a current system' (no additional cost). These cost possibilities are independent of how usefully powerful a strategy is, in delivering results to your project. Sometimes you can get *more* than you pay for.

But, don't bet on it.

Some strategies might even have a *negative* cost: pay for themselves and earn money.

Are you feeling lucky today?

Cost is not *always* directly proportionate to the ambition levels.

Design to Cost.
Smart, knowledgeable, motivated, strategy planners can consciously seek and find powerful strategies with low costs, and fast delivery timing. Engineers call this 'design to cost' [23, Miles, URL11].

In general, any given strategy can be replaced by another much-cheaper strategy (but still sufficiently effective), which would then change the costs substantially.

External suppliers, or internal Empire builders are not motivated to reduce costs by planning better. The more outrageous the cost, the more they 'win'.

Management needs to hand the 'strategy planning' to smart, motivated, strategy architects, with an assignment to find smarter strategies, and cheaper ones; before entering into dialogue with builders and suppliers.

Maybe the notion of 'estimating a large project cost' is irrelevant?

Getting necessary values delivered, at costs we can afford, by managing the design and feedback dynamically (design-to-cost) seems to be a much more useful paradigm.

'What will the project cost?' is a narrow bean-counter's view of the world.

'Can we profitably invest in getting more profitable and valuable?' sounds like a business-person's question, to me.

Finding Cost-Effective Strategies needs to be an Evolutionary Process.

This 'strategy architecture' process is not simply an up-front, one-time process.

You might well want to integrate 'strategy architecture' into an incremental project delivery process.

Then you can see how good the architecture is, before it is too late.

Get realistic feedback, learn, and change strategies when necessary.

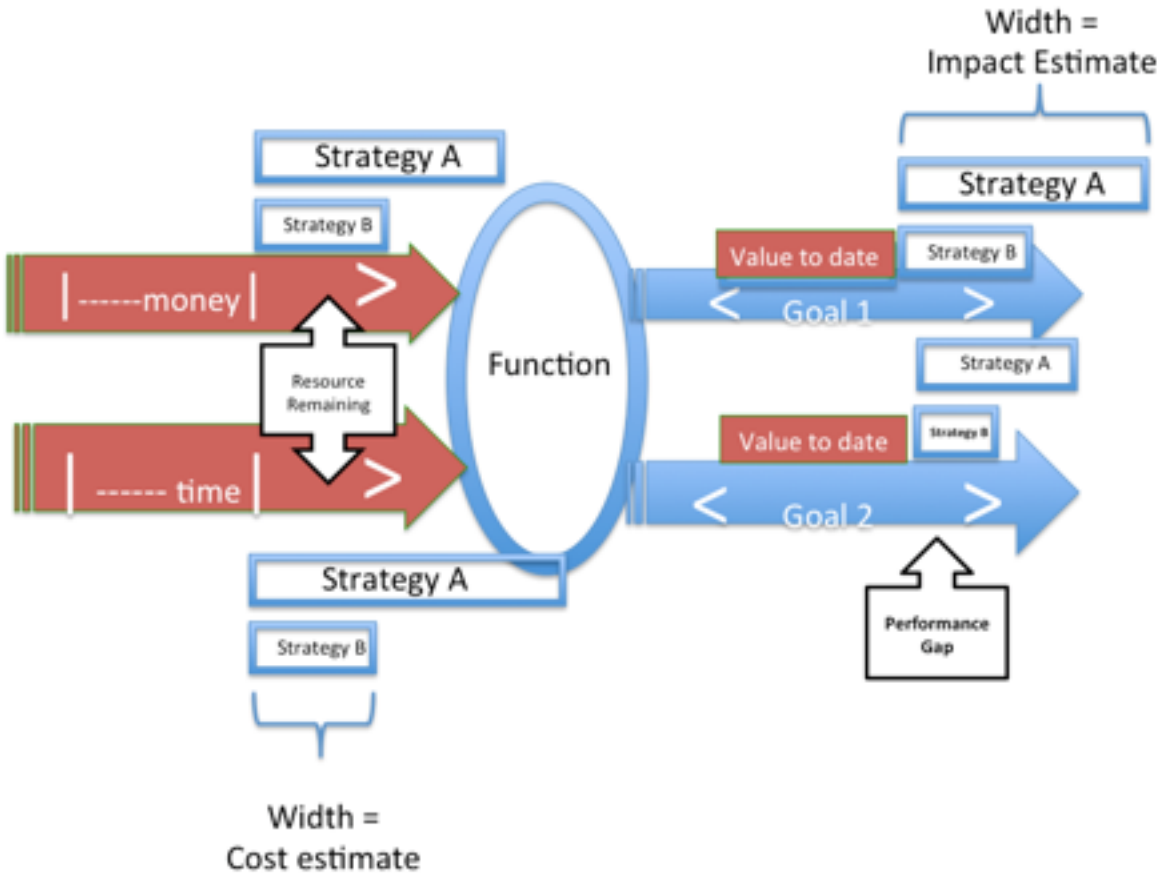


Figure 4.5 B ***Quantitative multi-dimensional strategy evaluation.***
(used earlier in 2.5).

The really-implemented strategy (the [rectangle] symbol here, with 'Strategy A' and 'Strategy B 'text in it) determines not only the 'degree of the value delivered' (width of rectangle), but also the multiple types of consequent costs, and their degree (cost proportional to rectangle width, a Planguage Icon).

'Performance Gap' is the remaining distance to the Goal. This is not a 'simple' picture, but it is more realistic. Note that both Strategy A and B will both reach our goal levels (get past the > Goal 1&2 icons).

But B is cheaper, and fits our Budget levels.

So, you can *try* to estimate the cost of a well-defined set of *strategies*, to hopefully meet a well-defined set of *objectives*; within defined *constraints*.

And with large-scale high-tech, or complicated organizations, your estimates might still be wrong by a factor of 'only' three.

There must be a better way?

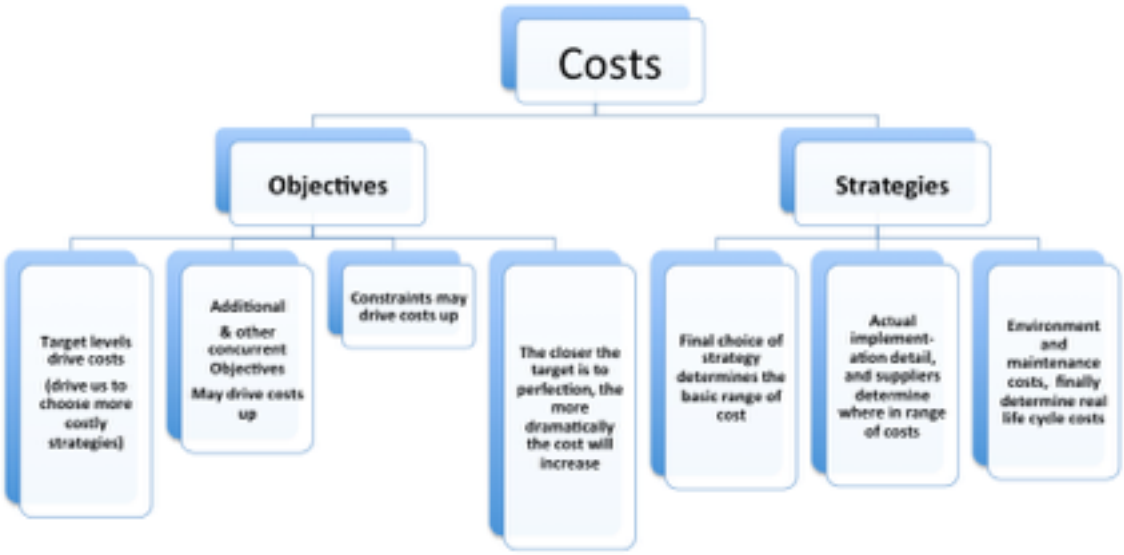


Figure 4.5 C ***Some factors affecting costs.*** *The more you know, the better you can estimate. But still, it is very difficult to be accurate. 'Especially about the future'.*

We need to adjust our plans, to fit our pocket. Dynamic Design to Cost.

The only way I *know about*, that actually has a documented industrial record of success, for delivering a project *on time* and *under budget consistently* is to do '***dynamic*** design to cost' [Case 2.5, URL11, Quinnan, IBM FSD [URL41], 'Cleanroom'].

Few people know about the 'Cleanroom' project management method. IBM never 'marketed' it. People seem to learn ideas that are packaged, patented/copyrighted, and sold; rather than what is truly useful knowledge.

Dynamic Design to Cost, in 'IBM Cleanroom' is an 'Evolutionary' process: defined as 'increments with feedback and learning'. Very much like the 'Plan Do Study Act' Deming Cycle [URL42]

That means, we start off with a pretty good cost estimate (or, in IBMs case a stipulated, contracted, and accepted estimate), and then frequently (like 50 times during a project of 1 to 4 years), we measure *actual* costs, continuously when we increment value delivery, 2% of cost at a time.

Then re-design, if necessary, at each value delivery cycle, so you can move nearer to the cost targets you *have to live with* (budget and deadline). [URL12, IBM FSD Cleanroom].

The 'dynamic design-to-cost method' does *not* work by 'improving our ability to 'make an estimate'. It does *not* work by 'big up front estimates'.

Dynamic Design-To-Cost works by being agile (learn true costs early and frequently, and change rapidly), and 'lean' (face your problem early, upstream). It learns to manage costs, to keep them within your budgets and deadlines, by *design (finding cheap enough strategies)*: not by estimation.

For example: here is a simplified model of the decision-making.

If you have done ten 2%-cost-steps (20% of project) of fifty planned value-delivery steps and the results are (chart below):

- you have used 40% of your cost budget (2x too much, proportionally)
- you have used 80% of time to deadline (4 x too much, proportionally)
- and achieved 20% of your value performance targets (proportional, to planned costs).
- The strategies did deliver the proportional value expected (20% Value for 20% cost), but they are in fact too costly!. They are not cost effective strategies.

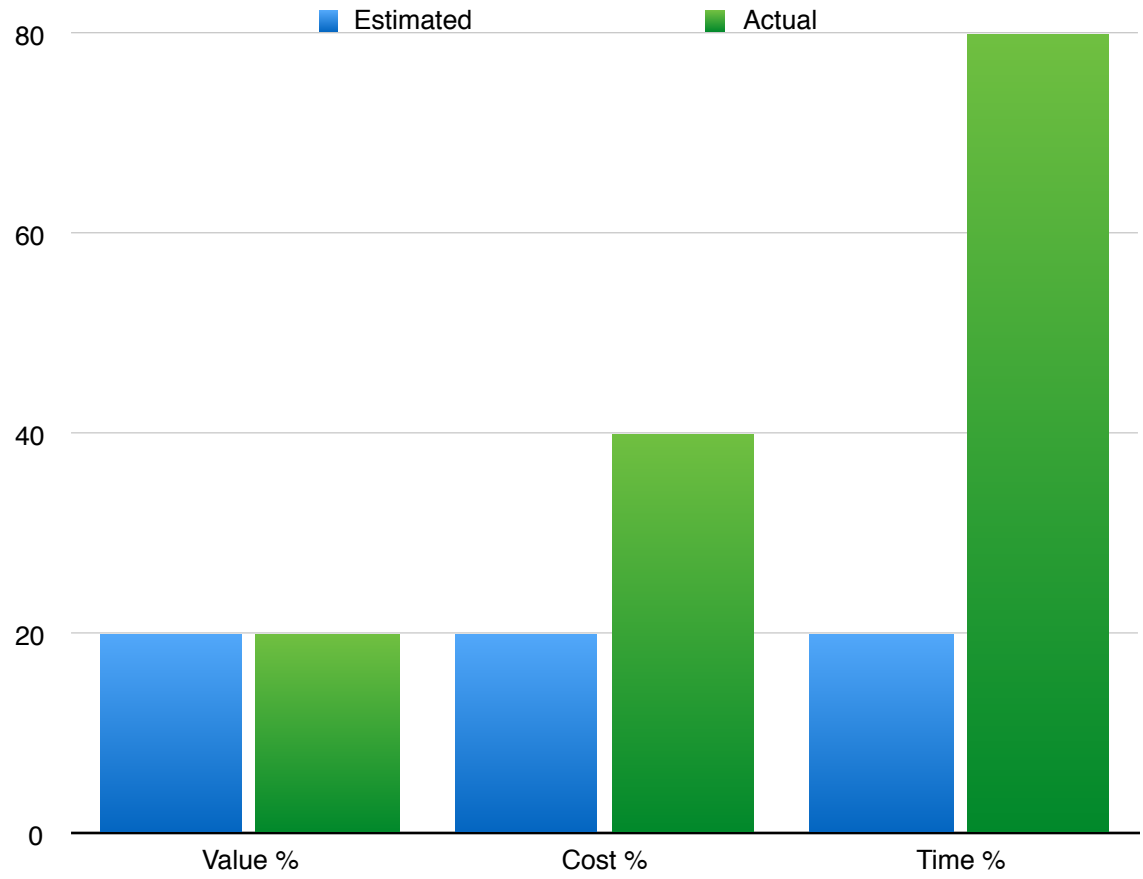


Chart 4.5 D. ***A snapshot 20% of the way through the project budgets. The basic process of 'multidimensional evaluation of progress towards Goals within Budgets'.***

Theory*: At 20% of planned value delivery cycles (10 of 50 planned 2%-cost iterations), we delivered 20% of value, as planned with current strategies. But* ***actual*** *incremental costs are far too much. And if we do not act decisively* ***now****, meaning 'change to cheaper strategies', we will fail to deliver planned value by the deadline, and and/or fail to deliver planned value, when we run out of budgeted money.*

Practice*: For a real example of one of our clients using such a snapshot to evaluate relative progress towards numeric goals, see Figure 4.10 Confirmit Case. This dynamic design-to-cost method enabled them to normally reach as many as 25 simultaneous value Goals within the quarterly product release cycle, 12 weeks of value delivery. Fixed deadline, and fixed staff; but using this feedback to decide where to deploy their resources in order to meet all 25 Goals by the deadline. For years on end. A very competitive successful international business.*

The 2 Estimation Elements in 'Design to Cost'.
OK, I admit there is *some* 'estimation' built into this 'design-to-cost' process. But **not** one big upfront estimate.

1. You estimate, and then re-estimate, repeatedly, based on 'costs to date', you *extrapolate* and say something like 'if we continue with these strategies, then we will run over budget, and past the deadline. So, we must change strategies, and we must do it **now**.'

2. In addition to the cost and value extrapolation, based on incremented facts, and on hard credible evidence, we use a *second* sort of estimation:

'what will candidate strategy X cost, in time and/or money?

This is 'Impact Estimation' [URL21, and Section 5.3 below].

If you do not solve your 'cost overrun' problem *early* and *incrementally*, the long-term cost-to-budget *deviation* will become far too large, before you even realize that it is unacceptable. And it will be too late to *recover* spent money and time.

Dynamic Design to Cost as a defence against arbitrary budgets and deadlines.
'Dynamic design to cost' as a management process, is particularly interesting to understand, when you do not have the luxury *to estimate how much you need or want, for your own scheduling and funding purposes.* You are not *asked*, you are *told* the costs and deadlines.

The government client, or other powerful forces, set a deadline for you; and they allocated a fixed-cost budget. Your salespeople 'happily' won, as low bidder of a fixed-price contract. *You*, however, are then stuck with the problem of 'making it happen', on time, under budget.

Salespeople don't care about your problem, they got their bonus based on the sale, not the profitability of the contract! Organizational disconnect.

If the ambitious quality levels (example: 99.998% for availability, 99% employee retention) that were actually expected, or assumed, are not defined, and not formally required; your answer is easy. Cheat!

Reduce quality and performance, until you can deliver the system or project 'on time' and 'under budget'. Who will notice?

And that is what people unfortunately do – cheat - reduce qualities to deliver within cost constraints - since well-stated qualities are rarely specified, and even more rarely *quantified [Chapter 1, Part 1]*. So it is easy to cheat on unquantified quality, without being noticed.

But, the moment you actually *have to* do it cheap, fast, *and* contractual world-class quality; *you have a problem, Houston*. And that is where incremental *dynamic design-to-cost* can help you, as it did IBM Federal Systems Division [URL41]. "Always on time and under budget, for years" they were!

The Conventional Project Model Is the Problem

We think, along with many others, that the old project notion of the 'fixed cost', or the fixed deadline, is obsolete. But it still dominates our practice.

The 'new' model is that we focus on 'managing value-for-costs', and we focus on getting 'sufficient value' to serve our purposes. Let us call that 'Value Efficiency Management'. But it requires a 'new' talent: ability to *quantify and measure values*: just as 'numerically' as we quantify, and do accounting for, money and time.

If we assume that each value-delivery cycle, can be planned, so that it actually has a value exceeding its' costs. We can just 'keep on rolling along', as long as most results are delivered early, and most all increments are really profitable.

This presumes that a system/product/service development, or organizational improvement, can be delivered in increments. We believe this is generally possible, as discussed elsewhere in this book [Chapter 5].

Case 4.5 Dangers of Committing to Qualities without understanding costs.

A Norwegian client of ours signed an international contract to deliver a civil-aircraft telephone system, in one year, for a fixed price of $20 million.
But the hardware and software, each independently, had to have 99.998% availability, according to the signed contract.

No one knew the cost of this availability level, and no one of them had ever achieved such a level.

No one knew *how* to do it, technically, either, the CTO, Kjell N. admitted.

No one even in their larger parent corporation (ABB) had *ever* delivered such quality, especially for software components, he told us, when asked.

We knew that the AT&T 5ESS Telephone Switching system cost about 2,500 engineers, 8 years to deliver, citing as a primary value of the entire project, '99.980% availability'. The state of the art at that time.

The client's new CEO (assigned to save the company from going bankrupt) agreed with me, that the contract commitment was foolish, and that they were *gambling the existence* of the small (100-person Oslo division) company, and were in fact seriously bound by these contract conditions.

The technologists and salespeople did not understand that, yet.

We got them doing Evolutionary value increments very quickly, and they actually made a profit in the present and future years. The happy CEO told me later.

But we are pretty sure that if they had used their normal methods, they would have lost money, as indeed they had, for the previous several years.

Case 4.5. The Fixed Contract that threatened company existence. Fixed price, fixed quality, and fixed deadline, but unknown strategies. By gradually trying to deliver expected qualities, and learning fast, they managed to get sufficiently acceptable deliverable quality.

Quote 4.5 B Data First.

"I never guess.
It is a capital mistake to theorize before one has data.
Insensibly one begins to twist facts to suit theories, instead of theories to suit facts."

Sir Arthur Conan Doyle (1859 - 1930), "The Sign of Four, A Scandal in Bohemia."

I leave it to the reader to deduce who this is portraying.

Practical Tip 4.5: No Cure No Pay Contracting.

- *Ask your potential suppliers if they are willing to guarantee fixed prices for delivering a defined set of high quality results measurably; not merely to get paid for **building** a defined strategy.*
- *Ask them if they will do this, contracting in cycles, on an incremental basis (flexiblecontracts.com) – stopping when they can not continue to deliver incremental results and value, for agreed fixed costs.*
- *Ask why you should trust them, if they will not, or cannot, do these things? [URL6]*

Policy 4.5 Strategies Determine the Costs, Not the Objectives.

- **We will practice that 'estimation of costs' needs to be based on specific, well-defined, and correctly implemented strategies, not objectives alone.**
- **And then, costs need to be *confirmed* evolutionarily by 2% implementation-steps, and measurement. Before strategy commitment.**

Why?

Based on experience and facts, not guesses or beliefs:
we need to quickly and incrementally, get results confirmed, by our own experience, before we go too far in the wrong direction.

Section 4.6 When the map differs from the terrain, believe the terrain.

Principle 4.6. Observation Beats Estimation.

**It is easier to observe
values actually delivered,
than to estimate them correctly
in advance.**

If you have decided to have a continuous series of short (1 week for example) value delivery cycles, as we recommend, then it quickly becomes *easier to measure reality*, than to *speculate about it* in meetings that take a week to schedule, and get repeated because nobody knows the right answers, and thus dares not make decisions. Or worse, makes bad decisions under pressure, but without facts.

The penalty for not knowing the value delivery effects, and the real costs, can be far more than a week's loss or delay. So it makes sense to spend a week *doing* delivery, and thus getting more reliable facts. And then, if successful, commit to keeping the successful value delivery strategy, and scaling up.

*Figure 4.6 A Plan, with many options, versus a single reality. Informed Birth Planning assumes nothing and prepares for anything. You create a birth map, you know your options along the way. http://www.bellabirth.org/informed-birth-preparation.html
In some situations it might pay off to plan for all possible options. But if the event is infrequent, like once, and the options are infinite, it might just pay off to plan the next step based on immediate past experience along this path.*

The clear advantages of 'experimentation', of trying out a strategy and measuring its effects, are:

- It forces your team to think through the entire **system-level** requirements (people, motivation, integration with current system) – all things needed to make value real
- It gives you more **credible data** about all the multiple values delivered, and the costs incurred.
- It actually moves you **towards** your value **objectives**
- It gives your team **experience, confidence, and pride**.
- It gives you, as a manager, much better **bragging rights** than your colleagues. Like those who mainly pontificate in meetings, while you deliver the goods, and get on with it

Figure 4.6 B. Christopher Columbus' misleading map and reality adjustments. *For thousands of years Southern Europe had bad maps. One quick experimental trip or so over 8 years, and reality confrontation then gave us much better maps.*
http://en.wikipedia.org/wiki/Voyages_of_Christopher_Columbus#mediaviewer/File:Conceptions_Colomb_map-en.svg

S1: <u>Automated Sales Aid</u>.

Type: Proposed Strategies for Automated Sales Aid Goal, with Impact Estimates.

<u>S1A: Package C:</u> saves 10 minutes per employee (50% of the goal), cost £1,000 once, 1 week to develop and implement. ROI over 1,000 to 1, over 3 years, for 1,000 employees.

<u>S1B: In House development D.</u> increases sales by 10% (80% of our Goal). Cost 1 week to develop and implement. ROI at least 500 to 1. For UK Market.

Planguage Example 4.6 Two possible strategies, and their estimated impacts.

So with estimates like the ones above, which one would you chose, and how can you be sure the choice is right?

Personally, we think it does not matter. Both are very promising. So get cracking with them, both as soon as possible, and in parallel if you can.

If at least one works pretty well as advertised, you are a winner. If you put off the decision to do something, you are a loser – until you do something.

You might wonder, if these short-term cycles of value delivery put you in danger of sub-optimization. The answer is that you are in little danger of that, because your primary day-to-day criteria for deciding what to do, is directly related to your longer-term numeric objectives.

Making real progress towards the longer-term objectives, early, and continuously, is a really good thing! Both in the short term, and the long term.

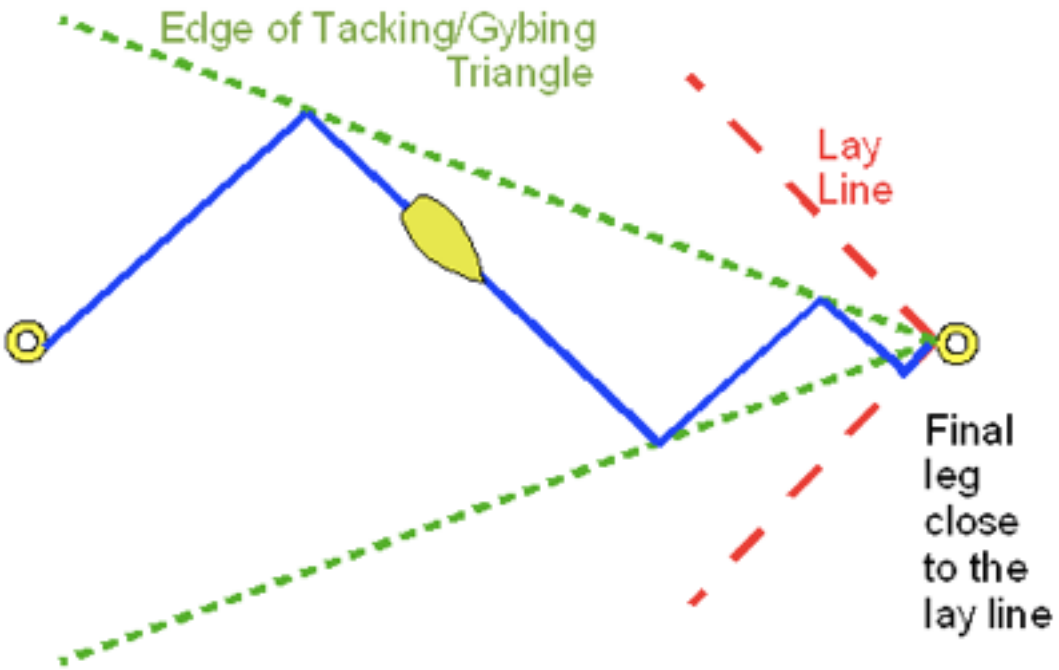


Diagram 4.6
Short incremental steps in the direction of the long term Goal.
http://www.oceansail.co.uk/Articles/TackingTriangle.gif

Your other option might be the famous 'analysis paralysis', awaiting the 'big bang' wonderful future.

"Projections are bullshit. They're just guesses. Wait until the real thing happens and then when it happens, then you can make the decision. Make decisions when you have a lot of information to make the decision. Not when you have to guess about what the decision is going to be or use data that doesn't exist yet."

Jason Fried, Keynote Speech, SXSW 2006

QUOTE 4.6 A

Practical Tips 4.6 Experiment, don't refine estimates.

- Try out the winners of rough quick estimates, in a quick rough way. Go with the results. Scale up. And measure again.
- Dump losers fast.

Policy 4.6 Just Do It.

- **We will use short meetings, and rough quick estimates, to quickly determine what might be promising to do in the coming week.**
- **We will then do it, and learn; and continue in this mode, 'as long as there is competition'** (W. E. Deming saying about PDSA Cycle)

Why?

This is probably the best way to get realistic feedback about strategies before scaling up.

Amelia Erhart:
The most effective way to do it is to do it.

QUOTE 4.6 B.

Section 4.7 Focusing on real value-for-investment has rewards.

Principle 4.7 Real Value Gets Real Budget.

**If you are good at delivering values -
big, early, frequently, consistently, efficiently,
you are more likely to get
additional support and funding,
because you show good results.**

Did you ever feel that you do not get enough budget, time, or power, to do things 'properly'? That your boss does not 'understand you' and the great capability or potential you actually have?

But, you know '*actions* speak louder than words': and so do *numbers*.

Case 4.7 The Raytheon Case: How to Succeed in Getting enough Budget to be the Best

Raytheon Corporation [URL27] invested in their software engineering process improvement for several years, for a 1,000, later 2,000, person division.

At a fairly early stage they reported:
"The return on investment (ROI) was thus 7.7 to 1 ($4.48M/ $0.58M)." [URL27] – They then remarked, that since they had the best return on investment to show, of any other corresponding group in the corporation, the culture of the company was, that they got a lot more additional money to invest in continuing their efforts, than any other group.

Among several interesting documented value-delivered achievements, was the systematic and long-term reduction of project (missile software for government) cost overruns.

We can promise you that a detailed study of their case [URL27] is a powerful management-planning 'case'. I Learned Many Lessons From It.

It discusses a number of critical values they managed to quantify and control, like the cost overrun factor below.

Case 4.7 Raytheon wins addition budget by demonstrating good ROI.

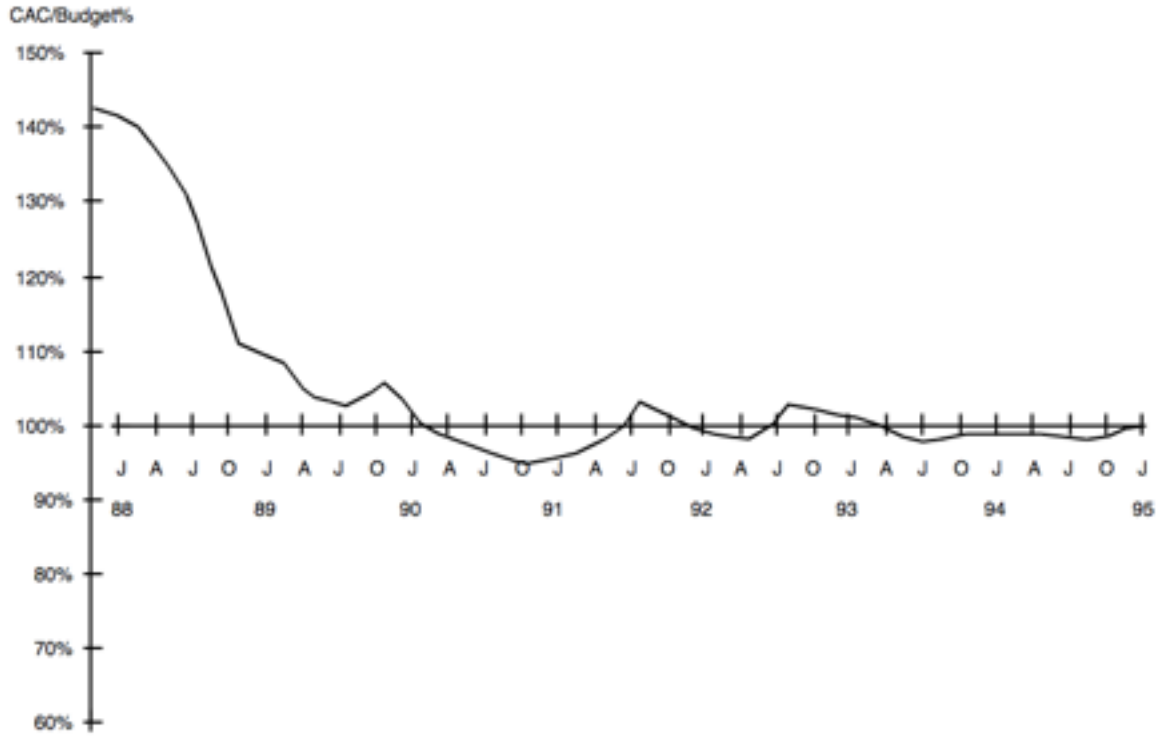


*Figure 4.7 from Case 4.7 Raytheon. Project Cost At Completion, was reduced from 42% over budget, to (long term!) close to zero for years on end. This is part of the Return On Investment Case. This can be compared with NASA project overruns of 50% to 200% for half of all projects, and never less that 20% more than budgeted. [**F4**: Gruehl chart, NASA in 2.2 and 7.7]. They used Spec QC (Inspection), Defect Prevention Process (DPP) and multidimensional quantified management objectives, as basic methods.*

Wow! Imagine if you could pull that off, as a manager and planner.

Einstein said:

"Try not to become a man of success.
Rather become a man of value."

And he immediately follows this with:

"He is considered successful in our day
who gets more out of life than he puts in.

But a man of value will give more than he receives"

Quoted from an Einstein interview, some months before his death, by William Miller in Life Magazine, May 2, 1955 [URL72, original source!]. Quote 4.7.
Albert Einstein. Born: March 14, 1879, Died: April 18, 1955

A. E.'s Favorite Photo

FINAL WORDS OF COUNSEL were offered by Einstein to Pat Miller (*right*) standing on porch with Dr. Hermanns as the visit to the scientist ended.

From the interview session.

Practical Tip 4.7 Brag about your value for money

- *Take the trouble to both estimate, and if possible measure, the return-on-investment of your projects. Like Raytheon, Case 4.7.*
- *Don't hesitate to present these opinions, or findings, to people who have money to invest in your projects.*
- *(they like to think their investment will be worthwhile!)*

Policy 4.7 Support Winning Teams

- **We will give increased resources to teams who prove they can deliver value for money, or for other limited resources.**

Why?

- To motivate efficient projects
- To reward delivery of measurable values

Section 4.8. How to protect the long-term values.

Principle 4.8 Entropy.

Short-term values' levels
are no guarantee
for *longer* term levels

We all know that romantic infatuation is no guarantee of long-lasting love.

So how can we make sure that initial levels of performance, which are satisfactory, will be maintained in the long term?

Figure 4.8 A. The hidden long-term problem: some of the strategies that must be planned to maintain system performance levels in the long term.

There are many threats to good levels of performance:

- Any new strategies added, can sour the mix, even if it is just an unintended side-effect of the new strategies
- Resources for maintaining the initial 'good enough' levels, might not materialize as assumed, or ever have been planned, or might be diverted to higher priorities
- Nobody takes responsibility for the performance level
- Nobody continues to measure it, or report it, or react if it goes bad.

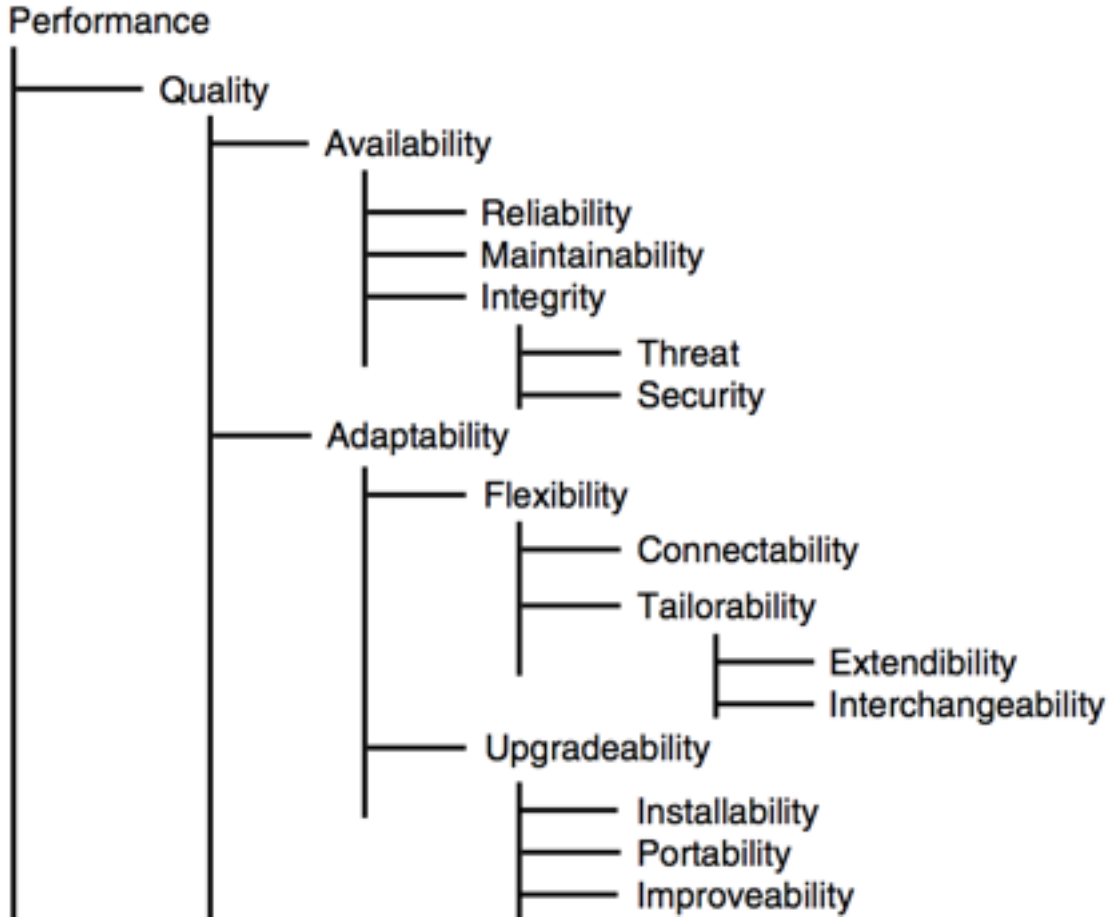


Figure 4.8 B from [1, B] ***Some of the many long-term plan-able quality characteristics of any system.***

Adaptability and Upgradeability are examples of long term values that can be consciously planned for, if you choose to take responsibility for the long term. Good companies, managers, architects, and engineers normally are aware of their long-term responsibilities. But many immature organizations, managers, and planners do not seem to have a long-term culture. Others might like to take responsibility, but they do not know how to measure these longer-term qualities, so they cannot develop serious plans [URL76].

These quality characteristics can all be quantified as objectives, and planned into the system [1B]. Organizations will end up fighting 'system entropy', decaying - costly to maintain systems, for the life of the system.

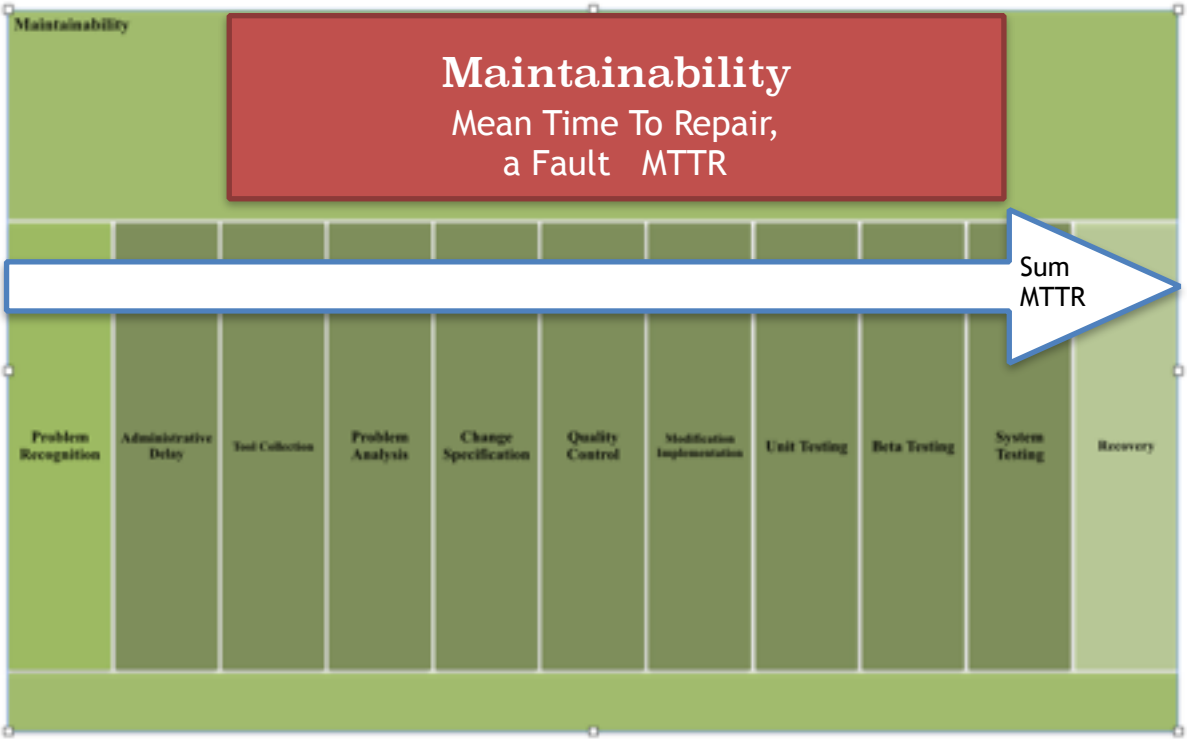
They will blame 'complex systems' and other people. But will rarely acknowledge 'we screwed up the initial planning, and took a short term view only'. But it takes great maturity, and long-term vision, to plan to get control over long term economics, seriously, *initially*.

We have one client, Confirmit in Oslo, who 'matured' *10 years* into their product maturity (in 2005), and started a serious effort to engineer these product values into their product [URL28, A & B]. They were able to continue being competitive, for years after that, to this day. Maybe it is never too late to plan for the rest of your future.

So, that threat list alone, above, suggests some tactics:

- Make sure you estimate, and then measure, threatening side effects, when you implement new strategies.
    - do not let side effects destroy ability to maintain your system.
- Make sure adequate resources are allocated for maintaining desired levels of performance, and that they are earmarked, and people are aware of the consequences of not using them as intended.
    - You need to 'fund a strategy', in order to maintain levels of performance.
- Exactly *who* is going to be responsible for the performance level?
- Exactly how will the level of performance be measured, and how will we report deviation?

Figure 4.8 C. When planning the long-term levels for just one key quality objective of a system, such as 'maintainability', it may require specifying these 11 sub-objectives, for each and all different components of the system. And then comes the process of defining, funding and creating corresponding strategies to deliver the capability. [see 1, B for many different adaptability type of objectives and how to quantify them] No wonder we avoid the task, and hope we will 'get lucky'. See also Template 10.7 B for detail on maintainability.



User Attention:
Type: Marketing Product Line Value Objective.

Ambition: Users attention to our product is maintained over time.

Scale: average minutes per day throughout the year, that defined [**Users**] spend doing defined [**Tasks**] on our product line.

Meter: Marketing Statistics.

Goal [This Year, Users = Paying, Tasks = Playing] **10 minutes.**

Goal [All Following Years, If Product is still Renewed, Users = Paying, Tasks = Playing] **at least 10 minutes±5 minutes.**

Responsibility: Marketing Director.

Planguage Example 4.8 Some plans to get, and keep, a long term performance level.

The example above makes a clear statement that we have a long-range ('All Following Years') objective to maintain. Hopefully that will direct and prioritize attention to that Goal. We also decide and document who ('Marketing Director') will take responsibility for it.

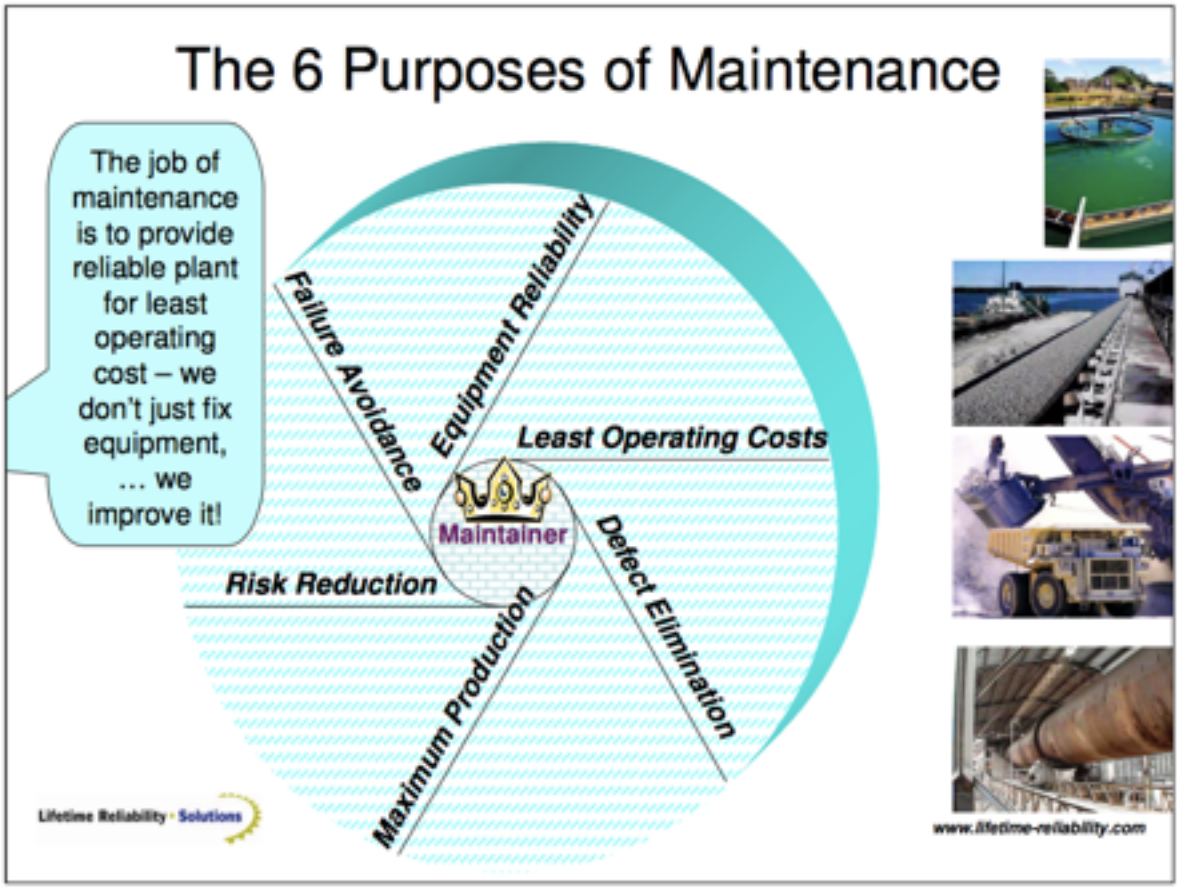


Figure 4.8 D. One view of maintaining the long-term performance levels of a system.

Practical Tip 4.8 Plan for long term success.

- *Assign specific people (positions, not individuals) or groups, responsibility for long-term performance*
- *Budget specific resources for maintaining the performance levels.*
- *Plan regular methods or tests to make sure the performance level is maintained.*

Policy 4.8 Built to Last

We want our product and organization values to last into the long-term future,
so we have to specifically plan for them,
and plan for their maintenance.
The *initial* value-deliveries are doomed to entropy (decay).

Why ?

- Because we cannot expect things to keep being successful under many other competing pressures and priorities.

Quotation 4.8 Quality by Reputation.

> ***"Men acquire a particular quality by constantly acting a particular way... you become just by performing just actions, temperate by performing temperate actions, brave by performing brave actions."***

— Aristotle, Greece, 384 BC to 322 BC

Section 4.9 Values conflict with each other because of the 'common purse'.

Principle 4.9 Value Conflict.

**Attempts to improve a value
will conflict with other values,
for
limited resources.**

The basic conflict amongst your value objectives.

If you have *infinite* resources, 'you can have it all'.

You then can push all objectives towards the state-of-the-art limits.

In real life though, if you want more of one value, you *might not* have the resources left, to get as much as you want, of another value. You might run out of time, people, or money.

Every attempt to deliver an improved value, *normally* requires *some* resources, and few of us have infinite resources.

A note: conflict for resources is not the *only* value conflict. For example extreme security can destroy usability. But I want to focus on the resources conflict here. Potentially, *everything* is in conflict with *everything* else!

There is no 'Iron Triangle of Project Management' [URL85]**: sometimes you *can* have it 'both ways'.**
This tradeoff (A for B) is not however *inevitable*. When you want more of Value A, Value B will *not necessarily have* to *suffer*. That would be possibly due to unimaginative strategy planners. They cannot identify cheaper alternative strategies.

The reasons, you might be able to get it 'both ways', is that
- costs reside in the **strategies**,
- and it is, in principle, *possible* to find an alternative strategy which improves *both* Value A and Value B, while reducing costs at the same time.

This 'finding cheaper but still effective strategies' is called 'design to cost' [URL12]. That means finding a design that reduces the cost, while maintaining, or achieving, your Goal-level values.

In addition, it is worth noting that giving 'excessive' priority for Value A (allocating resources *mainly* for Value A, increasing Value A well beyond its Goal, towards some Stretch or Ideal), does not *have* to result in *disaster* for Value B.

We have a process, described below, based on scalar constraints constraints (Tolerable, OK), for balancing resource use, and thus avoiding sub-optimization disaster for one value.

The Role of Scalar Constraints in keeping resource allocation balanced.

We can quite simply set limits on the minimum we require for Value B (a Tolerable or Ok level; a scalar constraint), *before* we are *allowed* to improve Value A, beyond its own minimum level. This process is done dynamically after each value delivery cycle.

This is a simple principle, and one employed by your own human body with regard to sleep, hunger, breathing, and any other attribute competing for our attention.

Here is a simplified example of specification of constraints and targets.

Sleep:
Scale: Hours deep sleep per night, average.
Tolerable: 0.2 .
OK: 4.
Goal: 7.

Hunger:
Scale: % of belly full.
Tolerable: 1%.
OK: 50%.
Goal: **100%.**

Air:
Scale: % of clean air needed per day.
Tolerable: 1%. (below this is intolerable)
OK: 50%. (at this level or better, things are satisfactory, but not good)
Goal: 100%. (at this level, we have 'success')
Stretch [For Pensioners and Athletes] 130%. (this level is valued, but marginal, luxurious)

Planguage Example 4.9 Three concurrent human body objectives.

Two Target Levels: 1. Goal and 2. Stretch.
In these simplified statements of objectives above, for three human body critical needs (values), you will find that:

1. Each of the above objectives has a **Goal** level, a *target* to reach, deemed quite sufficient, successful, and most satisfactory. Happy body.

The Goal level is also defined, from a project management point of view, as the 'we promise and commit' level. 'Goal' means that we believe that we have practical and economic strategies good enough to get us to the Goal level. So, we can promise and commit to get there.

2. **'Air'** has in addition a **Stretch** target level. 'Stretch' is a defined *value level* that would be 'of value' to *some* stakeholders; if it does not cost more than the value delivered.

The Stretch level is *not committed to delivery initially.* Stretch is certainly not 'promised and committed' like Goal is. We have to wait and see if it is 'possible' (if strategies and resources exist), later (after all concurrent Goal levels are satisfied).

But 'Stretch' does say that 'not all stakeholders are completely satisfied with the Goal level'. Some want more. Stretch is a tool to keep the team in a competitive mode. It also helps is consider special stakeholder's needs, at a lower priority than a Goal level.

The Stretch level of a value is specified as a Target in the objective;
- but *action* towards Stretch levels is delayed until we reach all 'concurrent' Goals, meaning 'all Goals that are deadlined before or at same time as the Stretch target'.
- and value delivery action is delayed, until we have enough residual resources, to work on the Stretch levels.

Goal levels should be *competitive*, Stretch levels try to improve on that competitiveness, if possible.

Figure 4.9 Do you think he is most interested in eating, sleeping, drinking or breathing?

Two *Constraint* levels: 1. OK and 2. Tolerant.
1. Each human body objective has an '**OK**' level, a *minimum satisfaction* level, we can live with, while we work towards the Goal level. 'I am breathing ok, but I would be happier with more fresh air.'

2. Each human body objective also has a **Tolerable** level, which means 'bare survival': anything less, you are dead. 'I am almost starving to death, surviving on leaves on trees'.

Let us imagine, that one value, say Air, is at the 0% (no air today) level (well below Tolerable Level). Within minutes our body is 'dead from suffocation', and it does not matter how much food or sleep is available.

'Tolerable' levels are the level, just above which, the system (body) dies or stops functioning; becomes 'Intolerable'. So, clearly we need to 'make sure' (in planning and management) that our system, regarding all critical objectives, is above that 'intolerable' level.

Plan for survival, first.

The Value Race: while balancing access to resources.
Then, when all critical levels are *Tolerable*, the race is on to the OK level; the set of OK levels for our critical objectives. This is balancing access to resources to that one value alone, does not cause death of the system.

But we must not let any one of the Value-'horses' race *so* far, as to exceed the OK level, while any *other* value is still at the 'gate' of intolerability. Can you reason why this is logically necessary? We are still 'balancing' resource use.

Value increases, need to all move at a reasonably even rate, to the 'OK' level, or beyond.

When all our three 'horses' have arrived at the OK level, we can start a new race for the Goal levels. Again, we need to restrain our horses at, or around, the Goal level, until *all* critical objectives have reached their Goal levels.

Otherwise, any attempt to move one value, much further ahead, can deplete resources too much.

These resources might be needed to move one of the other objectives, to its Goal level.

This is because we might, without intending to, deny us reaching that Goal level *altogether*, in return for some *marginal* benefit with *another* value, if we misuse the resources prematurely (before all are at their concurrent Goal levels).

Finally, all our horses assembled at the Goal level, we can restart the last stage of the race to a *Stretch* level, and if reached, we can stop; there being 'no further value defined', for 'claim on our limited resources'.

O---------Catastrophe|Tolerable--------Fail|OK---------|Goal----------|Stretch---?
Dead |*Alive* | *Good* | *Success* | *Incredible*

Constraint Levels help Balance Resource Use.
So, the specification of these Constraint Levels (Tolerable, OK) are two ways to make sure, certain values (Objectives), get bare minimum satisfaction, before other 'greedy' Target values take all the limited resources away.

It is a way to make sure, there is a 'balanced' approach to reaching objectives; avoiding extremes. Like *'The operation was a success, but the patient died'* as the old saying goes.

This entire process is as logical and natural as nature's process for helping your body to survive and thrive.

Priorities of Levels of Objectives.

These levels of objectives are priority signals:
1. first priority is survival (getting to **Tolerable** levels).
2. second priority is comfort (getting to **OK** levels).
3. third priority is satisfaction and success (getting to **Goal** levels).
4. fourth priority marginal luxury, if we can afford it (getting to **Stretch** levels).
5. there is no priority reaching perfection (**Ideal** levels). We can not afford infinite costs.

"Problems are hidden opportunities, and constraints can actually boost creativity."

***Martin Villeneuve**, Montreal-based screenwriter, producer, director, art director and TED speaker.*

Practical tip 4.9 One Level is not enough.

- *Always assume that it is useful and enlightening, to describe any objective, in terms of both a 'target level' **and** a 'constraint level', at least.*
- *This will lead to better balance in use of resources to manage the objectives.*

Policy 4.9 Distinguish minimum levels from targets.

- **We will clearly state our *minimum* value levels, like 'OK'**
- **so as to help us *balance* the priority,**
- **that any one value gets,**
- **with the priorities of *other* values.**

Why?

- to avoid the danger of satisfying some Goals before the constraints are satisfied
- and before allocated resources run out.
- the 'OK' level implies that the Tolerable level is also reached, earlier.

Section 4.10 The Never-Ending Quest for Value Improvement

Principle 4.10 Eternal Lust:

**When one set of values are successfully delivered,
you will need to turn your attention to *other* values,
and *other* stakeholders,
incessantly.**

In competition, business, charity, or war there is no end to the need for competitive improvement. Standing still on your achievements is not an option for winners.

I asked Dr. Deming when the Plan Do Study Act cycle ended: his answer was

"Never, as long as you are in competition."

Dr. W. Edwards Deming (1900-1993). The answer was given orally at his training course about 1984.

When all outstanding Goal levels are achieved, with perhaps some tweaking towards potential Stretch levels, there should always be a new set of competitive Goal levels specified, on the time horizon. Either by increasing levels of performance from just-achieved Goal levels, or by setting Goal levels in entirely new areas.

Case Study 4.10 Confirmit.

One client of ours delivered impressive measurable results in 25 of their product areas, in 12 weeks, 10 of the areas were in Usability, like 'Intuitiveness'. Then, having done that, took a deep breath and delivered another 10 impressive results in the next 12 weeks, using 4 parallel small teams. They released their enhanced product to the worldwide users every quarter. But they designed, produced, delivered and measured increments weekly, to pilot trials of real systems. They continue this process with improvements for over ten years: wiping out their competition worldwide. [URL28, Confirmit]

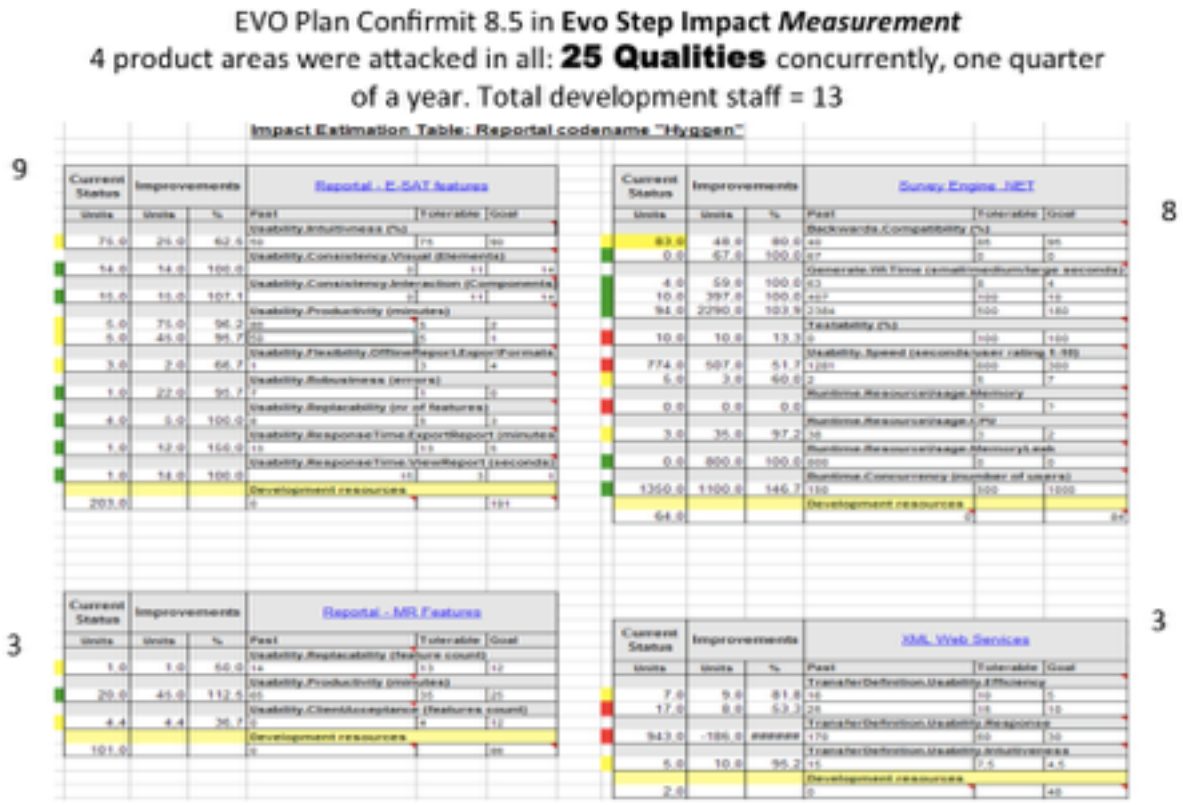


Figure 4.10 Confirmit Case. Comment below

This is a snapshot of the 9th week of 12 weeks value delivery, before International release.

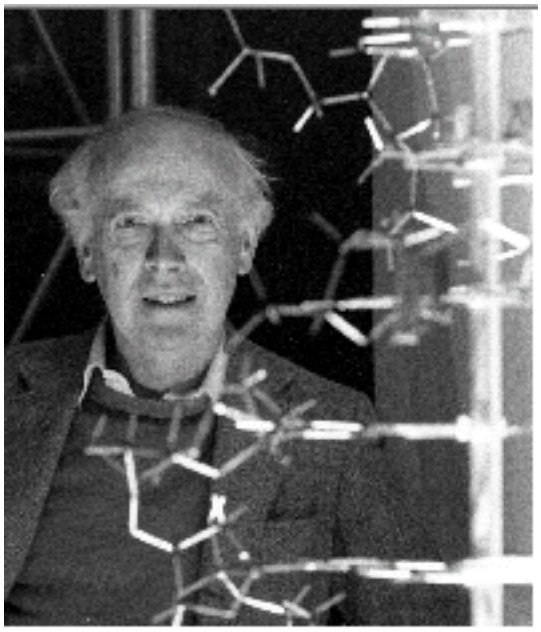
There are 25 product values being worked on by 4 small teams in parallel.

If you look at the "Improvements %" you can see the % of way to the Goal level, that the teams have made in 75% of the time (9 of 12 weeks) before Quarterly release. You can see they are way ahead of the curve. Using dynamic prioritization of the less-then-100% values, they usually manage to bring in all values at, at least, 100% of the Goal. The Green/ Yellow/Red indicators are 'calculated priority signals'. Green = at or better than Goal. Yellow = Tolerable, but not Goal. Red is 'not even tolerable': WORRY NOW!

Case Study 4.10 Confirmit.

"Take young researchers, put them together in virtual seclusion, give them an unprecedented degree of freedom and turn up the pressure by fostering competitiveness"

***James D. Watson**, Academic, Biologist, Geneticist, Zoologist, Scientist (1928–)*



Practical Tip 4.10 Assume you can never rest

- *Do not ever assume you are really 'done' planning for increased value.*
- *Assume there are new unsatisfied value objectives just waiting to be discovered, at all times, with **some** stakeholders – including stakeholders you do not know about yet.*
- *Or assume that competition will force you to up the game, in a never-ending Value Race.*

Policy 4.10 Eternal Competitiveness.

- ***When we achieve all our initial Goal levels, we need to turn our attention towards* other *competitive Goals,* eternally.**
- *Why?*
    - *the world refuses to stand still for us.*

Edit note:
26 August 20 2015

I have deeply edited chapters 1-4. using 4-6 passes of edit.

Main making things clearer, but adding new text and diagrams, and quotes.

Surprise me: send an email to tom@gilb.com *saying*
I have read…..
and my comment or question is….

Chapter 5. Decomposition (by value, by responsibility)

We have discussed **strategy** decomposition here and there earlier, but now we will focus on it in more depth.

Decomposition seems to raise problems of sub-optimization. but that is not true using the methods in this book. We are constantly keeping our eye on the 'performance of the whole' (see quote below): by keeping our eye on the top-level objectives.

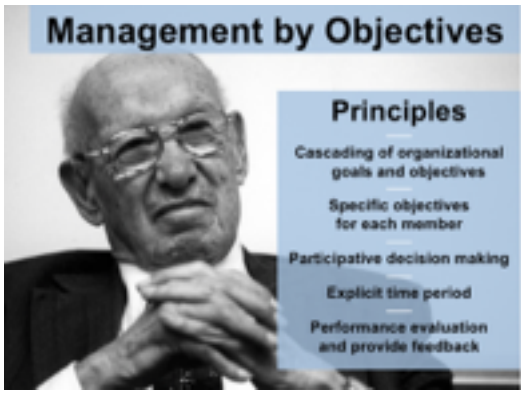
> ***"Throughout management science - in the literature as well as in the work in progress - the emphasis is on***
> - ***techniques rather than principles,***
> - ***on mechanics rather than decisions,***
> - ***on tools rather than on results, and, above all,***
> - ***on efficiency of the part rather than on performance of the whole"***

(Drucker, 1973: [25], page 509).

Drucker is also clear about quality as being the 'experience of the stakeholder', not the 'cost or strategy' put in to the product or service. that is our focus in this book too.

> ***"Quality in a product or service is not what the supplier puts in. It is what the customer gets out and is willing to pay for.***
>
> ***A product is not quality because it is hard to make and costs a lot of money, as manufacturers typically believe. This is incompetence.***
>
> ***Customers pay only for what is of use to them and gives them value. Nothing else constitutes quality."***

Drucker: [URL86]



[URL86] Peter Drucker. MBO Principles summarized.

Section 5.1 Strategy decomposition for early value delivery.

Principle 5.1 Mini Strategies.

Strategies can be split up
into earlier partial implementations,
and you can plan a regular flow of value delivery

Some strategies are too darn big.
Any given strategy idea, by whatever name, 'Architecture/tactics/ solutions/means/plans', might take an arbitrarily long time to deliver any value. **Years** is one possibility, before any value for a strategy is delivered. That is too darn long.

A bad planning mistake is to agree to such a 'delayed action' strategy, and then wait until it is 'done', perhaps years too late.

We need to harvest all the results we can, as early as we can. We need to be certain that we will harvest results *at all* – that the strategy *will* work. If our strategy will *not* work in practice, we need to find that out ASAP.

We need to learn how to harvest results with a given strategy, in *practice*.

All strategies can be split up into earlier deliveries.
Fortunately, there seems to some 'law' of systems, that our experience confirms all the time;

- A**ll strategies *can* be decomposed** (split up, divided, chopped up) **into smaller sub-strategies.**
- Well, almost all.

Not only that, but it seems, that we can almost always subdivide into ***very small*** incremental value-delivery steps.

Extreme Decomposition.
The 'Lean Startup' method, reports conducting 60 experiments per day, live with real potential product users [URL29].

So that means a decomposition, into about 10 'attempts at value delivery' per hour.

This is done in an internet software situation, high degrees of automation, and lots of parallel production and delivery of attempts at value delivery.

Not everyone is working in *that* internet environment, but it does show the level of decomposition possible, when the environment is right.

2% Decomposition.
In general, over a wide range of projects, we like to try to decompose into *weekly* value delivery increments; or increments of magnitude 2% to 5% of total budget.

In general, such decomposition works most of the time.

The Backroom For 'Longer than 2% Steps' for Building a Value Delivery.
If you cannot do that, and sometimes you cannot, you need to have a 'backroom' to build larger strategies in, when they do not admit to time-wise decomposition. For example, when there is a waiting period before something can be delivered.

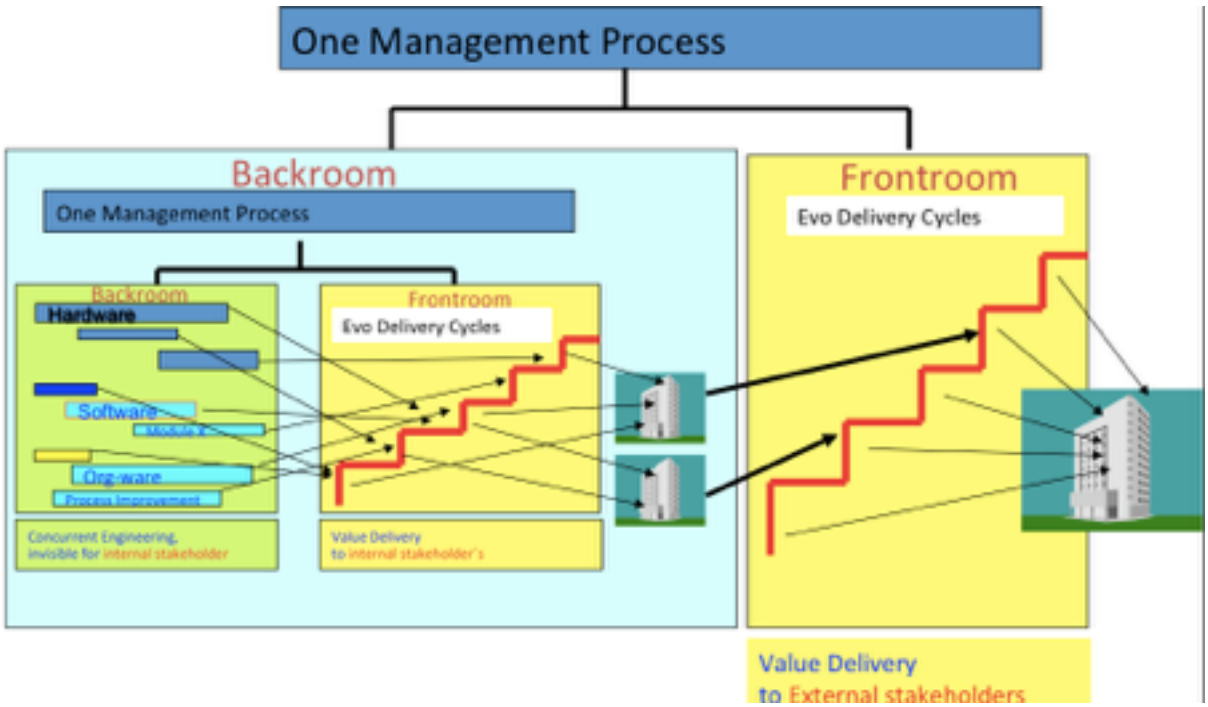


Figure 5.1 Philips Corporation in Holland use of our Backroom/Frontroom concepts, at 2 levels. See also Fig. 6.3 for Backroom visualization.

Case 5.1 Philips:

Our 'Evo' Project Management strategy tries to deliver some value, and to test strategy ideas, frequently.

But not all strategies can be decomposed into weekly chunks of development/integration/testing.

Our solution is to put all such strategies in the 'backroom' and let them take whatever time they need to be completed or developed, in the sense of 'ready for delivery to stakeholders some week', as the 'weekly value delivery'.

In this case, derived from work by Simon Porro, at Philips of Holland (our client), 2001, they took our ideas one step further.

Philips had *two* levels: value delivery internally to Philips internal stakeholders, then value delivery externally to external stakeholders, like customers. See Fig. 5.1

This is applicable to the complex products and markets at Philips Corporation.

More on this Philips case at: Case 5.6 and Figure 5.6.

More Detail about 'Backroom' Planning.
Please notice carefully what we are claiming here, the subtle concept of 'weekly *value* delivery'.

We don't mean 'weekly' *building* increments, if they cannot really be done. It is easy to 'misunderstand'. Value Delivery is *not* the same as increment building.

We are claiming that:

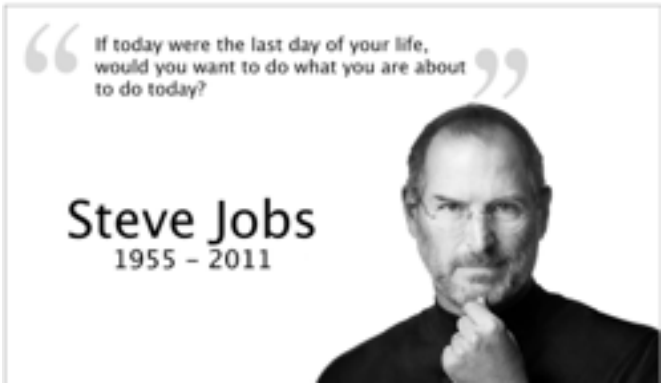
- *Most* strategies can be divided into smaller increments, than 'doing the entire strategy', *before* any value delivery
- In *many* cases, strategies can be subdivided, by independently deliverable value (maybe Security initially and Performance value later), so that they can be partially started, and delivered to stakeholders, in a useful short cycle, such as a week.
- But, in the cases where that is *not* possible, we are saying, we can put the strategy, or a big subset of it, into a 'Backroom' development process, of for example, several weeks or more.
    - But at some point, the Backroom development will be done well enough, to allow us to schedule the Strategy for value delivery to some stakeholders.
    - At that point, an upcoming value delivery cycle, of say, a week, is used to deliver to the stakeholders, and measure value delivered: *quite independently of how many weeks it took to develop the strategy* to that point.

- In the meantime, while the 'backroom' is developing any number of strategies in parallel, we will still, in general look to find *some* strategies, which we can deliver, each and every week to stakeholders:
  - Either because they are ready, and 'on hold' in the backroom. Or because -
  - they can be completed, from start to value-delivery in for example a week.

See also Fig. 6.3, and Fig. 5.1 for Backroom visualization.

"I have a great respect for incremental improvement, and I've done that sort of thing in my life, but I've always been attracted to the more revolutionary changes. I don't know why. Because they're harder. They're much more stressful emotionally. And you usually go through a period where everybody tells you that you've completely failed."

Steve Jobs



Practical Tip 5.1. Don't give up on finding small value delivery increments.

- *Keep the pressure on planners, to find 'weekly' value deliveries.*
- *If you give in to their excuses, you will not get the value.*
- *But you are going to have to train and coach them as to how to find small value delivery increments!*

Policy 5.1. Weekly Value Deliveries.

We will deliver value every week.

- **But if a strategy needs development time,**
  - **before it is ready to be delivered in a week,**
  - **do that development elsewhere (Backroom), and**

    - **find something else, which is ready-to-deliver 'this week'.**
- **Keep a regular stream of value flowing to stakeholders.**

    - Why?
        - To delivery value early and continuously
        - To learn about problems early and fix them before similar increments come on stream
        - To avoid large bureaucracies that tend to come with big delivery packages
        - To reduce risks of being wrong, about the 'big strategy'
        - To build confidence of your delivery team, and your stakeholders, in your ability to deliver.

Version 27 Aug. 2015 TsG 18:35

Section 5.2 Decomposing strategies into independent implementations.

Principle 5.2 Independent Values.

You can usually decompose strategies into *independently implementable* values

If we are going to get a flow of value going early, and continuously, we have to find ways of extracting *partial* strategies, from the set of large strategies, so that value delivery in a continuous stream is practical.

These are the **conditions for independently implementable steps**:

- 1. *There can be **no dependencies***; such as permissions, meetings for approval, lack of budget, and other things that need to be done, or implemented, so that value can be delivered. We need to be sure that if the strategy works, we get measurable value by the end of our cycle.

Once at Boeing (McDonnell Douglas), a director said nice things about Tom's attitude of getting cross-functional teams to work together on quality control of engineering drawings; the aircraft engineers and the assembly workers in the same meeting. Then he politely told him that before we invited the assembly workers to the engineering design QC meetings again, we had to clear it with the trade unions. Otherwise we could easily provoke a strike!

Case 5.2 A. Boeing Unexpected Complication

o

- *2. We have considered (in writing, in a written plan)* ***the entire system,*** all hardware, software, data, people, legal, and cultural elements (like Unions), so that value can indeed be delivered and measured.
    - *We cannot excuse failure by saying 'we delivered our component', but it did not work, because (of someone or something else, that we should have included in our plans)*
    - Participants have **agreed to spend the time** as planned, and prioritize it as a team effort.

We learned the hard way that getting the individual employee to agree, is not the same thing as getting their managers *to agree or prioritize anything.*
At a London (US) bank we trained Business Analysts in Planguage for 3 days, to reduce the unintelligible 'requirements' they sent to IT, by a factor of 8.

But, as soon as our students were back on the trading floor, where they 'lived', they promptly had no time or motivation to practice what they had learned.

We had to solve the problem by doing their quality work for them.

Case 5.2 B. Trained, but Unmotivated

At a London Telecoms, we trained the Marketing people to specify products and services more clearly in Planguage.

And the Marketing Director claimed she was with us on this.

But she had other priorities, and it took the CEO firing her, to get a new marketing director who really supported us, in practice.

Case 5.2 C Trained Staff, but Unmotivated Director.

- Machines needed to test or implement are **in fact available**.

In new product development, we have experienced that the few copies of the new product are booked up for other purposes, and cannot be used as planned.

This was the case in Philips, Eindhoven Holland, where there was really only one working New Product machine, and it was very expensive and complicated.

So you could not just get extra machines to test the software on.

But the marketing people were naturally using up its time for sales demonstrations, and they had more power than product development.

So we had to really worry if our delivery cycles would get time, when we wanted it on the New Machines.

We had to make specific agreements that we could access the prototype product long enough, on the exact days of our one-week cycle that we needed them. if not, we could not deliver the weekly value.

But it did work out, see Figure 5.6, where the value improvement is delivered on a regular weekly cycle.

Case 5.2 D. Limited Access to Test New Products.

- *3.* ***We have planned a safe and quick retreat***: if the delivery fails in any way at any time, we need to have planned how to remove it, so things are no worse off than before we implemented it.
  - *Irreversible acts like turning off the old system, removing old machines, removing people running the older systems prematurely, signing irreversible contracts, are a very bad idea; until the new improved system is really working in a stable way, and is reasonably scaled up. We understand the impatience and need to actually 'make the savings'. But, do not expect everything to go as planned, until it really does! Assume it will fail, until clearly proven otherwise. Keep old systems for a little longer as backup insurance. We hate it when people complain loudly about the new system being very much worse than the old one, and 'they will just have to live with it; because there is no way back. A sign of very bad planning indeed.*

- *4.* We need to have planned **continuous observation** from the moment of implementation*,* so that if anything goes wrong we can stop it, and/or correct it early. If you discover things are not working at the *end* of an implementation cycle, you have lost a whole cycle.

- 5. **Measurement itself** is planned, at the end of this short cycle; at least 'leading indicators' (e.g. the *time saving in a sample* of users, that precursors the *cost saving* later for the larger user set).

- 6. **Longer term measurement** is planned:
  - To make sure the *initial results hold steady*, as time goes on, and as we scale up.
  - To make sure that not only the leading indicators are good, but that the *intended improvements are really there* too.
  - In our dreams, we would hope that someone, if not you as planner, is planning to measure the *effects one level up*, your Fundamental Objectives. And maybe this will give you some feedback on whether your efforts are, in fact, working.

These are the conditions for sub-strategies that can actually can deliver value:

- Your expected primary **value improvement** for this current cycle is **estimated**. [URL21] Maybe with ± uncertainty, evidence, sources.
- **Side effects** on other objectives, and on resources, are estimated
- We have plans for suitable **Meters, ways of measuring** what is happening in at least the primary value improvement objective.
- There are **no** clear, unexpected, drastic **negative side effects**: such as people very disturbed, confused, getting bad results of some kind; which disqualify the value delivery attempt, even if 'our' value is delivered satisfactorily.
- The value delivered is **reasonably useful**, even if not quite as good as initially estimated. It should at least be clearly **worth the cost.**
- We are **prepared to learn**, analyse, and change in the short term (during or immediately after this delivery cycle) and the long term (in weeks ahead). We plan the time to do this. Even if it just a half-hour Defect Prevention Process [URL14] per week.

- **Critical plans are formally reviewed** against our standards, and policies, using a reasonably rigorous item by item review process (not a meeting and a management sign-off in haste) [URL15, URL23]. Plans do not Exit until they are close to defect free.
    - Plans do not have to be perfect, but they have to largely work as planned.
    - A review, for a weeks implementation, should take 2 to 4 people no more than one hour if organized well [URL23]. If you fail to do this quality control, it will normally cost you a lot more loss than that hour.

Case 5.2 Late discovery of Garbage In to Serious Project.

Sad Story: A Swedish Client, was contracted to deliver an Air Traffic Control Training System for Germany.

They had let air traffic controllers write technical requirements for the software engineers to program. They had written 40,000 of a contracted 80,000 pages of requirements.

We did a simple quality control [URL23] of 3 random pages, mainly against an intelligibility standard (clear and unambiguous) as judged by 6 director-level people.

These directors had signed off on the 40,000 pages as 'good enough to go' to the programmers.

But the defects level they found, was 20 major defects per page (unintelligible specs).

This was already causing major delays.

But the sad part was that they could have discovered this much earlier, and stopped it, by learning clearer specification.

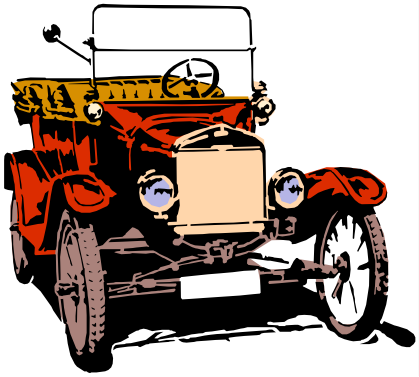
This incident taught me never to respect management approvals of technical quality.

"Nothing is particularly hard if you divide it into small jobs."

"There are no big problems, just a lot of little problems"

"Obstacles are those frightful things you see when you take your eyes off your goal"

--Henry Ford, 1863-1947



Ford in 1919

Practical Tip 5.2 It is about value for money!

- *Keep your eye on a main point of 'independently implementable' strategies.*
- *They each have different value-to-cost, and you are looking for the highest value-to-cost package you can implement at each new value-delivery cycle.*

Policy 5.2 Plan thoroughly for Successful Value Delivery

- **Make thorough plans, considering most everything necessary to succeed well, most times, for each value delivery cycle. There is no reason why the plan, in essence should exceed a page (or so).**

- Why?
  - Reduce unnecessary bureaucracy
  - Do it one step at a time (not for all steps at once)
  - If you set a plan limit to a page, it is always possible to do that.
  - If you don't the bureaucracy will explode uncontrolled.

Version Aug 2015 tsg 21:20 Pass 2

Section 5.3 The Impact Estimation Table as a value decomposition tool.

Principle 5.3 Value Analysis Table.

An impact estimation table decomposes by value

An Impact Estimation table (sketched earlier [Part 2, Part 4, Chapter 6, URL21, URL7]) serves many purposes in planning. Primarily it gives us a connection between any two levels of planning. Your level, Strategic Objectives, and the Fundamental Objectives above you. Or, The Means Objectives below you,

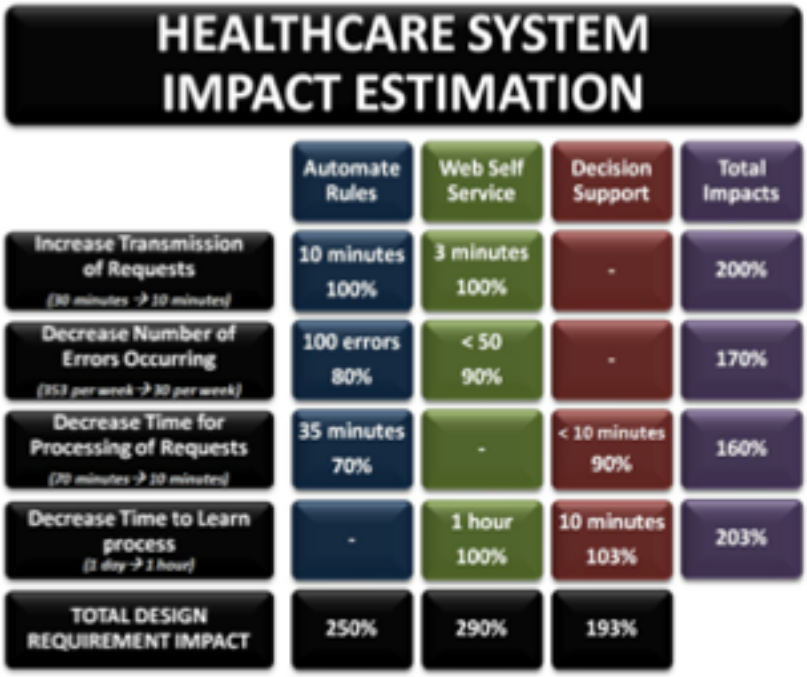


Figure 5.3 Healthcare Impact Estimation [URL30]
On the left side, strategic objectives, the top line, strategies (means objectives). The numbers in each cell: Real estimate of impact, and % of way to Goal on time.
Edit note: the arrow on the left side should be a <-> arrow according to Planguage.

The initial purposes of Impact Estimation Tables (IET) are to force us to think clearly, and to reason logically, about our selection of Means (solutions, architecture, strategies) for reaching our clear objectives.

One of many uses for the table, is to allow us to *visualize possible sub-strategies*, for our early, frequent, and continuous value delivery stream.

***Elon Reeve Musk* - Chairman & Chief Executive Officer Tesla.**

"Okay, I think that's a pretty open-ended questions, but – we have a philosophy of just continuous improvements, so every week there are approximately 20 engineering changes made to the car.

So it's not nearly as discrete as you're alluding to. With other manufacturers, they tend to sort of bundle everything together in a model year.

In our case, it's a series of rolling changes. So model year doesn't mean as much. There are cases where that step change may be a little higher than normal as, for example, with having the Autopilot camera, radar, and ultrasonics.

But we try to actually keep those step changes as small as possible.

And so that – I mean, essentially like the common questions that I get is from friends, they say, "when should I buy a Model S?" and my answer's always "right now," because – and they say, "well, aren't you going to make a better one in six months?" I'm like, yeah, of course.

But if their goal is to only buy a Model S when there aren't significant improvements happening, then they will never buy one."

Quote 5.3 A. Musk on eternal continuous Tesla improvement.20/week x 50 weeks = 1,000 improvements per year. Source: http://seekingalpha.com/article/3642146-tesla-motors-tsla-elon-reeve-musk-on-q3-2015-results-earnings-call-transcript?page=2&p=qanda&l=last
November 3 2015.

If we manage, by accident or intentional design, to find a subset of a strategy (top row, 4 columns), for impacting a single objective (left side column, 4 rows), which can be implemented (at *some* stage, *then* we have in fact 'decomposed the strategies' into smaller sub-strategies.

For example, we can ask, what is the subset of the 'Decision Support' strategy which can impact the "Time to Learn Process" objective of saving time. We want to move it from 1 day (Past level) towards the 10 minutes Goal.

If we find an answer, a part of the strategy that will help there, then we can try to implement it early.

We might well have to make a *very specific design* for the sub-strategy, so that it addresses exactly the Scale and Goal specified.

In some ideal world, every single 'Objective X Strategy' estimation cell, gives the estimate of how much value we would get, if we just implemented a subset of *that* strategy, for just *that* objective alone. We could then do the value delivery cycles in the sequence that gives the highest possible value.

We could, for example, begin with a subset of the Strategy 'Decision Support', which was exclusively tailored to save about 10 minutes, at least, for the 'Time To learn Process' objective. We could then see if that worked, more or less. If so, we could turn our attention to something else, like one of the 100% scores, for the strategy 'Automate Rules'.

The exact sequence, or priority, is not very critical, as long as fairly high-value things are done earliest.

Let us imagine a typical 10 Objectives, by 10 Strategies table. There are 100 cells estimated there. In simplified thinking, they are each 1% of the whole project. For a 2 year project, we have roughly one-week increments, if we do one cell every week.

Things are not quite that neat. But we are close to a reality which our clients have experienced for decades [URL31, HP experience is well documented]. We can get weekly increments for years on end.

The IET cells give us one *powerful clue* of possible decompositions.

We have, of course, additional strategies [URL16, URL18] for value decomposition, including just using domain knowledge, common sense, creativity and intelligence.

"1. Now. Right now. Take some little - tiny! - element of your project. Corral a surrogate customer. Talk to him/her about it. That is ... test it. Now.
2. Your immediate goal: "Chunk up" the next three weeks. I.e.: Define a set of practical micro-bits ... that can be subjected to real-world tests..
<u>Observation</u>: There is no situation - even at Boeing - where you cannot concoct a sorta-real-world-micro-test of some piece of your project .. Within a few hours to two or three days.
Quick Prototyping Excellence = Project Implementation Excellence. (No kidding... it's almost that basic!)"

Quote 5.3 B. Tom Peters, ***Reinventing Work, the project 50. Pages 138-9***

Quote 5.3 C.

Practical Tip 5.3

- *A 10 strategy by 10 objective Impact Estimation Table gives potentially about 100 implementation step possibilities.*

Policy 5.3 Value Decision Table

- **We will use the Impact Estimation table to give us insights into possible decompositions of strategies.**

- Why?
  - It forces us to think very systematically
  - And to argue logically about value potential
  - And gives us a complete picture of value opportunities
  - And it can help us understand the credibility and risks of value steps.

Version 27 Aug. 2015 21:53 Pass 1

Section 5.4 The 'Scale' parameters as a tool for decomposition of strategies

Principle 5.4 Scale Variations Decompose.

Scale parameters
help decompose
by value

Another practical tool to help you see possible strategy decompositions, is the use of Scale parameters, combined with the use of 'Qualifiers' in Goal statements. It helps you decompose in any number of dimensions of space, time and conditionality.

A Scale parameter, for example, could be:

Scale: time for defined **[People]** doing define**d [Tasks]** in defined **[Environments].**

People, *Tasks* and *Environments* are 'Scale parameters'. They are general dimensions of the above Objectives attribute.

Here is a 2 item decomposition.

G1: Goal [**People** = Service , **Task** = Refunds, **Environment** = Shop] 10 minutes.

G2: Goal [**People** = Salesperson , **Task** = Sign Contract, **Environment** = Home Sales] 30 minutes.

Imagine that there are 10 defined realistic and interesting Goal instances of each. Not just the 2 above.

Imagine all theoretical combinations that are possible for the 3 Scale parameters. We then theoretically have 10x10x10 = 1,000 possible combinations.

Even if only 100 of them were valid, we have a 1% decomposition set of parameters. We can choose to do any one of them. Then, when value is delivered, do the next one etc.

One advantage of this method, is that we can work out all possible theoretical combinations, then all possible realistic combinations. We can then ask if we have any plan for all combinations, and if not, which combinations we have missed; and what are the consequences.

To simplify we can 'group' things; to make sure all angles are covered.

G1: Goal [People = Service , Task = All, Environment = **{Shop + All Not Done Yet }** 10 minutes.

G2: Goal [People = **{Salesperson + Sales Assistant}** , Task = Sign Contract, Environment = All] 30 minutes.

The main point of this section is that Scale parameters are a useful tool for decomposing the problem, at the level of objectives, so that we can find simpler strategies to deliver high value early for critical stakeholders, and critical situations (Emergencies, Bankruptcy)

"For strange effects and extraordinary combinations we must go to life itself, which is always far more daring than any effort of the imagination."

Arthur Conan Doyle

Practical Tip 5.4 Scale Parameters are decompositions

- *Your 'Scale: _____' specifications should normally have at least 2, and up to about 5 parameters included inside them.*

This will enable you to decompose your system into to smaller problems, some of which can be solved earlier than others.

Policy 5.4 Identifying high value-to-cost and low risk value steps

- **We will systematically explore all interesting combinations of parameters or dimensions, relevant and useful for our planning.**
- **We will explore high-value and critical combinations with a view to implementing strategies for them early.**
    - Why?
        - It will help us be more complete in our planning, to see all combinations systematically
        - We can focus on early value delivery for interesting combinations of Scale parameters, without incurring delays, costs and overheads of worrying about all less valuable combinations at one time.

Version 27 Aug. 2015 22:08 Pass 1 tsg

Section 5.5 The value of the 'bad old system' as a starting point.

Principle 5.5 Start With What You've Got.

Adding value to existing systems
is smarter than
building new systems.

Building entirely new systems from scratch, is not a good idea.

But considering how awful the old system seems to you, it may seem a good idea at the time.

History shows that the 'big revolution' is probably going to fail totally.

History shows that successful revolutionary improvement in organizations and products is, if you look back, best accomplished by an incremental improvement process over the long term.

Tom Peters' early books on management had this theme, 'big fails', 'incremental succeeds'.

Why are revolutionary, 'big step', 'long project' changes dangerous ?

- There is too little, too late, *realistic* (field, user, customer, stakeholder) *feedback* to tell you that you are on the wrong path.
- You have a temptation to make use of *untested technology* or latest-fashion ideas, that have *not yet* been discredited
- There is too little *responsibility for res*ults, tied in to the associated management
- There are too many different ingredients, strategies, operating at once, on the final result. The ones that are negative are indistinguishably mixed with good ideas. If even one strategy turns the mix to poison, the project is dead. And you might not know this until it is too late.
- The project is so big that you are tempted to outsource large parts of it. But the outsourcing instance has little involvement with your real organization and stakeholders. They are even motivated to make the project much bigger, and last a lot longer than it need be.

- And much more: read 'case studies of failed large projects' from the internet. 62 million hits (August 27 2015); we mass produce failures with our bad planning methods.

What are the tactics to help an organization or product line grow fast towards competitive capability ?

- You just *do not start* 'large change projects'.
- You organize even the largest long-term visions, as a sequence of small steps (think, 'weekly')
- You have a clear long-term vision [Part 1, URL1] , but perhaps continuously updated, with respect to competition, technology and economics.
- You make internal resources *empowered and responsible* for making changes, and for getting results. They know the organization and its' stakeholders.
    - Outsourcers do not know. And trying to make good 'requirements' for the outsourcers, to convey that knowledge, NEVER WORKS. Maybe because we are terrible professionally at requirements analysis, and specification (we are). But more to the point, implementers need to be in close touch with reality continuously. With 'skin in the game'.
- You need tight Plan-Do-Study-Act ([9] Deming) cycles:
    - Measurable, quick feedback, quick change, maximum loss is one cycle – maybe a week (not the whole project).
- But, because things are small and visible, because all previous value delivery has given us cumulative, detailed, practical knowledge of what works and what doesn't, we tend to 'get it right' for each new short term plan, forward.
- There is really *no danger of short term sub-optimization*, when we do a series of short-term increments of change; as long as we keep in mind that the *definition of progress*, in the short-term, is that we are actually, measurably, moving **towards our long-term** quantified objectives. And these long-term objectives are constantly, being adjusted to fit necessary long-term directions, and new opportunities.
- People adjust and learn. One reason for failure of the big bang approach systems, is simply that the real stakeholders involved are forced, overnight, to learn far too much, to deal with the new revolutionary system.

- They don't want to. They cannot. You will not give them adequate training (they have a day-job to do). They will protest by failing, and rightly blaming the system.
- With small increments of system change, all related stakeholders have a fair chance to make small learning adjustments, as the changes roll it.
- In any case it is good systems design, to *allow* them to use their old interfaces, and habits, if they want to. The *new* interfaces and habits should be an *option*, for them, that we *hope* they will adopt, because these options are really *better* for that stakeholder, in their humble opinion.

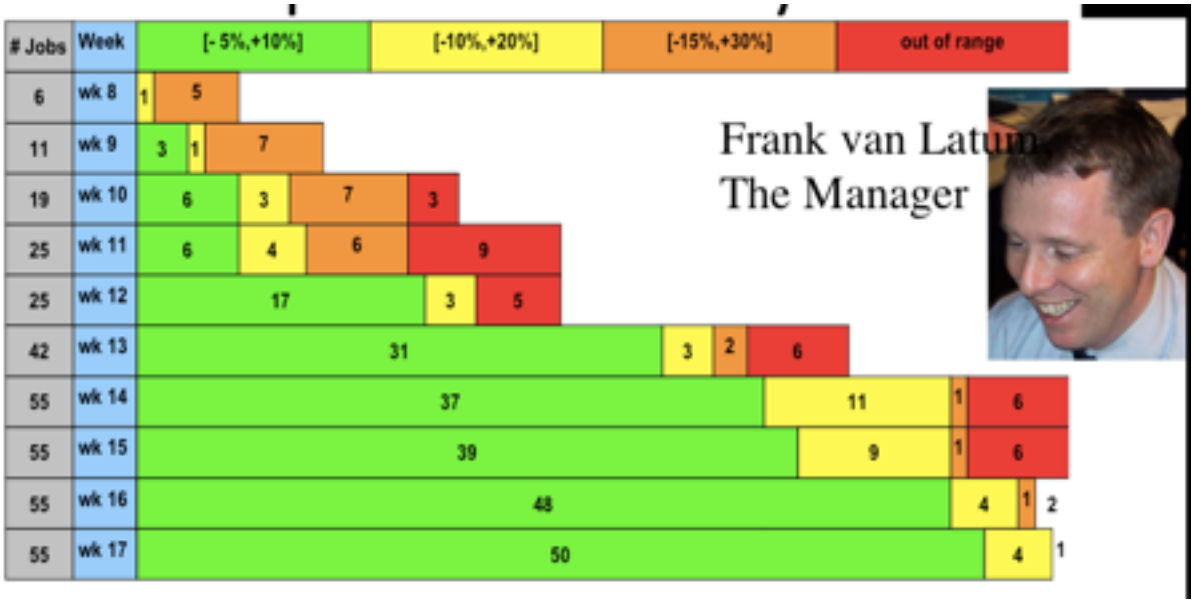


Figure 5.6 Philips Value Delivery Cycles Results. The % is the accuracy of predicting a production run of electronic circuits, before that actual run. Green is good, red is bad.

Case 5.6 Philips

[F1, for more details]

Figure 5.6 was made for a Philips client presentation, where the attribute delivered was ***accuracy*** *of pre-production prediction of production job run-time for a complex Philips production machine.*

We experienced great scepticism from Frank's Director about delivering small increments of results.

But Frank persisted, and his results were later received with spontaneous applause, even by the sceptical director.

The project had been stuck without any results, for a long time previously.

Incremental value is much better than no value.

The main point here is that we start from the existing system:

- We focus attention on the 'squeaky hinges': improving things that have the highest value.
- We avoid using 95% of project cost to replicate the current system functionality, which is usually quite OK as it is.
- We gain credibility with stakeholders by focussing on what they really value. Not, 'excuse us while we build this great new system for 3 to 9 years, which may be worse than the old one.'
- We do *small investments*, which, if we have selected, prioritized and designed right, have far greater return on investment (I'd look for minimum 10X, hopefully 100X when you can cherry pick).
- We look for funding and approval of our change process; based on an early, credible, measurable, track record of delivering value for money, that very vocal stakeholders rave about.

"That which remains quiet, is easy to handle.

That which is not yet developed is easy to manage.

That which is weak is easy to control.

That which is still small is easy to direct.

Deal with little troubles before they become big.

Attend to little problems before they get out of hand.

For the largest tree was once a sprout,
the tallest tower started with the first brick,
and the longest journey started with the first step."

Tao Te Ching (500 BC) From Lao Tzu in Bahn, 1980

Practical Tip 5.5 Evolve Revolutions.

- *Always assume that value delivery increments can best be delivered using the 'old system' as a base.*
  - *Do not listen to seductive arguments about 'building a great new future system' all at once, to replace it.*
  - *There is no proof of that argument, and the result is too often very much worse.*
  - *Mr. Nobody will be responsible, for the big bang project.*
  - *Make teams prove they know what they are doing, in practice, early. Make them confront measurable reality, and learn, learn and learn, early.*
  - *The most successful revolutions, are made by a series of Mountain Goat steps of change.*

Policy 5.5 Evolve the Revolution by Continuous Improvement

- **We will not *ever* start major projects, with high investments, and a long time (even before longer delays) until we supposedly get promised results.**
  - We will undertake the largest-of-scope projects, as a series of incremental proven improvements.
  - We will start with our existing organizations and products, and enhance their capabilities step by step, until we reach the competitive objectives we needed in the longer term.
  - We believe we will get needed improvements faster, earlier and safer by this policy.

- Why?
  - It is safer and surer
  - It is almost always possible, in practice
  - It gives far earlier results

Version 27 Aug. 2015, 23:13 tsg pass 2

Section 5.6 Specific tactics to enable you to decompose strategies

Principle 5.6 Decomposition Principles.

There are at least 20 principles to help decompose your project by value.

How to decompose systems into small evolutionary steps: (a list of practical tips)

1. Believe there is a way to do it, you just have not found it yet![4]
2. Identify obstacles, but don't use them as excuses: use your imagination to get rid of them!
3. Focus on some usefulness for the stakeholders: users, salesperson, installer, testers or customer. However small the positive contribution, something is better than nothing.
4. Do *not* focus on the design ideas themselves, they are distracting, especially for small initial cycles. Sometimes you have to ignore them entirely in the short term!
5. Think one stakeholder. Think 'tomorrow' or 'next week.' Think of one interesting improvement.
6. Focus on the results. (You should have them defined in your targets. Focus on moving *towards* the goal and budget levels.)
7. Don't be afraid to use temporary-scaffolding designs. Their cost must be seen in the light of the value of making some progress, and getting practical experience.
8. Don't be worried that your design is inelegant; it is results that count, not style.
9. Don't be afraid that the stakeholders won't like it. If you are focusing on the results they want, then by definition, they should like it. If you are not, then do!
10. Don't get so worried about "what might happen afterwards" that you can make no practical progress.
11. You cannot foresee everything. Don't even think about it!
12. If you focus on helping your stakeholder in practice, now, where they really need it, you will be forgiven a lot of 'sins'!
13. You can understand things much better, by getting some practical experience (and removing some of your fears).
14. Do early cycles, on *willing local mature* parts of your user/stakeholder community.
15. When some cycles, like a purchase-order cycle, take a long time, initiate them early (in the 'Backroom'), and do other useful cycles while you wait.
16. If something seems to need to wait for 'the big new system', ask if you cannot usefully do it with the 'awful old system', so as to pilot it realistically, and perhaps alleviate some 'pain' in the old system.
17. If something seems too costly to buy, for limited initial use, see if you can negotiate some kind of 'pay as you really use' contract. Most suppliers would like to do this to get your patronage, and to avoid competitors making the same deal.
18. If you can't think of some useful small cycles, then talk directly with the real 'customer', stakeholders, or end user. They probably have dozens of suggestions.
19. Talk with end users and other stakeholders in any case, they have insights you need.
20. Don't be afraid to use the old system and the old 'culture' as a launching platform for the radical new system. There is a lot of merit in this, and many people overlook it.

[4] Working within many varied technical cultures since 1960 I have never found an exception to this – there is always a way!

Figure 5.6 A 20 Principles of Decomposition [URL18, page 314].

Our experience in actually teaching this is, that if we use a real case study of the application of a principle, we can teach one or two principles.

But that means there is a 10 to 20 hours lecture course on this alone. And there is never time for that.

But hopefully if you take the time to read through the principles, you will get the spirit of it. Which is: *there is always a way. Don't ever give up, or take no for an answer. Use your imagination.*

After decades of solving problems of breaking down strategies into smaller strategies, usually being told in advance that it was 'impossible', Tom reviewed this history case studies, and derived 20 principles of decomposition [Fig. 5.6 A] that he had used in practice. This makes decomposition into a 'teachable discipline'. [URL16 A & B]

We also observed that half the smart people, on the same project, who we asked to find decompositions, 'just did it'. Half were still in denial 'it cannot be done'.

The real problem is lack of management leadership*: 'we as a team will decompose into smaller value increments'. 'We will persist until it is done'. Don't make me come in, and embarrass you, by showing how obvious and simple the solutions always are.'*

Case 5.6: UK Radar System

Once, we were confronted with a sceptical Radar Lab Director, on a London training course, who clearly said that he did not believe our ideas applied to his problems.

His lab in South England was building a new radar system for ships, characterized by dual radar antennas, rather than the usual one antenna.

His system was contracted to be mounted, when a particular new ship was launched in about 3 years time.

He did not see what we could do in practice before then, to deliver value incrementally.

We changed the perception of the problem, to which he agreed.

We suggested that his problem was not to construct a radar system and put it aboard a ship. His problem was to ***'increase the accuracy of perception'*** *(for ships and planes at sea). He agreed.*

And we further suggested that he need not wait for the new ship launch, to get results, and proof of better perception. There were plenty of UK and NATO ships out there now, doing peacetime exercises, that could trial the increments of perception improvement, at sea with real ships, real planes etc. He agreed.

His new task was to gradually build the data bases, of the dual radar profiles, of friendly and enemy ships and planes, so that accurate identification was faster and surer. Most dangerous planes first, we assumed. This he agreed to.

He then said. "It is now obvious to me what we have to do. But the thing that amazes me is that not one single engineer at our labs has ever thought a thought like this!"

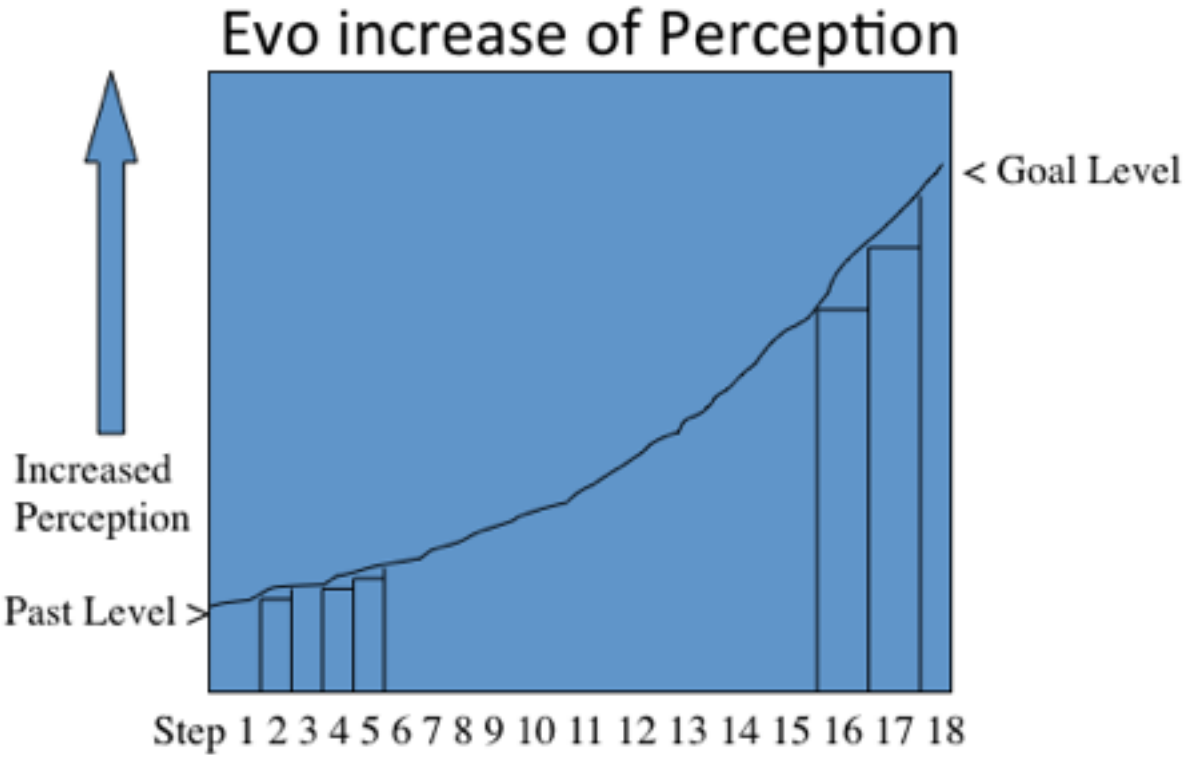


Figure 5.6 B. Case Study. Each step delivers higher accuracy of perception of the surrounding planes and ships.

Conceptually each 'weekly' step for the next years, would add data profiles of planes and ships, along with algorithms for processing them quickly enough to know if they were friendly, before feeling forced to 'shoot in case of enemy attack'.

Hopefully they could avoid the embarrassment of shooting down a friendly civil aircraft. Think South Korean Air Flight KAL007, Iran Air Flight 655 by USS Vincennes (1988), Malaysian Airlines MH370, over Ukraine (2014)

Practical Tip 5.6. Competing Decomposition.

- *Try 'competing teams' to find interesting decompositions.*
- *Never give up.*
- *Do the decomposition yourself, if your staff fails you.*

Policy 5.6 Use 2% of budget and prove you can use it wisely.

- ***No Evo (value delivery) cycle shall exceed 2% of total budget before delivering measurable results to a 'real' environment.***
- ***No Evo cycle will exceed 2% of total project time (= one week, for a 1 year project) before it demonstrates practical measurable improvement, of the kind you targeted.***
- ***Evo cycles which deliver the most 'planned value to stakeholders, for the 'resources they claim', shall be delivered first, to the stakeholders. Do the juicy bits first!***

  Why?
  - *Because decades of experience have demonstrated that this is almost always possible*
  - *It results in an early and continuous flow of value*

Version 27 Aug. 2015 tsg pass 2 23:51

Section 5.7 A simple checklist for practical strategy decomposition.

Principle 5.7 The Unity Method.

**The '111111' decomposition method
will help you decompose
into very small value strategies**

We were looking for something simpler, and even more general than the 20 decomposition principles, discussed above. And we observed a useful simplification which we now call the 111111 principles, or if that is too complicated, the 'Unity' Principle. [URL16 C].

We think people get *overwhelmed* by the volume of detail and consequent complexity of the project, and system, they are working on. Quite reasonably so, they *are* complex [URL87].

So the Unity method is a way of saying. Avoid Analysis Paralysis. Don't do 'nothing except planning and meetings', until you feel everything is 'mastered'. Mastery won't really happen anyway.

Just acknowledge that, '1 step forward' is better than 'zero steps forward'. One slightly happier stakeholder, is better than none.

Here is a summary of the method, the decoding of the 111111:

- 1% increase at least
- 1 stakeholder
- 1 quality or value
- 1-week delivery cycle
- 1 function focus
- 1 design used

At each value delivery step, for example weekly, you should plan to get

1. At least 1% (or more) value progress towards some objective.
2. And you should satisfy needs, partially, for at least 1 really important one, of your many, stakeholders. Don't worry about the others.
3. And you may have 10-top objectives or 50 but, just deal with at least *one* of them, rather than none.
4. The road to full satisfaction of your objectives may be years away, but get something done *this* week. No excuses, you have a lot of choice to pick from.

5. And the system may have 500 or 5,000 functions (*what* it does). But don't let that overwhelm you. Choose a frequent and critical one, of maybe more than one, if you can do it in a week, and just do *that* one.
6. And you may have a top ten strategies, and 100 sub-strategies to implement, but pick *one* you can actually implement in your 1 week cycle, and just do it. See what happens.

Some real progress, for 1 important thing, speaks louder that 10 meetings. There are doers and talkers; who are you?

Case Persinscom [URL16 C]

At the end of the First Gulf War, victorious Commanding General Stormin'. 'The Bear', Norman Schwarzskopf (1934-2012) had one complaint. And he got people's attention.

The Army's Personnel system, 'Persinscom', was so bad, that we might have lost the war, if we were dependent on it.

Well, it had been operational 8 years. Millions had been invested in it. But, when the General (I am imagining this detail) said 'How many Bridge Engineers do I have in Kuwait right now ? One day before launching the counterattack, the 3-day war. The system said '25 engineers', 3 hours later. But the General, while awaiting the delayed answer counted 100 engineers, outside his window.

We were invited to help solve the problem. We used a Startup Week [URL5] to identify objectives, strategies, and do an impact estimation table, Monday to Wednesday. Thursday, I told them, we were going to find a way to deliver value next week.

They were incredulous at the idea. "That's not going to happen. This is the Pentagon Bureaucracy etc.".

We could not resist the jibe that the General completed the war in 3 days, but they could not make the smallest improvement in his Personnel system in 5 days. We suggested they were out of touch with the new military reality. They were sure *we* were crazy.

But, at Thursday end, they said, 'we found it'. 'What', we said? Something we can deliver next week for value! Good, that was the idea. What is it?

'You won't believe we never thought of it before!' What is it? I asked.

'We discovered that the General got a delayed response, because he was put at the end of a queue, of thousands of other soldiers preparing the war using the same system.'

The solution was obvious. If General, move to head of queue. And it was actually done during the planning week itself!

So, **111111.** We improved, **1** Value, the response time; for **1** Stakeholder (the General), using **1** design (if General move to head of queue), for at least **1** function ('How many Soldiers here?'), at least **1**% of the way to the improvement Goal. In less than **1** week. Mission accomplished.

"When the Normandy Invasion was planned, a very specific strategic objective was given, and that strategic objective was the basis upon which the plan for the Normandy Invasion was derived."

Norman Schwarzkopf

General Schwartzkopf, at the head of a queue. Where he belongs.

Bono has composed a song about our method:

<u>One</u> love, <u>one</u> blood
<u>One</u> life, you got to do what you should
<u>One</u> life, with each other
Sisters, brothers
<u>One</u> life but we're not the same
We get to carry each other, carry each other
<u>One</u>
<u>One</u>

© POLYGRAM INT. MUSIC PUBL. B.V.

Practical Tip 5.7 Take a slice and try it out.

- *Start by specifying your currently most critical stakeholder*
- *Then ask what is their highest most-critical value objective*
- *Go for it*

Policy 5.7 111111 Simplification

We will do at least a little something towards our goals regularly. No excuses for zero results.

- Why?
  - The 111111 method is about giving people a practical checklist to help them decompose strategies into value delivery steps. It works!

Version 28 Aug 2015 11:44 tsg pass 1

Section 5.8 The 'Project Startup Week' Process.

Principle 5.8 START WEEK: [URL5]

The Project Startup Week
will kick start your
'decomposition by value' process.

How long a time do you use, to do feasibility studies, planning, and system building, before actually delivering any results or improvements?

The Initial Week for most all projects.
We have a fairly radical approach to most any project. We think one week is enough for initial long term planning of objectives and strategies. It does not matter how big the project is. Project duration or size is almost irrelevant as far as we are concerned.

OK we are not fanatic. Planning World War III, we'd allow a month. But it might be dangerous to use so much time. Even 24 hours might be dangerously much!

The reason is that your 'best' planning is *during* the implementation process, where you can get realistic *feedback* from your initial plans.

You need quick results too, and planning is a pure overhead. So focus on results, and let the planning activity be there in parallel, in the background. Interacting with real life. Planning right, is good, but good planning needs feedback from reality.

Over the years, we have developed and refined an initial planning cycle of a week. Maybe it suits us as international consultants, in and out in a week. But clients have asked us to teach their own corporate teachers so they could use the method on all their projects. One example was Boeing, El Segundo, after we did it on 25 aircraft projects.

The big idea is simple:

1. Identify, agree and quantify the top 10 objectives, in a day 1.
2. Identify the top 10 most powerful strategies, in a day 2.
3. Use a day 3 to draft an impact estimation table, to see if strategies are roughly good enough for meeting the objectives.
4. Use a day 4 to find one or more sub-strategies, that can get you delivering real measurable value next week.
5. Get next week's value delivery approved by the 'boss', day 5.

That's it. Disciplined time-boxing works. If you don't constrain the planning time, then it can go on forever. The degree of quantification, clarity, and practicality we achieve in the week, is, we think, clearly superior (clearer and right level) to what many people have planned, who have used far more time to do their initial planning.

At one extreme, we *could* make an argument for dropping this first week's planning, and just start doing something useful; while building up top-level plans in the background. But that might be *too* radical, and it is not our experience.

See Section 9.10 for more detail on the Project Startup Week.

CASE 5.8: LOO WATT THE STARTUP [URL32]

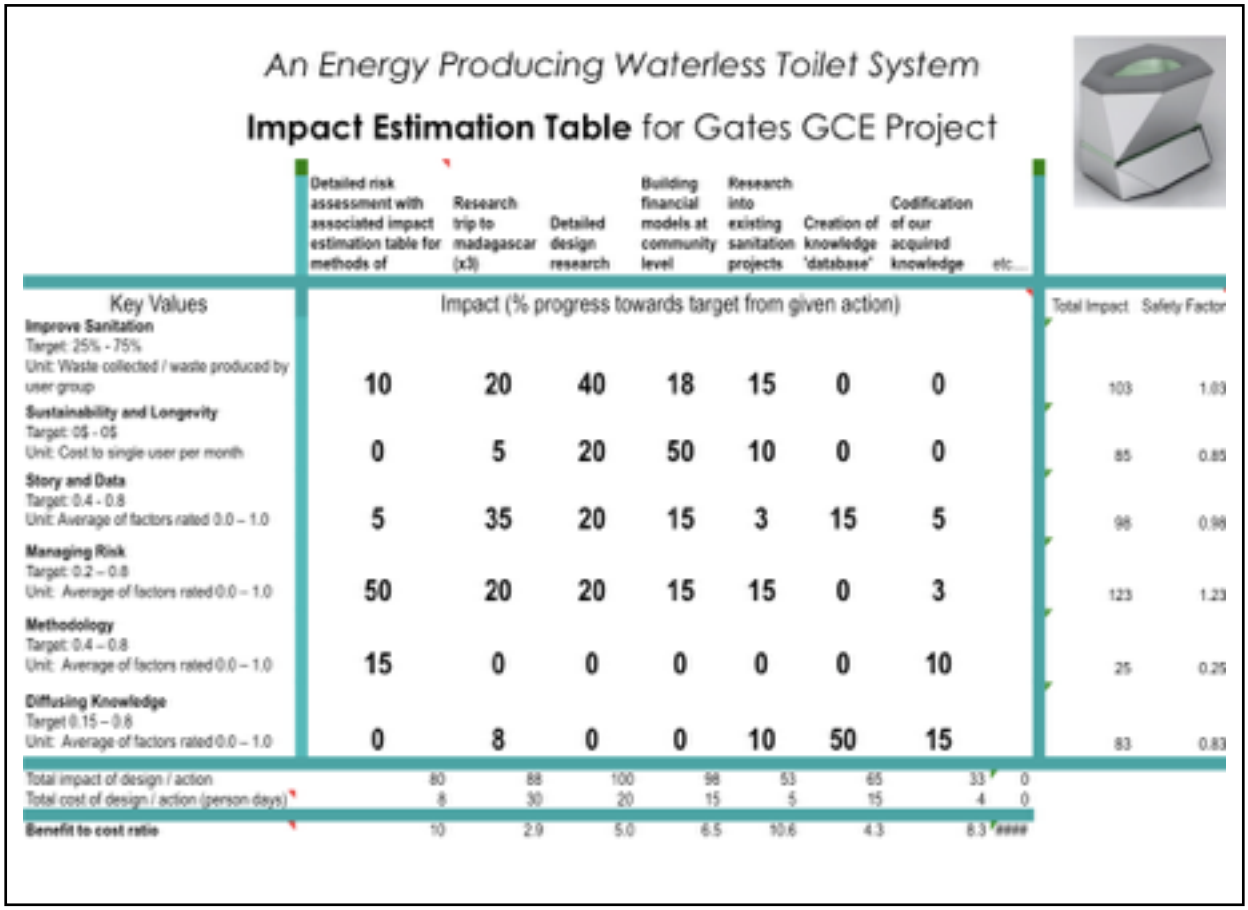
A start-up business, spawned in an 'Innovation Incubator' ('Design London') at Imperial College London, asked for help to get a clear high level perspective of their objectives and strategies.

We only used a single day, to do our 'Start-up week'. But that was sufficient in this case.

Kai and Tom met with the 'Loo Watt' team (for free, we could not resist adding 'Toilet Development' to our CV) of about 5 people, as well as their Advisor (Nicholas Coutts).

We divided the day into 3 parts:

- Objectives, Strategies, and an Impact Estimation Table for those Objectives and Strategies. Essentially the first 3 days of the Start-up week, compressed into a single day.



An Energy Producing Waterless Toilet System

Impact Estimation Table for Gates GCE Project

Key Values	Detailed risk assessment with associated impact estimation table for methods of	Research trip to madagascar (x3)	Detailed design research	Building financial models at community level	Research into existing sanitation projects	Creation of knowledge 'database'	Codification of our acquired knowledge	etc...	Total Impact	Safety Factor
	Impact (% progress towards target from given action)									
Improve Sanitation Target: 25% - 75% Unit: Waste collected / waste produced by user group	10	20	40	18	15	0	0		103	1.03
Sustainability and Longevity Target: 0$ - 0$ Unit: Cost to single user per month	0	5	20	50	10	0	0		85	0.85
Story and Data Target: 0.4 - 0.8 Unit: Average of factors rated 0.0 – 1.0	5	35	20	15	3	15	5		98	0.98
Managing Risk Target: 0.2 – 0.8 Unit: Average of factors rated 0.0 – 1.0	50	20	20	15	15	0	3		123	1.23
Methodology Target: 0.4 – 0.8 Unit: Average of factors rated 0.0 – 1.0	15	0	0	0	0	0	10		25	0.25
Diffusing Knowledge Target 0.15 – 0.8 Unit: Average of factors rated 0.0 – 1.0	0	8	0	0	10	50	15		83	0.83
Total impact of design / action	80	88	100	98	53	65	33	0		
Total cost of design / action (person days)	8	30	20	15	5	15	4	0		
Benefit to cost ratio	10	2.9	5.0	6.5	10.6	4.3	8.3	####		

Figure 5.8 The Impact Estimation Table, summarizing the first 3 steps of a project start-up week: Objectives, Strategies, and estimation of how well the strategies will help meet the objectives.

Their advisor (N. Coutts) told the team, that if they wanted to be sure of getting the $1 million promised by the Gates Foundation, for a successful Madagascar Trial, they should use our 'Planguage' methods to model their project.

The team embraced the methods, after our 1-day coaching. Coutts later told me they were spreading the word, to all their 20 fellow incubator projects, that *they* should also use Planguage to think about their projects.

They also told Coutts that the method helped them to *keep their eyes* on the main high-level systems considerations. For example that it was not just about an Eco Toilet, but it was about convincing people to change their culture, by adopting them in their homes; and ultimately about reducing disease.

After the Madagascar trials, The Gates Foundation gave them the $1 million grant ! As you can see from their website (loowatt.com), they also won a number of innovation awards. We were proud to have given them a tool to guide their excellent work.

End of Case 5.8 Watt Loo

"After two weeks of working on a project, you know whether it will work or not."

***Bill Budge**, Pinball Construction Set creator*

The Pretotyping Manifesto

Make sure you are building the right it before you build it right

innovators ***beat ideas***
pretotypes ***beat productypes***
data ***beats opinions***
now ***beats later***
doing ***beats talking***
simple ***beats complex***
commitment ***beats committees***

Source, URL80, Alberto Savoia, 'Pretotyping It'

Practical Tip 5.8 Quick Startup.

- *Try a quick run through in one day of top ten objectives quantified, top 10 strategies and an Impact Estimation table.*
- *Limit them to one page each, 3 pages in all.*
- *Get one or two people to work out a draft (objectives, strategies, IE Table) in a day or 2 beforehand.*
- *Then use your larger project group meeting day to discuss, improve and approve these first 3 outputs*
- *At the end suggest the top 5 value intersections, as basis for planning what to do next week.*
- *Delegate that to a small team to carry out, deliver 1, next week.*

Policy 5.8 Plan a week, then deliver value next week

We will plan the main objectives, strategies, estimates and initial value producing actions in a week; then start delivering real value the 2nd week of the project.

Why?

- The Startup Week, is intentionally constrained in time to give you high level management control over value, without too much bureaucracy
- You will learn more and know more if your team dives in on week 2, and tries to deliver real value to real stakeholders, than any initial additional 'feasibility study' will give you.

Section 5.9 Value delivery steps are also a 'hypothesis' we need to verify.

Principle 5.9 Trial & Error.

A Value delivery step is
a hypothesis, an experiment,
a piece of research
needing to be proven correct,
or not.

Experiment and Know.
A value delivery step is like a scientific experiment, the value of an experiment, is knowledge; what works, and what does not.

So, 'value delivery steps' seek knowledge, but if they work; we are quite happy to keep those 'delivered value steps' in place, and to scale up.

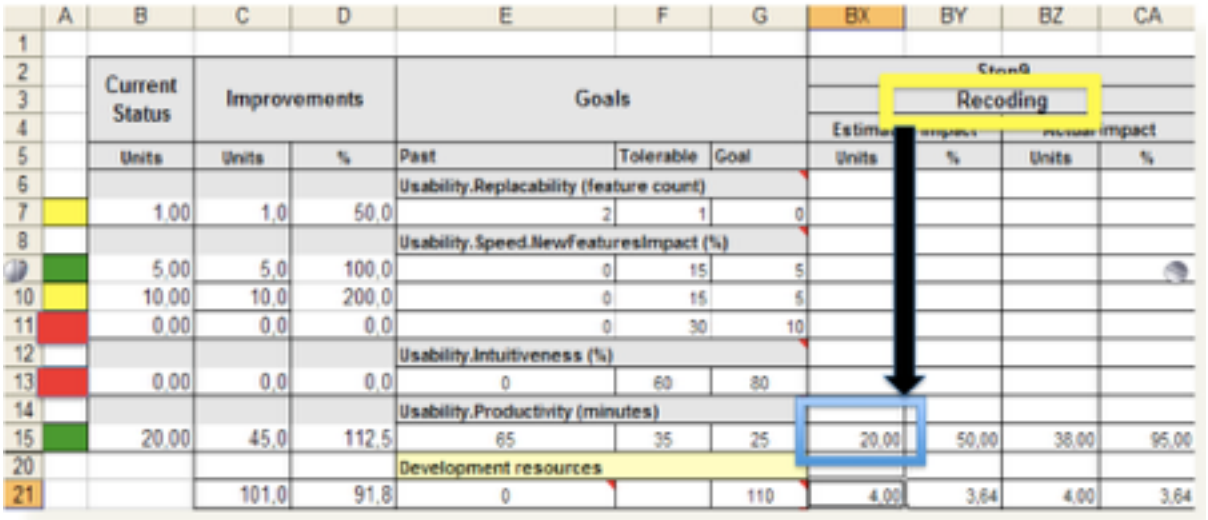
We cannot really know if a planned value delivery step will work at all. Anything can happen to invalidate it.

But in the long run, most steps will tend to deliver reasonable value; especially if our planning methods tip the odds in our favor. We do not just pick 'any sub-strategy at random', and try it. We need good reason, such as estimated impacts.

Using 'Impact Estimation', we prioritize sub-strategies that will probably pay off the most. [URL21]

- We estimate the primary impact on an objective we are focusing on.
- We estimate side-effects impacts on the other top objectives.
- We estimate the ± uncertainty, or range of expected impacts.
- We document the 'evidence' and 'sources' for our estimates.
- We assign a 'credibility level' to the estimate, based on the *quality* of the *evidence* and *source*.

So by the time we have done this for some sub-strategies, we have a fair idea of potential winners and losers. Even if a winner is not all that is expected, it will usually give pretty good value for money, compared to a 'loser'. Losers, in our ratings, rarely surprise us with 'fantastic unexpected value for money'.



	A	B	C	D	E	F	G	BX	BY	BZ	CA
1											
2		Current Status	Improvements		Goals						
3								Recoding			
4								Estim		Actual impact	
5		Units	Units	%	Past	Tolerable	Goal	Units	%	Units	%
6					Usability.Replacability (feature count)						
7		1,00	1,0	50,0	2	1	0				
8					Usability.Speed.NewFeaturesImpact (%)						
9		5,00	5,0	100,0	0	15	5				
10		10,00	10,0	200,0	0	15	5				
11		0,00	0,0	0,0	0	30	10				
12					Usability.Intuitiveness (%)						
13		0,00	0,0	0,0	0	60	80				
14					Usability.Productivity (minutes)						
15		20,00	45,0	112,5	65	35	25	20,00	50,00	38,00	95,00
20					Development resources						
21			101,0	91,8	0		110	4,00	3,64	4,00	3,64

Figure 5.9 The Confirmit Case [URL28].

Case 5.9 Confirmit. Decomposing Value Delivery, One Strategy at a Time.

Here is a snapshot of the end of week 9, of 12 weeks, before international product update release.

The team decided to attack the 'Productivity' objective (EFG14). To reduce the time needed to produce a marketing report from 65 minutes (15E) to 25 minutes (15G).

Amongst 10 brainstormed strategies, in half an hour, the one tagged 'Recoding' got the best estimated impact, 20 minutes saving (BX15), or 50% (BY15) of the way to the Goal.

This was the best option, that would fit into a week cycle of work (BX21 = 4 days duration).

So the team went for it.

At the end of the week, Microsoft Usability Lab, measured a saving of 38 minutes (BZ15). So the sub-strategy 'Recoding' was twice as good as estimated.

Confirmit then used the following weekend to tweak Recoding, getting down to 20 minutes.

The final accounting by the beginning of week 10, was 12.5% more than the Goal (D15).

Leonardo da Vinci

"Although generally recognized as the greatest genius of all time, Leonardo made many colossal mistakes and staggering blunders."

"Despite mistakes, disasters, failures, and disappointments, Leonardo never stopped learning, exploring, and experimenting.

He demonstrated Herculean persistence in his quest for knowledge."

[11, Gelb]

Leonardo himself wrote:

"I do not depart from my furrow.

"Obstacles do not bend me"

"Every obstacle is destroyed through rigor"

Practical Tip 5.9 Just Do It!

- *If you can't yet figure out the 'smartest thing to deliver', then just **do something rather than nothing**.*
- *Then hopefully you will get better insights into stakeholders and values, and the practicality of value delivery.*

Policy 5.9 Test Good Stuff Fast.

- **We will test the most promising estimates for strategies, on our objectives, quickly.**

Why?

- There is only one way to know if things really work. Do. Study.

Version 28 August 2015 19:28 Pass 2

Section 5.10 Proving you can really deliver value: one step at a time.

Principle 5.10 Keep It Small.

Decomposition,
and trying things out,
is a good measure of our real ability,
and our real knowledge;
as well as a rapid teacher and manager.

Our real world is complex and full of surprises, as you are well aware. The only way we know of, to deal with it, is to confront surprises and complexity as soon as possible [URL87].

If you can measure real progress towards the numeric goals you set, then that is the only proof, that you know what you are doing.

If you can successfully decompose larger strategies, then you understand those strategies. If you can successfully make your strategies work in a complex human environment, like an international market, then that is proof that you understand that domain.

We have given some decomposition tools earlier [Chapter 5], but here are some additional thoughts or tools:

Here are some additional ideas for **decomposition of the strategy:**

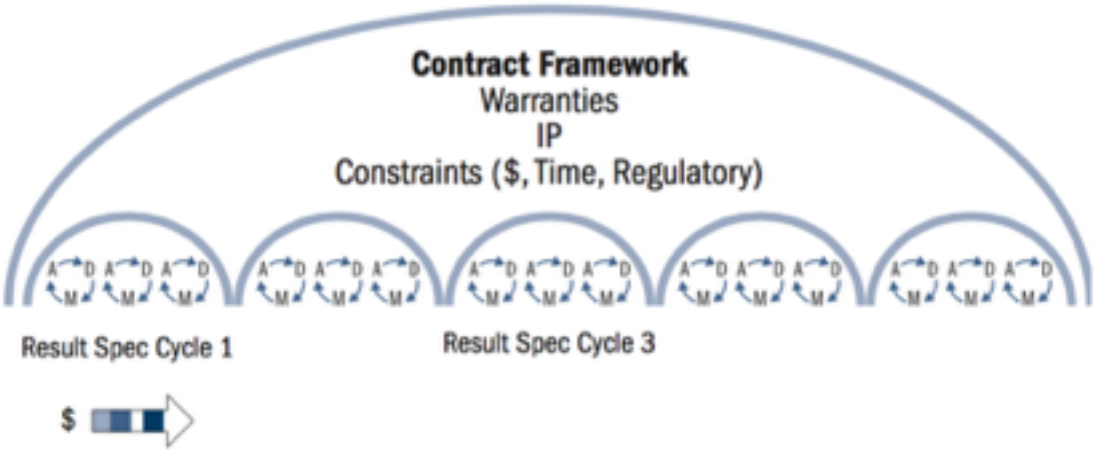
- Start with the old strategy, but add something to give additional value. Do not replace or alter the older strategy initially.
- Start with a bare bones core strategy, addressing limited function (what the system does), and build additional function gradually
- Give customers or users a *free choice* between the old strategy and your new strategy; and see if your new strategy is good enough to tempt them *voluntarily*, and via word of mouth, to choose and remain with your new strategy. If not, maybe it is not good enough; and that is *your* responsibility. Forcing them to the new stuff, is a bad idea.

Here are some additional ideas for step by step **deployment of a given strategy:**

- Deploy it on yourselves, locally, so that if it fails, you can keep a lid on it

- Keep it in your local area before you go national. People you know and who know you, and you can easily observe and meet.
- Master your home country before you try to go international.
- Do one additional new country at a time; do not try to release the strategy in multiple countries simultaneously, even if you already have a presence there, with older products, services, systems.
- Try your new strategies on volunteers.
- Try new strategies on present customers before new markets.

Here is an example of a decomposition, by Objectives, and a choice of what to do next week.



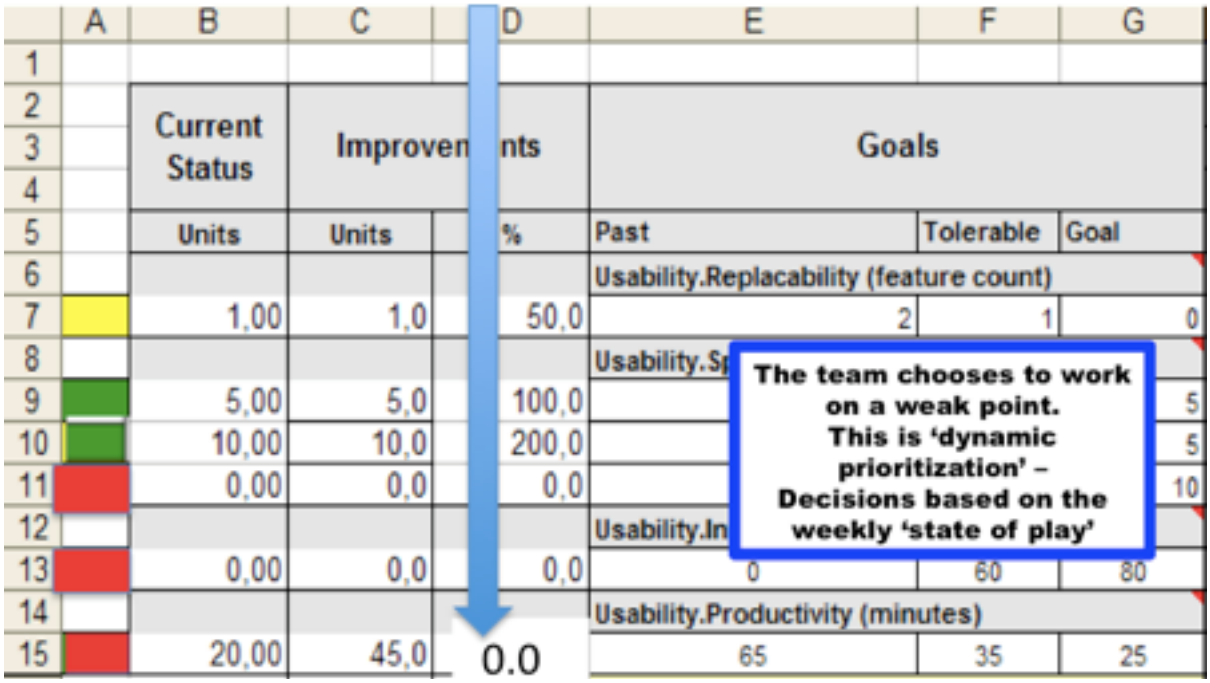


Figure 5.10: From the Confirmit Case [URL28].

Case 5.10 Confirmit.

The situation for the 4 person team, at week 9 of 12 before International Product Update Release, is shown in the column D (Improvements %).

Two of the goals are already reached (100%, 200%). Priority color Green.

One Goal is halfway there (50%), and has reached its Tolerable level (F7). Priority color Yellow.

Three goals are not at all improved from their Past Level (0%).They are priority color Red.

We have 4 choices to work on: one (or more) of the % , or the 50%.

Traditional Contract Model	Result Contract Model (Agile)
Requirements are contractual and specified up-front in the main contract.	Requirements are specified at the start of each result cycle.
Changes are managed by means of the change control mechanism.	Requirements are more resistant to change than traditional output requirements. Target outcomes are only specified at the start of each result cycle, are operational for shorter periods of time, and therefore are exposed to less change.
Analysis, design, development, and testing occur sequentially. Big Bang or Waterfall.	Each cycle must deliver value, so design and development occur concurrently. A systems view must be taken, providing real results in real life.
An all or nothing solution.	The solution evolves as a serious of result deliveries.
Constituent modules of software are worked on independently until integration takes place.	There is continuously working and stable software and hardware system.
Testing is used as a contractual tool at the end of the development process.	Testing occurs throughout the development process, providing feedback for improvements.
Success is measured by reference to conformance with the change-controlled contract.	Successs is measuered, cycle by cycle, by requirements delivered, driving value to the customer.

They chose 'Usability.Productivity', currently at 65 minutes, for 'time to setup a marketing report' (The Scale).

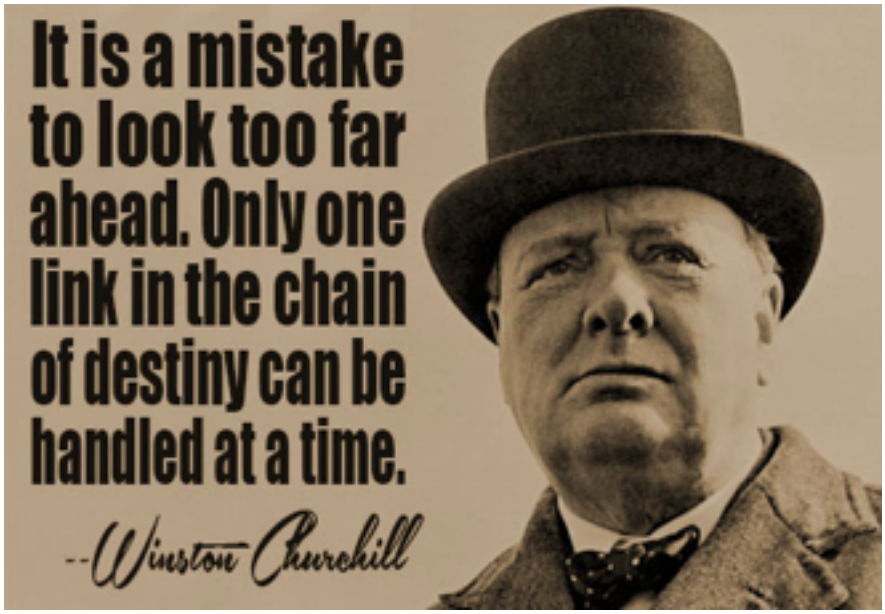
And decided to see if they could improve that, to at least a tolerable 35 minutes (F15), and perhaps to the Goal of 25 minutes (G15).

They did in fact achieve, that week, getting it down to 20 minutes: a 112.5% impact.

So, this is a decomposition based on dynamically available choices for 'filling the gap' to the Goal levels.

"It is a mistake to try to look too far ahead. The chain of destiny can only be grasped one link at a time."

Sir Winston Churchill



Practical Tip 5.10 Learn Fast.

- *Get some results fast. Learn from it. Get better. Get more.*

Policy 5.10 The Mountain Goat Approach

- **We will work towards our larger long-term goals, one practical step at a time: as sure-footed as mountain goats.**
-

Why?

- Reduce scale of losses
- Improve track record of value delivery: the whole main point of value planning, projects, management, and most of our work

Churchill: 'Sooner or later we will get an iPhone, maybe even an Apple Watch! Check with Dick Tracey, he has one"

Winston Churchill smokes a cigar as he holds a radio to his ear and watches the descent of American paratroopers during his visit at Fort Jackson, South Carolina.

Version 28 Aug. 2015 19:49

Chapter 6. Prioritization, Evaluation

[URL33, 2 PAPERS AND TRAINING SLIDES]

Chapter Introduction

Management is about making choices. The moment you recognize you have limited time, people and money, you know you have to make choices.

Another word for `choices' is *prioritization*: choosing one thing ahead or, or instead of, another.

So, this raises the question of whether there are some methods of prioritization, which are better than others.

We have some specific tools and ideas for the reader, in the chapter that follows.

Much of the basic ideas necessary, about clear objectives, clear and decomposed strategies, and impact estimation, have been covered in the earlier chapters.

You will get more out of this chapter, if you have studied the earlier chapters. However we hope you can dive in an read this chapter independently, initially.

But this chapter will focus deeply on prioritization itself.

Section 6.1 A Better Way to Prioritize, and make decisions.

Principle 6.1 *You Might Get It All, If…*

If you have *infinite* resources,
you can have it all –
'choices'
are not necessary

How do you prioritize? How do other people prioritize?
Some people, and *all of us* sometimes, do not *consciously* prioritize, in the sense of a 'well-documented decision-making process'.

We just let a 'stream of events' hit us, and react to them, as they come, and perhaps with some sense of urgency (crisis, danger, threat) or importance (the boss, your near family, your favorite hobby)

If you are happy, and that works for you: fine. Nature has given us some pretty good prioritization methods as individuals, to help us cope and survive. Societies, and cultures, have also worked out more-or-less good prioritization methods, to help them survive.

Our concern in this book is with the following environments:

- Competitive international business; everything from start-up businesses, to large multinational corporations.
- Government planning, Town to Multinational.
- Non-Profit Organizations, such as emergency aid, research.

These environments have the following in common:

- A massive stream of new technology and economics.
- International ability to cooperate, to work together on projects, and to fund projects from non-traditional sources.
- Competition constantly emerging. Not just the usual business competition, or the ancient military competition, but competition for ideas and paradigms; at breath-taking speed.

So, **we,** in the above environments, could hopefully make good use of a '*better than traditional* way' to 'decide what to do, next'.

The 'better way to prioritize' would need to have all or some of the following characteristics [URL74, Priority Papers]:

- *Ability to cope with a **large number of concurrent objectives.***
- *Ability to cope with a **large number of concurrent strategies** for reaching the objectives.*
- *Ability to deal with **rapid change** in everything: objectives, strategies, constraints, and related environments.*
- *Easy to **scale** up and down, to smaller and larger problems.*
- *Ability to model **large and complex** systems.*
- ***Easy of learning** and using; for any level of planner.*
- *A clear **logic**, that can respond to any priority policy, that people might choose to use.*
- *The **ability to quantify** priority levels in objective ways, not using emotive or manipulative 'marketing' language.*
- *Capability of **automated tools** [URL73] to help reflect changes quickly and cheaply: not least ability to ask 'What If ..? ' questions.*
- *Capability of easily being **extended or tailored** to any interesting planning purpose.*
- ***Intelligibility** of results, to decision makers.*
- *The ability to integrate considerations of **risk, uncertainty**, bad data, dubious assertions, with better risks and certainties.*

We hope we have such a set of ideas for you here, or at least, a basis for your extensions in any interesting area.

Planguage, surprise, meets all of the above conditions.

Figure 6.1 Stages of collecting data, in order to make priority decisions, using Planguage.
These data (stakeholders, objectives, strategies, etc.), detailed earlier in this book, are the foundation for better decision making.
Courtesy Tse and Kahlon [URL30].

The 'system factors text is quite small, so here is an enlargement:

Step 1: The key stakeholders should provide a vision of the existing vision view of the existing issues to be addressed 6 expectations required.

Step 2: Some stakeholders are unidentified from the system, therefore identification of all is required.

Step 3: Identify the requirement from the stakeholders which provide insights towards the scope of the system.

Step 4: Pinpoint existing measures present within the organisation. Planguage commends using all values to facilitate constraint.

Step 5: These are scale performance specification, setting parameter target area(s) for achievement.

Step 6: Identification of key designs to the requirements and establish possible credential impacts.

Step 7: The specification of performance should be reflected to the requirements and likely impact.

Here is a *simplified* version of our Planguage prioritization method:

1. **Quantify** a critical objective
   Like:
   '**Goal**: Increase Number of Heavy Users in China by 30% in 1 year.'

2. **Identify** some potential strategies:
   Such as S1, S2, S3. These are the name tags of 3 strategies
   Like :
   '**S1**: Reduce 1st year User cost by 90% for 2-year contracts.'

3. **Estimate** the Impact of each strategy in terms of % ability to reach your specified Goal (1. above) (on time)

   S1 -> 100%
   '->' is a Planguage arrow, meaning 'impacts'.

   '100%' means 'we estimate we will get to the Goal level on time'.

   S2 -> 50% 50% means, 'we expect to get halfway to our Goal level on time'.

   S3 -> 10% 10% means 'we expect to get only 10% of the way to the Goal by the deadline'.

This is the **simplest structure** for making a quantified choice.

S1 wins; it is 'prioritized'.

Because it is expected (estimated) to give the highest impact.

- *'This is incredibly simple!'*, we hope you feel *that* way, and are saying it to yourself.
- But!
  - Most people do *not quantify* most of their critical objectives at all: *they have no basis for prioritization.*
  - Most people *name* their strategies, but do not *define them well enough* to *understand* their impacts

- Most people do not attempt to *quantify the impact* of a strategy, on a Goal.
  - Especially not for *qualitative* objectives ('User-friendly', 'Secure').
  - Even cost estimation is poorly done, if at all. But cost is not our biggest worry, or concern just here.

"That's what building a body of work is all about.

It's about the daily labor, the many individual acts, the choices large and small that add up over time, over a lifetime to a lasting legacy.

It's about not being satisfied with the latest achievement, the latest gold star, because the one thing I know about a body of work is that it's never finished.

It's cumulative.

It deepens and expands with each day you give your best.

You may have setbacks and you may have failures, but you're not done.

You haven't even started."

Barack Obama (1961 -), *Arizona State Commencement Speech, 2009*

Practical Tip 6.1: Goal Priority, not Power Plays.

- *Prioritize using strategies that satisfy your defined objectives and constraints*
- *Do not accept any 'prioritization' from opinionated people.*
- *If they get pushy; ask them if we can* ***'fail to respect agreed and specified objectives and constraints', with their formal documented permission.***

- *They will 'chicken out'.*

- *Get people to:*
  - *either constructively suggest changes in the agreed objectives and constraints*
  - *or, try to contribute, and specify, facts about the expected impacts of strategies on the agreed organizational-or-project objectives and constraints (resources in particular)*
  - *or to somehow prove or guarantee, with their 'skin in the game' that their ideas will 'work and deliver the expected value, at the expected costs'.*

Policy 6.1 Logical Priority.

- **We will prioritize, based on clear and logical, ends and means (objectives and strategies).**
- **We will re-prioritize regularly and continuously: THINGS CHANGE.**
- **Our prioritization process is about 'efficiency': values/ resources.**

Why?

Most individuals, and most business decision-making cultures do NOT prioritize based on a 'well-defined set of objectives and constraints'.

The inevitable result of this, is bad decisions. Decisions which are not based on the agreed objectives, budgets, people-resources and deadlines.

Decisions are then based on personal opinions, prejudices, power.

All well intended, but not very wise.

Section 6.2 Priority is dependent on many changing factors

Principle 6.2 DYNAMIC PRIORITY:

Static initial prioritization is unrealistic – things change

As soon as we mention 'priority', many people have a knee-jerk reaction: 'like weights?', they say.

They are familiar with the use of weighting numbers (high-medium-low priority, or '1 to 5' weight) in many situations, like school grades, and Balanced Scorecard.

Weights have their uses, in indicating (static, subjective) priorities. But we are going to share our opinion with you that *they are not a good planning tool*. Not for complex, ever changing, complicated systems.

In fact, weights are more dangerous than useful, for the kind of planning we are dealing with here.

The simple reason is: Weights are *fixed*; but priorities are *dynamic*.

But there are more reasons.

We hope to convince you that you should *totally avoid initial fixed weights*, to indicate your priorities, in *serious* planning.

Let us begin with the arguments for this position, 'against fixed weights'.

This is a series of statements which help our argument against weights. *If you want to you can skip it until you need to know the argument against them.*

- Priority means 'claim on remaining limited resources'.
- The 'claim' means that we will prioritize using resources for those strategies, which will best help us deliver, our critical objectives.
- We assume that we are engaged in a continuous process of trying to reach competitive goal levels.
- We cannot continue to prioritize doing strategies for objectives, that we have *already* reached (100%+), at competitive enough levels.

- We can ill afford to prioritize strategies for objectives whose goals we have *almost* reached (99%+).
- We need to use limited remaining resources (like 5% of our budget, 10% time to deadline) to do strategies which help close the *biggest* gaps (like 50% to Goal) towards our set of goals.
- The amount of remaining resources is usually constantly declining (100% -> 80% -> 60% -> 30% -> 10%)
- We are assuming that our strategy-delivery process, is trying to get us to reach all our multiple critical Goal levels, through a series of value delivery cycles.
- And we assume that we are re-planning, after every value delivery cycle: trying to find the smartest way to maximize progress towards goals, using still-available resources.
- Anything can happen, we assume, at any cycle. You can make any arbitrary progress towards any one goal, or several goals.
- You can make negative progress too. You can consume any amount, and number, of available resources, to surprising and arbitrary degrees.
- In short; you know almost nothing for sure about the final outcome, in relation to your goals and resources.

The conclusion of these observations is:

- The facts needed to determine your current priority, are constantly and arbitrarily changing
- The facts needed are: remaining limited resources, and remaining distance to Goals
- Only when these facts are available, can you search for a suitable strategy: one that will move you towards your Goals as much as possible, in the (weekly) cycle duration, with as little use of other resources, like money, as possible.
- We can prioritize any strategy, which we can find, that gives best progress, towards residual Goal levels, at the lowest consumption of residual resources.

If this seems complicated, remember it is essentially the way your body prioritizes its' needs! Like, air, water, food, warmth.

Figure 6.2 A. What is your instinctive answer?

Most people answer 'Air'. Which is the correct answer to *another* question: 'which of these 3 if totally denied, would kill a human fastest?'

But, that was *not* our question!

Imagine you are in the Sahara desert. Plenty of air. But you are running low on water, food, and you are about to die of overheating.

You get the picture.

It depends on:

- How many critical factors can kill you,
- What levels of those factors will actually kill you,
- How close are we to each life-threatening factor, and
- at what rate are we moving towards death levels.

In short, you need 2 or 3 facts about each critical factor.

Sort of like being hooked up to medical measuring instruments during a surgical operation!

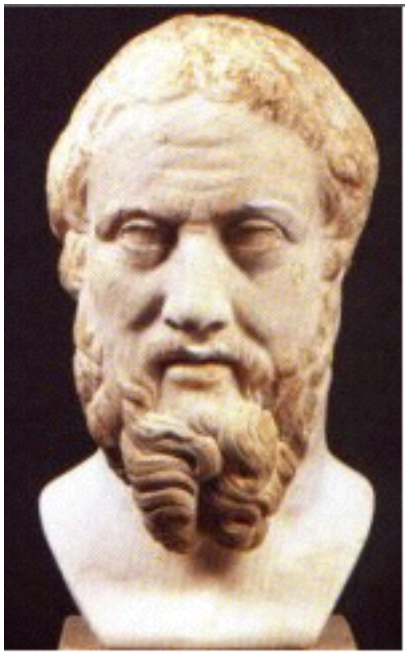
Figure 6.2 B

Dynamic prioritization, the human body method, is a pretty smart prioritization method, and keeps you alive in changing conditions.

We could do worse than use this dynamic and logical method for management planning.

"Knowledge may give weight, but accomplishments give lustre, and many more people see than weigh".

Herodotus, Historian, Greek, 484BC – 409 BC



Practical Tip 6.2 Good Value for Resources Used

- *avoid all forms of early fixed-priority weights, both numeric and words*
- *avoid 'deciding priorities' until the last responsible moment: when you have far more facts, to make a good decision*
- *keep your focus on attaining your critical Goals, within your continuously depleted resources.*

Policy 6.2 Prioritize Dynamically

We acknowledge that prioritization is complex and dynamic, but we will use reasonable planning effort, to make sure we get the best value for money.

We will not oversimplify, and thus get bad value, for too much cost.

Why?

- to avoid unintentionally delivering too much of *some* critical objectives.
- to avoid using limited resources in vain.
- to increase the probability of achieving critical goal levels on time.
- to respect the fact that we are the constant hostage, of many types of changes, in the factors that determine our best interests

Version 30 Aug. 2015 13:43 tsg pass 2

Section 6.3 Articulate your priority policy, and adapt it.

Principle 6.3 PrioritIes Policy

You need a clear prioritization policy, and it can change and vary.

If management fails to take a position on methods of prioritization, the evidence is that individuals will prioritize, somehow, and badly.

Quote 6.3 A

"As to methods, there may be a million and then some, but principles are few.

The man who grasps principles can successfully select his own methods. "

- *Ralph Waldo Emerson 1803-1882, Mass. USA*

Limits to Choice.
If you have multiple objectives, which we all have, you cannot simply prioritize strategies that favor a *single* objective.

If you have reached *one* objective's Goal, then you cannot favor strategies that *merely exceed* that Goal, while other objectives can *not reach* their concurrently deadlined Goal levels.

You cannot favor strategies that are far more expensive, or time-consuming than other comparable strategies, which would only serve to reach the *same* goal equally well.

You cannot favor strategies that have nasty negative side-effects, either because of your ignorance of them, or worse – with your full knowledge of them.

There is a lot more that is unwise to prioritize. But we constantly see management planning, that has not even thought through these ideas. And these bad practices are the *rule*, rather than the exception.

At the same time you might wonder why projects fail, or are late, or over budget. *Of course they are,* with such illogical prioritization as a basis!

Again, as stated in this book: we assume an environment of a steady stream of value delivery cycles. Nobody is expected to prioritize everything up front and at once. We need to deliver results early. At the same time as we need to learn the realities of value, side effects, and costs; and make adjustments to reflect those realities. There is no other sensible way in the complex, unfamiliar, unstable environment to which we are all subjected these days.

Here are **some alternative potential prioritization** policies:

Examples of alternative prioritization policies:

1. Best Value (1 objective) for Money
2. Best Values (several objectives) for costs (Capital, Maintenance, Support, etc.)
3. Best Long Term Goal Value for Capital Costs, and initial delivery time
4. Best Long term Goal Values for Long Term Costs (ie including maintenance and service, warranty, replacement in the life cycle)
5. Most Popular Values for selected stakeholders
6. Values for Resources, with regard to risks

Etc. There are a very large number of potential combinations

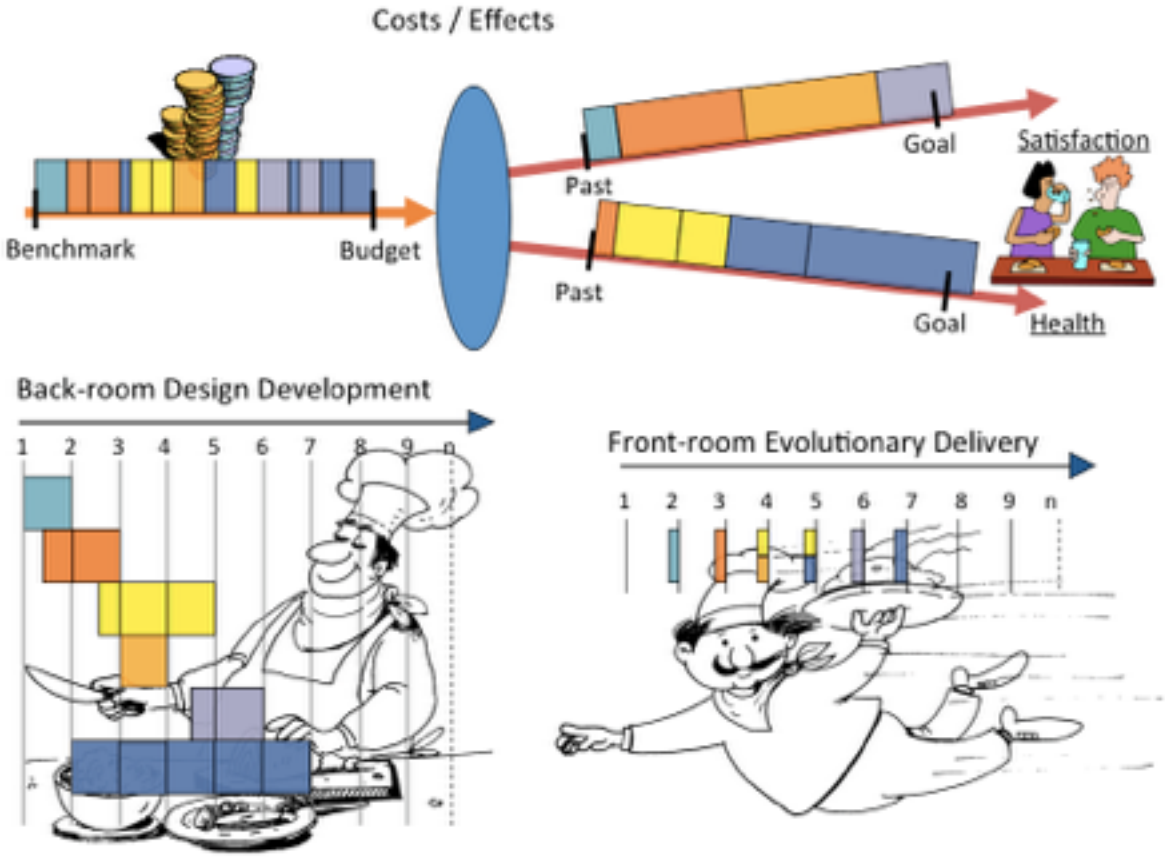
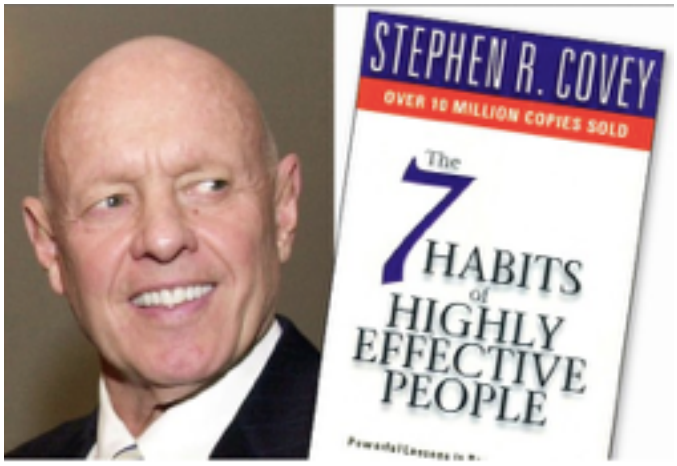


Figure 6.3 One visualization of the prioritization problem.
On the one hand, we are investing up front in the back room, consuming limited budget, and not immediately getting any value back. Is this a wise investment? A necessary evil? But we can track incremental value delivery from Past to Goal, and see the value build up. We need to figure out the lowest-cost set of sub-strategies to reach our Goal levels. Reality is of course at least ten times more complicated than this simple model.

Quote 6.3 B

"The bottom line is, when people are crystal clear about the most important priorities of the organization and team they work with and prioritize their work around those top priorities, not only are they many times more productive, they discover they have the time they need to have a whole life."

Stephen Covey (1932 - 2012), Utah, USA



Practical Tip 6.3 Results first, costs second.

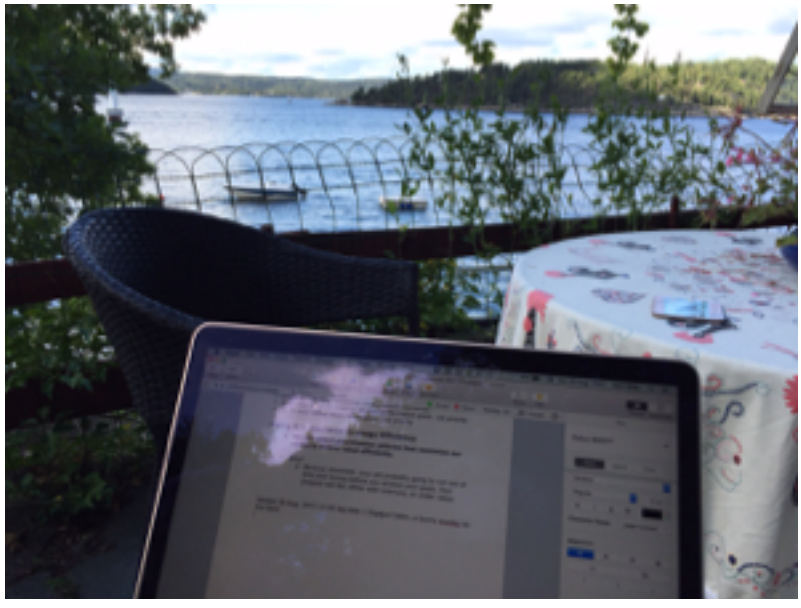
- *keep your focus on meeting all your performance goals: 1st priority*
- *try to keep within resource budgets: 2nd priority.*

Policy 6.3 Maximize Strategy Efficiency

- **We will select 'prioritization policies' that maximize our ability to deliver value efficiently.**
- **The default prioritization policy is do the 'best overall values for overall costs'.**

- Why?
  - Because otherwise, your are probably going to run out of time and money before you achieve your goals. Your projects will fail, either with overruns, or under value.

Version 31 Dec 2015 *minor edit.*
Version 30 Aug. 2015 14:00 tsg pass 1 Digerud Cabin, a Sunny Sunday by the Oslofjord. Nice 'office' to work in. :).
May I share it with you? Right photo. Left photo is from earlier visit by my colleague Nils Malotaux and his Shigemi. All 4 of us in the photo.



Section 6.4 Quantify *All* Your Critical Objectives As A Basis For Decision Making.

Principle 6.4 Bangs For Bucks: The Efficient Priority Principle

A good, general, prioritization policy
will deal with the *critical few* objectives,
the *most powerful* strategies,
and their *numeric relationships*
of values-to-resources.

It would be *nice* if prioritization could be done with respect to a *single* dimension of a strategy's effect on an objective, or on capital cost.

But this will not work, because then we are going to destroy other 'unseen' objectives, or misuse other 'unconsidered' resources, without even being conscious of our damage, until it is too late.

There are plenty of failed projects, and this is one reason they fail.

In short, we have to consider all critical aspects, simultaneously.

All, does not mean 'most', or 'all that are easy or convenient'.

'Consider all critical aspects' means that the ONE critical dimension of objectives, resources, or constraints, that you *fail* to plan for, and manage, is likely to be the 'weakest link' – the one that will be your downfall.

The definition of a *critical* factor is exactly *that*, **critical**; *it alone* can destroy everything, if it gets out of control. Critical factor management is not a luxury, not an option. It is a *necessity* for survival. Life critical.

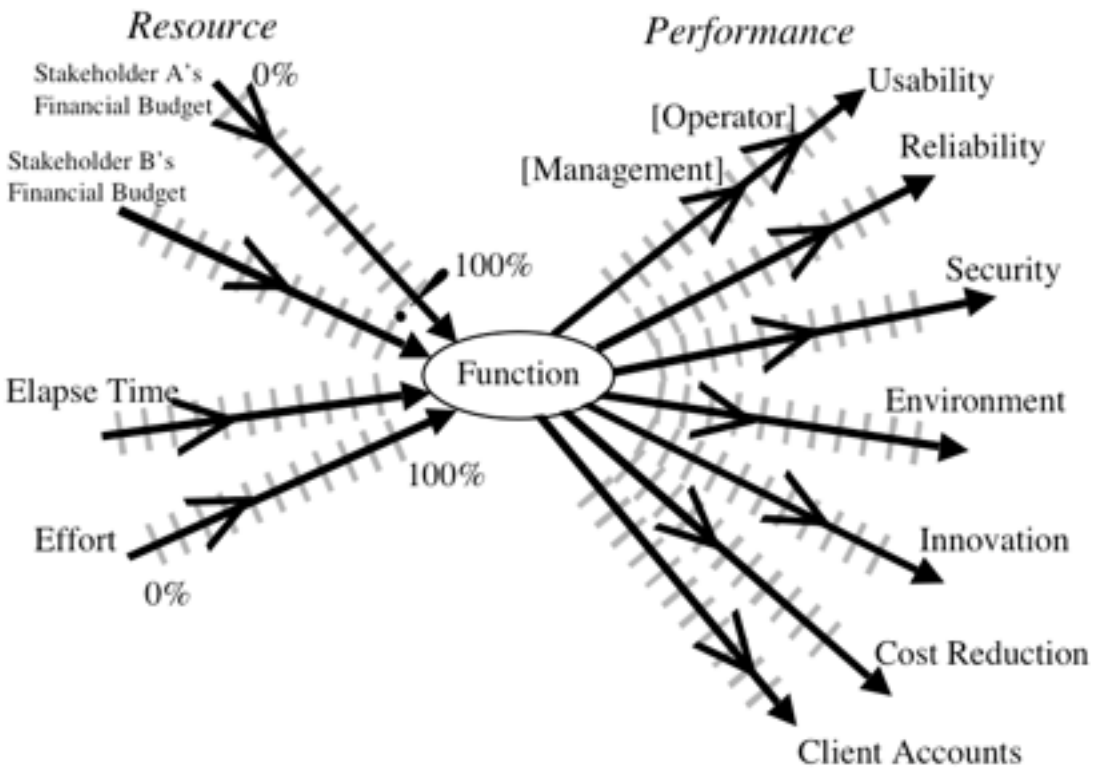


Figure 6.4 A symbolic representation of the multiple critical factors problem.

Comment on Fig. 6.4 Using Planguage icons [1, URL34]. Oval shape is 'function' (*what* we do), right-side out-arrows are variable dimensions of **system performance** (how well/good it functions), left-hand arrows represent scales of measure, for any variable, and limited, **resources** necessary to produce the performance levels. The Arrowheads on the arrow represent **Goal** levels (right-side levels), or **budgeted** levels (left-side arrows). The Value Planning process needs to try to achieve *all* the performance goal levels, without running out of budgeted resources. The **'!'** level, represents a constraint level. A 'Fail' level, not OK, even though the budget level has not been formally reached. The '!' is a pain, or danger, signal.

So here is your **checklist of what you have to do, in order to plan as well as you can,** to *prevent a critical factor from hurting you*.

If you have been reading the book thus far, this will come as no surprise, and be a kind of summary:

- **Define targets** (Goals, Stretch) and **constraints** (Tolerable, OK) for your top 10, or so, critical performance objectives
- **Lesser priority objectives** (number 11 to infinity) are not immediately under control. So they are *at risk*. But, first-priority objectives, are *first*. We hope the second wave of priority objectives can await the accomplishment of our first wave.
- The **critical resources** need to be budgeted. These are usually about 2 to 6 types, but can be any number. You need one-or-more Budget levels (the official plan for time, money etc. in time and space [Qualifiers]), and perhaps you also need some *constraints* information (OK, Tolerable), to warn you when the 'official budget' is overrun, and to help decide how much tolerance of Budget overruns is acceptable.
- A set of **strategies** needs to be identified, and thus 'prioritized'. Top 10 most powerful strategies is a good start, which we *initially* hope and believe will be sufficient to meet our performance goals, within our resource limitations. And, probably they should be decomposed into implementable sub-strategies [2.7].
- Initial **estimates** should be made, of the expected effects of **each strategy**, or sub-strategy, on **all critical performance attributes**, and on **all critical resource attributes**. For this process, you are going to need the tool we have introduced earlier, and will discuss later, the **Impact Estimation Table** [5.3]. This '*strategy impact'* estimation should give confidence, and insight, that you know how (*which* strategies) to reach your objectives, within your limited resources.
- There will also be a set of **overall constraints** (in *addition* to the 'scalar constraints', such as Tolerable, OK, mentioned above) which apply to the strategies. These can be legal, cultural, corporate policy, standards, related contractual obligations. These need to be *formally* listed, and used to *initially validate* all suggested strategies. If an *otherwise powerful* strategy is *illegal,* in the European Union for example, you cannot plan to use it *there* (as Microsoft (bundling) and Google (Privacy) learned.
- If you have the energy and time initially, you would *also* begin the process of acknowledging **elements of risk, and uncertainty.** A subject we will treat in more detail later in this book (Chapter 7). But we have, with the above tasks, made a good start in managing risks.

Now, you have the requirements for a thorough plan. As you realize by now, that is just the beginning. You have done the same work as in the first 3 days of the Initial startup planning week [URL5].

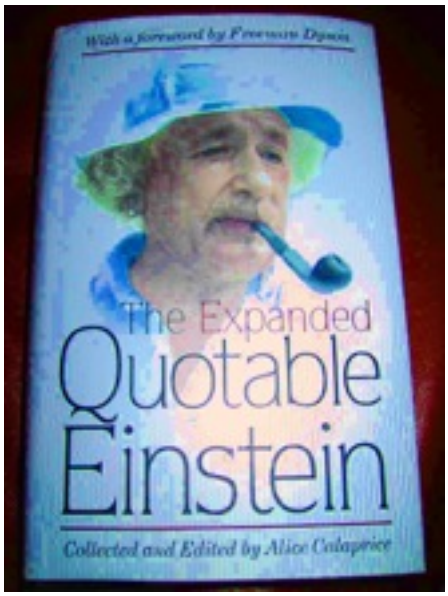
But establishing these critical top level objectives, strategies, and estimations *is* **our first priority**, in the first 'big bang' moments of a project.

Now you are ready to try out your ideas early, frequently, continuously; until 'goals are achieved', or 'when resources are depleted'; this is the Evo Project management process [URL4 B, 1 C].

"One should not pursue goals that are easily achieved.

One must develop an instinct for what one can just barely achieve through one's greatest efforts."

Albert Einstein (1915)



[24] Quotable Einstein, Ed. A. Calaprice.

Practical Tip 6.4

- *Quantify your critical objectives.*
- *Use them to dynamically decide your priorities.*
  - *especially when they are changed.*

Policy 6.4 Prioritize Critical Factors

- **We will quickly pin down the top-priority objectives, and probable strategies, quantitatively.**
  - **The values-to-costs ratio of strategies should be reasonable.**

- Why?
  - This will give a healthy framework, and overall view, for the detailed process, of both planning and value delivery, to follow.

Version 30 Aug. 2015 15:50 pass 2 tsg on Oslofjord

Section 6.5 Value Management Information for Prioritization.

Principles 6.5 Data Driven Decisions

A Value Decision table
gives you a systematic overview
of the many related facts
about your potential priorities.

If you are going to be able to get, a complete overview, of even your top-level objectives and top-level strategies, we recommend an Impact Estimation table (IET). [URL21]

The IET is also called a *Value Decision* Table, when you use the basic IET tool in the mode to *make decisions*, based on both initial estimates, and perhaps on later feedback from value delivery.

All other known or similar methods fail, as far as we can see, to show rigor in definition of the objectives. Methods like Balanced Scorecard, and Quality Function Deployment, for example, demand no *clear quantified objectives* at all ('Reliable' is acceptable), and become useless for clear-thinking purposes, based on that criticism alone (there is much more, see references) [URL35 QFD, URL75 Brodie PhD, URL62 BSC].

Here is a summary of the discipline of Value Decision-Making using IE Tables:

We estimate the expected impact of a strategy, on a Goal in a **single table cell** of their intersection. We can initially put such estimates in the **Strategy** specification itself, and later move them to a table. This transfer is automatic in Planguage tools [URL73]

- The initial estimate uses the Scale defined for the objective to define the meaning of numbers on the scale. We call this the **Real Scale**. For example, if you estimate the strategy would save 10 minutes, we specify that directly as '+10 minutes'.
- We can, *optionally,* estimate the range, or the **uncertainty** of the result. For example 10**±2** minutes (or 10: +10, -5).

- We usually *convert* the Real Scale estimate into a '**% of the way** towards 100% of the Goal'. This assumes we start from some Past level, a baseline called the 0% point. So if the 10 minutes real scale impact, was one half of the way to the Goal (say Past = 60 minute, Goal = 40 minutes, 100% (60-40) = 20 minutes) we would specify a 50% impact for the 10 minute impact. The % estimate is useful for several purposes, such as adding up impacts from different scales of measure, in a 'common currency'.
- Next, we need to document a *basis* for the estimate, even if that basis is an acknowledged wild guess. We call this basis, **'evidence'.** Same concept as legal evidence, or experimental evidence. The factual basis for the estimate.
- It is useful to know the **source** of the evidence, so we can decide if that source is credible, and can review, question, or audit the source.
- Finally, we, optionally, calculate an estimate-**credibility** level, based on the *quality* of the evidence, *and* of the source. 0.0 = no credibility, 1.0 = perfect credibility, in 0.1 increments [URL35 for detail, Table 9.5 The Credibility rating]
    - This forces us to 'be aware' if our decisions are 'shaky' (0.0 to 0.3), or are quite 'solid' (0.8 to 1.0). Credibility ratings *motivate* us to find better strategies: strategies with better evidence and sources. Overall average Credibility ratings (example 0.6 average of 100 ratings in a 10x10 table) allows us to claim a degree of 'credibility' when we present IET evaluations to other people.

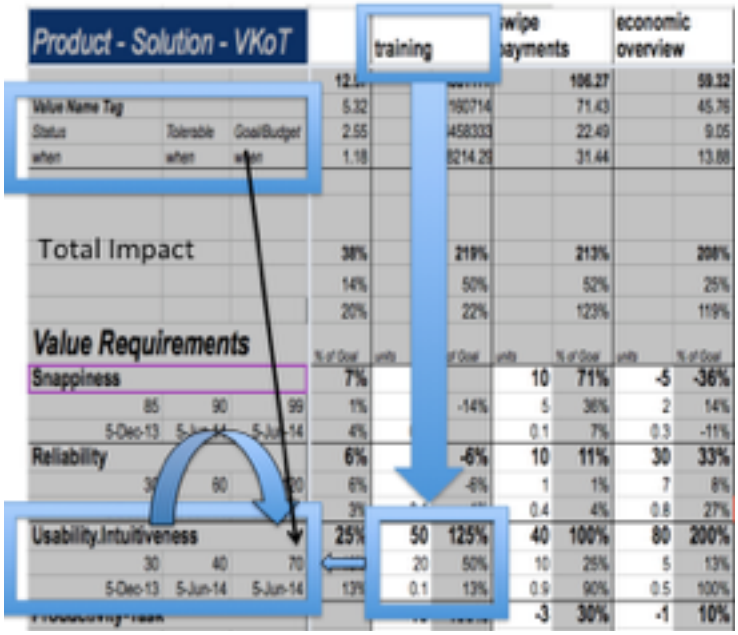


Figure 6.5 A . A Value Decision Table from a group class-exercise, at a Nordic Bank, using Kai Gilb's IET Tool [URL68].

*In the 'Training -> Usability.Intuitiveness' intersection, you can see the 'real value' (**50** units on a defined scale), the % of the way to the Goal (= **125%**, when Past = 30, Goal = 70). The ± uncertainty is '30', so the lowest level is 50-30 = **20** minutes saved, or only a **50%** impact. The Credibility is stipulated at **0.1** (very low), so the original 125% is reduced (0.1 x 125, rounded) to **13%**. A pessimistic estimate adjustment.*

Here is a summary of what we can do, with the set of estimates in each cell, and all cells of a table:

The overall picture for all Objectives and strategies, or any selection of them, is derived this way.

- We can vertically add the **% impact of a single strategy**, on any *set of objectives* (like our top 10) we choose. This sum gives us a measure of the **overall impact** of a *single* strategy, on an entire set of goals, like your top 10. See Fig. 6.5 B.
- We can add up the horizontal **% sum of any set of strategies**, on a *single Goal* for an Objective. This gives us a measure of 'if we have enough strategies to possibly reach the Goal level'. If the sum is, for example only 10%, we realistically cannot hope to reach the Goal. If the sum is 200%, we possibly can reach the Goal, with a **safety factor** of two. Rough, but useful, feedback. See Fig. 6.5 B.
- We can ***combine*** the corresponding **± uncertainties** with the estimates, to get a rough idea of the **riskiness** of the strategy, or *riskiness of our hopes* to meet a particular goal.
- We can **multiply the estimates by the Credibility** factors (range of 0.0 to 1.0, Table 9.5 The Credibility rating) to reduce shaky estimates to a more modest size.
- We can include **estimates** for the **resource costs** of each **strategy**, and thus get some idea of the '**efficiency'** of the strategy. Strategy Efficiency is the 'values for resources' ratio. This is also called 'Cost-effectiveness' of a strategy.
- We can also link impact estimation tables **'hierarchically'** to model the related stakeholder levels. Typically, these levels are the *organization*, the *stakeholders*, and the *process or product* supporting the stakeholders [URL36, 'Bring' case, with hierarchical tables]. See Fig. 6.5 C below.

US DoD. Persinscom **Impact EstimationTable**:

Strategies

Objectives	*Technology Investment*	*Business Practices*	*People*	*Empowerment*	*Principles of IMA Management*	*Business Process Re-engineering*	*Sum Requirements*
Customer Service ? <-> 0 Violation of agreement	50%	1	5%	5%	5%	60%	185%
Availability 90% <-> 99.5% Up time	50%		5–10%	0%	0%	200%	265%
Usability 200 <-> 60 Requests by Users	50%	5–10%	5–10%	50%	0%	10%	130%
Responsiveness 70% <-> ECP's on time	50%	10%	90%	25%	5%	50%	180%
Productivity 3:1 Return on Investment	45% 50%	60% 5%					303% 251%
Morale 72 <-> 60 per month on Sick Leave							
Data Integrity 88% <-> 97% Data Error %	42%	10%					177%
Technology Adaptability 75% Adapt Technology	5%	30%					160%
Requirement Adaptability ? <-> 2.6% Adapt to Change	80%	20%	60%	75%	20%	5%	260%
Resource Adaptability 2.1M <-> ? Resource Change	10%	80%	5%	50%	50%	75%	270%
Cost Reduction FADS <-> 30% Total Funding	50%	40%	10%	40%	50%	50%	240%
Sum of Performance	*482%*	*280%*	*305%*	*390%*	*315%*	*649%*	
Money % of total budget	15%	4%	3%	4%	6%	4%	36%
Time % total work months/year	15%	15%	20%	10%	20%	18%	98%
Sum of Costs	*30*	*19*	*23*	*14*	*26*	*22*	
Performance to Cost Ratio	*16:1*	*14:7*	*13:3*	*27:9*	*12:1*	29.5 :1	

Estimated Impact of Strategies -> Goal Levels, in %

Figure 6.5 B. A real IET used as a Value Decision Table [URL16 C, USA DoD Case].

This table is presented without underlying details, which are the basis for the % estimates. Which strategy gives best value for resources? Notice the vertical sum of % impacts (Sum of Performance row). Notice the horizontal sum of impacts (Sum Requirements) with a safety factor of about 2.

Here is a summary of *applications* of Value Decision-making:

- To force yourself to **think clearly** about both your strategy ideas, and the clarity of your goals.
- To enable a team to discuss their various inputs (strategy ideas) and their evaluations of those ideas, in a **very systematic way**; making sure they are *aligned with the strategic objectives.*
- To facilitate **presentation** of strategy proposals to management, and steering committees, or even investors in startups.
- To enable **independent audit** or **quality control** of strategy proposals.
- To **track progress** towards goals, at each value delivery cycle, [URL28 C & D] during project management.
- To enable logical fact-based selection of competing mutually-exclusive alternatives.
- To collect **feedback** about *real versus estimated* impacts, during a value delivery cycle, and **adjust**; using Dynamic Design to Cost, or by adjusting expectations, or budgets and deadlines. [URL11, URL12].
- They can be the 'accounting system' for managing your contracted suppliers [URL6].

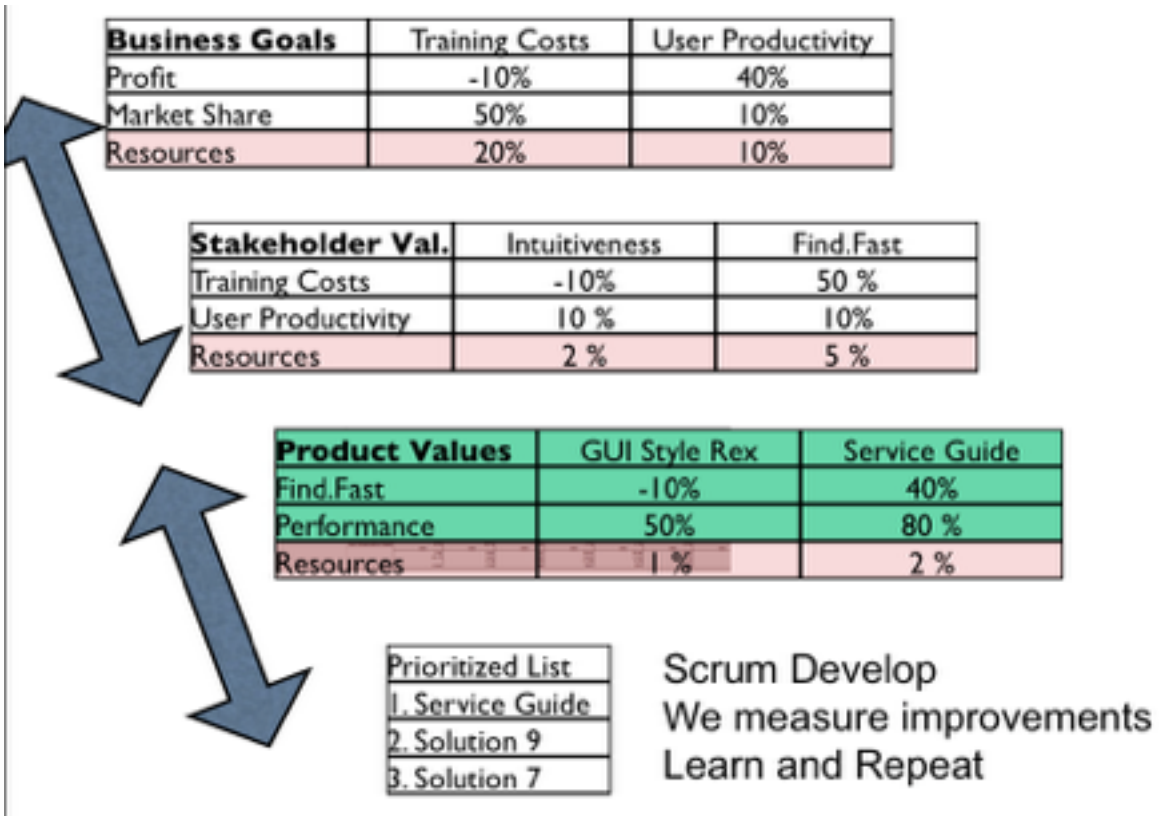


Figure 6.5 C. Simplified hierarchical Value Decision Tables. Source 'Bring' Case, Kai Gilb [URL36]. The arrows indicate feedback in both directions.

Figure 6.5 D Impact Estimation is a tool which combines Planguage for a variety of purposes, such as Risk-Based Value-Based Decision-Making. Source: [URL30, Tse & Kahlon]

"No matter how complex the situation, good systems engineering involves putting value measurements on the important parameters of desired goals and performance of pertinent data, and of the specifications of the people and equipment and other components of the system.

It is not easy to do this and so, very often, we are inclined to assume that it is not possible to do it to advantage.

But skilled systems engineers can change evaluations and comparisons of alternative approaches from purely speculative to highly meaningful.

If some critical aspect is not known, the systems experts seek to make it known. They go dig up the facts.

If doing so is very tough, such as setting down the public's degree of acceptance among various candidate solutions, then perhaps the public can be polled.

If that is not practical for the specific issue, then at least an attempt can be made to judge the impact of being wrong in assuming the public preference.

Everything that is clear is used with clarity: what is not clear is used with clarity as to the estimates and assumptions made, with the possible negative consequences of the assumptions weighed and integrated.

We do not have to work in the dark, now that we have professional systems analysis."

Simon Ramo [12, URL38], Founder TRW Systems.(May 7 1913- , 102 in 2015!).
He built a very successful business on his ideas: which are exactly the ones we are writing about in this book. I used to visit Barry Boehm, Chief Scientist at TRW, and share ideas, like Impact Estimation Tables, with his group.

Practical Tip 6.5 Simplify Value Decision Making

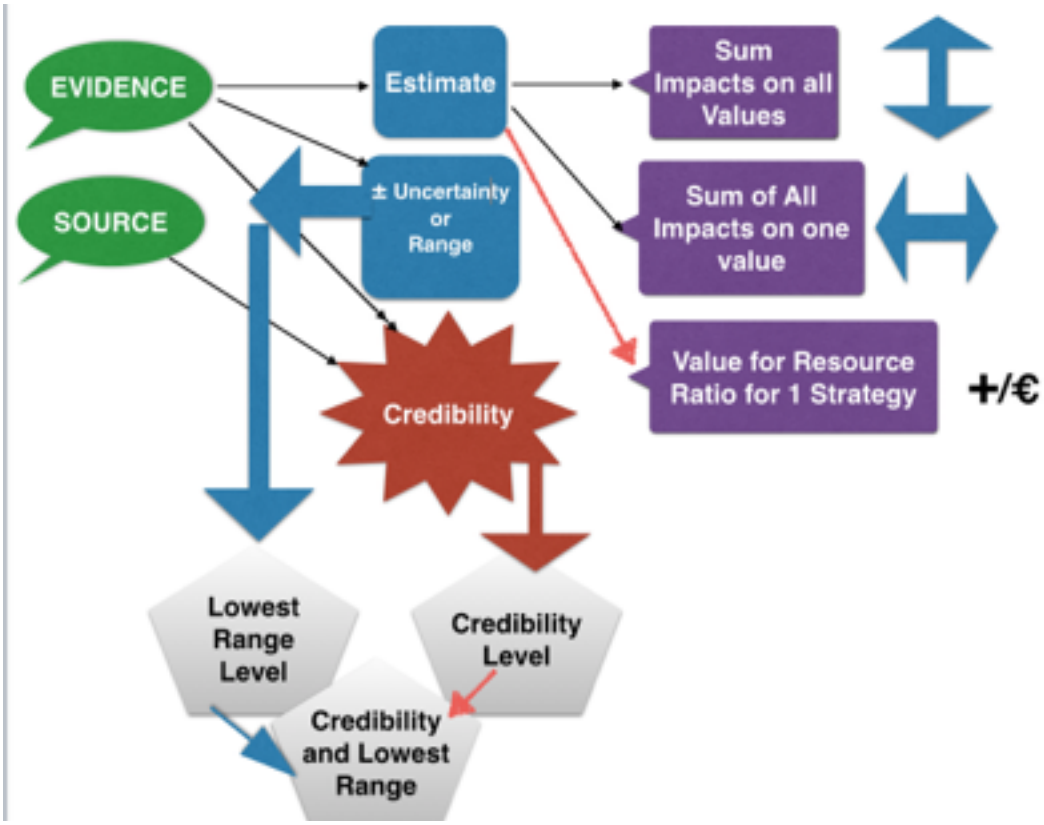
- *Even a very simplified Value Decision Table forces us to think in several critical dimensions*
- *Even the most detailed Value Decision Table can be presented in simplified ways: like a bar chart diagram, showing two or more competing choices.*
- *You can 'drill down' in the tables, and in related specifications of objectives and strategies, for detail to gain confidence, or to analyze the conclusions.*
- *Value Decision tables allow us to be as clear as possible about our knowledge, yet aware of risks and uncertainties.*
- *Impact estimates are not trying to correctly predict the future. They are trying to give us a better idea of the promising efficient strategies that we can confirm or deny by early trials.*

Policy 6.5 Articulate the ends-means relationship, quantitatively

- ***We will articulate our strategies, and their impacts on our critical objectives, using applicable tools, from the Value Decision Table.***

- Why?
  - To force *ourselves* to think clearly about complex systems
  - To improve our ability to communicate complicated ideas with other people,
    - so they can review them,
    - approve them responsibly,
    - react constructively to them, and
    - work as a team towards well-understood purposes.

Version 30 Aug. 2015 pass 3 18:31



Section 6.6 How to Prioritize in Both Long and

Short Term

Principle 6.6 Long And Short Term Value

**An impact estimation table
will help you prioritize
in the short term
and long term**

A single objective, can have Goal statements *simultaneously* for the long term and the short term.

In fact we can plan any time-series that interests us; we can draw a curve of our planned improvement of any objective.

This includes the fact that we can, using Qualifiers, also plan specific dimensions of interest at those times, such as Employee types, and Markets.

Productivity:
Scale: Average Time for Average Salesperson to Make Sales Activity Report, Daily

Past: 60 minutes

D1: Goal [End this Year, New Salespersons] 30 minutes.
D2: Goal [Within 3 Years, Top Salespersons] 15 minutes.
D3: Goal [Within 5 Years, All Salesforce] < 5 minutes.

Planguage 6.6 A. Short and Long Term Goals.

Here is a simple example of short term and longer term Goals for a single objective (**Productivity**).

The tags D1 etc. are used to cross reference the Goal elsewhere, such as in an Impact Estimation Table (as below).

Notice the different categories of *people* that the target applies to, as we move through time.

We can have any interesting number of space dimensions simultaneously (example [New Salespersons, China, Under 30]) and independently of the time dimension.

	Strategy1	**S2**	**S3**	**Sum -> Goal**
Deadline1 "Short"	**10%**	**20%**	**70%**	**100%**
D2 "Medium"	**15%**	**0%**	**30%**	**45%**
D3 "Long"	**50%**	**-30%**	**80%**	**100%**

Planguage 6.6 B. Strategy impacts on short, medium and long term. On an Impact Estimation Table. (IET)

In this simplified IET, we can visualize the planned impact we expect to get for the 3 strategies, in the different time frames.

We can see that the D2 Goal, lacks sufficient Strategy effectiveness to reach 100% (15%+0%+30% = 45% only). Medium term has bad strategy coverage.

So we need some more-effective strategy in the plan, either a S4, or a redefined Strategy1, or S2, or S3, so that they are more-effective for achieving D2.

Which strategy would you use first?

P.S. don't you think just scraping in at exactly 100% for D1 and D3 is a bit nerve-racking ?

What can go wrong?

Here are some reasons this multi-time-range IET planning capability is useful:

- It effectively *decomposes* the deployment of a strategy *[URL16]*, so that we can identify, and focus on delivery of *high value*, do-able, things first. We need not wait for the later things to be deployed
- It is *easier to estimate* [URL11] smaller and more-specific increments at one time: then *learn by deployment* what the performance effects are, and what the costs *really* are. Then we can apply this feedback to the *future* steps, to get better estimates for them, or learn to redesign the current strategies to be more cost-effective (efficient) in future deployment.
- When conditions radically change, we are in a *better position to insert new strategies*, as they become available, or target new, and different, categories of stakeholder.
- It enables us to synchronize strategy investment and deployment, with external considerations such as management change, budgeting cycles, product launches. *Alignment !*

I don't think that you can invent on behalf of customers unless you're willing to think long-term, because a lot of invention doesn't work.

If you're going to invent, it means you're going to experiment, and if you're going to experiment, you're going to fail, and if you're going to fail, you have to think long term.

Jeff Bezos, Amazon (1964-)

Practical Tip 6.6 Try a short and long term Impact Estimation Table

- *consider the option of planning both a long term and a short term Impact Table: to avoid just thinking short term*

Policy 6.6 Short term planning feeds forward to long term plans.

- **Clearly plan both short-term and longer-term results, for your objectives.**

- Why?
  - Whatever feedback and learning you get, from the short term, will be helpful adjustments for the long-term goals and strategies.

Version 30 Aug. 2015 22:11 pass 1

Section 6.7 Using the IE Table for Measurement and Feedback: The Meter.

Principle 6.7 Estimates Are A Good Start, Measurements Are Better.

**An impact estimation table
can help keep track of
real feedback
and incremental progress.**

Estimation is better than not.
An *Impact Estimation table*, as the name implies, is *initially* a place to *estimate the effects* of strategies, on critical attributes.

But the accuracy of such estimates is only rough. Sometimes we would be happy to get the right order-of-magnitude. The same, of course, holds true for *any* estimation or forecasting process.

However, rough estimates are better than none. Especially if we are conscious of their roughness, and we have many tools to do that in Planguage. Explained in this book.

And rough estimates are useful, as long as they generally guide us better, to finding reasonable actions in the short term.

At one extreme we could say that the primary purpose of estimates is to tell us what *might* be worth doing and measuring.

But measurement should evaluate estimations.
However, once having chosen a sub-strategy, to deploy, in a current *value delivery cycle*, we need to measure what *really* happens, learn from it, and take appropriate action.

The Impact Estimation Table (IET) starts project life as a 'budgeting system'.

But this rapidly (within a week, in practice) is combined with corresponding short-term measurement, of the very things we just estimated. So we can rapidly, before unnecessary damage is done, *reverse* badly-estimated implementations.

We might even *claim success* in discovering a bad risk, before it could do damage to the larger project! In fact rapid feedback, week by week, is a great *risk management* method.

Here are some problems of short-term *measurement itself.*

- You might not yet be able to measure:
  - longer-term effects,
  - effects of scaling up later,
  - effects of later changes and strategy implementations,
  - or stability of the system, under volume or pressures.
- You might not find it worthwhile to measure, as *accurately* as you'd like. You might have to 'make do' with sampling, and *not quite realistic* test subjects, under *not quite realistic* test environments
- But, difficulty in *ideal* measurement, should never stop us from doing *worthwhile* measurement, especially in the short term.
- There is a simple long-term guideline: **the cost of measurement should not exceed any benefits from it**.
- You will have difficulty in the short term, in measuring the knock-on effects of your strategy, especially at the next level up (your Fundamental Objectives).
- You will have to schedule the longer-term measurement for the longer term, and make-do, as well as you can, with 'leading indicators'.

CASE 6.7: The Experienced Professionals at Confirmit.

One of our clients, Confirmit [URL28, Source: Trond Johansen] reported, that when individuals had done this discipline of weekly numeric feedback, compared to their weekly estimations, they, in the longer term, became so skilled at high-quality estimates, that they chose to dispense with making initial estimates, and the corresponding short-term measurements: but they did still take the trouble to measure the cumulative effects of 12 weekly increments, before quarterly release of their product to their customers.

In short, they really knew what would work and how well.

That was good enough.

This is probably true in many disciplines, we think of sports first.

After the famous 10,000 hours of experience (Malcolm Gladwell, "Outliers" book, Violin Practice), you attain a Master's Level, and can successfully do intuitively, what was a painful, step by step, coached learning process initially.

In the Confirmit case the original team that learned our methods and applied them from 2003, stayed with the company, and grew to leadership positions in the expanding company.

The danger is when these people are replaced with people *without* such practice and knowhow.

Then, you probably need to go *back to basics*, at least under changing conditions of economics and technology.

You need to get the new people to estimate and get rapid feed back, until they reach the understanding of an experienced master.

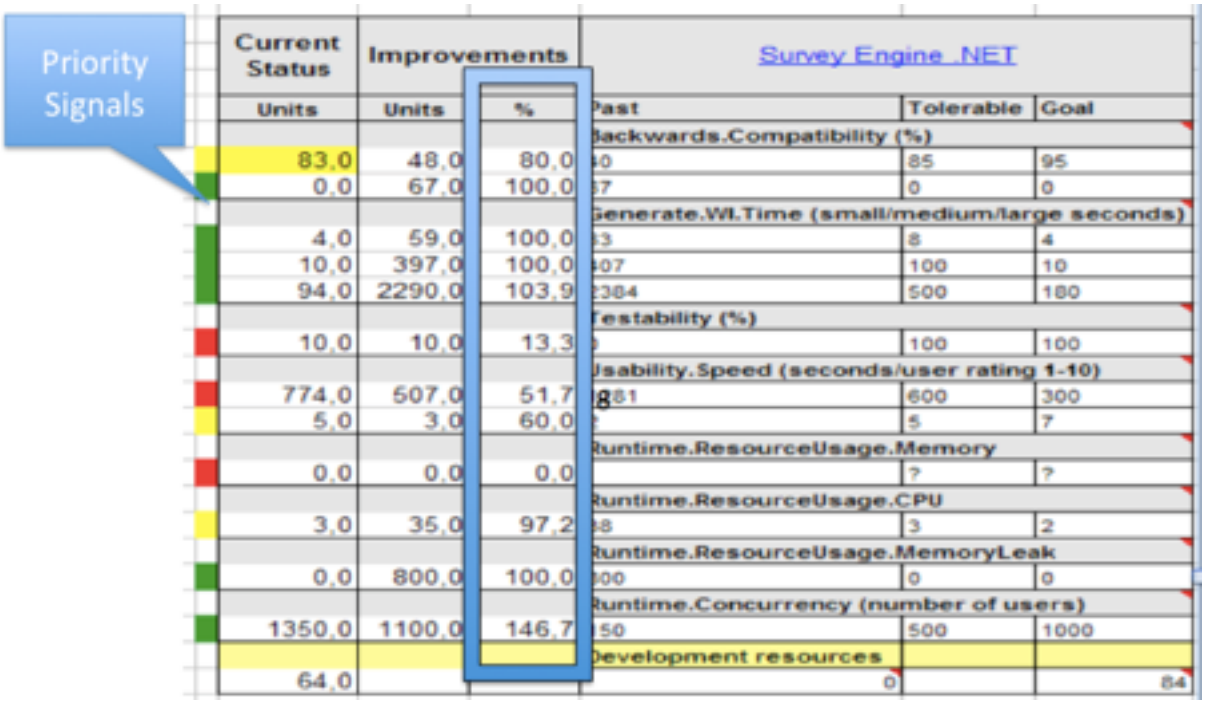


Figure 6.7 . Incremental Value Tracking at Confirmit.

Case Study 6.7 Confirmit, Tracking Incremental Value Delivery Against Goals.

Comment on Figure 6.7
This chart is from one of 4 parallel 4-person teams at Confirmit, in their first quarter, of 12 weekly value delivery cycles.

The % column shows the cumulative measured progress towards Goal levels, after 9 of 12 weeks, before Quarterly release of product, 3 weeks later.

They are clearly on track to about 100%+ in all objectives.

Especially if they prioritize 'designing to Goal', for the objectives that are less than 100%, in their remaining 3 weeks (10 to 12) of value delivery.

They are clearly good enough at estimating, and delivering what they estimate.

The color priority signals at left are computed:
Red = not Tolerable,
Yellow = Tolerable but not at Goal,
Green = 100% of better, no priority.

You can always quantify and always measure critical values.
There is a principle, discussed earlier (Section 1.1) that we can invariably find some useful Scale of Measure (or set of them) for our critical objectives.

There is a follow-on principle, that, for any given defined Scale of Measure, you can find one or more ways of measuring along that scale, as opposed to 'none'.

In fact you might need 2 or more ways of measuring in practice, for different time and conditions.

Example; short term during development, then another way for product handover, and perhaps a third way at customer site or remote help facilities.

Meter Attributes.
What is the difference between the various specifications of 'Meters' (the defined measuring processes for an objective or a resource) that you might make?

You can certainly guess that *some* Meters have higher quality levels.

Qualities like accuracy, reliability, predictability, ease of learning, ease of doing, cultural acceptance, prior familiarity; in short they have sets of *performance (including quality)* attributes, and sets of *cost* attributes.

Meters need good enough quality at lowest cost.
When we select a measuring process, we are generally looking for satisfactory measurement qualities, at the lowest costs.

So, in fact, if we defined *our objectives* for the measurement, we could evaluate all measurement candidates using an Impact Estimation Table, and pick the best one!

Meter selection is in fact a multivariable decision. It is technical: an engineering decision.

But normally people will not find it pays off, to get *so* formal about the Meter decision.

We just wanted to point out a possible *formal* structure of our usually informal (intuitive, using custom) decisions, about measurement processes [URL37, How Good is a Process?].

Productivity:
Scale: Average Time for Average Salesperson to Make Sales Activity Report, Daily

Past: 60 minutes.

D1: Goal [End this Year, New Salespersons] 30 minutes.
Meter: Stopwatch by Trainer.

D2: Goal [Within 3 Years, Top Salespersons] 15 minutes.
Meter: Self Timing reports.

D3: Goal [Within 5 Years, All Salesforce] < 5 minutes.
Meter [After App is used by everybody] Automated measurement in the reporting app.

Planguage 6.7: here is a simple example of planning a set of Meters. Meters are a process for measuring in practice, along a single defined Scale of measure. The Meter statement is usually a rough outline only. The detailed 'test' planning to be done by others, such as system testers.

But the **Meter statement,** a rough outline, has **several purposes**:

- To draft, at the *planning* level, what *scope* of measurement, would be roughly able, to give satisfactory feedback for purpose, at a cost that seems reasonable.
- To declare that we fully intend to measure results, in practice
- To remind us to build the *costs of measurement* into the cost estimates for the strategies.
- To trigger a discussion, in meetings and reviews; about whether the Meters (as informal as they might be, in initially-specified detail) would be *acceptable*, with respect to accuracy/credibility, costs, and cultural/legal acceptance.

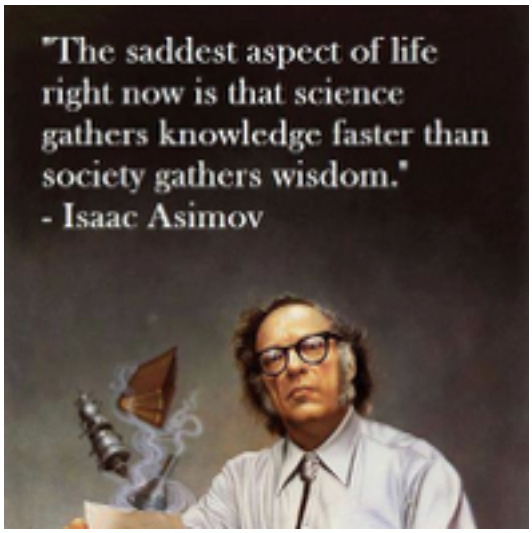
"I believe in evidence.

I believe in observation, measurement, and reasoning, confirmed by independent observers.

I'll believe anything, no matter how wild and ridiculous, if there is evidence for it.

The wilder and more ridiculous something is, however, the firmer and more solid the evidence will have to be."

Isaac Asimov (2 Jan 1920 - 6 Apr 1992).



Practical Tip 6.7 The IE Table as your Planning Accounting Method.

- *use a spreadsheet or Impact Table tool [URL73] to keep track of real feedback or measurements, compared to your estimates*

Policy 6.7 Measurement Confirms Our Ignorance: hopefully early.

- **We will commit to comparison of 'measured reality' to our 'original estimates', so we can responsibly change bad ideas into better ones.**

- Why?
  - To get early warning of bad ideas, that need changing
  - To avoid endless meeting discussions and opinions, and settle matter by 'experimentation'; using an Evo value delivery step, and measurement as the decisive argument.

Version 31 Aug. 2015 22:04 pass 3

Section 6.8 Our Priority Decisions Can be Computed

Principle 6.8 Priority Logic.

Priority can be computed
by an Impact Estimation Table spread sheet

Most people, you and us included, normally have an *intuitive* approach to prioritization of daily choices. That's natural, and serves us well.

But, in complex, multi-person, multi-team, large-scale, long-term decision-making, what is *good enough* for the *individual*, is *too simple* for the larger situation.

Mind you, **nature's principle** of *'dynamic prioritization'*, which is based on *many* factors, and on the degree of fulfillment of our critical needs, which we all use daily, is an excellent general prioritization model. It just needs to be *enriched and formalized,* to cope with the fact that we must deal with more people, more factors, and a fast-moving environment (technology, economy, politics).

Here are the basic conditions for *computing* our priorities, for making prioritization an objective logical act:

- The **critical objectives**, that we must use, to evaluate the *strategies* we might prioritize, are **identified**.
    - If one really-*critical* objective goes **un**identified, the *entire* prioritization decision is potentially *invalid*. That missing objective could threaten the entire system/organization/ product/service.
- The identified critical *objectives* are then **defined** with **quantified goals** (success, total satisfaction), and **quantified constraints** (minimums for survival, and for being OK).
- The scales-of-measure chosen are **appropriate** for reflecting that which really is critical.
    - we don't choose 'easy to quantify and measure', we choose to measure things which are really decisive: even when they are difficult to measure.
- The critical and limited ***resources*** are, similarly, identified, and relevantly quantified, with both Budgets (mirror image of Goal) and Constraints (OK and Intolerable budget excesses)

- Any **external constraints**, which might invalidate choice of a strategy, need to be identified, and clearly stated, so that they are easy to use as a manual checklist.
    - We do not see, at present, the possibility of automation of 'checking that external constraints do not invalidate a strategy'. That is as tricky as determining a court case; because some of these external constraints are exactly about the legality of a strategy. And these are regularly tried in the courts.
    - So, we need some kind of filter, like a human manual evaluation, to see that any winning strategies, are probably not in violation of the external constraints (legal, cultural, corporate policy, ethics, and their like).

- Next, we need, for all candidate strategies, ***estimates*** (as in Impact *Estimation*) of the *immediate* (next value-delivery step, on top of the current existing system) *effect* of the strategy being implemented.
    - For this we need a sufficiently clear and detailed description of the strategy.
        - We need a *foolproof* strategy description. So that the estimator cannot fail to understand and consider, how the strategies *relate* to objectives and resource constraints.
        - And, the implementer cannot fail to implement *exactly* what the estimator has considered.
    - If there is ambiguity, or lack of clarity, on points affecting the impacts of the strategy, then either the estimates, and thus the choice (the prioritization) will be wrong: or the implementation will not be what was estimated, and the results will disappoint us.
        - *Lack of clarity, and too much ambiguity, are big problems that we find when reading 'strategies'.*
        - *Our many, varied, and long-term measurements, of management and technical plans, show that about 30% of the words used, are either unclear or ambiguous [URL23].*
        - *This is enough to invalidate any method of prioritization; to make it worthless.*

Computing a valid strategy.
At this point, if the above conditions, regarding us 'clearly describing the problem', and 'collecting estimates on the impacts of the strategy candidates' are met, we are ready to compute a **valid strategy**.

A valid strategy that does not violate any known external constraint; like, 'not *illegal'.*

And a strategy that best fits the remaining gaps to Goals, for the objectives; and can make use of the remaining resources.

All of this is a computation for *this current value delivery step only*.

Unfortunately we cannot draw valid conclusions for these strategies, with regard to their effects on *later* steps, because the conditions will necessarily have changed.

That is, we will have fewer resources remaining, and we will have smaller gaps to fill, towards the Goal levels of the objectives.

Prioritization Strategies.
We also have to decide on our prioritization strategy; meaning exactly *how* we will choose to prioritize, given the information we have at hand.

In this we have a large number of choices, which will depend on politics, personalities, organizational culture. Tricky, yes, but we can at least be clear, open, and conscious with ourselves about these choices.

There is one 'general prioritization policy', which by default is recommended instead of others. We will focus on that one below, but it can be necessary, and desirable, to deviate from that general policy, from time to time.

The *general policy* is that
"we should aim to reach *all of the goals of our critical objectives*, with the *smallest consumption of limited critical resources*."

Analyzing strategy alternatives.
A simple tactic is to compute which strategy gives the overall greatest impact on the critical objectives, in terms of moving them to the yet unreached Goal levels.

- If one strategy clearly scores highest impacts
    - indicated by a simple vertical addition, in the strategy column, of the % impacts on a set of Goals, using the IET is sufficient. See Table 6.8 below, 'Cumulative Total for Performance Requirements',
    - *and* if the set of % impacts on *resources* is smaller than any other; we have a clear winner.

- This means that strategy gives best **overall value, for overall use of the critical set of resources** (in simple terms both 'time and money') **remaining**.
- At this point we have a 'hypothesis' that this is the smartest strategy to implement, in the next value-delivery cycle.
- We experiment, by implementing the strategy, and measuring its effects.
- If the *actual* effects on goals are very disappointing, or the actual costs are very bad, then our *experiment* has *failed* (to confirm our estimations).
- There may, or may not, be a better strategy amongst the ones we chose *not* to use. But, at the extreme, this current strategy needs dumping or radical reformulation for greater effect. Try again.
- Sometimes, the results are disappointing, but the capital cost and time are *already spent* (we hope it is only a week's worth!), and *some* gain towards Goals is hopefully made. Some knowledge is acquired. So, we may decide to *keep* the result, and continue, to try to find something in addition, but less disappointing.

Our favorite example of automatic computation of priority has been given twice earlier (Confirmit examples; Red, Green, Yellow color coding, and % total to goal to date, Figure 6.7 above).

Prioritization Policies.
Here is a sketch of some of the *innumerable variations* on this potentially, and usually, automated prioritization process, which you might need to consider:

- **Different Policies**:
  - Go for **one extremely critical objective's Goal, early.** 'To heck with' the costs, and to heck with with the results for the *other* objectives.
  - Go for anything with reasonable effects, if it **costs practically no capital outlay**. Assume you are, for the moment, until you show results, budget/cash *poor*.
  - Focus on meeting the **constraint levels first** (Tolerable, then OK level). Do not allow use of resources to move some other objectives towards their Goal levels, as this might result in an intolerable, or 'not OK' system, in other objectives. Sub-optimization. Not enough resources left to get all critical objectives to an OK level.

- **Allow violation of an external constraint** (say 'the system would be illegal in a limited geographical area', maybe) betting that we might be able to change or ignore the constraint, or at least get a much better result everywhere else.
  - *Elon Musk, 2014 , ignoring some US State laws which forbid selling Tesla cars, directly, not through a dealership, is one example that comes to mind. [F2]*
- The list of variations to the policy is long, so we hope the sample above suffices to illustrate it.

Stakeholder Value							Key: s = seconds m = minutes d = days w = week	Designs by expected Increment with design dependencies			
Regulator	IT Dept.	Customer	Rule Admin.	Business Unit	Back Office	Total Value / Benefit	Bank System By End Date: dd/mm/yyyy Requirements	1 D1: Automate Rules + Manual Testing	2 D2: Back Office Loan Decisioning	3 D3: Web Self-Service	4 D4: Automate Rules + Automate Testing
		4				4	R1: Time for customer to submit request 30 min <-> 10 min	-	-	10 m 100%	-
					3	3	R2: Time for Back Office to enter request 30 min <-> 10 min	-	-	0 m 150%	-
		9		9		18	R3: Time to respond to customer request 5 days <-> 20 seconds	-	1 d 80%	20 s 100%	-
					1	1	R4: No of Back Office complaints 10 per week <-> 0	5 50%	<1 90%	0 100%	(2) (80%)
		1			5	6	R5: No of customer complaints 25 per week <-> 5	-	15 50%	5 100%	-
1			5	4	8	18	R6: Time to update business rules 1 month <-> 1 day	2 w 50%	-	-	1 d 100%
1			3	4	6	14	R7: Time to distribute business rules 2 weeks <-> 1 day	1 d 100%	-	20 s 103%	-
2		14	8	17	23	64	Cumulative Total for Performance Requirements	200%	170%	280%	50%
							Development Budget 2.5M <-> 300K	2.3	2.0	1.0	0.5
							Development Cost for Design	0.2	0.3	1.0	0.5
							Cumulative Performance to Devt. Cost Ratio	1000	567	280	100
							Cumulative Stakeholder Value to Development Cost Ratio	23.5/0.2 =117.5	17.8/0.3 =59.3	13.7/1.0 =13.7	9/0.5 =18

Table 6.8. ***An Impact Estimation table based on mostly real data from a Bank*** *(modified for confidentiality).*
Based on PhD research by Lindsey Brodie, Middlesex University [URL75]. We can see the factors discussed in the text above, such as '% Total for Performance', and others. One innovation, explored here is on the left side, the separation of ***individual stakeholder value****: thus giving yet another potential variation when considering priority. [URL38]. The numbers used in the example are artificial value to stakeholder-benefit scores. A device to keep real data confidential.*

- **Additional computations to clarify risks**:
  - The **± uncertainty factor** is introduced, and the *least optimistic* end, of the range, for objectives and costs is used. 60±40 = 20, 20 is the 'least optimistic' end, if low is bad.
  - The **Credibility score** (0.0 none, to 1.0 perfect) is used, and multiplied by the estimate, to get a more pessimistic estimate. Example: 0.5 credibility x 60% = 30%.
    - The Credibility score is based on factual evidence for the estimate, and the source of those facts (Illustration 6.8 below).
  - **Impacts on more long-range goals** (some years from now) for the same objective are considered, not just shorter range (this year, initial deadlines).
  - Final prioritization is based on the initial experimental **measures of 2 or more competitive strategies** actually being applied in parallel to find a winner, based on real results, not just estimates.

Concept --->	Real Impact	% Impact	% Uncertainty	Evidence	Source	Credi-bility
Example	30 seconds	25%	±5%	All 9 uses are < 32 seconds	Report X 22 June	0.5
Purpose	Estimate probable impact	Express impact as % of way to Goal [t,p, e]	Estimate best/worst case	Rationale for estimate	Allow check of evidence	Risk analysis

t = time, p = place, e =event qualifiers

Illustration 6.8 Some concepts about Impact Estimation, which can be used to vary, and make more realistic, or more conservative, the priority evaluation.

The point is that *there is no one simple computation*.

- There are lots of potential tools and variations.
- The intelligent planner, is aware of interesting variations for their particular situation, and applies variations to suit their own needs.

The main point we want to bring out here is that you can build a model of your priority process, and the priority decision *can be computed*, fairly automatically, according to those factors you choose to consider, and those prioritization rules or policies you choose to apply. Several automated tools do this today [URL73, Kai Gilb's Tool, Confirmit's Tool)

***Automatic* Priority Computation is *most useful* when there**:

- is a need to look at **several scenarios**, asking "what if?", so that even a 'losing strategy' proponent sees *why* they lost; and maybe accepts defeat gracefully, or fights back constructively.
- is a **frequent need** to re-prioritize (like weekly for each project, or for many projects)
- are **many factors** (10 objectives and 10 strategies – 100 cells - for example) need to be evaluated simultaneously
- is a management need, to make sure that **various projects** have a **sound approach** to their decision making
- is a need for **transparency** of decision making: accountability to Boards, Steering committees, electorates, press, media and public.

"Facts are stubborn things;

and whatever may be our wishes, our inclinations, or the dictates of our passions,

they cannot alter the state of facts and evidence."

--John Adams (1735-1826), 2nd President of US

Practical Tip 6.8

- *Use the 12 Tough Questions (Section 1.5) on your next attempt to prioritize strategies [URL15 for detail]*
    - *Especially 'Why is this 'improved X' not quantified?'*
- *Setup your own spreadsheet to do some of the computations above*

Policy 6.8 Responsible Prioritization

- **We will use a conscious and transparent method for prioritization decisions, with clear alignment to our critical objectives, and constraints.**

- Why?
    - To improve the probability that we will achieve our goals.
    - To avoid wasted time and money, due to illogical, un-estimated decisions.

Version 31 Aug. 2015 22:53 pass 2 tsg

Section 6.9 The Five Levels of Priority

Principle 6.9 Nature's Priority

**First priority is survival,
to get to tolerable levels
and avoid constraints**

Most planners when stating an objective, limit themselves to stating one primary idea: the *main goal* itself. Represented by one single number.

We believe that it is very useful to state *other* additional levels of satisfaction, or dissatisfaction too.

One primary reason for this is because stating target levels, and constraint levels, of a set of objectives is a way of understanding their priority.

But there are several other good uses too for stating constraint and target levels of all your top-level critical objectives.

If I say: *"I wish the room temperature was better, but I am grateful it is not below freezing"*. I am stating two related but independent ideas.

- My Goal: a better room temperature.
- My Constraint: not freezing

Almost all critical objectives, and resources are of this nature:

- There are levels which are quite satisfactory
- There are levels which are quite unsatisfactory.

In fact the very definition of the notion of 'critical' implies that there is some level which is 'deadly'. Like 'freeze to death', or 'boil to death'.

It is obviously **first** priority to survive, to not die or disappear.
Then it is **second** priority to avoid discomfort and suffering.
Then it is **third** priority to reach a level of satisfaction.
It is **fourth** priority to go beyond satisfaction, to some luxury level.

So, we can, in Planguage, state these levels *directly*:

- **Tolerable**: survival, borderline, threatened with death constantly
- **OK**: absence of pain and discomfort, but much improvement desired

- **Goal**: quite satisfied, successful, only marginally useful desire to improve this
- **Stretch**: a level better than Goal, but which has *some* marginal value, for *some* stakeholders and instances, *if* the price is right
- **Ideal:** a theoretical state of maximum perfection, not really attainable, or not sustainable in practice, or not economical.
    - *Like 100% availability 24/7*

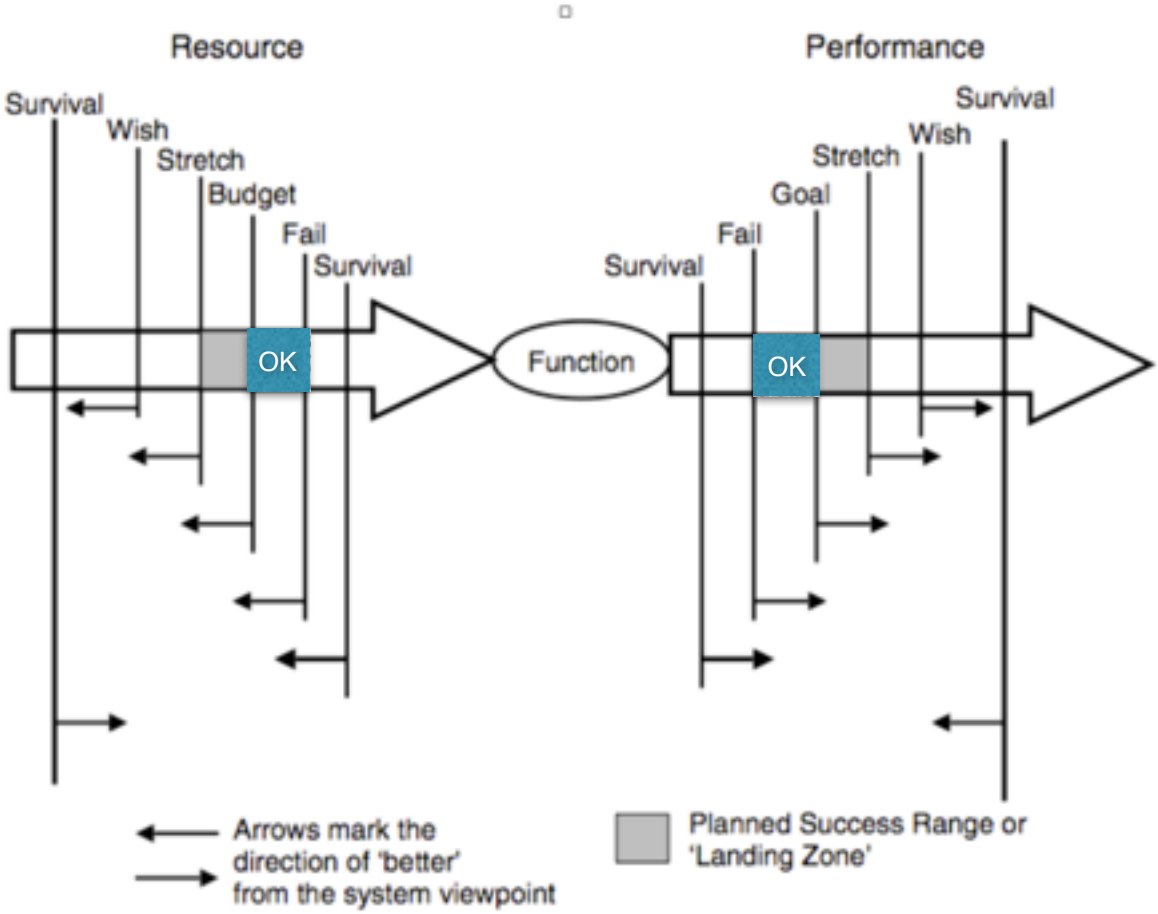


*Figure 6.10 Some levels of priority. A 'Tolerable level, and the beginning of the Tolerable Range (**<->**) is a synonym of the 'Survival' level. towards the Goal/Budget level. The OK Level , and the OK Range is a bit better than the Fail level. Source [1]*

By stating these levels explicitly, as part of our planning of an objective, we create *extreme logical clarity* about our priorities.

The exact numbers for a given level, like a Goal level, might well be somewhat *subjective*. We are usually guessing about what will be a successful level in the future. If we are somewhat wrong, we will often be able to adjust to a better number, with hindsight. But sometimes the only way to know what a 'good' number is, is to deliver it and see.

The reality might be that we plan for a range, a comfort zone or 'landing zone' (as Intel calls it). Rather than an exact number.

But at least it, once specified, becomes a *sort* of clear agreement between the stakeholders who agree to it, and the planners and suppliers who will help them achieve it.

In some cases these levels will be applicable to *contractual* agreements. Partial payment for the constraint levels, full payment for Goal, and perhaps a bonus for a Stretch level. [URL6]

Here is an example:

Service Availability:
Type: Critical Product Line Objective.
Stakeholders: Service Personnel, Product Users, Consumer Associations.

Ambition: highest service availability in our business.

Scale: % of [**Instances**] for defined [**Users**] for defined [**Products**] and defined [**Problems**] where service can begin within the hour, and be completed within an hour.

Tolerable [Instances = Life Threatening, Users = Medical, Products = Body Monitors, Problems = Total Malfunction] 99.0%

OK [Instances = Life Threatening, Users = Individual Patients at Home, Products = Body Monitors, Problems = Total Malfunction] 99.5%

Goal [Instances = Unable to Modify, Users = Medical, Products = Body Monitors, Problems = Pain or Discomfort] 99.98%.

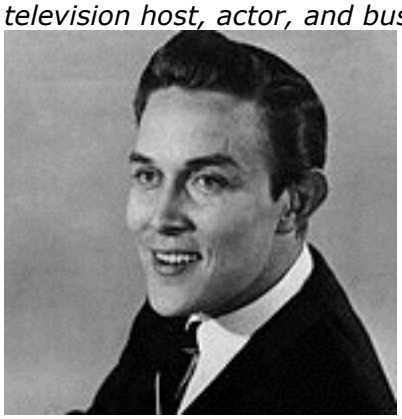
Stretch [Instances = Health Threatening, Users = Medical, Products = Body Monitors, Problems = Any Malfunction] 99.99%.

Planguage Example 6.9. The example is constructed to show variation. Hopefully the reader can see the possibilities for any useful variation.

This quote sounds to me like 'Dynamic Design to Cost' [URL11, URL12].

> ***"I can't change the direction of the wind,***
> ***but I can adjust my sails***
> ***to always reach my destination."***

Jimmy Dean (August 10, 1928 – June 13, 2010) was an American country music singer, television host, actor, and businessman.



Practical Tip 6.9

- *do not just state a target-to-reach (Goal, Wish, Stretch).*
- *Specify at least one constraint level (Tolerable, OK, Fail)*

Policy 6.9 Clarify Priority levels

We will clarify the relevant and useful numeric levels, of not just our targets (Goal, Stretch, Wish), but also our constraints (Tolerable, OK, Fail), when we specify our objectives.

Why?

- It will help us prioritize.
- It can be useful in setting contractual levels of performance.
- It gives us some short-term levels (constraints) to work towards.
- It will help avoid failure conditions, due to failure not being defined, specified, and managed.

Version 1 Sep. 2015 22:56 pass 2

Section 6.10 Accepting A Stakeholder Wish As A Committed Goal.

Principle 6.10 Dreams are not enough

**You cannot commit to mere stakeholder wishes,
they need to be implementable in practice.**

Minimum Levels.
As discussed above, our first priority is Survival (Tolerable level range) and then our priority is to getting somewhat better than the Tolerable range, but still 'just surviving', getting to the OK level range.

These (Tolerable, OK) are the 'constraint levels', the worst acceptable cases for survival, and then being OK.

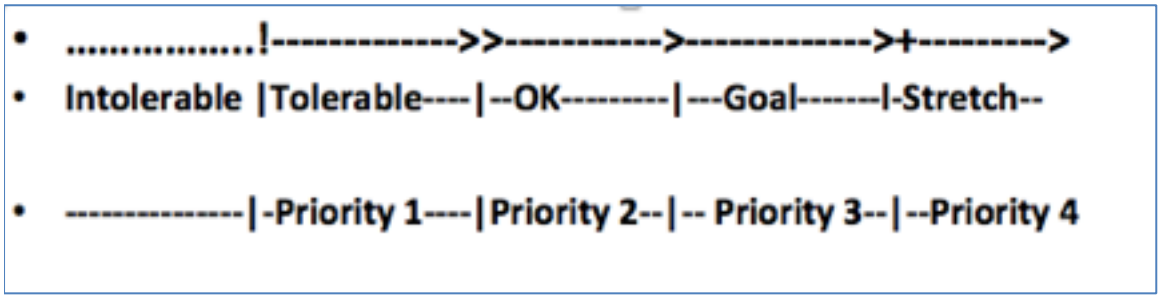


Figure 6.10 The levels on a Scale viewed as priority levels.
Note: The keyed icons (!, >>, >, >+, and -> are defined symbols in Planguage. They are human language independent, and can be keyed on a digital device). Sort of like emoticons :) and music notes, or electrical symbols. For the purposes of this book they are a side note. Some keyed icons are so convenient that they are in Planguage daily use; for example, the [Qualifier Square Bracket], and the <- Source Arrow.

Main Target Levels.
After fulfilling constraints, getting to the 'Goal' level, is the next priority.

'Goal' defines 'Success'. Or 'full satisfaction'.

Perhaps, in contracting, 'full payment'.

But Goal levels are not necessarily the 'end of the road'.

The Stretch Level.
The *last priority* level, to be allocated our project resources, if any remain, or can be obtained as extra resources, when all currently due (dead-lined) Goal levels are reached, is the **Stretch level**.

This is a kind of 'marginal' satisfaction. *Some* stakeholders place *some* value on Stretch levels.

In simple terms the Goal level means *'success', 'enough', 'stop using resources', and 'full satisfaction'.* But Goal also means **'We Promise, we commit to delivery of this level'** concept.

If a project is going to commit to a Goal level, then it needs to feel *confident of having the capability of getting to that level.*

Not just to the Goal level *sometime*, but **on time**, and consistent with the other [qualifier conditions] associated with the Goal:

For example:

Goal [Deadline = New Years Day next year, Area = Canada, People = Teenagers, If Trade Agreement Valid] 30%.

Planguage 6.10 A. A a Goal commitment, with [qualifiers]. To make it clearer exactly [when, what places, people and events] the promise is valid for.

Here is the set of conditions that we think are logically necessary for a project team to commit to a 'Goal level' value delivery.

The Goal Level should:

1. be technically possible - within the state of art. There are known strategies to get there.
2. be economically possible - resources exist, or can be obtained.
3. have costs consistent with other requirements. This is not the only objective!
4. be effective, and the effect necessary to satisfy stakeholder needs.
5. be profitable: value exceeds cost.
6. be prioritized: by agreed rules of priority, such as Effectiveness, Profitability, or Politics.
7. have all [Qualifier Conditions] in the Goal statement as 'true'.

List 6.10: the conditions for converting a Wish plan, into a committed Goal plan.
*In the meantime, if you wish to make it clear that these conditions are NOT yet fulfilled; or we do not know if they are yet, we can use the **Wish** statement.*

For example:

Wish [Deadline = New Years Day next year, Area = Canada, People = Teenagers, If Trade Agreement Valid] 30%

*Planguage 6.10 B. A Wish statement, indicating **some** stakeholder value, for reaching the numeric level ('30%'), under the [qualifier conditions].*

The 'Wish' statement means:

- We *acknowledge that there is* some stakeholder value, in achieving this level, under stated [qualifier conditions].
- We would *like* to be able to promise to deliver this level on this project; but we *cannot* yet. We have to fulfill the 8 [conditions] *first* (above example) in order to change the Wish to a Goal commitment.
- When we know we can do this, technically, resource-wise and consistently; with all our other simultaneous commitments in the project, to other Goals, by that Deadline, we will then re-classify this statement as a Goal.
- We can, in the meantime, *document, keep track of,* the status of the Goal conditions in the Planning. Or we can do a periodic evaluation of all conditions to see if we can commit to a Goal.

For example:

W1: Wish [Deadline = New Years Day next year, Area = Canada, People = Teenagers, If Trade Agreement Valid] 30%.

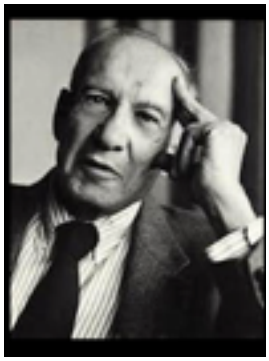
Planning Status: [W1, December 24, 20XX]
PS1: Strategies: S1, S2, and S5 are estimated to get us to about 150%±60% of the Goal <- IET ABC.
PS2: Capital Costs of the strategies are estimated to be 5% of total project budget
PS3: We do not know if we have qualified people available to carry out S5.
PS4: We do not know, at this stage of planning, if this Wish will have sufficient priority ahead of all current commitments at that time; and there are still possibilities of additional commitments.

Planguage 6.10 C Documentation, integrated in the plan, explaining why the Wish status cannot be converted to a 'we promise' Goal in the plan.

Although we are discussing prioritization, at this moment. You will observe we are **also managing risks** (of failure). More in next chapter on risks.

"Work implies not only that somebody is supposed to do the job, but also accountability, a deadline and, finally, the measurement of results —that is, feedback from results on the work and on the planning process itself,"

Drucker wrote in Management: Tasks, Responsibilities, Practices. [19]
Peter Drucker,: Business Author (1909 Vienna - 2005 Claremont, California).



Practical Tip 6.10 Using 'Wish' as Blackmail for Resources.

- *You should always initiate your statements of stakeholder needs and values using a Wish statement, rather than a Goal statement. Otherwise your stakeholders will tend to misunderstand, and believe you have promised them such results.*
- *Explain that we all need to be sure that the 'conditions are right', before you can commit to a Goal.*
- *This is a tactic for avoiding perceived project failure.*
- *You are also trying to tell them to not, foolishly, tell their colleagues that this 'Wish' will really happen, yet.*
- *You can choose to reveal, or not, the fact that 'Wish' is also your way of making sure (pressuring the resource allocation powers that be) that you really have the necessary budget, people, and time to 'promise' delivery.*

Policy 6.10. Let stakeholders 'Wish', all they want: don't promise until you know you can!

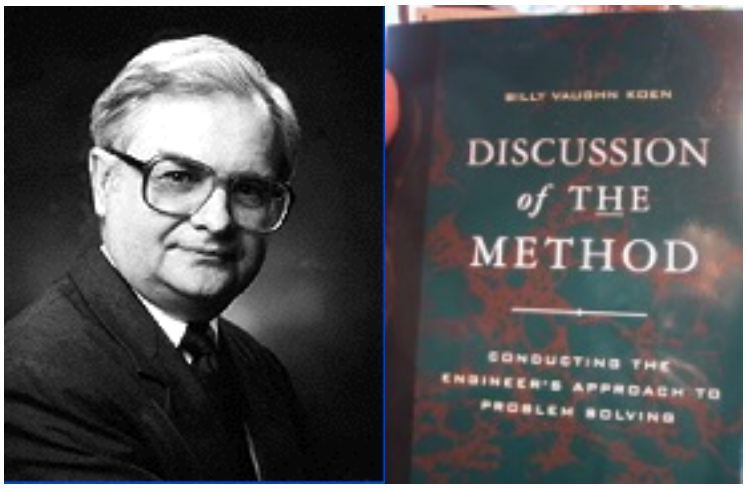
- **We will not commit (a 'Goal') to planned value deliveries for our stakeholders, until we are pretty sure we have all our ducks in a row (all conditions for moving from Wish to Goal).**
- **Use 'Wish' until we are ready.**
    - In the meantime we will keep written track, of all conditions Planning Status, so we know what is keeping us from commitment, and can work on these problems, or use information about Planning Problems to motivate others (for example to get necessary resources, to give it a go).

Why?

- To build and keep credibility for keeping promises. You need that to succeed in the long term.

Chapter 7. Risk Management

"Make small changes in the sota"



'Sota' = Engineering State Of The Art Heuristics

Source: Prof. Billy Koen, [20] Discussion of the Method, p. 48

Always give yourself a chance to retreat; and
Use feedback to stabilize the design process

GILB's Law
If you don't attack the risks, the risks will attack you.

There is some amount of risk in every project.

Further, ignoring risks does not negate these risks, but means that they will not be taken into consideration and managed.

If these risks are not monitored, they can evolve into real problems, that are likely to "attack" you.

When something has risks, these risks should not cause you to avoid pursuing the project.

Nor should many risks cause you to pursue the project but ignore the risks.

A stitch in time, saves nine.

(Source POSEM [5])

Risk management is an *economic* matter. The cost of risk planning is usually trivial compared to the costs of delays and failures. But only experienced and responsible people are motivated to 'take the trouble' out of the plans.

Version Sept 19 2015 13:33 Odessa

Section 7.1 Risk Management is universal and continuous.

Principle 7.1 Universal Risk Responsibility.

Risk management is 'everybody's' job, at all times.

The whole point of all management, engineering and planning work is to succeed in our projects; or put differently, to avoid risks that can cause degrees of failure.

We liked Swedish Ericsson's Corporate Quality Policy in the 1990's, which clearly stated that risks were the responsibility of every engineer and manager.

Here is a 2014 public presentation of those ideas, unchanged for decades.

"Risk management:
Risk management is integrated with the business and its operational processes and is a part of the Ericsson Group Management system to ensure accountability, effectiveness, efficiency, business continuity and compliance with corporate governance, legal and other requirements. the Board of Directors is also actively engaged in the company's risk management." [URL39]

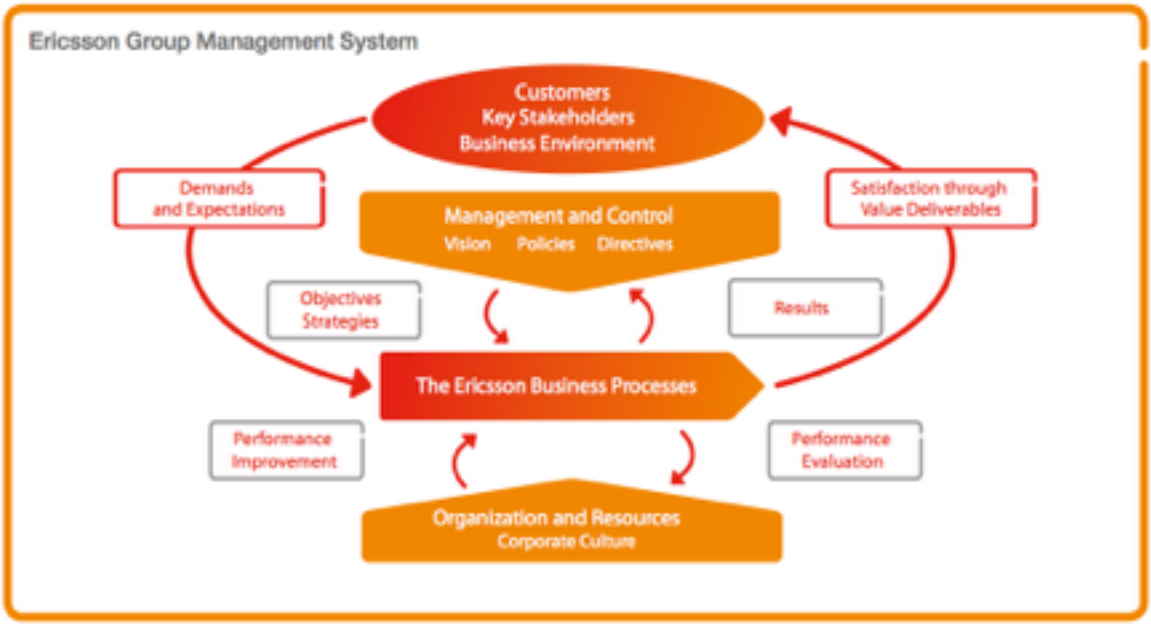


Figure 7.1 [URL39] . This is a pretty good representation of what this book is teaching. Notice the 'Value Deliverables'. 'Performance Evaluation/Improvement', Results, Key Stakeholders: even the terminology corresponds well with our concepts, and actual terminology. We worked for Ericsson as teachers and consultants for over 10 years. Mutual admiration, perhaps?

Not only is risk everybody's responsibility. There is risk in just about *every detail* in planning.

So we need everybody to be conscious of the risks (bad things that can happen), and to do risk prevention (to avoid bad things happening), and to plan risk mitigation (what to do if bad things happen).

What can go wrong in planning?

- **Some '*Objectives*' Risks**
  - You can fail to state, and manage, a critical objective
  - You can commit to a Goal when you actually do not fulfill the 8 logical and basic conditions of Goal readiness discussed earlier (Chapter 6.10)
  - Your target levels are unreasonably high, or your deadlines are unreasonably short (same ideas as 8 conditions, 6.10) above.
  - Your Scales of measure do not reflect the critical reality well enough. They are not correlated well with real success.
  - Failure to consider the needs of all critical stakeholders. Just one stakeholder forgotten is enough to hurt.

- **Some '*Strategy*' Risks**
  - The planned strategies are not really powerful enough to reach the targets on time. Even when estimates say they are.
  - The strategies are not decomposed in smaller steps of implementation, and thus fail on too-large a scale.
  - The strategies are misinterpreted by implementers, because they are unclearly specified.
  - The strategies are not planned for, regarding all critical systems aspects, for example motivation, culture, legality, long-term support, service and user support, and international considerations.

- **Project Planning Risks**
  - Failure to plan to do small, for example 'one week', increments (trials, pilots, experiments).
  - Bad communication with sub-suppliers, they misunderstand, or falsify progress reporting.
  - Your own well-intended bureaucracy denies the value of smart practices and insists on big bang, big risk, exclusive suppliers, long term contract, or layers of approval

- Focus on doing *tasks*, *not* on moving the critical top ten values to Goal levels.

What can we do in 'planning' to *avoid* risks?

- Just about **every detail in this book** is practical advice in dealing with risks; all of it. That is the whole point with planning!
  - We could title this book: *'Advanced Risk Management'.*
- **Focus** on getting **rapid feedback** (weekly) and necessary correction regarding numeric progress towards your **top ten critical objectives**
  - Including when *subcontracting*! In your contracts.

What can we do in planning to *mitigate* risks ?

- 1. Make specific plans to 'mitigate' all identified risks. I.e. fix the problem that the risk exists.

Goal [Europe, If New Tax Law is in Force] 30%.
Implied Risk: Tax law not in force.
Mitigation: stop all investment in this Goal. Status: Approved by CTO.

Planguage 7.1 A The risk is implied in the [qualifier]. But the planner makes it explicit, 'Implied Risk:…', and suggests a Mitigation.

- 2. Use [Qualifiers] to automatically 'un-plan' things, when conditions are not 'true'

Goal [Deadline = 1.1.20xx, If Merger with XYZ in Force, Activity = Recall**]**
35%.

Planguage 7.1.B If the merger does not occur, as expected (a risk) then this planned Goal is automatically NOT a goal any more. All numeric-level-specification statements have qualifiers, that must all be true for this numeric level of the plan, to be in force at all.

- 3. Pay suppliers for *results*, not *work*. Then if risks kick in, it is mainly *their* problem, and they are highly motivated to solve it for you [URL6], or motivated to prevent it. Write **no cure no pay** contracts.

What can we do in planning to detect risks?

- 1. Have standards (specification 'Rules', like below) which expressly demand risk specification, and mitigation planning. Lack of this (risk and mitigation explicitness) is deemed a major defect, and is the fault of the planner, and plan Owner.
- 2. Then make sure all plans are reviewed against these Rules, and that major defects are not accepted (no 'exit' of plan) [URL23]

Rules:
R1. All implied or obvious risks will be brought out explicitly (Risk: …).

R2. All risks will have an explicit Mitigation: specification, including source (<-).

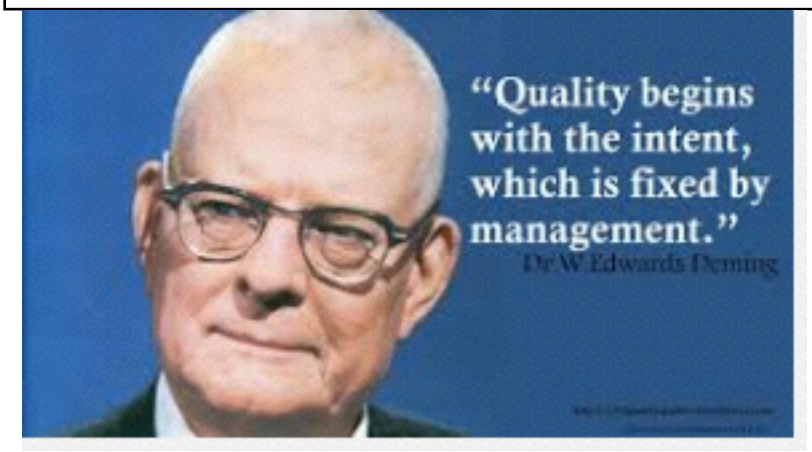
R3. When there is any risk of a numeric specification (a Goal level, an IE estimate) deviating more than trivially (±2 %), then an explicit ± *uncertainty*, or *range*, will be used in the specification (30±10, 20 to 40).

R4. When a numeric uncertainty is indicated (R3) the cause, evidence or background for this assertion, even if it is just a rough intuitive guess, shall be specified, so that we understand what it means. The source (<-) should be given to permit credibility and audit .

Planguage 7.1 C. A sample of Planguage Rules for risk specification. These are then used in a review/audit/inspection/QC as a best practice standard to check the planning specifications against. There are many Rules you can use as templates, to be found in Competitive Engineering [1]. You need to develop your own rules, to enforce the planning rigor you want and need. When management enforces a high standard of just 'a few major defects per plan page', our industrial experience is that planners really learn to take these rules seriously, as part of their job.

"Long-term commitment to new learning and new philosophy is required of any management that seeks transformation. The timid and the fainthearted, and the people that expect quick results, are doomed to disappointment."

Dr. W. Edwards Deming (1900-1993)
http://en.wikipedia.org/wiki/W._Edwards_Deming



Practical Tip 7.1 Sample the risks in your real plans.

- *take any current plan, perhaps just a one-page sample, and identify as many risks to success as you can. Just underline them quickly.*
- *identify at least one mitigation, for each risk identified: a word or two will do to keep it simple, but show there is some kind of help available.*
- *then ask: does this need to be done on the entire plan?*
- *Or are you willing to sign off on it, as it is?*

Policy 7.1 Risk Pervasively Permeates Planning

- **Risk planning is everybody's responsibility, in every detail of planning, and execution of the plans.**
  - Why?
    - Because even small details in a large plan can ruin your whole project.

    - ***"One nuclear bomb, can ruin your whole day!"***

      - Car bumper sticker spotted in London.

Version 19 sept 2015 13:50 Odessa.

Section 7.2 Reducing Risk

Principle 7.2 Kill Risk Early.

The best way to deal with risk is
early, frequently, measurably
– delivering a value stream and learning fast.

The sheer infinite quantity of risks, means that we cannot imagine all possible risks in advance. We cannot plan for them all.

We can identify and plan for some risks. We can hope to keep the volume of threats down. So we will not be *so overwhelmed* by problems, that we *can* have capacity cope with surprises we did *not* plan for. We can also have some *general mitigation strategies* for most any risks, even ones we could not pre-plan. For example rapid frequent early numeric feedback is very general.

Some basic concepts:

- A **Risk** is any factor that could result in a future negative consequence. It is a combination of {(Threat or Attack) & Mitigation}.
- A '**Threat'** is something (person, thing, situation, circumstances) that can potentially cause some defined class of project failure or lack of success.
- An **Attack** is when the Threat becomes real, when it really emerges, and potentially does, or can do, some damage to a system.
- **Damage** is anything contrary to plans, including lack of performance, lack of value delivery for objectives, or increased unexpected resources (time, money, staff), **Damage** is any negative consequence of an attack on a system.
- A **Mitigation**, is any act, pre-planned or not, which reduces or eliminates the damage caused by a threat happening.

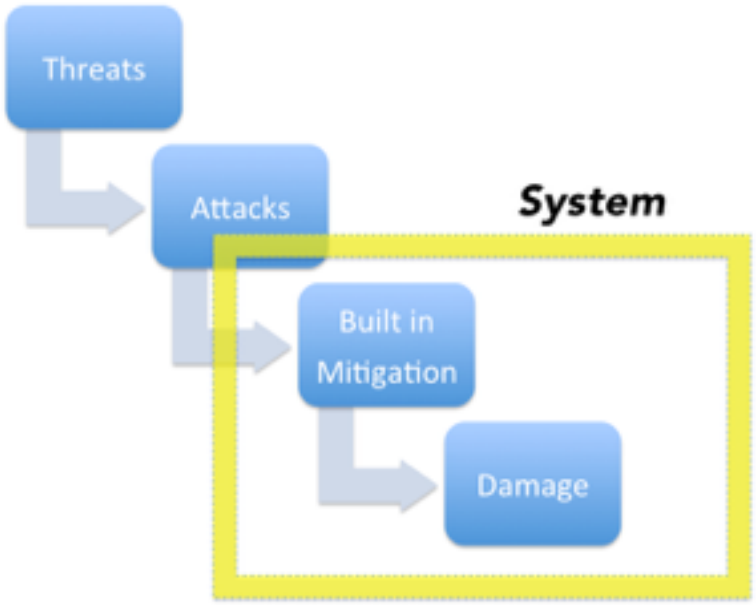


Figure 7.2 The risk environment. Threats may turn into real attacks, which get confronted with our planned mitigation. If the mitigation plan is unsuccessful, damage results, of various kinds. Notice that 'built-in to the system mitigation' is one type of mitigation. We can also mitigate risks at earlier stages, such as in planning, contracting, building trials, tests etc.

Risk Planning tasks include:

- *Identification* of interesting threats, and *specification* of those threats in the plans

> **T1: Threat: software changes by outside vendors could open up security back doors inadvertently.**

- Analysis of the probability and frequency that a threat type will actually attack a given system, under given conditions

> **T2: Threat** [Environment = Hacker Attack, Type = Service Denial] *Several Times Annually <- According to Banking Statistics.*

- Identification of suitable strategies, *mitigation* strategies, for dealing with the attacks, and evaluation of their cost effectiveness.

> **M1: Mitigation [T2]: Avoid External Suppliers, and Public Domain Security Software**

- Prioritization, in the implementation sequence, of the most useful **built-in mitigation strategies**, and *learning from feedback*, how good they really are.

> ***Evo Step Planning***
>
> **Sequence:** Step 3, for Version 9.5 Release.
>
> **Security Package:** M1, M3, M5.

- Planning potential post-damage mitigation strategies, for dealing with successful attacks.

Post Damage Mitigation Planning

> **Post Damage Mitigation [T1, T2] PDM1, PDM2**
>
> **PDM1:** Compensate Service-Denied Customers Financially.
> **PDM2:** Activate Insurance Policy compensation against Service-Denial Financial Losses by Customers.

What are 'generic' risk-mitigation ideas in particular? These are the sort of things which give general mitigation against bad things, even though you cannot know exactly *which* bad things – attacks – will happen, and cannot plan *specifically* for them.

- **Short feedback cycles**: the faster you can experience the first attacks, the faster you can implement specific mitigation for them.
- **Flexible Contracting with Suppliers** [URL6]: the ability to quickly decide later, exactly what your contractors must do for you, in order to mitigate surprising attacks. Even if it is not already specified in their work requirements.
- **Open-Ended Architecture**: [URL28 A&B, 5 C, 1 B] invest early specifically in things that make your system easy to change quickly, for unforeseen threats. You can consciously engineer *ease of adaptation and change*, into your system [URL76].
- **Buy financial insurance**: this is built-in to the system, post-damage mitigation. But you can look into it in advance. Plan it.
- **Keep your focus on top level final critical objectives:** not the initial strategies that actually are a source of some attacks.

"Risk comes from not knowing what you're doing"
— Warren Buffett (1930-)

Practical Tip 7.2

- *assume absolutely everything can go wrong, potentially*
  - *Murphy's Law is true! [URL55A]*
- *If there is ONE single tactic to bet on, to deal with this omni-risk situation, it is to get early realistic feedback, as you build your new system, product or organization*
  - *'Early feedback' includes both 'measures of value and cost' -*
  - *and 'observations, stakeholder feedback, and complaints'*

- *One related tactic is to schedule for some level of implementation your 'worst fears' first: your highest risks with the worst potential consequences. Pilots, experiments, trial marketing, small-scale before you scale up. Limit the damage. Get experience, rather than 'plan more'.*
- *'Planning' can help you find out where you need to get practical experience with risks, first.*

Policy 7.2 Plan a low-risk system: Provoke attacks early

- **We will plan to encounter attacks early, with a view to quickly learning how to mitigate them.**
- **We will specifically plan by identifying risks, threats, attack possibilities, and at least, planning specific and general defenses.**

- Why?
  - A reasonable investment in risk planning should easily pay off, in fewer costly problems.
  - The cost of integrating mitigations at early stages is probably less than patching-them-in later under pressure.

Version 19 sept 2015 14:43 Odessa

Section 7.3 Continuous learning, improving and grass roots power to change.

Principle 7.3 Improve Constantly.

You need to learn how your organization is performing, and adjust it constantly: delegate that power [URL14].

We all realize that the best of organizations need constant improvement to remain competitive. I think most managers would recognize improvement as their responsibility: especially the higher the rank they have.

I do not believe that the best way to accomplish the improvements, is that management makes the decisions about *exactly what* should be changed (the means, the strategies). [URL14]

One danger is that managers, the higher up they are in rank, are farther from the reality on the ground. They are likely to make decisions that do not work well with their troops in the field.

We suggest that the role of management is to motivate, enable, and support the do-er level to 'make' the changes *themselves*. Management should *set ambitions* for improvement (objectives, motivation) and *provide a framework* (money, time, competence, power) to *enable* those objectives to be met.

This means that the Grass Roots will take the following actions:

- Analyze problems
- Select problems to work on
- Generate ideas of how to solve problems
- Try out their own ideas (experiment, pilot)
- Publicize their results
- Take credit for their successes
- Take responsibility for their own working environment
- Assist larger parts of their organization to spread their successful ideas
- Monitor the longer-term results, and make changes, or further improvements

The consequence of this, for the Grass Roots, is that:

- The grass roots will need *training* to do this
- They will need *access to facts*, experts, consultants, coaches
- They will need *budget* to do this

- They will need *time at work* to do it
- They will need *management support* to do it
- They might need to be 'protected' from their immediate supervisor's short-term pressure, by enlightened managers, higher up.
- But, it should be a *part* of their job to improve, not their full time job.

  Otherwise they are no longer the grass roots.
  And I do not believe in using hordes of expensive big name outside consultants to do this job, ever.
- Improvement work should be looked at as continuous investment, with continuous returns: and you should be looking for large returns [URL27 A, Raytheon got 7.3 to 1 ROI].

The Role of the Manager then becomes:

- Design and improve the *framework* for Grass Roots improvement.
- Budget time and money as investment.
- Review return on investment, and help those with low returns.
- Keep feedback data regarding the results, and costs.
- Celebrate successes.
- Make sure good results are available to others. Spread to others in the organization.
- They should NOT *mandate* the improved methods to be spread to the larger organization! They should spread the information to *motivate* others to adopt, if relevant for them.

Case Study: IBM DPP Method and its application at Raytheon..

Robert Mays, Linda Jones, and others at IBM, invented the Defect Prevention Process (DPP) [URL 40, URL14, URL27 Raytheon Case].

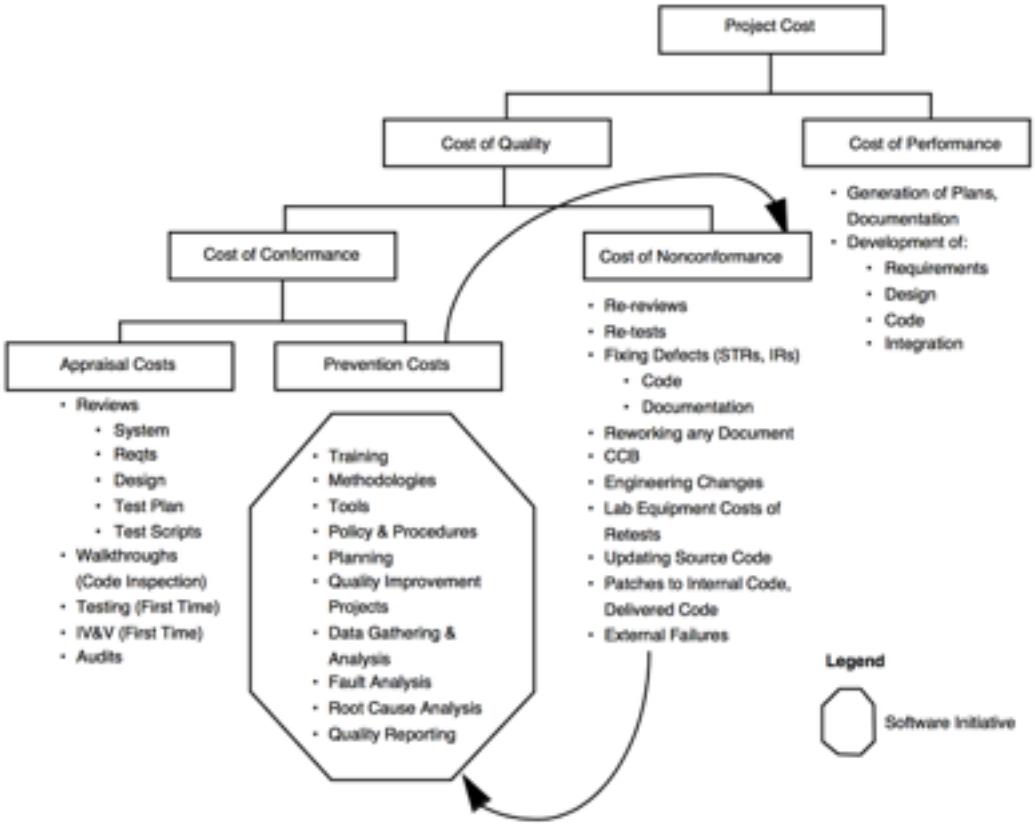


Figure 7.3 A In the Raytheon Case [URL27 A], where the Defect Prevention Process was a major improvement force, this diagram shows the attention paid to reducing waste (Non-conformance costs) by investing in prevention (Prevention Costs). The Return On Investment was about 7.3 to 1. This is a clear example of 'Lean' thinking.

This essentially delegates power to the grass roots, to improve their own work environment. Mays reported that the IBM Return On Investment of DPP was from 10:1 to 13:1, internal ROI was 3:1 to 2:1.

In one IBM site application (Minnesota. Tom visited them, and also visited Mays in Research Triangle Park, NC) in a 30 month period, there were a reported **2,162 Defect Prevention Process Actions implemented** in a 30 month period (Steve Kan, IBM SJ, 1994, URL77).

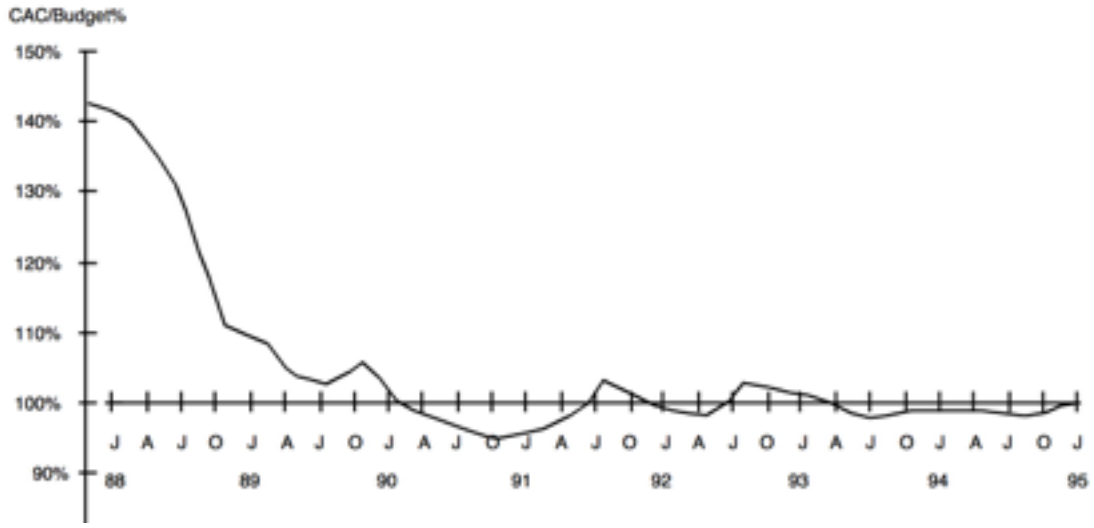


Figure 7.3 B. From the Raytheon Case [URL27 A]. The waste-prevention effect of the DPP effort was strong. Within about 1 year normal project budget overruns of 140% were essentially down to zero for years on end. Quite an amazing result for the DPP process.

There were about 2,000 employees in IBM Minnesota. That number, 2162, of implemented improvements, for 2,000 people, in that 30 month time frame, says a lot.

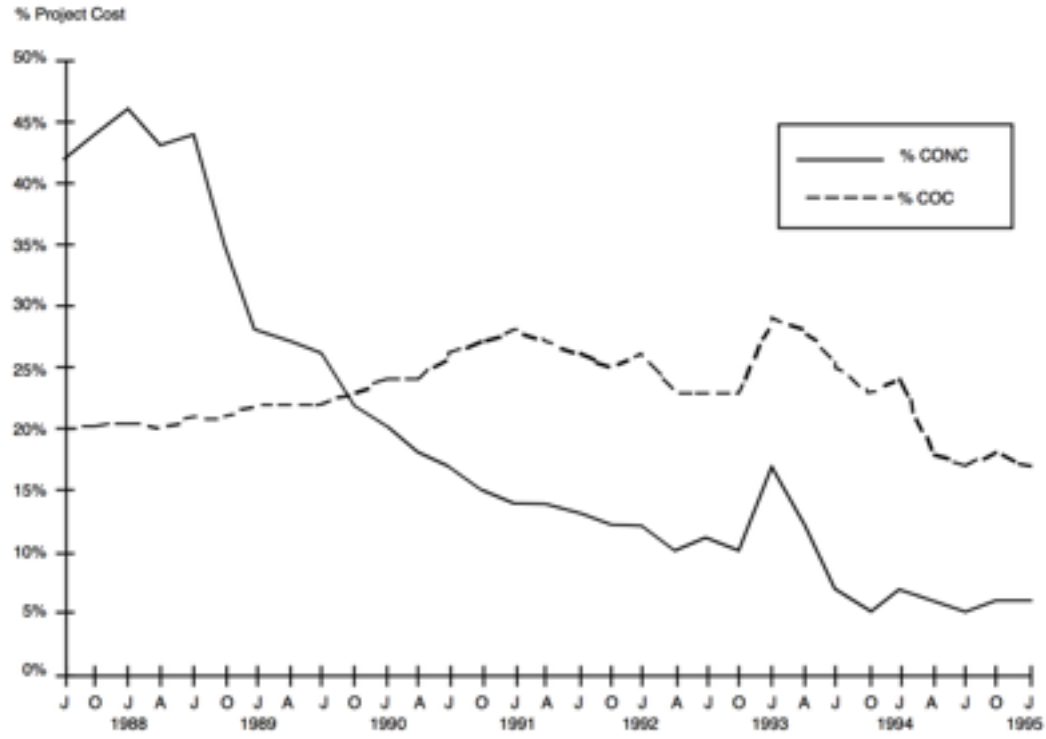
It was very *decentralized* stream of process change, done by very many different smart engineers.

Management was smart enough to empower them.

This DPP method was later the core of the US DoD SEI Capability Maturity Model (Software Engineering Institute, CMM) Level 5 (adopted by DoD from IBM), the highest level of 'maturity' for a technical organization.

Not easy to reach that high level, but it is very competitive if you really do 'DPP' for the long term.

Figure 7.3 C. Dramatic Wasted Effort Reduction For 1,000 Software Engineers.



At Raytheon [URL27 A]. In Figure 7.3 above, CONC is Cost Of Non-Conformance (of wasted effort). COC is Cost of Conformance (of minimum normal effort needed to get stuff done at all and check it). The Cost of Quality. The % CONC curve over the years shows the dramatic reduction of the 45% wasted effort (of total effort) , by 10X. It continued to improve towards 2-3% the next 2 years (Bill Curtis, SEI, Software Engineering Institute, at CMU).

It took many years of persistent DPP effort. But quite dramatic results were obtained in the first year. That is when they mandated 'following Raytheon's defined standards'. The first year result was 'bad' (reduced to 27% waste), but better than previous anarchy (45% waste). This (following the standards) level corresponds to the Deming/Shewhart SPC (Statistical Process Control) level of a stable, predictable system [9]. The basis for systematic improvement.

Raytheon was then ready for a multi-year stream of process improvements to drive the waste down further, systematically, and 'permanently'.

I find it interesting that few managers know about this DPP process [URL40], or its industrial results. It was never 'marketed' and sold as are so many 'silver bullets' today; none of which can show such dramatic and persistent results. SPC, its' cousin method is well known in manufacturing [9], but less as a method for improving a *human organization's capability*, like this.

The negative reversal spike in the chart 7.3 C above (at 1993) was due to an 'improvement' idea that failed, and was removed (my source, a Raytheon employee, orally to the author). This illustrates the importance of following up changes, by measurement, to validate their expected results, or to discover early that they are not working as intended.

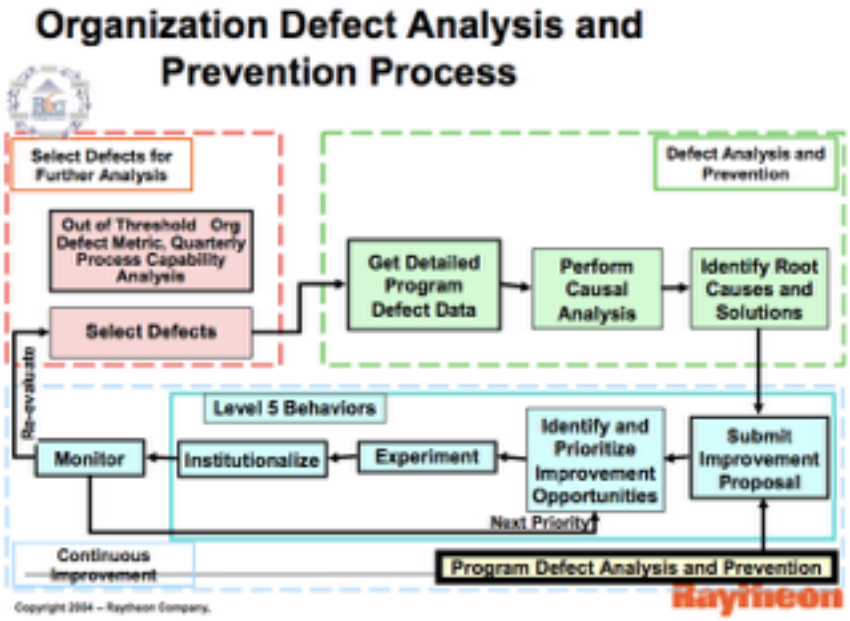


Figure 7.3 D. A Raytheon view of the Defect Prevention Process.
Source: Experiences in [URL27 B]
Root Cause Analysis and Defect Prevention Methods
Kelly L. Lanier Raytheon Network Centric Systems

It is worth saying that 'decentralization of decision-making' to the field (grass roots), and 'decision-making about the environment and organization', is not a new idea. It is ancient. In fact when organizations were smaller (Farm, Blacksmith, Pub Owner) that is the way it was done. In some military organizations [F3], the US Marines also come to mind, delegation to the soldier-in-the-field-operation is necessary.

But sometimes managers lose their way, and need to be reminded what works best. They are under pressure from outside services and methods suppliers, who are very good at convincing managers what to do, to use the supplier's products, services and methods. To make the 'big' management decision, 'outsource everything'. The history of these big centralized decisions is not a pretty one.

Here are the words of a great leader, General Robin Olds, 'Fighter Pilot' (and he kept on flying real missions (grass roots contact) long after he became the boss):

"Here's what I learned over the years.

Know the mission, what is expected of you and your people.

Get to know those people, their attitudes and expectations.

Visit all the shops and sections.

Ask questions. Don't be shy.

Learn what each does, how the parts fit into the whole.

Find out what supplies and equipment are lacking, what the workers need.

To whom does each shop chief report? Does that officer really know the people under him, is he aware of their needs and training?

Does that NCO supervise or just make out reports without checking the facts?

Remember, those reports eventually come to you.

Don't try to bullshit the troops, but make sure they know the buck stops with you, that you'll shoulder the blame when things go wrong.

Correct without revenge or anger.

Recognize accomplishment.

Reward accordingly.

Foster spirit through self–pride, not slogans, and never at the expense of another unit. "

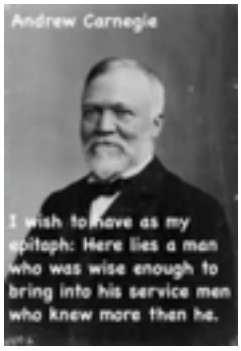
Robin Olds, in 'Fighter Pilot' [13, see more detail F3].

Robin Olds

From a planning perspective, management needs to set the top level objectives clearly, then get out of the way of their troops, and let their troops find and evaluate strategies to meet objectives.

"No person will make a great business who wants to do it all himself or get all the credit."

Andrew Carnegie (1835-1919)



Practical Tip 7.3 Delegate finding the means, to meet your aims

- ***Become fanatic on delegating responsibility for finding good strategies, to match your objectives and constraints***
- ***Do not get pulled in to giving your opinion on particular strategies (you might get what you favor, but not what you really need)***
  - ***'if it works, I support it.' is your reply***
  - ***'if it does not, you need to find a better strategy'***

Policy 7.3 Let the Grass Roots find the right strategies

- **We will focus management on planning top-level clear objectives.**
- **Management will then delegate the power to find the strategies needed, to their organization.**
- **The closer to that real action, the better; to the grass roots, the young recruits.**
-
- Why?
  - It pays off
  - People like it, a lot
  - People hate management imposing methods on them

Version 1 Oct 2015 in Air London to Kiev

22OCT MARK WHILE SORTING BACKUPS

Section 7.4 Navigating projects over troubled waters

Principle 7.4 Steer Projects Towards Objectives

You need to learn how any project is going, *early*, *measurably*, and adjust it *continuously*: *delegate* that power, too

Time and time again we see for ourselves, or read in media, about large projects, public and private, way over budget and deadlines. Usually with no results delivered at all.

The project that delivers as expected, on time, under budget is so exceptional that it should get headlines (Fig. 7.3 B, Mills, 1980 below quotation in 7.4).

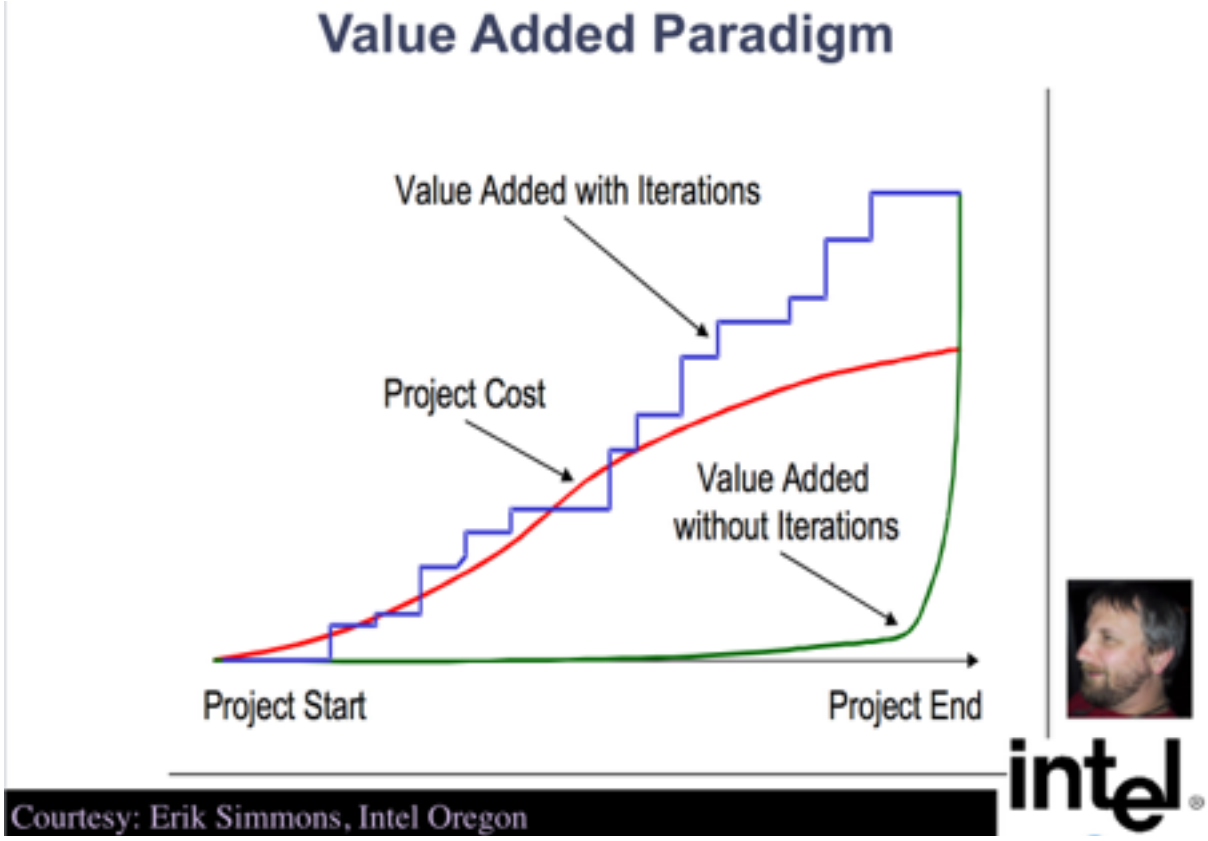
Here are the reasons we fail: Each one of the sub-process failures listed below increases risk of project failure.

- We attempt *too much at once*, and learn that it fails, too late.
- We get focussed on '*building the new system*', instead of focussing on delivering the top-level critical *results; the improvements [URL8, A].*
- We do not seem to understand how to specify the *top-level critical results quantitatively*, and clearly (as discussed earlier); so we have no clear destinations to navigate towards.
- We do not seem to know how *to decompose major strategies into smaller incremental deliveries*; so we can get real results early; and real feedback about our ability to deliver bigger things gradually (as discussed earlier here [URL16, Chapter 5 Decomposition]).

Here are the conditions for success:

- You need to *assume that all plans are 'risky* theory', until proven otherwise
- Management needs to be somewhat *fanatic* on the policy that 'they *expect*, without fail, early *measurable improvements*, in the critical objectives direction; and continued progress in the direction of final goals, on time, under budget; as the norm, not the exception'.

- *Staff need to be trained and coached* to do these things. They are as-good-as-never trained to do that, especially not in Business Schools, which presumably 'train' our managers.
    - I cannot find a Business School exception, but if there is, they are hiding well from me, and the media [2, The Puritan Gift].
    - Where have you heard *"We train management to deliver measurable results early and incrementally; leading to delivering projects always on time, and under budget".*
    - If I have missed such a faculty, let me know! So until academia catches up with reality, we are going to have to train ourselves internally[1].
- All manner of reward, for your own staff, for contractors: money, promotion and praise needs to be *on the basis of measured*, and **lasting,** *improvements actually delivered.* That is not the current reward system! [URL6].



[1] in Oct 2015 I accepted a teaching assignment at Lviv (Ukraine) Catholic Business School, so that there is at least one Business School that teaches these things. This book was the textbook, and we planned a Mars Mission using Richard Smiths Tool [URL73]. Any Business School that wants to teach these ideas is welcome to a free digital copy of this book for all students. A Friend, Dmytro Mindra, has started translating this book into Ukranian too! (Nov. 2015). Translations welcome!

*Figure 7.4 A. One aspect of dealing with value delivery early and frequently is that the value is **in place much earlier**. The main point however is that you can be sure you **really did get value** from your strategy ideas. You do not risk that they totally fail you, as so many projects actually do. I like Erik's remark in his Foreword to my 'Competitive Engineering' [1] book: 'This stuff works!'. Erik Simmons and his staff have trained over 20,000 Intel Engineers to use Planguage in their daily work. They volunteer to learn it, it is not a 'standard'.*

What specifically can we use, of Planguage Planning tools, to help us succeed in projects?

- You can *quantify critical objectives* (Chapter 1). The 'Foundation'.
- You can *detail* strategies better (to assure their value and cost effectiveness), and *decompose* strategies by value, and *estimate* the effects of strategies better (Chapter 5).
- You can get an overview of *relationships* of Objectives and Strategies using Impact Estimation tables (Chapter 5 & 6). You can see the quantified expectations, and later the results, for each strategy in satisfying your objectives.
- You can specify the *risk* elements and levels more explicitly: for example ± uncertainty, evidence, sources, issues, dependencies, safety margins etc. (Chapter 6.5).
- You can prioritize the strategies that promise to give the best effects, and lowest costs (Chapter 6).
- You can track the value deliveries, step-by-step, using the IE Tables. And guide the project towards your goals, while keeping your eye on resource consumption (Chapter 7, 2.3 Quinnan).
- If results deviate from good expectations, you can quickly try better options, and get back on track again. Confirmed by feedback to the IE Tables (Chapter 7, 2.3 Quinnan).

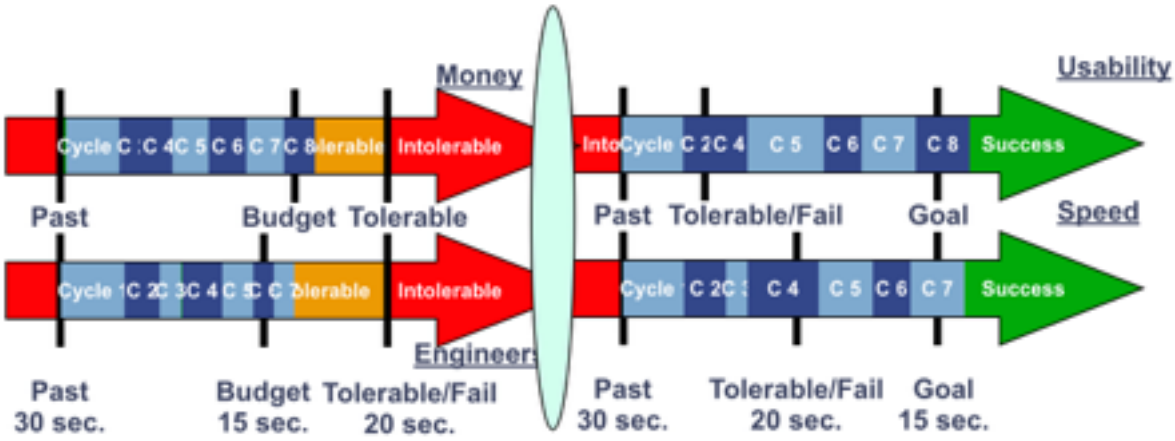


Figure 7.4 B. (Source: Kai Gilb) Each value delivery cycle, might simultaneously deliver value to more than one value objective. At the same time each cycle will consume scarce limited budgeted resources. We need to track all of this incremental feedback in order to: 1. stop when we have success, 2. stop using resources which are 'used up', 3. choose new strategies which do not need resources which are used up: i.e. which use the actual resources still available. Design to cost, dynamically, like IBM Cleanroom and Quinnan (Chapter 7, 2.3 Quinnan).

Case Study: Cleanroom at IBM Federal Systems Division:

You met Robert Quinnan, in Chapter 2.3, regarding 'Dynamic Design to Cost'. This is a major engineering art, with respect to this sub-chapter's Principle 7.4 ('**adjust it continuously'**): Quinnan is making adjustments to the system design, and keeping his eye on *both* performance effects, and costs cumulation.

Let us look at the larger *management* picture, in which Quinnan worked.

Let me introduce Harlan Mills. The 'Leonardo da Vinci' of Software Engineering. He was given a very difficult problem to solve, by his IBM Federal Systems Division management.

> ***" Every time we win a government contract, as lowest bidder, with fixed prices, fixed high quality and fixed deadlines, we lose money. Can you fix that? "***

His *Cleanroom* team cracked the problem, over a ten year period. Proving on real projects, that they could deliver the most advanced technology, military and space, 'on time and under budget' – every time, years in a row.

Not bad. In fact, 'perfect project management'!

Why can't we all do as well? We can if we follow their recipe! The key ideas are the same as this book is preaching. His world was software, but the principles we are offering for consideration here are *universal*.[URL13, URL41].

Again, as with the Defect Prevention Process (DPP) method above, IBM was not in the marketplace 'selling' this 'Cleanroom' method (as they do 'Rational' and other method products, more widely adopted). IBM was not in the 'methods' market at the time. So most managers never heard of it, and its remarkable results. Sometimes the best things in life *are* free!

Harlan Mills

Let Mills speak for himself *[IBM Systems Journal 4/1980, URL41]*

"Software Engineering began to emerge in FSD" *(IBM Federal Systems Division, from 1996 a part of Lockheed Martin Marietta, and other mergers later****) "some ten years ago, in a continuing evolution that is still underway:***

Ten years ago general management expected the worst from software projects – cost overruns, late deliveries, unreliable and incomplete software

Today management has learned to expect on-time, within budget, deliveries of high-quality software.

A Navy helicopter ship system, called LAMPS, provides a recent example. LAMPS software was a four-year project of over 200 person-years of effort, developing over three million, and integrating over seven million words of program and data for eight different processors distributed between a helicopter and a ship in 45 incremental deliveries." *(Note, that is about 2% of time to deadline for each delivery step). [URL88]*

"Every one of those deliveries was on time and under budget.

A more extended example can be found in the NASA space program" *(Space Shuttle Ground Software)." [URL89]*

- "Where in the past ten years, FSD has managed some 7,000 person-years of software development, developing and integrating over a hundred million bytes of program and data for ground and space processors in over a dozen projects.

- There were few late or overrun deliveries in that decade, and none at all in the past four years." *[URL41, URL12]*

Wow!

Perfect Large-scale long-term high-tech project management. Are *you* that good?

It is worth noting, that everybody else in the FSD business, space and military at that time, was doing 'big bang' (aka Waterfall) projects, and failing; at taxpayer expense, as IBM also did previously.

One of my Naval sources (Norm V. Brown) then, 1980s, told me that of $40 billion software spend at US DoD annually, half was known to be totally wasted on such non-evolutionary projects. He invited me to teach DoD my evolutionary project methods. He said they needed it desperately, but were not conscious of that fact.

I volunteered to help. A mosquito attacking the elephant in the room.

The major innovation in Cleanroom, we believe (there were many, some a bit over the top for your purposes), is the small, 2% (monthly for 4 years) value delivery steps.

The 2% value delivery steps act together with (people like Quinnan) examining (measuring) real impact, real cost, cumulative impact, and cumulative costs. Dynamically redesigning, if necessary, to keep on track for performance, quality and cost.

It works! The 'secret' of this successful method is 'well kept'. Even most of the rest of IBM, I found in my visits to the other IBM research labs, did not know about Cleanroom, I found.

A note. *Cleanroom* was for sure *iterative* and *incremental (I&I)*. But Cleanroom was one *decisive* concept *more* than I & I; it was ***evolutionary*** [1 C, URL13]. That means it had *feedback, learning and change.* It had the 'SA' (Study, Act) in the Deming [9] PDSA Cycle [URL42].

Any 'agile' project management method which does not have these 'learning' aspects, and most do *not*, cannot deal with risk, on the fly, as Cleanroom, Evo [1 C, URL13], HP [URL31], and Confirmit [URL28] do.

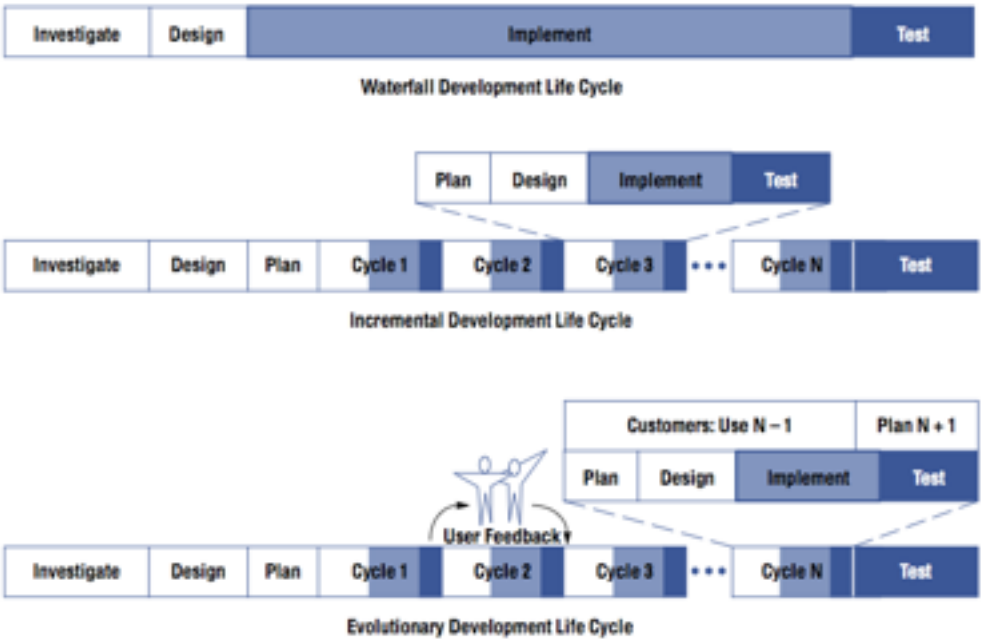


Figure 7.4 C [URL31 B, HP] 3 approaches to delivering projects. The Evolutionary cycle (bottom) is characterized by stakeholder feedback and change. The others just hurry up to be late with bad quality.

"Life is pretty simple: You do some stuff. Most fails. Some works. You do more of what works. If it works big, others quickly copy it. Then you do something else. The trick is the doing something else."

Leonardo da Vinci 1452-1519

Practical Tip 7.4

- *do not ever commit to a strategy, in a contract, by paying up front, or by 'getting top management approval' for the contract, until you can measure that it really delivers the **value** you are expecting it to.*
    - ***Value** both at the immediate level (like 'User Friendly'), and the expected knock-on effects (like saving training time and expense).*
- *Take the attitude:*
    - *we are happy to pay well for real value delivered*
    - *and we are not willing to pay at all if the expected value fails to materialize for any reason*
    - *and it is OUR responsibility to plan to get value for money. [URL6]*

Policy 7.4 Do "more of what works", Like 'Leo' advises.

- **We will monitor our project value delivery, early and frequently; and correct our course, as soon as we can, to get critical values delivered.**

Why?

- Because well-intended things fail, for obscure reasons, and it is irresponsible of us, as planners and managers, to allow those inevitable failures to ruin our final results.
- We cannot just declare the project to be 'complex', as an excuse for failure. We need to manage the initial complexity [URL87] by
    - decomposing strategy implementation into small value-increments (Chapter 5, Decomposition, [URL16])

- measuring the effect of each increment
- changing strategies to avoid failure, and reach our objectives on time. Like Quinnan in Cleanroom [URL12].

EDITED 3 NOV 2015 14:31

Section 7.5 Asking powerful questions about risk.

Principle 7.5 Bad Risk-Managers are the main risk

The management of risk
is primarily a management responsibility.

Of course, management needs to make sure 'everybody' deals with risks at their own responsibility level. Management leads by action and interest, stimulates, motivates, and takes responsibility.

When 'technical' projects go wrong, it is natural for non-technical managers to assume (hope?) that something 'technical' (outside of their own manager competence) has gone wrong. Even managers with a *technical* background, are constantly confronted with new technology, and larger scale, which effectively obsoletes their University Engineering/ Science training [URL43, Undergraduate Basics]. Not to mention that the new challenges often obsolete their personal practical experience of 5 to 20 years ago.

Well, I have concluded that the main problem in technology is management itself.

- If the management planning and decision-making is bad, then great technologists, reporting to them, will fail.
- If the management is good enough then, poor technology will be spotted early, and great technology has a fair chance to emerge.

The 'big idea' is that

- if management **focuses on the end results**, in the form of their own measurable and critical objectives,
- and uses these results as a *constant filter* on all technology,
- then the wrong technology *cannot* easily emerge or survive,
- and the right technology has a better *chance, an opening, to be discovered* and invested in, before it is too late.

At the very worst, when it becomes clear that no suitable technology is available, or known to us, to date, we can

- Change our over-optimistic objectives and budgets to be more realistic
- Give up early: give up seeking impossible objectives or impossible deadlines.

- Await improved technology in the future, or *provoke those inventions into our reality*, as the greatest of entrepreneurs seem to consistently do (Jobs, Edison, da Vinci, and Elon Musk come to mind)

We can plan systematically to uncover the problems early; before we have even wasted any time on experiments, or smaller investments.

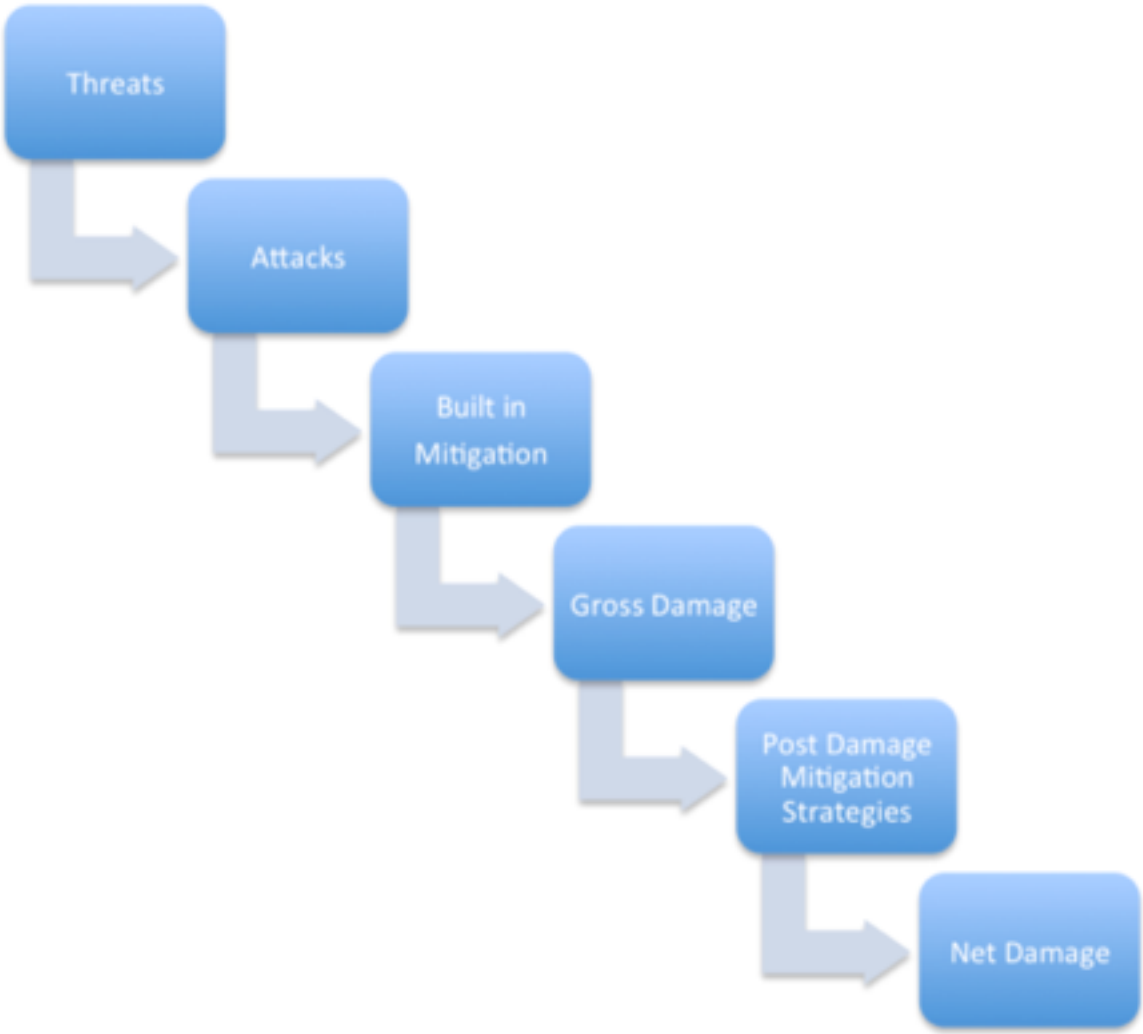


Figure 7.5 A. The risk management territory. Areas for managers to plan.

- *Threat Mitigation:*
  - *Strategies that reduce the existence of certain threats in type and volume*
- *Attack Mitigation:*
  - *Strategies that reduce the existence of certain Attacks in type and volume*
- *Post Attack Mitigation*
  - *Strategies that reduce the Damages of successful attacks to a system*

Let us re-introduce (Chapter 1.10) two Planguage 'tools', you can use as background information in your planning.

- The **Record level**: the state of the art, the best known.
- The **Trend level**: where we think things are going in the future.

Customer Service Availability:
Scale: % of 24/7 a customer gets a qualified answer without waiting or failing.
Past [Last Year, Our Main Service System] 95% <- Service report.

Record [Last Year, Our best competitor] 98% <- Their PR.
Record [Worldwide, Last 10 years, Similar Customer Service Systems to Ours] 99.98% <- Industry Surveys.

Trend [by Next Year, Based on Last 5 years, Our Main Service System] 93% ?
Trend [Next Year, Our best competitor] 99% ??

"Based on these benchmarks – what is a reasonable planned level?"

Tolerable [by Next Year, Our Main Service System] 99% ? <- Mkt Dir.
Goal [by Next Year, Our Main Service System] 99.5% <- CTO.

Planguage Example 7.5 A. Examples of the use of 'Record', and 'Trend' statements.

The information contained in the Record, and the Trend statements is a technical responsibility. Technical experts should know the answers. If they do not, they are NOT technical experts!

As a manager, your responsibility is therefore:

- Make sure that these *questions are asked*, and the answers are *integrated* into the future planning!
- Make sure you have access to good enough expertise to get reasonable answers to these questions (Trend, Record)
- Test the answers. Make sure they are not technical bluff. [Planguage Example 7.5 B]. Let the techies know you will not tolerate misleading and incompetent information. It is *your* decision basis, and you don't need 'Garbage In'.
    - Use the 12 Tough Questions! [URL15]
        - *'Would you 'bet your job' on those estimates of the Record level?''*
    - Do a systematic *review, audit, or Specification Quality Control* (SQC, more later) against sharp teeth *rules* like:

R1: All Trend and Record estimates will be backed up with specific Evidence, and Sources
R2: Uncertainty (±) degree will be specified for all Trend, and Record estimates.

    - Ask for evidence for the estimated numbers, and ask why it is <u>not</u> yet *written down, together* with the estimate. Let techies get used to the fact, that you *are interested* in the fact background, and expect to get it - in writing, *without* asking.
        - It is OK, for the moment, that they do 'not have a clue', and 'did not have time to investigate': as long as YOU are made explicitly aware of this (in *writing*, of *course!* Orally or implicitly is NOT good enough!)

Trend [Within 5 years, Our Biggest Competitor] 50%.
Evidence: none
Source: none
Credibility: none
Cause: no time to get data yet.

Planguage Example 7.5 B. Example of adding more information to a Trend specification, clarifying that it is a wild guess. And inviting something more substantial.

Case 7.5: The Power of a Powerful Question

- Tom was ('secretly') consulting for the technical directors of a very large and world-famous telecoms company. Monthly to Quarterly meetings, half a day. Their chosen agenda ("What can we do about quality?" for example). My job; come up with useful ideas for their problems.
- These same senior directors, 6 months earlier had been unceremoniously told off, in front of me, by their CEO, for having *listened* to my advice for 6 months, but *not actually tried it out* yet. They had 'excuses' for that, of course.
- The CEO said they *could not afford such delays* if the ideas were as worthwhile as they seemed to him. So, the CTOs were thereafter taking me seriously. In fact after every meeting, they really went out and *tried* my suggestions in practice, seriously! (Of course, the suggestions worked pretty well!) CTOs knew the CEO was going to be asking for results, and at least, serious evaluations of the improvement ideas. 'The stock market price was at stake , damn it' (he had mentioned).
- I had taught the 2 Older CTOs the '12 Tough Questions' [URL15] (by the way, have you tried the '12 Tough Questions' out in *your* meeting yet? What about tomorrow?).
- So at one meeting, a much younger, up-and-coming, Technical Director (with a PhD degree in Engineering) made a presentation to the 2 CTO's about a major new technology 'trend', that he recommended investing in. This would have impacted the work methods of thousands of employees.
- My 'students' (the 2 Older CTOs) decided to have a bit of fun, and asked the younger Director some of the 12 Tough Questions [URL15]. With me present, but silent.
- It turned out that the young PhD had *no serious basis whatever* for his recommendations. The senior CTOs immediately realized that there was a *very high risk*, for them, if they approved it.
- They were visibly angry, immediately, even with me there, and said that the PhD had greatly disappointed them. That they had hired him, and bet on him, hoping for trustworthy guidance, with substance behind it. They realized that they possibly had never gotten well founded recommendations from him, and had probably approved too much of his Fad ideas already.
- They sent him to Siberia. Well, a far corner of their empire. And it took about 2 years before they brought him back, in some role. (*He* did not become my friend, but the CTOs remained appreciative).
- I had given them a *tool* they could use in all such encounters, which must have been daily for them.

"I not only could not stifle controversy among your readers. I welcome it. This Administration intends to be candid about its errors; for as a wise man once said: 'An error does not become a mistake until you refuse to correct it.' We intend to accept full responsibility for our errors; and we expect you to point them out when we miss them."
John F. Kennedy (1917-1963)

Practical Tip 7.5

- ***keep on asking challenging questions until your employees get the idea that they need to research ideas better, before wasting your time, and before embarrassing themselves.***
- ***insist that replies to your questions are*** *documented* ***together with the ideas, with their source noted too.***

Mitigation:
are strategies designed to reduce quantity of threats, to reduce probability that threats turning into attacks, to reduce frequency of attacks, to reduce success rate of attacks, to reduce damage due to successful attacks

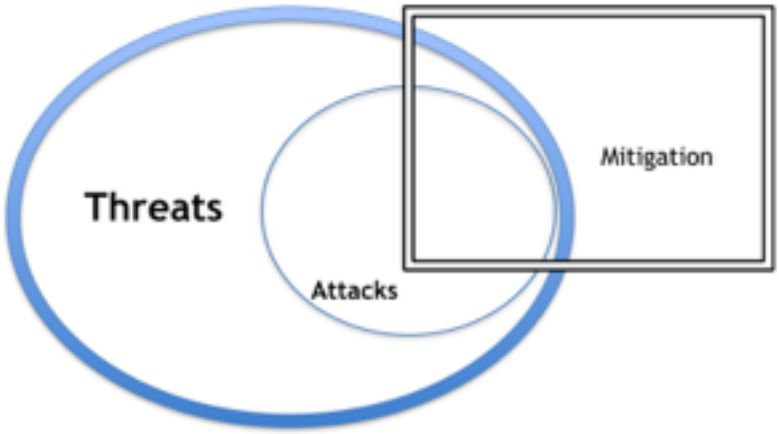


Figure 7.5 B. Three areas where managers can stimulate constructive thinking about risks:

- *Have we specified the main threats?*
- *Have we specified the main attacks we must counter?*
- *Have we planned or noted mitigation to reduce threats, and to counter attacks?*
- *Can anyone imagine threats, attacks and mitigations that we should be considering?*

Policy 7.5 The higher the manager, the more responsibility

- **Management is primarily responsible for making sure that strategies are credible, and tested, before betting heavily on them.**
- The planning processes will be such that, even bad plans cannot go very badly wrong, before we stop, change, and get better ones.
- This is a management responsibility: but it 'would not hurt' if power to do this is delegated to the technical and action levels too!
- But that *delegation* itself, is a management responsibility too.

- Why?
    - Our experience is that the techies will happily do stupid things, on a large scale, for years; as long as management is stupid enough to pay them to do it

Version: Edited 2 Nov 2015 17:43 tsg

Section 7.6 The holistic systems view of risk planning.

Principle 7.6 CONSTANT AWARENESS

"If you do not actively attack the risks they will attack you".
[5, p. 73]

Risks are, as we have defined here earlier, a combination of

- bad things that can happen (threats, which can become actual attacks), and
- strategies for coping with attacks; Mitigations
  - Mitigations that can prevent attack success
  - Mitigations that can recover from, or compensate for, successful attacks

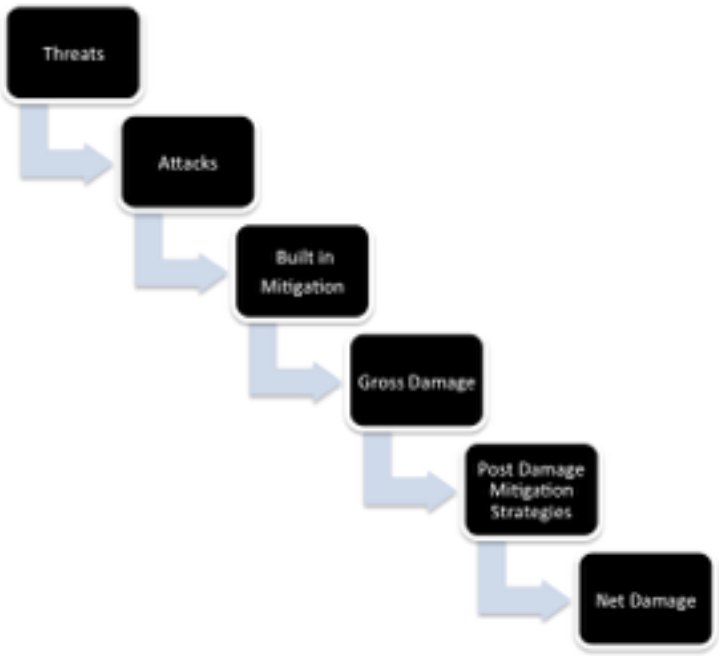


Figure 7.6 The sequence of 'Threat', to 'Net Damage', with two levels of mitigation. All of these factors become possibilities for management planning, to reduce damage. Same chart as in 7.5 A.

Here are the opportunities for the 'Value Planner' to proactively reduce the risks, and thus reduce the damage: thus improving competitiveness.

Let's start at the top of the chart and work our way down:

Threats:

- A planner can analyze (list, describe, prioritize) types of threats, and plan strategies (mitigations) for avoiding them, or for reducing them.

Threat Reduction:
Type: Risk Management Objective.
Stakeholders: Security Director, Marketing Director, Users.
Ambition: to cost-effectively reduce the threat types and quantity, by evasion, counter threats etc. Reduce potential.

Scale: % reduction compared to present levels for defined [Threats] and defined [Sources].

Past [Last Year] <various benchmarks, depending on Threats and Sources> <- No estimate available now.

Goal [Next year, Threat = Denial of Service, Threat Source = Eastern European Hackers] 90% Reduction <- Security Director.

Tolerable [Next year, Threat = Denial of Service, Threat Source = Eastern European Hackers] 60%-80% Reduction ??

Constraints: EU, and similar local Privacy Laws.

Potential Strategies: Pay Ransoms, Close-Downs, Legal Prosecution, ?

Planguage 7.6 A. An example of planning to reduce specific threats. The threat reduction objective. The strategies for doing this can now be planned and evaluated.

Attacks

- A planner can analyze types of attacks, and plan strategies for avoiding them, or for reducing them.
- Using the same type of planning as in the above example.

Built-In Mitigation

- A planner can plan strategies for dealing with the attacks which *actually* occur.

For example:

Availability Mitigation:
Type: Strategy for reducing unavailability of on line service, due to power or system failure locally.
Description:
D1. Triple geographically spread systems, coupled-in automatically with failure of main system, or also second system.
D2. Make local service staff mobile enough to reach secondary locations within 1 hour.
Impact: should bring us from 90% availability to 99.98% Guess <-TG
Cost: 3x single system cost, but 50% less than strengthening the local system against any attack.

Planguage 7.6 B An example of specifying a strategy for mitigating attacks.

Gross Damage

- A planner can use the Gross Damage, from successful attacks, as an objective, and seek to minimize it, using certain strategies. We can use the same structure as in the **Threat Reduction** objective above.

Post-Damage Mitigation Strategies

- A planner can identify strategies to reduce the Net Damage (an objective); things like recovery, insurance, compensation.
- Specified as a strategy, like the example above **Availability Mitigation**, except that the strategies will be aimed at recuperating for damage inflicted by the actual attacks.
-

Net Damage

- A planner can use the Net Damage, from successfully completed attacks, as an objective, and seek to minimize it, using certain strategies. Strategies that operate *after* initial damage is done.
- Net Damage will be stated as an objective, but the Scale may be in terms of some class of (after successful attack) damage, such as financial, reputation, or customer lost time. This will drive everything above; both reduction of threats, reduction of attacks, better system attack mitigation, and finally better Post Damage Mitigation (our 'last resort').
- In other words, this objective will look at the **overall chain of opportunities,** above, to reduce net damage; and will try to find a cost-effective set of strategies: as opposed to sub-optimizing, by looking at just one part of the above chain, such as attack reduction.
- Obviously Net Damage is a logical point to begin planning from. All the other objectives and strategies, are just one level, or another, of Means Objectives to minimize Net Damage.

"Take calculated risks. That is quite different from being rash."

4 Star General, George S. Patton (1885-1945)
(from Letter to Cadet George S. Patton IV (later Major General)
June 6, 1944)

"Never tell people how to do things.

Tell them what to do and they will surprise you with their ingenuity."
www.generalpatton.com/quotes/index3.html

Note the revolver "ivory gripped. Engraved, silver-plated Colt Single Action Army .45) ; that is risk mitigation. Not to mention the helmet, and 'swagger stick' with concealed blade! "He in addition frequently wore an ivory-gripped Smith & Weston Model 27 .357 Magnum on his left hip", presumably as post-attack mitigation.

Practical Tip 7.6

- *during meetings, try to snap-up risks from oral questions and answers.*
- *write them down immediately, and make sure they are integrated into the appropriate place in the planning documentation.*
- *For example:*
    - *Someone says "What if the supplier goes out of business within 10 years?"*
        - *Write, perhaps in the Strategy naming the supplier:*

Risk 23: Supplier can cease support within 10 years: <- TG, meeting 10 July 2015
Potential Damage: to Product Support.
Mitigation: Insurance, Alternative Supplier, ?

    - *Encourage others to do the same*
        - *and to make it easy for them, suggest that if they feed the notes to you, you will do the work of integrating them into the plan, with source credit for the observation, if they want it.*

Policy 7.6

- **We will consciously and explicitly plan to mitigate 'threats, attacks, and their damages', in the most cost-effective way.**

- In general, we expect the *best strategy efficiency* by attacking *upstream* (starting with Threat reduction or avoidance).
- We also expect that any attempt at obtaining a 100% effect, at any one level, will be impossible, or at least *not* cost-effective.
- So, we expect to look at the entire chain; from Threat to Net Damage as one single problem to be solved in the most cost-effective manner.
- We must certainly first recognize any scalar constraints (Tolerable, OK), and other constraints (legal, cultural, stakeholder pressure and expectation) as a minimum set of requirements before we optimize the total.

Why?

Smart planning. Cost-effective use of your time.

Version: 3 Nov. 2015 18:51 edit

Section 7.7 Spec Quality Control to Reduce Specification Defect Threats: 'very upstream' risk management

Principle 7.7 Devil In Details.

There is potential risk,
and potential risk *avoidance*,
in *every detail* of your planning
of objectives and strategies.

Any single planning specification detail, can be the cause of a far greater loss of time and energy, than the trouble it takes to write that specification in a disciplined, responsible way.

There will always be the need to strike an economic balance regarding how much investment, in good-enough specifications, we need to make at the planning stage, and how much grief we should be willing to accept later, due to sloppy or frugal planning. It is reasonable that your organization should be able to experiment with, and evolve, sound quality control (QC) practices through time.

However our practical observations, amongst our clients; and the very high incidence of failed and delayed projects, well-documented internationally, suggest that we have a very long way to go in improving practices of planning. Maybe it is time to go a bit 'overboard' in thorough planning practices, and get a feel for the effect of *that*, compared to your norm?

Refer back to the NASA experience, (Figure 2.2, repeated below for convenience). Clearly there was a substantial additional investment in Requirements, more than what many projects thought was good enough (3% to 14% of total project cost). But there was also a point where it did not seem to pay off to plan better requirements any more (20% investment on)

But the projects who thought they 'saved' 6% to 17% project effort in doing requirements planning, actually lost 30% to 180% in project cost overrun. Bad economics. Who is responsible? (Hint: not the planners)

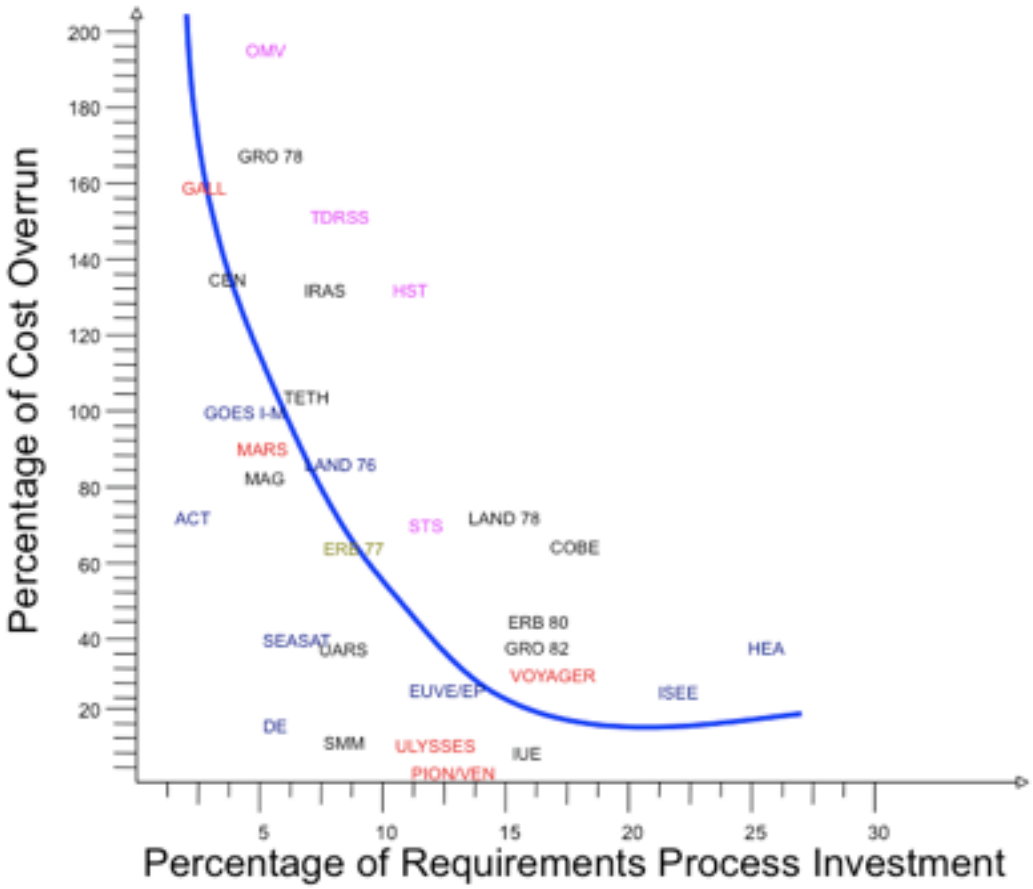


Figure 7.7 A (same as 2.2) The Gruehl Chart, NASA. [F4]. These data are not your data, but you could collect comparable data. We believe you will find the same principle. It pays off, more than people think and practice, to invest in rigorous quality and detail, on planning and specification, up to a point.

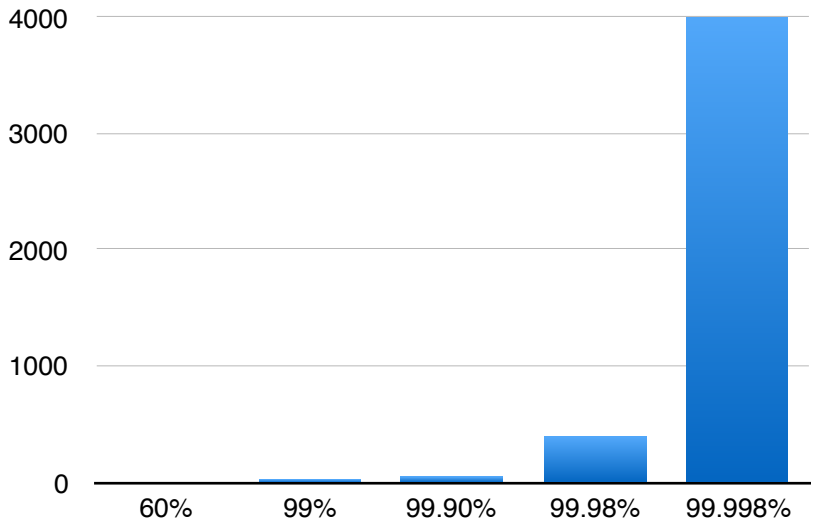


Figure 7.7 B A rough illustration of the relative development costs of higher quality levels, such as system availability. As we approach perfection, costs tend towards infinity.

CASE 7.7 A. Two projects, but; only *one* did heavy *planning* investment

We were discussing the need for up-front planning with a CTO, Thomas, at a large telecoms supplier. They had actually taken the need to learn from their big project experience seriously. So, they really assembled the teams after the project, wrote a small book on project experiences, and made those books available to all who were interested, including me (an outside consultant).

Thomas said to us, *"You do not have to convince me that it pays off to invest in up-front planning. I have read the Project Experience book for one project, which invested heavily in the requirements phase; even until very close to deadlines for delivery. People were really worried that it would ever move on, and deliver results. But it did, very successfully.*

By contrast there is a similar project, under my wings, unfortunately, that 'simplified' the requirements process. It was keen to get on with the 'real engineering' of the product (Mobile Base Stations). I don't think they have delivered yet, and they are very late."

Our experience, backed up by the chart and Case above, it that this is an issue that *management must manage*. Clearly, left to themselves, and probably under management deadline-pressure, planning teams will *not simply do the right thing*; even if they *personally* feel it is the right thing!

This means that management, if you want your planning organization to do the right thing:

- Needs to do the **research** needed to determine the right degree of investment in planning (like NASA)
- Needs to make sure there are **standards** for what the right thing is (Rules, Processes, Quality Control, Process Exit Limits) [1].
    - These can be evolved by the planning staff themselves.
    - But initially a good draft standard needs to be put into place, or recovered from your dusty archives.
- Ideally, there would be **training** in the 'new' standards.
- But, we have found (at Boeing, Ericsson, Intel and others) that if there is Specification Quality Control [URL23, 1 D, about SQC] seriously-sponsored by management, the technical staff will quickly learn how to follow the standards, even really good old standards (in the Case of McDonnell-Douglas) which had been ignored under time pressure.

Here is an outline of what you need to do:

- The SQC process, cheaply (< 1 hour), measures the defect density (defect = violation of a required specification standard) of every plan component. Potentially you do SQC for every larger (about 1 page or more) objective and strategy.

Status: SQC Exited. The Level < 0.25 Major Defects/page

Planguage Example 7.7: of a statement regarding the level of defects in a plan.

- The Exit level standard says, '**no exit**' if there is more than, say 1 defect/page.
- (Note that most people are at 100 defects per page, now!)
- Most people *will* learn what to do, and how to do it, quickly, at a rate of 50% learning-improvement per feedback cycle.

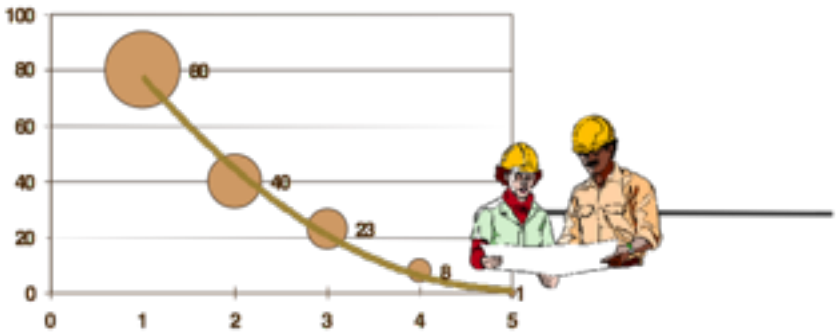


Figure 7.7 B. Typical (real case) Gary at McDonnell-Douglas [URL44] (Boeing).

Case 7.7 B. Motivating people to confirm to best practices, in practice.

Gary, an aircraft engineer, is motivated by a new management demand to have 'no more than 1 defect (standards violations) in released (process exited) work'.

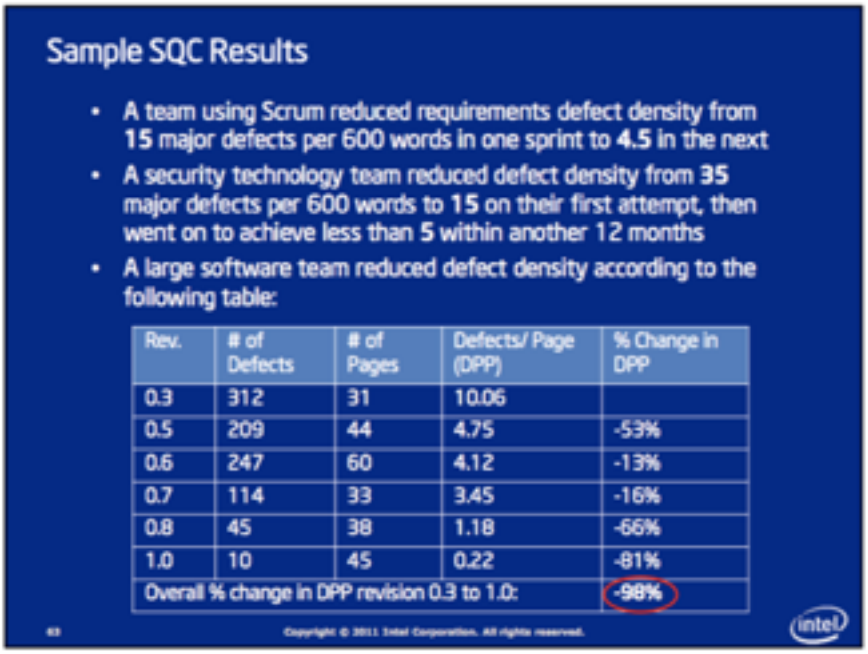
Gary delivered his usual 80 defects per engineering drawing. 80 times over the 'drunk drawing' limit. See Fig. 7.7 B above.

He was motivated by peers, to learn. He got 50% better at every cycle of submitting to SQC, until he reached 'under 1.0 defects', the exit level.

All engineers (hundreds were measured in the first weeks after we started this) managed to meet this standard.

The resulting manufacturing capability, the knock-on effect, was also seriously measured and remarkably better. [URL44]

Case 7.7 C Intel reports similar results with our methods, after long term use:



Sample SQC Results

- A team using Scrum reduced requirements defect density from **15** major defects per 600 words in one sprint to **4.5** in the next
- A security technology team reduced defect density from **35** major defects per 600 words to **15** on their first attempt, then went on to achieve less than **5** within another 12 months
- A large software team reduced defect density according to the following table:

Rev.	# of Defects	# of Pages	Defects/ Page (DPP)	% Change in DPP
0.3	312	31	10.06	
0.5	209	44	4.75	-53%
0.6	247	60	4.12	-13%
0.7	114	33	3.45	-16%
0.8	45	38	1.18	-66%
1.0	10	45	0.22	-81%
Overall % change in DPP revision 0.3 to 1.0:				-98%

Copyright © 2011 Intel Corporation. All rights reserved.

Figure 7.7 C. Reported defects per 600 words page in product requirements. [URL46, URL45 Terzakis] . We used this same table initially in Section 3.8

In the table above we see that the initial submission of requirements, using Planguage, is 10 defects / 600 words.

Under pressure to get down to a (NASA level) 'release' or 'exit' level (0.25 defects/600 words) they keep on 'learning' their corporate standards. The defect reduction is 98%. The productivity result is over 200% [URL46] see just below.

Downstream benefits of SQC at Intel in Terzakis case (the table above):

Scope delivered at the Alpha milestone increased 300%, released scope up 233%

SW defects reduced by ~50%

Defects that did occur were resolved in far less time on average

"This stuff works"

Erik Simmons (below), Intel, referring to Gilb Planguage, SQC and Evo in 'Competitive Engineering' book [1] Foreword

Practical Tip 7.7

- ***Use at least 30 minutes to do at least a 1-page sample Spec QC of requirements specifications.*** *Chapter 10, and [URL23]*
- *Reflect on the damage it will cause to proceed, unless you have very few defects.*

- *Hint: you can expect at least 10x more future costs, than the immediate costs of fixing the problem now.*

Policy 7.7 'Quality First' gives 'Quality Finally'.

- **We will not leave our project success to chance and deadline pressure.**
- **We will set high, and profitable, quality standards for planning; measure conformance continuously; and release plans only when they meet our quality Exit levels;**
- **so that we have a fair chance of meeting our deadlines and budgets.**

- Why?
  - Because planners need to be given clear and strong guidance, by management, about the quality of planning that the organization needs.
  - Planners need to be motivated, by management, to get planning quality right, and never be pressured to compromise short term, at the expense of the project's longer term fate.

Version Nov. 3 2015 tsg edit.

Section 7.8 Fighting the root causes of risks is good economics.

Principle 7.8 Root Out Risk.

**Better to fight the root causes of risk,
than to get a 'stream of late disappointments'.**

You cannot identify in advance, and directly plan to mitigate, *every* potential threat.

There are something close to an infinity of potential threats, and more will come into existence while you are planning: and even more *after* you have launched your new project's system.

So, forget such details, all 100 billion + of them.

Your only chance to do something useful about threats is to focus on '**root causes**' of risk, and deal with them.

Even then you are doomed to miss some root causes initially, and will have to recognize them after they first emerge, in testing, value step implementation, and operation of your system.

'Root causes' are the ultimate upstream reason why an error/defect/fault/ malfunction/threat/attack sequence occurs.

	->	->	->	->	->	->	
Planning	Error (in Planning)	Specification Defect	Spec QC	System Defect	Malfunction or System Failure	Measure Value Delivery	Net Damage
Security	Threat	Attack	Mitigation (Built in)	System Defect	Damage (Gross)	Post Damage Mitigation (Recovery)	Net Damage
Program -ming	Coding Error	Code Bug	Inspection	Bugs Remain	Bug Encountered	Remove Bug	Net Bugs Remaining
Pattern	Fault	Fault	FIX	Fault	Fault	FIX	Fault

Table 7.8 The planning chain of areas of threats and mitigation, compared to 2 similar security planning and software reliability planning scenarios. © gilb.com 2014.

This table gives a *list over places* to look for 'fault' root causes, and suggests some downstream places to stop their flow to further downstream .

For example: if there is a Security threat, we can seek its root cause (example 'known attractive rich customer data') and deal with the root solution (for example 'protecting the customer data') rather than the myriad of potential threats from all sources, and all times.

Why is the 'root cause' interesting?

- Because there are fewer of them to tackle
- Because by fixing the root cause, you in fact fix many related problems, including ones you do not even know about!
- Because if you fix anything less than a root cause, you are not defended against *change in the nature* of the threat; and especially not for changes from adversaries (enemies, competitors) who will try to get around your defenses (by specific mitigations).

The distinction between the common cause and the special cause.

Dr. Juran (Quality Control Handbook, Juran Institute) [15] made the point in one of his many books that the most significant single thing he learned from his teacher (also Deming's teacher), Walter Shewhart (originator of Statistical Process Control) [16], was to focus **management** attention on '**common causes'** of problems which are *inside* the set natural variation control limits (problems which occur repeatedly and are due to a single cause; they are well worth fixing).

This is where we can set objectives to make the system better overall, a better average performance, and less variation (more predictability) around that mean.

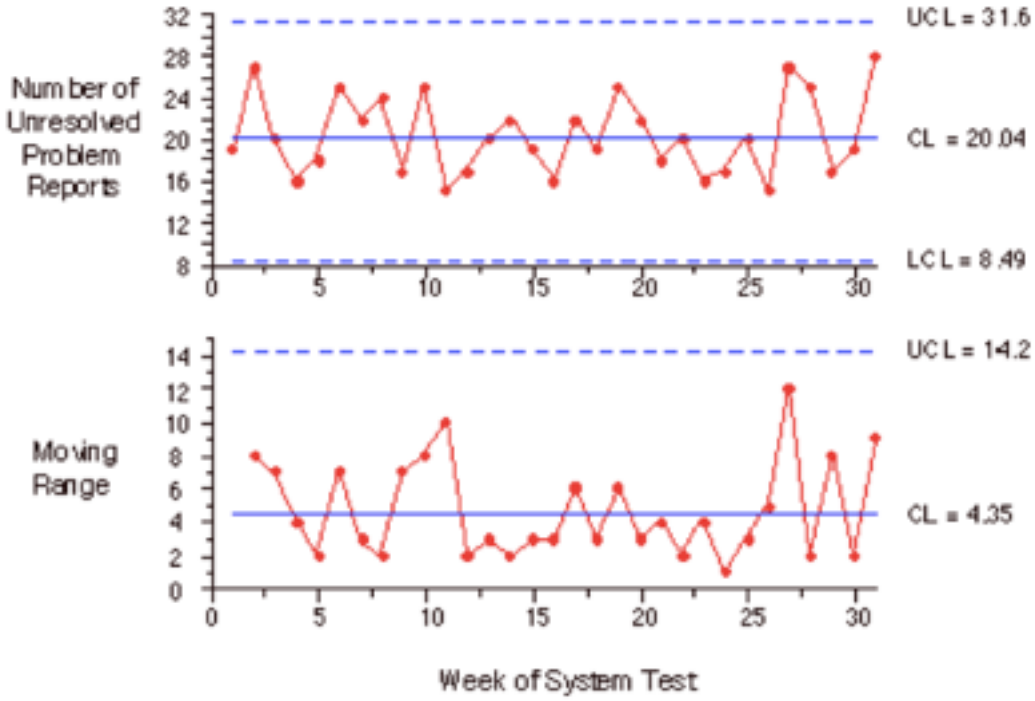


Diagram 7.8 A Statistical Process Control Charts [URL47]. This is the natural and accepted variation of a system around a mean of 20. The lower chart shows improved performance (average 5) of the system. This is, hopefully, due to management objectives supported by management strategies, to get better.

On the other hand, any measurements outside the established control limits, need to be handled without special 'planning'. They are called '**special** causes' (or assignable causes). The remedy (if indeed it is worth dealing with) is *local or 'one off'*. For example *inform an individual better*, *replace a faulty component or part (like a battery)*, *improve room temperature for people*.

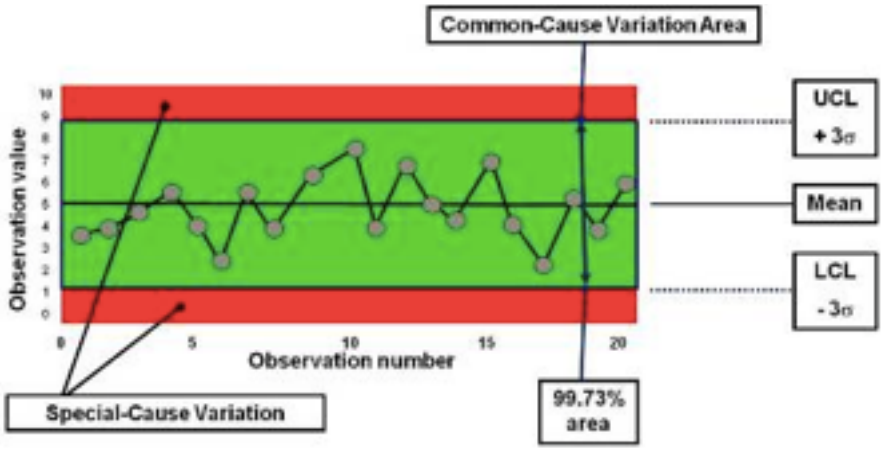


Diagram 7.8 B Variations, outside the acceptable performance range, the Upper and Lower control limits, are deemed due to 'special causes' (i.e. they are not normal system variability). They need to be tackled as one-time fixes, and tackled locally: on the shop floor, not by 'top' management: not by strategic planning. By supervisory management, by local teams, by the individual concerned. Source: *http://www.free-six-sigma.com/images/2.2-6SPC-Chart-Explained.jpg*

> ***"It is top (strategic) management's responsibility to address 'common cause' variation, and therefore it is management's responsibility to make improvements to the whole system."*** *(Deming interpretation, Wikipedia)*

Causes which activate problems rarely, and are outside the set control levels (outside 'normal' and 'acceptable' variation), are called 'special causes'.

> ***"Because 'special cause' variation is 'assignable' (to a specific cause), workers, supervisors or middle managers that have direct knowledge of the assignable cause, best address this type of specific intervention."*** *(Deming interpretation, Wikipedia)*
>
> "

This is very good advice in our search for a way to do some general planning, efficiently (value/money). It is also good advice on how to handle Threats that are 'unknowns' or the 'economically unknowable'.

- *Management* needs to focus on improving the long-term performance, including *reduction in variation* of performance. That means, making the results more 'even', more 'predictable'.

- All other problems, outside the scope of the norm, need to be dealt with locally, on a case-by-case basis, if at all. They are not a cost-effective area to do 'management planning', to make the system *generally* better.

- The interesting **threats** are the ones that *recur*, and which determine both the average performance level, and the variation degree from that level.[F5]

- If we can eliminate them, we have high leverage. We eliminate the source of bad performance in the longer-term future, not just one sleepy employee. The smartest way to deal with these Common Cause Threats is to find their *root* cause, not just their symptom.

So, how do we identify the *root causes* of threats, and what can we do about them?

- Look at a threat, or an attack and ask ***why*** it is there. Ask a series of *why* questions, until you sense you have a root cause which can be dealt with by your strategies, or initially, by your objectives,

Figure 7.8 A. a simple example of seeking a root cause using a series of 'why' questions.

A very interesting *combination* is a ***root*** *cause*, which is at the same time a ***common*** *cause*. This is likely to give a very good return on investment.

Objective 1:
Scale: Market Share in defined [Markets]

Goal [Next 5 Years, Market = {Europe & Americas}] 30%

T1: Threat: Market Downturn.
C1: Cause [T1]: International Economics.
C2: [C1] Cause: Power Struggle in East Europe.
M1: [C1, C2] Mitigation: Shift Investment Focus to South America, and Upmarket.

Planguage Example 7.8 a simplified example of the principles of working out a plan to deal with Threats, with logical cause chains. Once some root causes are suggested, it is easier to think of credible strategies for dealing with them. Since we believe we cannot directly affect the causes, we have to avoid the threat, by moving elsewhere.

*Figure 7.8 B Threats can have either a common cause, **or** a special, (assignable) cause. The built-in mitigation strategies need to focus on the common cause attacks, and at best, report the special cause cases to 'operations' for local one-time fixes. Similarly, post-damage mitigation strategies need to be primarily planned for the more-frequent stream of common-cause attacks, which cannot be contained by the Built-in mitigation investment strategies.*

Here is what we have to analyze: a *planning agenda*:

- **Threats**: plan to reduce them, or avoid them
- **Attacks**: plan to reduce them, capture and get rid of them before they can do damage (using Built-In Mitigation strategies)
- **Built-In Mitigations**: estimate their expected strengths (Impact Estimation) and follow up, by measurement, to make sure they are as effective as planned. If not, improve or change them fast.
- **Gross and Net Damage**: set specific objectives for Damage containment or reduction, with constraints and targets. Find strategies to get to these levels, and if that does not work, find *better* strategies.
- **Post-Damage Mitigation**: assuming some damage will be done; plan strategies, which can be activated *after* damage is done, if necessary. Or have strategies built-in immediately ready for action, for reducing the damage to critical objectives, time, money, and reputation.

The common question that gets asked in business is, 'why?' That's a good question, but an equally valid question is, 'why not?'
Jeff Bezos

Practical Tip 7.8

- *ask why, ask again*
- *treat a root cause*
- *that's good economics, or leverage*

Policy 7.8 Fight threats: upstream and generally

- **we will fight 'common cause' threats, by *planning*, and we will try to get maximum leverage in the uneven fight, by working from upstream first, and looking for *root causes* and *common causes, not special causes,* to fight with general strategies.**

Why?

- It is a good basic investment, so you don't get too overwhelmed by problems,
- so you then have capacity to start dealing with all the *other* problems that will come along later

Version: Edit Nov. 3 2015 19:39 Kolbotn tsg

Section 7.9 The value of overwhelming value

Principle 7.9 Overwhelming Value

Try to overwhelm
unavoidable disappointments,
with overwhelmingly successful,
cost-effective,
value delivery.

You are going to win some, and lose some. Your strategy for avoiding a big loss, and for getting a big win is disarmingly simple. Sort of like the military commander who knows that his attacking force has to be 4 times the defending force, to win. You have to get a cumulation of wins, at low cost, while you avoid big losses, by detecting bad strategies very quickly, with minimum loss or commitment involved.

We have discussed many of the basic ideas, for delivering great value, earlier: here is a summary.

- **Focus on the top-level critical 'few' objectives**
  - So when you win, you win something *important*
- **Pick strategies that have a large number of good impacts on many critical objectives** (using IE Tables) **at the same time**. Like a smart chess move.
  - So that if there is disappointment in some areas, there is some good news in the impact on other objectives.
  - So that the total sum of effects is a large value
- **Prioritise immediate short-term, next week, value delivery steps, so as to get the highest total value for the time and resources available**.
  - Get off to a flying start with reasonable results, at reasonable costs.
  - Avoid using resources without *some immediate* value to show for it, as opposed to no value whatsoever - which is embarrassingly common. 'Value delivered' over 'work done'.
- **Pick the strategies with best values-for-costs in the 'worst-worst' case.**

  - When estimating value, use the Impact Estimation method of risk management. Consider ± uncertainty, and Credibility
  - 'Worst-worst case' means the lesser estimated value in the ± range, *times* the credibility factor (60±20 = 40 x 0.5 = **20** <- 20 is worst-worst case). The *most conservative* estimate.
- **Make sure your supplier contracts are based on value for payment** ([URL6, No Cure No Pay].
  - Make sure you cannot continue to 'burn' money if things don't work well for you. Slam on your brakes, to avoid wasted resources.
  - Do NOT contract for *work done*, or *systems delivered*. *Contract for the results* you expect; both incrementally, step by step; and cumulatively in the long term.
  - That includes the idea of contractually binding a supplier to long-term maintenance, at a predictable cost.
    - So they do not get too tempted to do sacrifice long term *maintainability characteristics* [URL28, A B, 1 Chapter 5 'Adaptability']. Which you need to keep in your top 10 critical objectives anyway!

Case 7.9 IBM Santa Teresa Labs

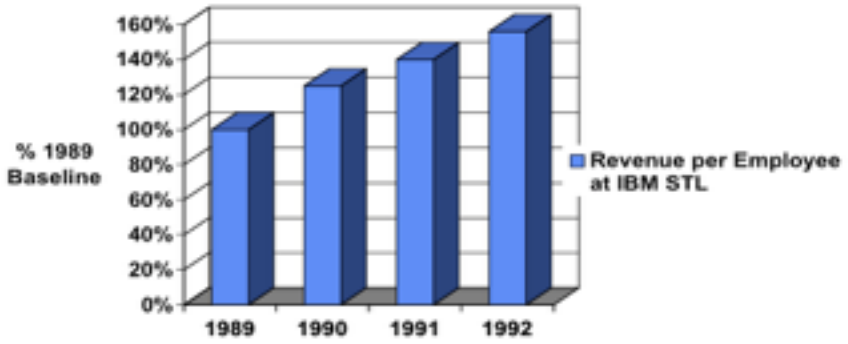
IBM Santa Teresa Labs, San Jose CA, (one of our clients, some years earlier) was rated one of the worst of IBM Labs.

The leadership from the best IBM Labs, Minnesota was sent out to sort them out.

The case study book of this organizational turnaround [17], 'Secrets of Software Quality, by Craig Kaplan et al, is wonderful in its detailed stories of the implementations (the blood, sweat and tears) of a large number of strategies (40, Table 7.8 E), to improve their top few (4 of them in charts below, Tables 7.8 A B C D) critical objectives.

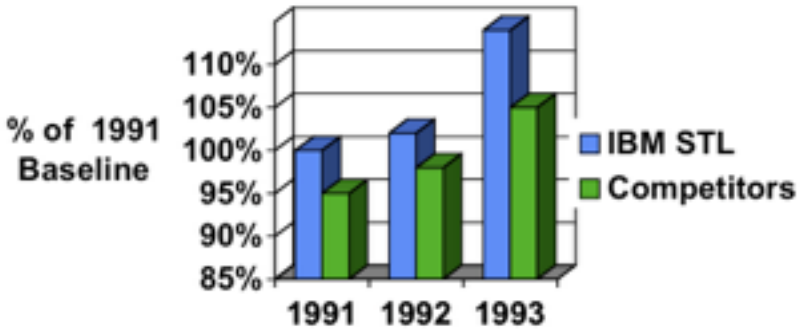
The year-by-year incremental improvements, as a result of the strategies, is shown in the tables below.

56% Increase in Revenue per Employee

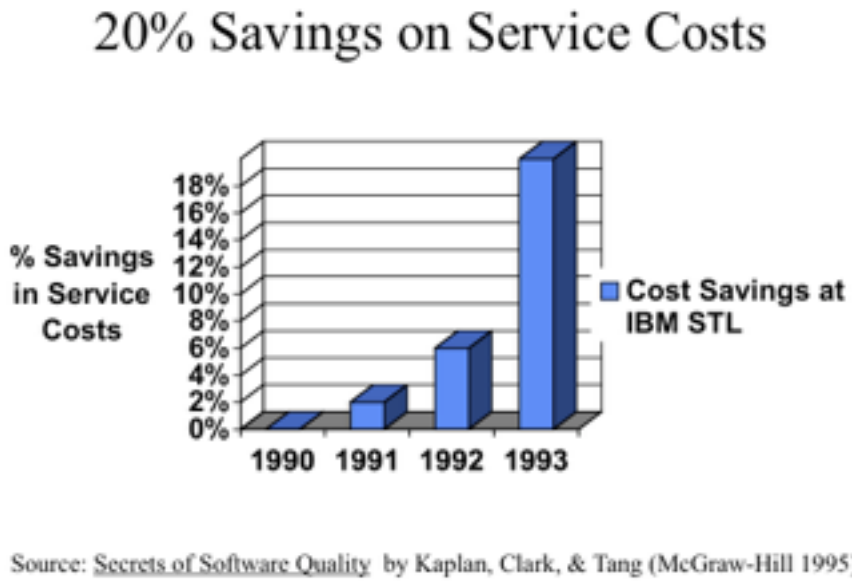


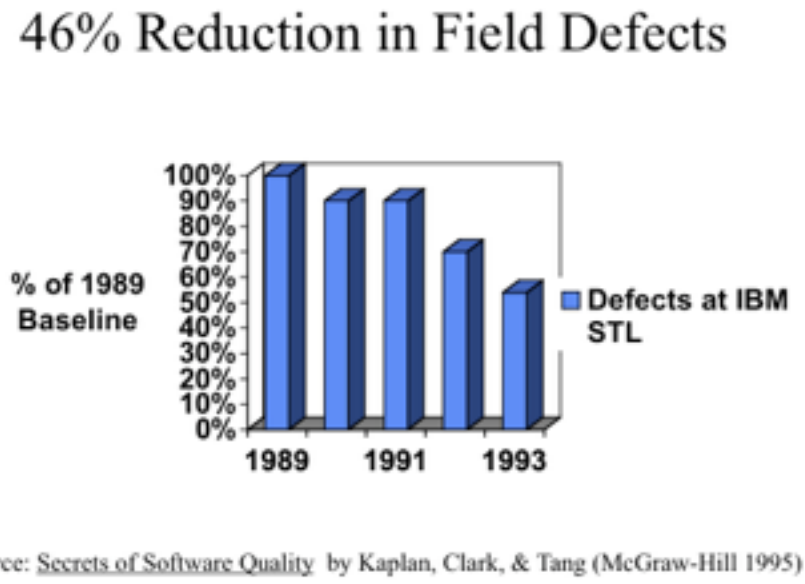
Source: Secrets of Software Quality by Kaplan, Clark, & Tang (McGraw-Hill 1995)

14% Improvement in Customer Satisfaction



Source: Secrets of Software Quality by Kaplan, Clark, & Tang (McGraw-Hill 1995)





Tables 7.8 A B C D The incremental results of turning around an organization using a number of strategies.

40 Innovations From IBM

Awareness	Coping	Management	Integration
Dept. Q. Strategy	Ctr. SW Exellence	Strategic Focus	Code.Int. Reviews
Excel. Council	Mgmt. Council	Empowerment	Workforce 2000
Q. Publications	ISO 9000	Quality Week	Q. Exchanges
Seminar Series	Code/Design Insp.	Comp. Understand	Q. Cust. Partner
Leadership Inat.	Early Test	Proc. Benchmark	Q. Supplier Partner
Extend Unit Test	C.L.I.F.T.	Defect.Prevent.Pr.	Perform. Mining
Dev. Handbooks	Error Prone Mod.	OO Design/Code	Orth.Def.Class.
Cust. Sat. Survey	H.A.R.M.	Knowledge MineCs	Q. Ret.On.Invest.
Joint Appl. Dev.	Cust. Sat. Linkage	Electronic Meeting	
Process Modelling		Online Code Rev.	
		LAN Lib. Control	
		Rapid Prototyping	
		Clean Room Tech.	

Table 7.8 E. These are the 40 strategies IBM used. Half are technical, half are social.

These (Table 7.8 E) are the 40 strategies IBM used, and to a considerable degree invented.

The Cleanroom process [URL41], and the DPP Defect Prevention Process [URL40], discussed earlier in this book are among them.

But, the interesting lesson here is the necessity of using *management* strategies to support the *technical* strategies.

These gave the results in the charts above. The book [17] 'Secrets of Software Quality' is one of the best studies I have ever encountered in terms of the depth of information behind the struggle to implement each one of these 40 strategies.

Notice the very large number of organizational, motivational strategies, mixed with the technical strategies. The 4 columns (Awareness, … Integration) are stages of maturity on the USA 'Baldrige Quality Prize' Scale, the USA Version of the Japanese 'Deming Prize'.

"I think it's very important to have a feedback loop, where you're constantly thinking about what you've done and how you could be doing it better. I think that's the single best piece of advice: constantly think about how you could be doing things better and questioning yourself."
Elon Musk, Co-Founder/CEO Paypal, Tesla Motors, Inc. [F6], Space X, Solar City

Photo From a TED Talk Video

Practical Tip 7.9 Impress Early

- *in your early project steps, the first few weeks, look for things you can do which cost practically nothing (a 'weeks effort' maximum) but which are 'guaranteed' to deliver measurable value: at least using early indicators.*
- *you can use Impact Estimation tables to spot such opportunities, with less than an hour or so of effort*
- *the point is to build your confidence and credibility at an early stage.*

Policy 7.9 High average value for resources

We will aim for an *early*, high, stream-of-value-for-resources, by
being conservative, cherry picking, and cutting losses.

But the stream of value will always be in terms of, incremental achievement of the long-range critical objectives. Not by means of sub-optimization, not sacrificing the long term, to satisfy the short term lust-for-results.

Why?

- To discipline ourselves to be very results oriented, very efficient, very total-systems thinking.
- To prove that we can achieve results, early and consistently.
- To be able to absorb small losses, as we experiment with promising new unknown-risk strategies.
- To learn as fast as possible; what it 'really takes', to deliver real results in our environment.

Version: edit Nov 3 2015 19:52 Kolbotn tsg

Section 7.10 Focussing on overall value progress.

Principle 7.10 Accentuate The Positive

Reward and focus
on net success,
do not punish
small accidents.

If managers keep their focus on the 'big picture', meaning 'the progress towards their top ten critical strategic objectives', and make sure that everybody knows that is where their focus is, then people will feel less afraid of trying new and promising ideas.

People should be quite clear that they will be judged on their *success in moving forward* to the major objectives. They will *not* be judged on failed experiments along the way.

Many wise observations are about the fact that if you are never failing, you are not trying hard enough.

"Far better it is to dare mighty things, to win glorious triumphs, even though checkered by failure, than to take rank with those poor spirits who neither enjoy much nor suffer much, because they live in the gray twilight that knows neither victory nor defeat."
Theodore Roosevelt (1858-1919) American President, in "Strenuous Life"

Here are some tools in Planguage which can help managers, planners and implementers **to 'dare' and not be stopped** or slowed in their pace towards the strategic objectives:

- The main tactic is the we implement the objectives in a **series of small steps** [URL18]**.** Nothing *big* can go wrong:

  - *if* you make sure you have **not** made long-term irrevocable commitments. For example with suppliers or partners, or by removing older systems too early.
- The **prioritization evaluations** (Chapter 6), discussed extensively earlier, with **worst-worst case num**bers ('60±30 = 30 x 0.5 credibility = 15'), and **highest value for resources** ('15:1') that is available, means you have less fear of big waste, or result disappointments.
- **Contracting for results**: don't pay until results are on the table.[URL6].
- **Specification Quality Control of plans**: use 10 to 60 minutes to examine your plans according to some rules [1, sample rules are in most chapters].
  - The many small mistakes people make initially in the plans can be cleaned up, or prevented from happening in the first place, for 1% or less of the cost of you moving downstream, with bad plans.
  - People do write 100 unintelligible things per page, so attacking that early, is going to save you from much bigger problems, caused by the misinterpretations of your ingenious plans.

Case Study 7:10: Product planning for Production in China, Major International Manufacturer, at European Research Center.

- Our client, a European subsidiary of a famous multinational electronics company, gave Kai and Tom a 'vision and scope' document for a product.
- Tom predicted , before seeing the documents that they would have about 150 Major defects per page or 300 words. Because that, in our experience, is normal.
- We agreed on the following specification Rules, which are very basic ones.

1. CLEAR: Clear enough to test
2. UNAMBIGUOUS: Unambiguous to Intended Readers
3. NO DESIGN: No inadvertent (accidental) design

Rules that we would use to decide what a 'defect' was.

We also worked out together who the 'Intended Readers' (Rule 2 above) were, for this plan:

- **Tester in China**
- **New Company Employee**
- **Subcontracted developer and Tester**
- **Marketing – Product Marketing**
- **Product development teams**
- **Product quality teams**
- **Company & Litigant lawyers**
- **Partner auditors**
- **Etc., many more, you decide sometime (for each class of document)**

List 7.10 A: Intended Readership List.

The client people then checked their plan against these rules, with these intended readers in mind, for about 20 minutes.

They classified all violations of the rules as either

- **Minor** (cannot ever hurt us anyway)
- **Major** (can possibly translate to product, and can hurt us (economically, time wise) if bad interpretation.

List 7.10 B. Definition of Defect Seriousness in the plan

We agreed, in principle, that such a critical planning document should not be released (should not 'exit' the planning process) with more than 1.0 major defects per 300 word (page).

Here is the text they checked: (for an advanced large scale printer)

"**Media Input.**
All media rolls, sizes and formats need to be loaded from the front, including heavy weight rolls up to 10 kg. In those "close to the limit cases" (weight or size), an intermediate support may be used to facilitate the roll loading process. Media load experience will be similar for both rolls.
Media information tracking using printed barcodes needs to be supported by the printer.
Reliability goals for roll switching will be defined in chapter 4.7.
Media cannot be loaded while printer is not in idle state (error, printing,...)
Roll switching time will be less than 20 sec.
Roll switching time will be as in GW ++. No d-skew algorithms should be used between roll switching except for the first load."

The plan, looks innocent enough, and is typical. It is 127 words.

How many unclear words can you find? How many strategies, instead of the intended objectives and vision do you find?

Our 2 checkers, found:

- Victor: 24 major defects
- Carlos: 32 major defects

Note that this is enough, by far, to reject release of the document. But, it gets worse.

We did a little extrapolation: [URL23] to get a more-realistic picture

- If the team had been larger, 4 people, this initial data from 2 checkers, indicates that the 4 checkers would have found, together about 44±15 unique Defects.
- 44 would be 1/3 of the defects that *actually exist* (such checks are only 33% effective)
- so, that indicates there are 44 defects (would be found by team of 4+88 defects (un-found) = **132** ±30 Major defect.
- Much more than the initial defect identification, much worse.

We found that his defect level was consistent with several other planning documents at the same Research Centre.

The client agreed (difficult to disagree with yourself) and continued working with us, to rewrite the plans, to remove *almost* all the defects.

Which we did.

And here's another idea to keep you from being scared of failure:

- **Brag about your failures**: a weekly blog or presentation: "Our most interesting misconception, and what we learned and can tell you about".
- "Popular myths and fads that do not really work: and how we proved it, before committing too much."

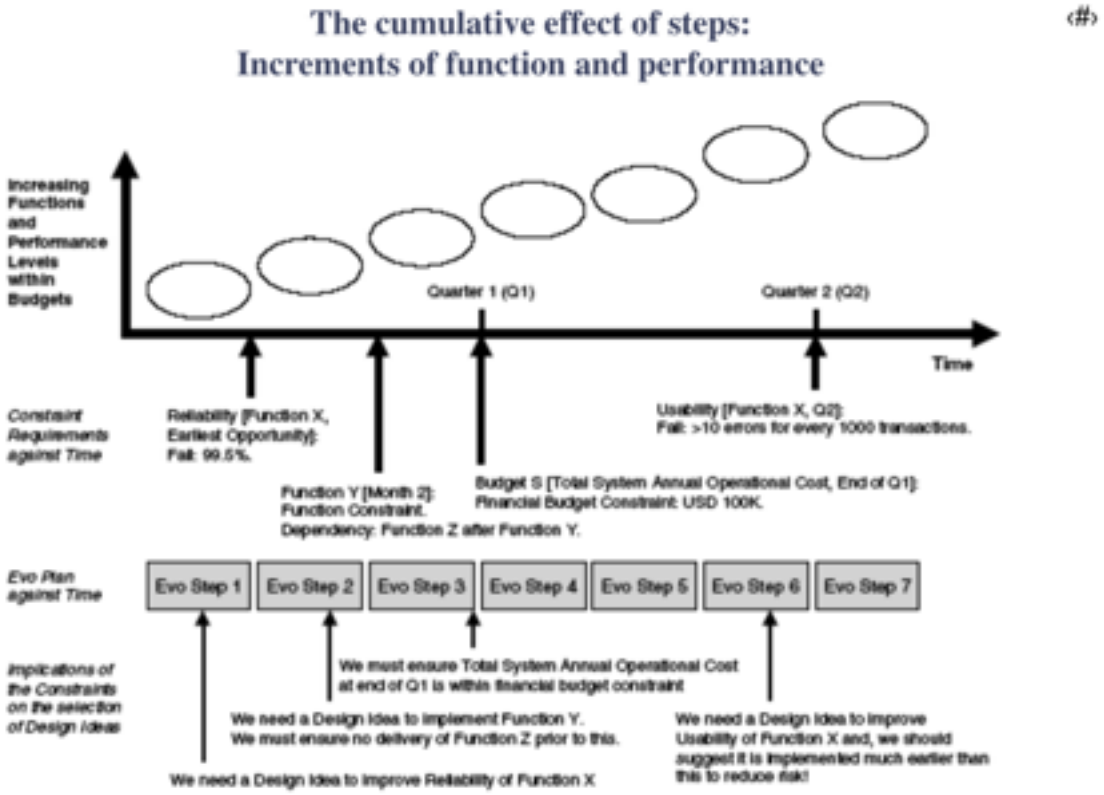
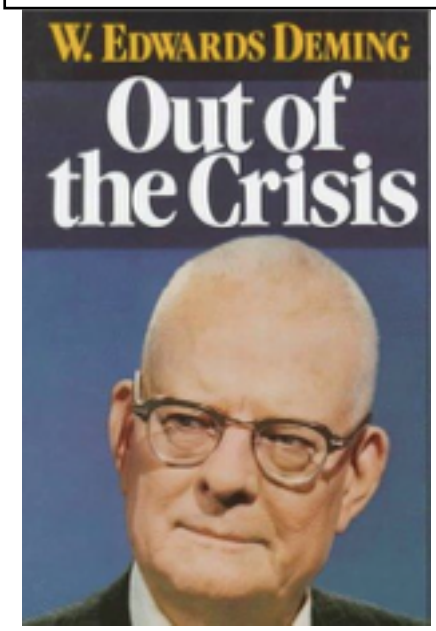


Figure 7.10 A hypothetical series of early value delivery steps. Some steps enhance functionality. Some steps enhance qualities like reliability and usability. Some steps deliver future economic performance. There is a mixture of bets here, and visible measurable process to a real system or organization. Source [1]

"You can expect what you inspect."

W. Edwards Deming, 1900-1993



"Deming emphasized the importance of measuring and testing, to predict typical results. If a phase consists of inputs + process + outputs, all 3 are inspected to some extent. Problems with inputs are a major source of trouble, but the process using those inputs can also have problems. By inspecting the inputs and the process more, the outputs can be better predicted, and inspected less. Rather than use mass inspection of every output product, the output can be statistically sampled in a cause-effect relationship through the process."

[from Wikipedia, http://en.wikipedia.org/w/index.php?title=W._Edwards_Deming].

Practical Tip 7.10 Make ten early bets on value delivery

- *in your first 10 weeks of your project, plan 10 increments of value.*
- *declare all increments as pilots, experiments, or trials (some might fail)*
- *but keep successes in place, and then scale up.*
- *The point is partly political, be seen as a 'value deliverer'.*
- *partly to get you in the mode of a 'stream of measurable value delivery'.*

Policy 7.10 Focus on progress, ignore other distractions

- **We will clearly put all emphasis on the real consistent progress towards our strategic goals.**
  - **We will not do anything to criticize experiments and small failures, assuming they are genuine, controlled, attempts to make progress.**

- Why?
  - To motivate people to try our new promising ideas,
  - and to avoid demotivation and fear.

Version: edit 3 Nov. 2015 tsg Kolbotn 20:16

Chapter 8. Delegation, Outsourcing, Contracting

We will sometimes need to delegate 'value planning' to internal and external suppliers.

Here are some ideas on delegation, while retaining your own *responsibility* for the value and cost results.

Section 8.1 Delegating Decisions

Principle 8.1 Delegate Clearly.

Management should
set a few clear goals,
and delegate the rest

see Quote 8.1

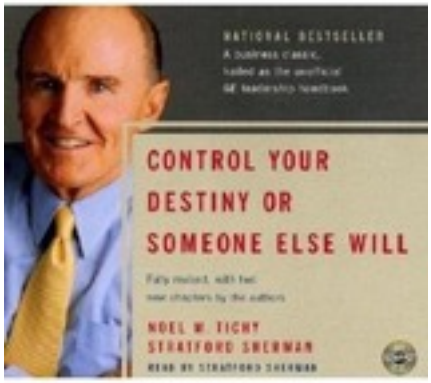
'delegate' (above) means that you are *responsible for setting your strategic objectives.* But you *do not get directly involved in the decision-making of your subordinates*, colleagues or suppliers, who are trying to help you achieve your Goals. Let *them* find a cost-effective means.

Figure 8.1 Delegation of the means. Retention of responsibility for the Ends [3, Keeney].

"Instead of directing business according to detailed...strategic plan, [Jack] Welch [General Electric CEO] believed in setting only a few clear, overarching goals.

Then, on an ad hoc basis, his people were free to seize any opportunities they saw to further those goals."

Quote: 8.1 Source: *Noel Tichy and Stratford Sherman, "Control Your Own Destiny or Someone Else Will"*



- **Why we get tempted to get involved, instead of delegating.**
  - We are trying to be helpful.
  - We are trying to show we have professional or technical knowledge.
  - Salespeople have *gotten* to us, and convinced us that their solution is the *right* way.
  - Our *solution* suggestions are *easier* for us to *express*. Solutions are concrete; *buy, do things.* We have not yet learned how to articulate our objectives measurably.
  - The solution, that our support team is responsible for finding is *inadvertently bundled* up together with our woolly objectives in the same sentence. Clue: 'link words'
    - 'We will get productivity using X *in order to* achieve Y"
    - '*By doing* X we will raise ability to do Y'.
  - You are *not* focussing on the *improvements themselves*, you are, out of habit, thinking in terms of the solutions.

- **Why we should *not* get involved giving the solutions because:**
  - We are, with *our* power and authority, going to get what we ask for (possibly the wrong solution, or a bad one).
  - We dismiss the possibility that our support team already know of better ideas, if we 'suggest' solutions. Insulting.
  - We miss the possibility that *they* will discover *better* ideas by measuring, and seeing that they can improve their initial ideas, to get even better results

  - We remove the support-team's *motivation* to find, and improve, solutions, that *really* do support *our* strategic objectives.
  - We will be viewed, behind our backs, as obsolete in our thinking, and lose respect.

- **How to keep our distance, from our helpers:**
  - Have a strict policy, for yourself, and your level of responsibility, to set the critical improvement objectives numerically, and support your support team with motivation, praise, interest etc. but to *never ever* meddle with *their* decisions on the solutions.
  - Always articulate *your* needs *only* in terms of *improvements* and *results* [URL1]. Do *not* (unless absolutely consciously necessary) constrain suppliers with *your* solution idea.
  - Humbly explain. You are not an expert in their field, you respect their knowledge, you will judge their ability on the speed and degree that they can deliver to your Goal levels.
  - Set standards (*Rule 1: Objectives shall not contain their solution, a design*) and make sure they are followed by Specification Quality Control of the key documents.[URL23]
  - When the 'salespeople' call (even *within* your own organization), ask them the 12 Tough Questions [URL15]
  - If you must hint at a solution, make it very clear, in writing, that it is *tentative*, and *not* required. You would *prefer* they find a way to get better results than your idea.

Corporate Agility:
Type: Critical Corporate Improvement Objective.

Ambition: All employees and suppliers can far more rapidly respond to problems and opportunities.

Scale: the speed in Average Calendar Time from a Response Opportunity occurs, until satisfactory action is completed.

Goal [within 3 years] <10 times faster than at Present>. <-CTO

Rough Solution Brainstorming: X, Y, Z. <- TG & KG, 13 July 20xx

*Planguage Example 8.1 A The last specification, "**Rough Solution Brainstorming:** X, Y, Z", allows us to collect ideas about solutions, and usefully to show that we believe there are some solutions at all. Adding the source people and date helps clarify who did NOT suggest the ideas, and how long ago the suggestion was made. But it is clear here, we hope, that the job is to reach the Goal, and the solutions are not any class of requirement. <fuzzy brackets> mean we are not more specific because we do not have the benchmark data for the 'Present' level of performance.*

Personal experience case 8.1 TG, Boeing

At Boeing, El Segundo, some young engineers I had taught came to me and expressed extreme frustration; told me of very heated arguments.
They had 'bought in to' on my advice, the need to clearly separate ends and means.

But some of them felt very strongly that they needed to express the solution ideas anyway.

I clarified my position, to their great relief,

" You can and should collect any potentially useful ideas for solutions, at the same time as your are discussing objectives and requirements.

That is natural, people cannot help thinking of solution ideas.

It would be a great shame to lose an idea that might be a great idea, just because we suppressed solutions totally, at this stage.

The important thing is the solution suggestions never get misunderstood as a 'requirement'.

That they must all be challenged, and stand the test of being estimated and measured.

Just keep them in a separate bucket labelled 'Potential ideas, we should not forget: a Checklist to review later'."

- **Exceptions: Constraints**
  - There will be rare (1%?) instances where there is a political or legal necessity for you to somehow stipulate a solution. This is called a 'Strategy Constraint'.
  - *However*
    - Do this 'gently'
    - Explain exactly *why* you are doing it.
      - **Source**: who said?
      - **Justification**: cite the law, policy, standard, regulation that needs this solution.
    - By clearly open for arguments to the contrary.

- **How to impose a solution 'gently'**:
  - Give the solution in *very general terms* ('Solutions will be within the EU, in Origin and execution').
    - Be clear that you expect your support staff to work out details, so that the *exact* variant they choose, supports your strategic objectives in the *best* way.

  - Be clear about *why* this constraint is specified, and specify the reasons for it (Source, Authority, and Justification or Rationale).
    - This gives your support staff reasonable grounds for accepting the constraint, or for arguing against it.

Common Architecture:
Type: Architecture Strategy Constraint.
Level: Project Omega.

Description: Our Project Omega 'preferred architecture' will, *as far as it is not in conflict with our major objectives, and resource constraints*, make use of the 'Common Corporate System Architecture' [Latest Version].

Authority: CTO and Chief Corporate Architect.

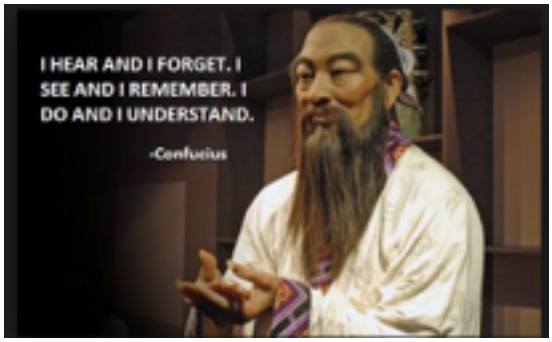
Rationale: this is a Corporate Standard (a strong guideline but not absolutely mandatory), and we are obliged to generally respect it.

Exception: We will use it, unless we can show that it, or any part of it, is a *serious impediment* to reaching Critical Objectives at all, or cost-effectively. We can then either Document an Exception, and the reasons why; or get written acceptance from the Chief Architects Office. We should make written notes, in our plans when we have accepted this constraint, for a defined aspect of this Architecture, but, where it has, to some degree, compromised our value-for-money ability.

Planguage Example 8.1 B: How to specify a design constraint 'gently'.

When it is obvious that
the goals cannot be reached,
don't adjust the goals,
adjust the action steps.

Confucius (551-479 BCE)



Practical Tip 8.1

- *keep your focus on the critical value improvements*
    - *limit yourself to 'how well', avoid 'how'*
- *do not be tempted to decide on* their *strategies*
- *strategies are to be approved by the 'authority of measured results'*

Policy 8.1 Objectives Responsibility: Solutions Delegation.

- **We will keep our focus on 'achieving clearly articulated critical objectives' *at 'our own responsibility level'*,**
    - **and will to the largest possible degree,**
    - **leave the *exact* solutions, to our objectives, open for others to find,**
    - **and to change,**
    - **whenever change 'clearly benefits our stated objectives'.**

- Why?
    - To get the best results for *our own* strategic objectives

    - To avoid misusing our power and authority to get particular solutions adopted: and therefore having to take responsibility for the possible failure of those solutions!

Version: Nov 3 2015 20:56 tsg minor edit.

Section 8.2 Supporting People who you delegate decisions to.

Principle 8.2 Enable.

Management should
work hard to enable
'supporting people'
to meet their management's goals.

Any given strategic planning level, the one that owns their own Strategic Objectives, has an opportunity to support their own supporting team (any people helping them to reach their objectives, including outside suppliers).

Here are some types of support you can give:

- Signal that you are interested in *providing resources*, and in r*emoving any barriers* to progress towards the Goals.
- Give a clear contact point (an email, or a responsible person), where anybody is welcome to send any advice, request, complaint, experience etc. that might be relevant to achieving, or failing to achieve, objectives. Allow anonymous reports there.
- Demonstrate that *you have time* to discuss barriers and problems that your support team might have.
- Demonstrate that you will consistently find reasonable solutions, to remove barriers.
- Lead by example: show that you are accessing help with *your own managers* or Board to achieve *their* Fundamental objectives.

Case 8.2 ICL: Adopting Planguage for Top level Corporate Planning

International Computers Limited, was once 33,000 employees, and Britain's answer to IBM. But they were badly led for years, and were losing money for about 8 years straight [URL49 A].

Then Prime Minister, Margaret Thatcher, the Government owned them and funded them, made sure they got an energetic young leader, Dr. Robb Wilmot, an electrical engineer by education, high flyer (named one of the youngest vice presidents in the TI company's history) from Texas Instruments (said to be considered the forthcoming CEO there).

Wilmot had gotten a copy of my current book manuscript via an old school friend of mine (John Andrews), running his European Office, when I bumped into 'Andy' in Bombay India at the airport. Wilmot liked the quantification ideas. I later saw a his printout of my book manuscript. Yellow high-lighted, densely, by Wilmot. Wilmot remarked to me once *"I never read books, they are bound to be out of date, in these fast moving times" [F7]*.

He asked me to advise him on why he had managed to move ICL to break even, after 2 years, but could not make further progress, in spite of his CEO Power.

Photos 8.2 Robb Wilmot, and The ICL HQ in Putney London he worked from.

After 2 weeks study, with his 'halo' on me, I reported to him and his Directors, that all his top managers, himself included, constantly spouted management woolly phases for objectives; like 'State of the Art Technology' [URL1]. And little else. I showed proof that nobody reading these 'Fundamental Objectives' understood them as intended, *But they acted on their misunderstandings anyway.*

I also showed that the *software* engineering staff, unlike their hardware colleagues, had no ability to interpret conventional hardware engineering ideas like '99.90% availability' (for their software component) at all. They also had no 'Design Engineering' ability.

And software, now 'half' the product, was a major problem. But that (Design) was a problem that took them 2 more years to acknowledge that it was mature for top management attention, Mike Watson CTO told me 2 years later.

It does not help to have clear quantified objectives, if your product designers do not know how to interpret objectives and turn those ambitions into viable Strategies.

After 2 years, we got started with the design-to-clear-objectives problem: but too late, as top management was, as usual, moving on.

For years, before and after that, I was seconded to a large number of ICL projects, to help change the culture, at the grass roots level (also a wise understanding, of top management) [F8 SQC] and [5 D Usability] are published examples.

This initial analysis resulted in my 15-minutes-of-fame presentation to the Directors. Wilmot immediately agreed. Told his 12 directors that I was right, and that he, the CEO 'was the worst offender'! And he would be the first to mend his woolly ways. True Leadership!

Wilmot immediately 'suggested' that all his directors would 'please' submit their top-level critical quantified strategic objectives, with Scale and Goal etc. He did not, the act implied, know what to hold them accountable for, *yet*. Financial budgets, were a lagging indicator: he needed *leading* indicators, like the 'Usability of the Software for Retail Shops'. This was the curse of the Unbalanced Scorecard, in practice.

Otherwise: *he would not be able to see clear reason why they should retain power or budget*. They knew he was dead serious, and capable of removing their power and budget. They quickly got the message!

Their *own* Director-level, Strategic Objectives, had to support *his* (for them) Corporate Level Fundamental Objectives (Scale and Goal etc.).

I had drafted these Corporate level quantified objectives for him, including an Impact Estimation table for the Corporate Strategies: and which he had already gotten Board Approval for, using my *handwritten* version!).

My favourite example was 'Corporate Viability'. Now how would you quantify that? I figured out an incorruptible measure, which was appreciated by the Board. We tied 'Corporate Viability' to the Stock Market share price! That beat the previous 'We shall be viable' objective.

The Board had remarked, the CTO (Mike Watson) told me, that this set of quantified Objectives, and Impact Estimation Table (Wilmot's Strategies, impact on quantified corporate objectives), was *the 'clearest Board presentation they had ever seen'.*

One of the 8 corporate strategies, which Wilmot insisted on, against my ethical-consultant advice, was using my Planguage corporate-wide. And the Board was happy with that, upon seeing the Board level example of quantified objectives, and Impact Estimation Table with Corporate Strategies, that Wilmot presented to them.

Corporate Planning Director, Peter Hall, and I were immediately sent round to the 12 directors to help them get the Objectives right. Not one of them had training or ability to quantify their management objectives [URL2]. But we sorted them each, in about half a day. They were glad for the help!

One of them (Peter Gershon, Marketing director) was later Knighted (Sir P. G.) and I like to invent the story that my 'pupils earn Knighthoods'. Which is possibly an exaggeration. But the reader is welcome to try. Even Robb earned a well deserved CBE (Commander of the British Empire).

The bottom line, was that ICL went into profit, and remained in profit for the next 15+ years (unlike some major competitors in the period). Until a financial 'bean counter' (Unbalanced Scorecard again) stayed at the CEO helm too long, and a new Corporate Owner (Fujitsu) failed to keep the imaginative improvements going. Nothing lasts forever, without re-energization. But it was a good run for the money!

Here are some *signals* that 'support of your team' is needed:

- Planned deadlines are not met: especially short value-delivery cycles
- Planned value-delivery levels are not cumulating at a rate consistent with meeting the longer term Goal levels

- Deliveries are 'met', but with complaints from the users or recipients. Like 'too difficult', 'too many faults', 'old way was better'.

Devices in Planguage that will help you communicate about the need for support team help.

- Use a 'Wish' (see 6.10) statement, rather than a 'Goal' statement, until all the Goal conditions have been met. This forces a dialogue about necessary resources and support.

- Indicate any types of *Risks, Threats, Damages, Issues, Assumptions, Low Credibility, ± Uncertainty, low estimates in Impact Estimation tables*, in the plans, which need resolution, perhaps with support to get them resolved. There are plenty of Planguage devices you can use, to signal plans needing help.

- If feedback about incremental progress, is clearly under the expected level. For example if you are 50% of the way towards a Goal level, and have used 90% of the time to Deadline for the Goal, then you have a resource problem, and people need help.

W1: Wish [Deadline = New Years Day next year, Area = Canada, People = Teenagers, If Trade Agreement Valid] 30%

Planning Status: [W1, December 24, 20XX]

PS1: Strategies: S1, S2, and S5 are estimated to get us to about 150% ±60% of the Goal <- IET ABC.

Mitigation: More Strategies will be developed to 300% Safety Margin.

PS2: Capital Costs of the strategies are estimated to be 5% of total project budget.

Mitigation: an independent conservative estimation will be made to see if we can trust such a low estimate.

PS3: We do not know if we have qualified people available to carry out S5.

Mitigation: a search for qualified people has begun, and funding is secured.

PS4: We do not know, at this stage of planning, if this Wish will have sufficient priority ahead of all current commitments at that time; and there are still possibilities of additional commitments.

Mitigation: We have requested CTO guarantee sufficient priority.

Planguage 8.2. Examples of giving support , Mitigation, to documented risks.
The ongoing re-planning is integrated in the plan,right under the Wish statement, explaining why the Wish status cannot be converted to a 'we promise' Goal in the plan.
Derived from Planguage 6.10 C

"For a business to strengthen its position on the market, its managers should become skillful at helping their subordinates to set and achieve specific and measurable goals with realistic deadlines and clear expectations. Managers should also mentor employees through challenges, helping them grow and develop new skills."
— Anna Stevens

Practical Tip 8.2

- ***send regular signals to your team: 'is there anything I can help you with?'***

Policy 8.2 All possible aid will be given

- **The support team, helping you meet your Strategic Objectives, will be given all possible support and resources to achieve the objectives. They will hopefully be open and forthright about any problems they have, that might need your management attention.**

Why?

- To make sure we reach our objectives
- To remove barriers, often unnecessary barriers, so as to deliver value

Version December 2 2015 13:10

Section 8.3 Smart 'Value Contracting'.

Principle 8.3 OUTSOURCE, BUT MANAGE

**You cannot outsource responsibility,
you can outsource in smart ways:
like no cure no pay,
and flexible contracts**

We outsource like never before, offshore, near-shore, on-shore: 'just get the work outside of the company'. So, a major planning need is to communicate with these outside suppliers. In principle, there is little difference, 'outsourcing' or 'in-house', regarding the need for good clear planning. We already need to communicate plans *far more clearly* to our own employees, so in principle we can simply communicate that same quality of plans to the outsiders.

Except:

- The outsiders, even 'on-shore natives', do not have the same *culture* that your organization does.
- They do not have the same *motivation* .
- There is a *legal* contractual difference.
- They *answer to other* top managers.
- They may not have your *language* as their native language.
- You *both* may be planning in a *common **foreign** language (like English).*
- If they totally fail you, their option may be to just *find another fool, as a client.*
- You may *never meet most of them personally*, and they will not meet you
- The hard *experience and learning* they get, does *not stay* with your people, your organization.
- Etc. the list is long.

Yet, it is obviously tempting. There are some clear advantages. Our suggestion here is that *extraordinarily much better planning* is necessary for dealing with outside suppliers. This is because of the exceptional factors above.

One simple reason is that if our plans are not clear enough, both sides will end up paying the legal profession to clarify, what we should have clarified, at less expense, in the first place, ourselves.

If you think planning is expensive, look at legal costs.

We get constantly reminded of this, one example.

Case 8.3: Failure in Supplier Communication: Both Ways

In Spring 2014 we were brought in by an offshore organization that was being paid £100 million per year to help build 'public transport systems'. They were, after 2 years into the project, in ***total failure*** *mode.*

We analysed the situation. Nothing surprising there for us. The causes of such failure are so common.

The Customer had written very detailed system requirements. But in fact they had no quality standards for clear unambiguous specification. They had no form of Specification Quality Control [URL23]. So the usual '100 unintelligible things per page were streaming towards the supplier. The supplier swallowed these specs, very far offshore. Using about 100 intelligent well-educated engineers, who were fed Garbage In. They had no direct communication with the customer, their 'final' product, 2 years later, was the first 'reverse' communication, to their client.

No quality control on the incoming specs to the supplier. They just guessed at the meaning, and built an expensive system; the delivered system just did not work when delivered, on thousands of practical details.

The supplier claimed they were doing some kind of (then faddish) 'cyclical delivery', but obviously, that delivery was to themselves, not to the customer, so that real problems could be not detected a year-or-so earlier.

The supplier's own tests just confirmed their local misinterpretations of the customers woolly specifications. 'We correctly built or misinterpretation'.

The supplier top management, we met, actually thought everything was fine with the customer!

But the middle management, in direct contact with the client, was left in no doubt by their customer: there was a catastrophe, from which recovery was, at best, expensive and painful.

The local Supplier project manager finally gave up on his <u>own</u> management, and left the company.

We see this miscommunication all the time. But maybe only an independent consultant, not citing names of the parties, will tell what they have seen.

But, let me guess, this is the norm, not the exception. There are plenty of studies of failed projects, and failure seems to be the norm, both for in-house and outside contracting.

Now to be fair, projects were failing long before outsourcing. But outsourcing did not seem to improve the failure statistics.

Diagram 8.3 The above Case study in a nutshell.

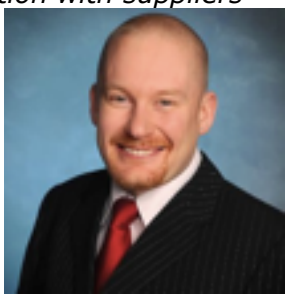
Quote 8.3

"Define clearly what you want.
Often suppliers are accused of poor outcome.

The accusations are unjust if the definition has been unclear or incorrect."

'Tips for successful communication with suppliers'

by Jussi Aromäki [URL50] 2011



Practical Tip 8.3

- *do not assume your legal advisors will help you make a clear supplier contract; they do not know* how*, and do not* care*.*
- *you have to do that yourself.*

Policy 8.3 Extraordinary Clear Planning & Pay when delivered.

- **We will invest in extraordinarily clear planning for communication with suppliers,**
- **and we will confirm correct understanding early, continuously, and frequently; using real outputs from the supplier, in our own realistic setting.**
- **We will contract with the suppliers to pay only when expected products and services are successfully delivered. [URL6]**

Why?

- it pays off tremendously to invest in clarity
- no plan survives contact with the enemy
- to motivate suppliers to work in *our* interest, not just to 'work'.

Version 22 Dec 2015 **5:29 PM Kolbotn**

Section 8.4 Managing by Results

Principle 8.4 GIVE GOALS, NOT MEANS

**Never tell your supporting teams, the answers,
focus on giving them really good questions:
like - your real top-level critical goals - quantified**

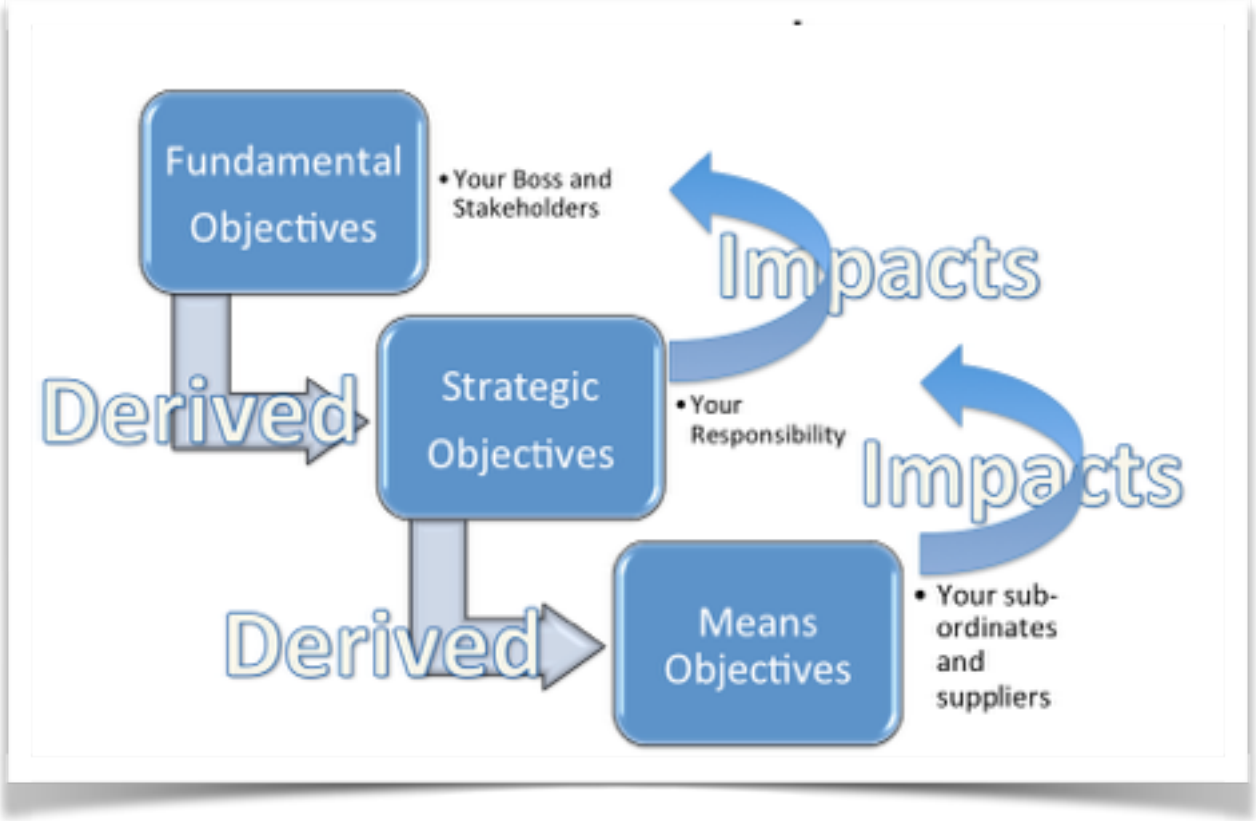


Figure 8.4 A hierarchy of objectives.

Our objectives should be derived from the objectives at the level above us, the one we we 'support'. From your boss, if you like. Reaching our strategic objectives should result in partial or full satisfaction of the Fundamental Objectives, those above us.

The same structure is supporting us too. Our support team's Means Objectives is effectively the same as our 'Strategies for reaching our own Goals'.

We can of course be one of several support teams supporting our 'boss', but it would be useful to know exactly what contribution, numerically, we are expected to make, amongst the many.

Case 8.4 The Ericsson Productivity Objectives Hierarchy [URL19 B].

Here are real examples of levels of planning, done at Ericsson Base Stations, years ago. More details at URL19 B.

The Fundamental Objective was to produce more product, to get more sales. But reaching this was constrained, because it was not possible to hire enough engineers on the world market.

We finally agreed that the Means could not be 'Hiring x,000 more engineers'. It had to be 'multiplying the productivity of the engineers we already employed'. The initial dream was ten times more productivity. But thi was a 'misunderstanding' of the 10X quality that we helped HP Corporate Quality and Engineering achieve earlier [URL31].

When we looked realistically at the productivity we could achieve, with known methods, we could set a conservative objective of productivity improvement factor 2, within 3 years. For example as Raytheon had done [URL27]. This effectively became our Fundamental Objective.

To Support this, we (Kai and Tom) worked on planning for 2 weeks with the CTO (Thomas Ericsson) Staff, and came up with the following set of CTO Level **Strategic Objectives:**

<u>Software Productivity</u>:
- Lines of Code Generation Ability
- Lead-Time:
- Predictability.
- TTMP: Predictability of Time To Market:
- Product Attributes: Customer Satisfaction:
- Profitability:

Case 8.4 A. The **names** *of the Critical few Strategic Objectives.*

These were each quantified in Planguage (see URL19B for detail)
But here is one example:

TTMP: Predictability of Time To Market:

Ambition: *From Ideas created to customers can use it. Our ability to meet agreed specified customer and self-determined targets.*

Scale: % overrun of actual Project Time compared to planned Project Time

Project Time: Defined: time from the date of Toll-Gate 0 passed, or other Defined Start Event, to, the Planned- or Actually- delivered Date of All [Specified Requirements], and any set of agreed requirements.

Specified Requirements: Defined: written approved Quality requirements for products with respect to Planned levels and qualifiers [when, where, conditions].
And, other requirements such as function, constraints and costs.

Meter: Productivity Project or Process Owner will collect data from all projects, or make estimates and put them in the Productivity Database for reporting this number.

Past [1994, A-package] < 50% to 100%> <- Palli K. guess. [1994, B-package] 80% ?? <- Urban Fa. and Palli K. guess

Record [IBM Federal Systems Division, 1976-80] 0% <- RDM 9.0 quoting Harlan Mills in IBM SJ 4-80
"all projects on time and under budget"
[Raytheon Defense Electronics, 1992-5] 0% <- RDE SEI Report 1995 Predictability.

Fail [All future projects, from 1999] 5% or less <- discussion level TG

Goal [All future projects, from 1999] 0% or less <- discussion level TG

Planguage 8.4 B. Example of one of the strategic objectives.

Supporting them were the *Means Objectives*, listed below (again see URL19 B for detailed quantification, of Means Objectives, in Planguage)

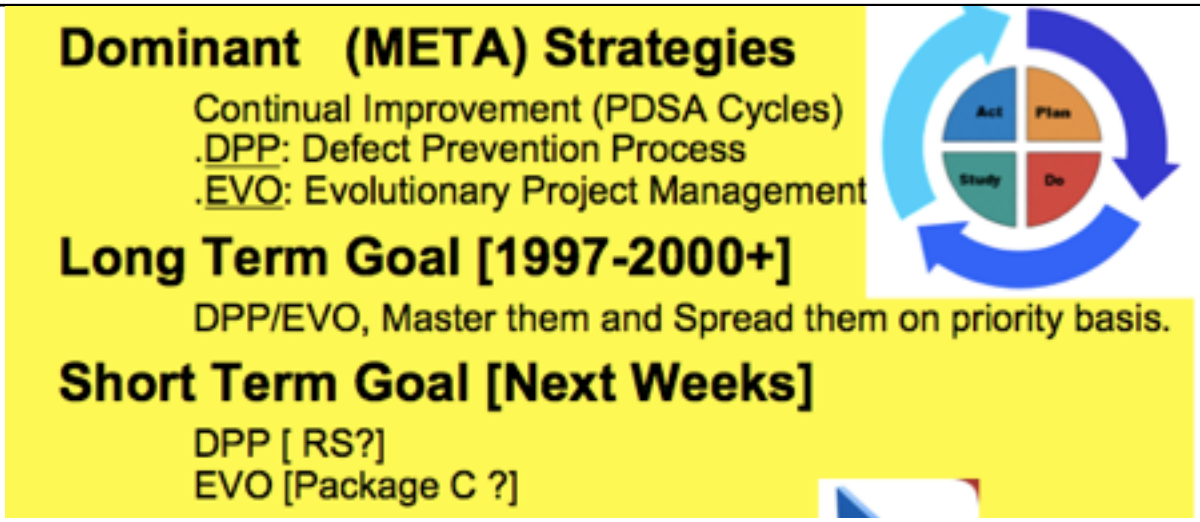
- **Complaints: Feature Production:**
- **Rework Costs:**
- **Installation Ability:**
- **Service Costs:**
- **Training Costs:**
- **Specification Defectiveness:**
- **Specification Quality: Improvement ROI:**

Case 8.4 C. Means Objectives names

In order to achieve these Means Objectives, and partially the Strategic Objectives, we had a long list of *candidate strategies*, which people suggested would result in achieving the objectives, partially.

Case 8.4 D. The full brainstormed list of strategies for reaching Strategic Objectives. Some of them, like 'Architecture Best Practices' had no merit, since no one could suggest specific practices or their effects and costs. These 'empty' generalizations were actually suggested by senior experts! This failed to impress the Ericsson management.

By doing Impact Estimation with regard to risk and cost, we identified 3 Meta strategies to get started with (next week. literally). We called them 'Meta Strategies', because they gave very good impact in themselves, and had done so at comparable corporations. But these Meta Strategies had the added attraction of helping us *measure the effects of all other strategies* we might use later.



Case 8.4 E. The management summary.

This plan agreed, we took the steps to immediate action.

We called in the Project Manager for 'Package C' the major new product development, and asked if he would immediately commit to Evolutionary, weekly delivery of value in his project. He agreed.

Ericsson Corp. had some very positive experience in Japanese Deliveries, indicating that this would work [Järkvik, 'On Succeeding', Internal Pub.] But it was very controversial, since it was radically different from their 'Props' U-model Big Bang current Corporate Project Method.

Then, to get the Defect Prevention Process [URL40] going, we chose the key upstream process, *'Requirements'*, from their dozens of other engineering processes, and decided to start with that.

It helped that we had seriously experimented with DPP, directly with the participation of the CTO (T. E.), for a year just previously, so he had confidence that it would work. That decision means we got a *week by week* tuning of the Requirements Process, starting the next week.

So, although this is a relatively large organizational project, we spent 2 weeks planning it, and went for execution of value delivery, and learning cycles, on a weekly basis, by the third week. [URL5, Startup Week]

"Let no man turn aside, ever so slightly,
from the broad path of honour,
on the plausible pretence that he is justified by the goodness of his end.

All good ends can be worked out."

Charles Dickens (1812-1870) Writer

Practical Tip 8.4

- *make a graphical map of the different levels of objectives. Separate level of concern clearly.*
- *see Figure 8.4 for the 3-level basic structure. It applies anywhere in a chain of command.*

Policy 8.4

We will keep our focus on defining our own objectives, and getting them delivered. We will not meddle in our support team's choice of strategies for getting there.

Why?

- Clear lines of responsibility
- We do not get blamed for bad choice of strategies
- We empower and respect our support teams

Version 22 Dec 2015 Kolbotn 5:43 PM

Section 8.5 Removing barriers to delivering value

Principle 8.5 Remove Barriers

Tell your teams that *your* job is
to remove any barriers to them delivering value;
ask them frequently what their current barriers are.

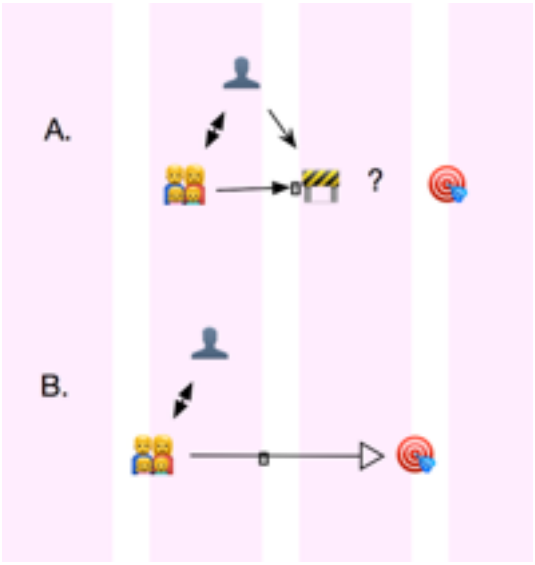
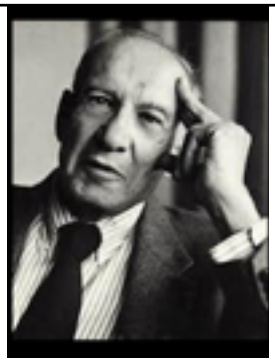


Figure 8.4
A. the manager in dialogue, or observation with the team, discovers a barrier to their value delivery progress.
B. the manager removes the barrier and the team can progress towards their objectives.

Quote 8.5 A

"Most of what we call management consists of making it difficult for people to get their work done."



— Peter Drucker *(1909-2005)*

The good CEO spends their time 'getting cups of coffee for his staff', so they can get on with their productive work. And also so the CEO can see the action in practice.

Quote 8.5 B

> **"***Dole out cream and sugar. During the busiest times of the year, executives at the Cigna Group push coffee carts around the office, serving drinks and refreshments to their colleagues",*
>
> *noted Steve Harrison, author of The Manager's Book of Decencies: How Small Gestures Build Great Companies.*
>
> *"As they serve, executives coach and encourage colleagues and hear about real consumer issues". [URL52]*

Symbolically, 'coffee serving' is a healthy metaphor.

Management can increase their own productivity by continuously worrying about things that can stand in the way, of their support team, of delivering value.

Here are some of the things that reduce ability to deliver value:

- A grass roots experience that management will not *really* listen or act on suggestions to improve their ability to deliver value
- Focus on tasks and deadlines, rather than the *actual value* to be delivered.
- Lack of grass roots freedom to find their own solutions, and measure their own ideas: being told what to do and how to do it; rather then being inspired to achieve results, and find imaginative ways to do it.
- Lack of built-in-the-organization processes, like Defect Prevention Process [URL40], and Statistical Process Control URL47, 9] - so people can work to remove barriers themselves
- The belief that 'stand-up meetings', and any 'talk talk' meeting can discuss their way to solutions, without a framework of *measurement*, and without a framework for *investment in change* (like Raytheon, URL27)

- As consultants we both meet with top managers, *and* we work in the trenches. We do not see enough of managers in communication with their staff; I don't mean, in a meeting. I mean right in the middle of real staff work.
- Blaming people (who are essentially helpless in the face of their working environment), instead of (the real culprit) blaming processes, one of Deming's favourite attack points [9]

Here are some things we can do in Planning, and with Strategies, to help support teams deliver value :

- Continuously asking, and really listening and acting on your support team's experiences:
  - "What could I do to help *you* deliver value better?"
  - "Has management taken the need for improvements seriously?"
  - "If you were in my shoes, what is the most useful change you would make, to help people get more value delivery?"
- Wandering the shop floor, MBWA, Management by Walking Around (Tom Peters [8]).
- Sitting down with real people and trying to find out how you can help them. Hopefully with things that can be used by many people over time.

The "Real Problem", I know you know this, is *communication*.

Communication between people. The methods and principles set forth in this book are really devices for improving communication between people about technological and managerial matters.

Case Study 8.5 The Bill of Rights for company communication

At one point in my work there was a large multi-national computer company, ICL UK, which had decided to adopt the "measurable objectives" ideas in this book, at the *management* level *and* the *technical* level (See related Case 8.2, URL49].

The Chief Executive (Robb Wilmot) had decided to issue a booklet to popularize the concept : "The ICL Way to Set Objectives". I became involved in the work of drafting the booklet, together with Personnel Division people.

One day, I got suspicious of the concept of telling all the bright professionals in the company *how they should act*. I thought it was about time we granted them some *rights,* instead of giving them *duties*.

So I drafted the Company Communication Bill of Rights (below). It was a 'hit' with the people I was working with, like Peter Hall (Corporate Planner) and with the Chief Executive (Robb Wilmot). But we initially ran into severe political problems, with the more-traditional Personnel Director (later fired by R. W.!) who felt that we were giving too much power to the masses!

Perhaps he did not appreciate that with each 'right', comes a corresponding set of 'obligations'?

Ahhh, the power of an idea whose time has come .

The Bill of Rights was published, put on the walls, framed, and the CEO constantly referred to it in his speeches.

I have no good way of tracing the cause and effect here, but as noted earlier in this book, the company *did* go into profit for many years; and I like to think this was a small part of that. Not every strategy is fully measurable.

The Bill of Rights for Company Communication

1. You have a right to know precisely what is expected of you.

2. You have a right to clarify things with colleagues, anywhere in the organization.

3. You have a right to initiate clearer definitions of objectives and strategies.

4. You have a right to get objectives presented in measurable, quantified formats.

5. You have a right to change your objectives and strategies, for better performance.

6. You have the right to try out new ideas for improving communication.

007. You have the right to fail when trying, but also to kill failures quickly.

8. You have a right to constructively challenge higher-level objectives and strategies.

9. You have a right to be judged objectively on your performance against measurable objectives.

10. You have a right to offer constructive help to colleagues to improve communication.

Case 8.5 The ICL Bill of Rights. This was my attempt to delegate power to the people, and lucky for me the CEO agreed heartily!

I don't think the 007 joke survived in final draft, but *I* enjoyed it. Please feel free to use these ideas as you please!. See also [URL51] ICL Quality Metrics

Quote 8.5 C

"The Four Keys of Great Managers:
1. "When selecting someone, they select for talent ... not simply experience, intelligence or determination."
2. "When setting expectations, they define the right outcomes ... not the right steps."
3. "When motivating someone, they focus on strengths ... not on weaknesses."
4. "When developing someone, they help him find the right fit ... not simply the next rung on the ladder."

— Marcus Buckingham, http://www.marcusbuckingham.com/

Marcus has other relevant things to say in terms of this sub-chapter, so while he is on stage:

Quote 8.5 D

> ***"You are at your most powerful***
> ***when you are showing people obstacles they don't see,***
>
> ***& then how the obstacles can be avoided".***

Practical Tips 8.5 Experiment with discovering barriers

- make a list, for your boss, of the 3 worst barriers to your *own* progress
- ask one of your teams or projects to make a list of the 5 worst barriers to their current progress
- remove one barrier for your staff
- figure our a way to remove one of your own barriers, and tell your boss you (already) did it.

Policy 8.5. Specify Barriers Explicitly

- **We will *document*, in our planning, all assumed barriers to achievement of our plans.**
  - We can use headings such as Assumptions, Issues, Risks, Dependencies.
  - We can also suggest mitigations for each.
  -
- Why?
  - Costs little
  - Might result in removing barriers, that will cost far more than the documentation effort, if not removed

Version 22 Dec 2015 Kolbotn 6 PM

Section 8.6 Getting early short-term feedback.

Principle 8.6 Measure Now

**Be prepared to measure 'leading' indicators,
before you can measure
longer-term indicators
of value delivered.**

Planners are quite naturally worried when I talk about *measuring* results of value delivery *every week*. Surely, some important results cannot *immediately* be measured, even if your target system can be modified and made ready quickly ('next week') for enabling the results to happen.

Surely some things take time to be absorbed by stakeholders, before we can *really* measure that we got the results we wanted, for example; *more sales, better reputation, receptivity to sales approaches, loyalty to a brand or a leader etc*.

Obviously some of these things take time, even when you have plugged in strategies that will impact them. Customers and employees need to get used to the new ways. To absorb new ways of doing things, to repeat the tasks to build confidence; and such things are not easily measurable on the Friday, after you plugged in your new strategies on Thursday, of your weekly delivery cycle.

Yet, we would love to be able to measure immediately, to know things were worthwhile, safe to keep in place; just to get a good nights sleep in the weekend after a value delivery cycle. So, how do we 'sort of' measure immediately, things that cannot *really* be measured immediately ?

The answer lies in fairly well-known concepts, though I admit that it took years before I became conscious of them. The key concepts are called 'leading indicators' and 'lagging indicators'.

As the terms indicate:

> **Leading indicators** are numeric feedback, that we hope correlate fairly well with the longer-term results, and values, that we *ultimately* hope to achieve.
> But, they may be easier to capture initially.
> Leading indicators capture information *before* a new trend or reversal occurs: maybe in time to use corrective action.

Lagging indicators are numeric feedback that measure *more realistically*, but *later*, what we wanted to achieve.
The lagging data might just not be available at all; or it might be more costly to capture properly.

Lagging indicators capture data after a new trend or reversal occurs. It may be too late to take efficient corrective action.

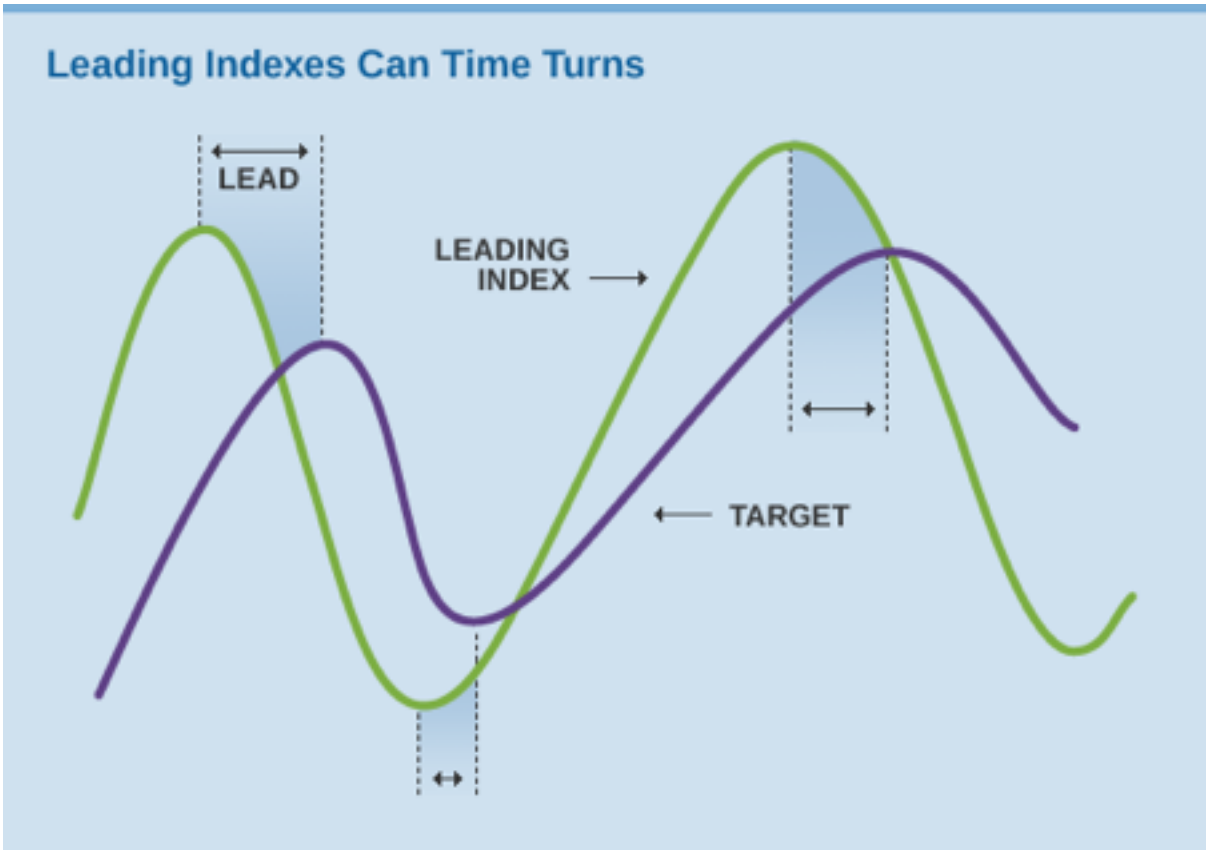


Figure 8.6 A. Leading and Lagging indicators can be used together to understand your system.

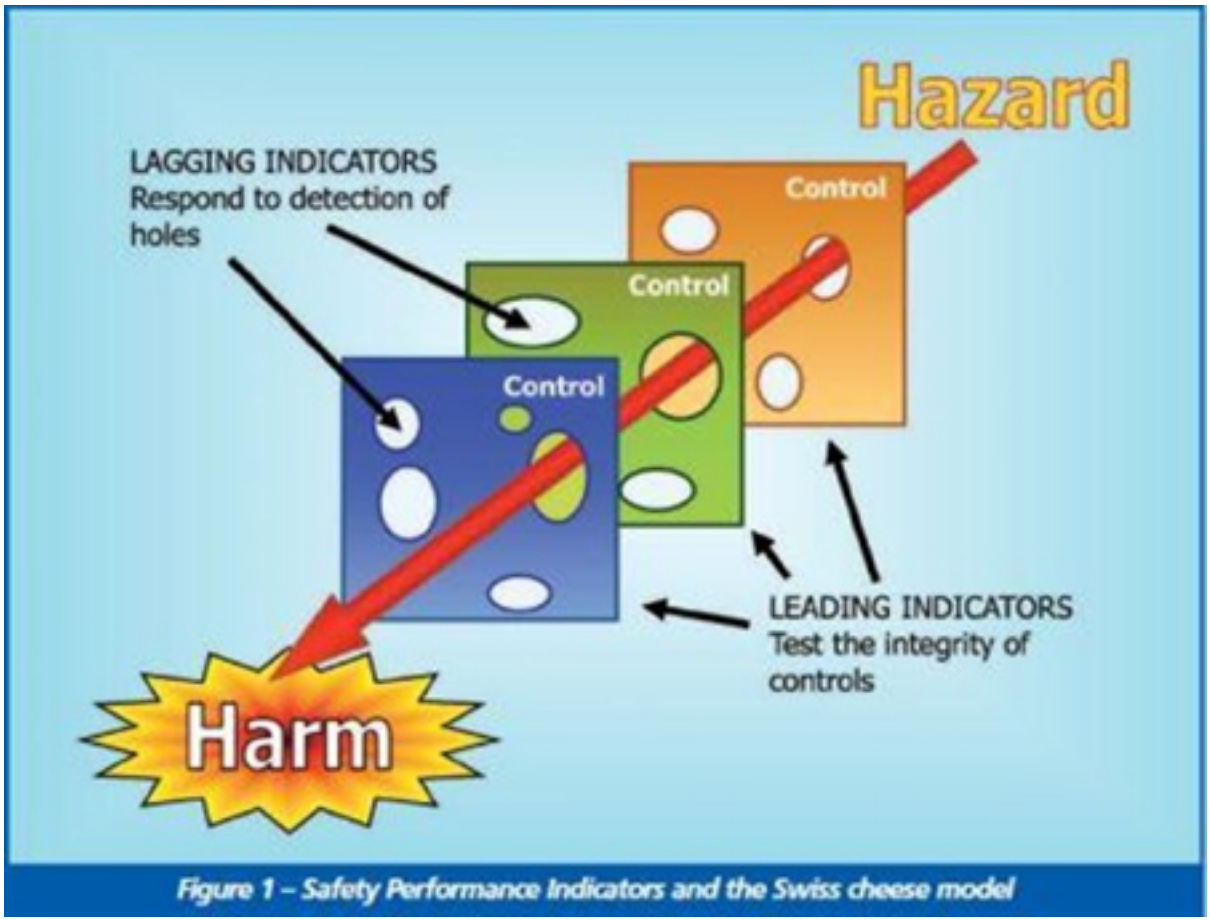


Figure 8.6 B Another interesting view is that leading indicators can give you an indication about whether your management processes are working properly. Lagging indicators can confirm or deny that early corrective action has succeeded or not. Safety environments have concluded that more leading indicators are necessary to give warning before incidents actually occur [URL78].

https://www.businesscycle.com/ecri-business-cycles/turning-points-leading-indicators

Both 'indicators' are a *combination* of a selected **Scale**, or set of scales; AND a selected method of measurement (**Meter**).

The Scale is a choice of **what** we want to measure. The Meter is a choice of **how** we intend to measure results, on the selected Scale of measure.

The more ***relevant*** **the Scale** is, with *relation to our values*, then - the *more useful* measurement, and therefore useful feedback from our strategy implementation, we are going to get.

The more *accurate, repeatable, credible, updated, and precise* (5 qualities of measurement processes) the chosen **Meter** (measurement process, test process) is designed to be, (and the chosen Meter is a matter of engineering design) – the better quality of information we will get, in order to reflect our chosen **Scale** of measure.

Otherwise, a bad quality-measurement process will 'corrupt' the information we wanted to get, about levels on the Scale we have chosen. We will, for example, get feedback which is not accurate. And this in turn will to some degree, possibly a critical degree, cause us to take bad decisions. We might believe things are working well, when they are *not*; or the opposite.

From a Planguage *planning point of view*, we can try to make sure we have Scales that correlate 'pretty well' with the reality we are trying to manage; for both leading and lagging indicators.

And we can plan the measurement processes (Meters), for both leading and lagging scales, to give a sufficient-for-purpose measurement process along our chosen scales. But final detailed planning might well be done by a specialist planner for testing.

Here is an example to show the principle of such measurement definition.

Sales:
Type: Critical Marketing Objective.
Ambition: order of magnitude more than today, within a few years.

LI1: Scale [Leading Indicator, Short Term Changes] Average % Internet 'Buy' Clicks per Deal offered, they click 'Make Me A Deal I Cannot Refuse'.

Meter [LI] Built-in website software counters and reports, with up to the minute feedback, and correlation with all variations and changes to the Deal Offered.

G1: Goal [Weekly, After Change = New Deals] 50% of Deals requested, as consummated, and are paid, for delivery (but subject to 30-day refundable, returnable by law).

Lag1: Scale [30 days after defined Changes made] net earnings, after returns, sales profit cumulated since Day of Sale, using a Deal of defined Change, per sale not returned or refunded.

Meter [Lag1] Corporate Sales accounting database reports, tailored as needed, for example by Deal, Geography etc.

G2: Goal [after 30 days from Sale Consummated, Change = New Deals] €30 (per Sale not refunded)

Planguage Example 8.6 *Illustrating the principle of deciding, for both leading and lagging indicators, the Scale we will use, and the corresponding Meter we will use to measure along a given scale.*

Comment on the Example Above
LI1 specifies the Scale of measure we have chosen for the leading indicator. We deem it to be sufficient to indicate the rough initial market response to Deals. We hope it correlates well-enough, with longer term final sales and income, to tell us if the deal offerings are *effective*.

Lag1 is our choice of longer-term measure of real net income. The real outcome we want to manage.

The two different Meters represent our design of practical and cheap measurement processes, which will give us a sufficient quality of measurement, so that the Marketing Analysts will be happy to accept and use the feedback data, and will not be willing to spend more resources to get greater accuracy, precision, credibility, repeatability, or up-datedness.

The Goal levels are, as always, on the corresponding defined Scale, and they are essential slightly different types (see [….]) of levels, so the Goals are necessarily reflecting the Scales above them, and the defined moment (immediate, month later) in the sales process.

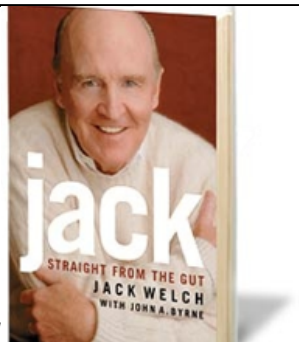
Aligning Rewards with Measurements

"You have to get this one right. … a universal problem: What you measure is what you get – what you reward is what you get.

Static measurements get stale.

Market conditions change, new businesses develop, new competitors show up.

I always pounded home the question 'Are we measuring and rewarding the specific behavior we want?' "



Jack Welch, former CEO General Electric '[jack', page 387]

Practical Tip 8.6 Leading and Lagging exercises.

- for your most critical current objective
  - define a **leading** indicator Scale
  - define a corresponding **lagging** indicator Scale
- for another objective, as an exercise,
  - define a Meter for the Scale, which is *easy to use on a daily or weekly basis*
  - then, also define a Meter for the same Scale, which would be *more useful in the longer term, and more credible.*

Policy 8.6 One Size Feedback Does Not Fit All

- **We will take the small effort necessary, to carefully distinguish between leading indicators, and corresponding lagging indicators; and specify the Scales and Meters intelligently. At least for *critical* objectives.**
- Why?
  - So we can get quick feedback from incremental changes, yet more-realistic feedback, more credible feedback, in the longer run, *about* the longer run.

Note: a strong guideline which this author largely wrote for a client, on the subject of corporate quality measurement, and Planguage concepts, is given in the ICL Introduction to Quality [URL51]. This is a strong set of Policy Guidelines, with practical templates.

Version 22 Dec 2015 Kolbotn 6:22 PM

Section 8.7 Crediting ideas and motivating responsibility.

Principle 8.7 Give Credit Generously

Credit individuals and groups,
with ideas of all kinds,
in writing:
that gives responsibility, motivation and pride.

I read a lot of plans. Very few plans specifically credit *any* individual with *any* detail of the content. Many plans have a named author. But that is not the same thing as crediting individuals in detail, for ideas.

Ill. 8.7 Music Credits.

One given plan might actually have dozens of different individuals, that have actually suggested strategies, estimates, and other details. I think it is a mistake to hide these sources. It may even be unethical ! Or illegal.

For example:

<u>Strategy X</u>:
Description: use triple redundant distinct software, with 2 of 3 voting corrections, and otherwise manual decisions after stop because of output differences
.
Estimated Impact [on Objective 1] = 30%±10%, Evidence = Similar levels found at Our Corporate R&D Labs.

Planguage Example 8.7 A . One detailed source credit. But what is missing?

Who suggested the Strategy X ? Who estimated the impact on Objective 1? Who dug up the Evidence, and the Source for the Estimate? Who cares?

Let us rewrite that, with Credits all the way, and a little more (artificial) technical detail:

Software Redundancy:

Type: Technical Strategy.

Description: use triple redundant distinct software, with 2 of 3 voting corrections, and otherwise manual decisions after stop because of output differences. <- *Wendy Bartlett.*

Estimated Impact on [Availability] = 30%±10% <- *Linda Queen.*

Evidence: = Similar levels found at Our Corporate R&D **Labs.** Using Distinct Software with 3 Versions <- *Tommy Scharf.*

Planguage Example 8.7 B. Reasonable levels of Source crediting, for the earlier 8.7A statement.

So, using the 'Source' keyed icon '**<-**' liberally, we make transparent the fact that 3 different people were involved in this specification.

1. *Wendy suggested the Strategy.* More exactly, she wrote the detailed description of it. Someone else could have suggested the basic idea.
2. *Linda*, not Wendy, made the *estimate* of the Strategy impact
3. *Tommy*, not Linda, actually dug up the best specified *evidence* for the estimates.

What a great *team* they are!

What do *you* think are the effects of 'specific detailed crediting of various ideas in a specification'. The *cost* of doing so is obviously trivial.

The effects of crediting are:

- Individuals (possibly Teams) will *refuse to let the specification stand*, if they have reason to believe it is not 'to their credit' because it is:
  - Unclear
  - Incorrect
  - Unsubstantiated
  - Can cause problems later for which they would be held responsible
  - Not properly *their* idea. *Plagiarism*, failing to credit the right person or originator: which might be discovered later.

- Anybody else reading the plan can see who has been contributing constructively to that plan. Creative and knowledgeable people can be spotted.

- If anything is unclear regarding that plan element, we know *exactly* who to ask to clarify it.

- The moment people see that their name will be integrated in the planning detail, they will try extra hard to make sure it is a 'high quality' specification, even if they have to do some extra study, and come back and re-edit it later.

- In fact, when their name is on it, they might well insist on annotations to indicate how shaky it is, such as:

  - Fuzzy brackets: <this is fuzzy, don't trust it>
  - ± uncertainty
  - ??
  - Issues: questions asked that do not have good answers yet
  - Risks: direct notice of particular risks involved
  - Assumptions: statement of pre-requisites, for this specification information, so you do not have to guess or know in advance.

- At the extreme, people will insist that their name is NOT to be noted as a source, at the present state of specification. Good!

What are the *disadvantages* of detailed source crediting?

- Shy individuals might feel that others will feel they are taking undue credit (but they can insist the others be named too)
- People might hesitate giving useful ideas, for fear of being held responsible for them, even when *badly* implemented.
- Or, they might resist giving good ideas, for fear of somebody using that to blame them, even if their idea does not bear any real responsibility for problems.
  - *We might consider allowing anonymity, as the price for noting an interesting specification.*

What are the *advantages* of detailed crediting?

- People will take pride in having their contribution acknowledged.
- People will make very sure their contributions are well formulated, and not misinterpreted by other people, noting them down.
- This should directly stimulate creativity, and better ideas [URL53].
- When Quality Control of plans encounters problems, we have a direct pointer to people we can get help to clarify the specification with.
- People who are credited, can get feedback on how the plans worked
- They can also get feedback on the quality of the plan, the quantity of their contribution, and on improvements or changes that were made by others, *later*, to *their* planning ideas.

- Plan readers will make use of the knowledge of the formal ***authority*** of the source (like CTO, Research Director) to decide how seriously they choose to take the plan element.
- Plan readers will make use of the knowledge of ***credibility*** for the individual cited to decide (or find out by asking other people for references) if that person has credibility; a reputation for being right, and giving good ideas.
- Individuals can build their reputation and credibility through being identified for their contributions
- Individuals can 'prove' that they contributed ideas; others cannot 'steal the glory'.

Since we are touching upon the stimulation of creative contributions to planning this seems like a good point to cite a paper I wrote on the subject, **Practical Purposeful Creativity** [URL53], which is my argument for Planguage as a 'creativity stimulator', and 'creativity protector'.

Some people, wrongly I believe, think that if we 'get formal', like using 'Planguage', we threaten to *destroy* creativity. Well, maybe we will destroy random invention of new stuff (sounds like 'Art' to me, and I enjoy Art).

But the moment we add the industrial, and management, idea of 'purpose', we can begin to see the necessity for *clear* purposes, to stimulate the creativity in the *right* direction.

'New', without purpose, may be *art*, but it is not *industry*, and probably not very useful to stakeholders. So let us not confuse art, with commerce or engineering.

Here are the fundamental ideas from the paper [URL53]:

Quote 8.7 A. Gilb **Practical Purposeful Creativity** [URL53]

> *"Ten fundamental principles of practical creativity.*
>
> *1. Practical creativity must have a defined purpose which is objectively measurable.*
>
> *2. Practical creativity must operate in multiple purpose dimensions at the same time.*
>
> *3. The result of practical creativity depends on the clarity of the stated objectives.*
>
> *4. The result of practical creativity depends on the nature of the creativity process and the agents employed to do it.*

5. Any creativity objectives initially defined, will tend to change as time goes on, due to changed perceptions, changed external world, and experience with delivering partial results.

6. The practical creative process follows the rules of any similar 'design' 'planning' or 'engineering process': it is merely a higher-level generalization of them.

7. The 'net value' of an additional idea for solving a defined problem, can be estimated, in relation to remaining unsatisfied objectives. How far will the idea move us in the direction of our final objectives, from where we are at the moment?

8. The degree of yet-unsatisfied objectives, for a problem being solved, determines the priority needed for continued creative effort. This (degree of yet unsatisfied objectives) is a function of previously accepted or applied ideas, and of any changed objectives since they were originally defined.

9. Seemingly 'bureaucratic' idea management processes can stimulate, protect and justify creative effort. Total freedom of thought is not necessarily the best way to get useful creativity.

10. If a creative effort fails to satisfy even a single real, defined or not, critical success factor then it is, in practice, a total failure. It serves no useful purpose."

© Tom@Gilb.com 1992&2014

Quote 8.7 B Da Vinci on The Rigor Needed for Creativity

"these rules will enable you to have a free and sound judgment:
since good judgment is born of clear understanding,

and a clear understanding comes of reasons derived from sound rules,

and sound rules are the issue of sound experience –

the common mother of all sciences and arts."

The Notebooks of Leonardo da Vinci. 18.

Practical Tip 8.7 Very frequent source annotation

- as an exercise, take a page or so of planning, and attempt to *source* every individual statement; using '**<- ?**' where unknown
  - difficult, to impossible with hindsight.
  - you need to capture 'when it is suggested', or it will be lost
- in meetings, capture the individual, or group of individuals, that are in fact responsible for having suggested a particular specification
  - capture visibly for the meeting, on screen, that you are doing this.
- add a dose of the 12 tough questions to each statement [URL15]
  - at least once in a while, just to make your point
  - but not so much as to scare people off!

Policy 8.7 Give Credit where credit is due

- **We will rigorously credit individuals, (at the exact statement or phrase of their specific contributions), for their contributions to our plans.**

- Why?
  - To reward active creative contributions
  - To encourage contributors to take responsibility for their opinions

Version 22 Dec 2015 Kolbotn 6:50 PM

Section 8.8 Delegation of planning activity and responsibility.

Principle 8.8 Delegate Responsibility

Delegate special responsibility for individual objectives and strategies, to owners, experts, and implementers, in the plans themselves.

Writing rich and useful plans is best done as teamwork. We have just now above (8.7) focussed on the idea of specifically crediting individual planning element contributions, as they occur; maybe at a meeting, maybe after a phone conversation or email.

It is a somewhat different idea to specifically delegate specific parts of the planning to individuals or small teams.

Illustration 8.8 By Andrey Pavlov. http://www.dailymail.co.uk/news/article-2111013/Antsy-fantasy-Russian-photographer-creates-fairytale-world-obliging-insects.html

Delegation can be done at the following levels of planning:

- Find **specific strategies** for one or more specific objectives and goals; essentially forming your support team (your team for finding strategies to meet your own Goals.
- Detailing **specific objectives**, or other requirements, such as legal constraints to
    - specialists, experts,
    - people who have more time,
    - people who need to *learn* these subjects,
    - or need to take *responsibility* for them
- Detailing the **multidimensional estimates** for certain strategies, across all critical objectives, and all critical resource budgets
    - Essentially building part of an Impact Estimation table [URL21], which other team members can build other parts of, in parallel.
- Take responsibility for setting **goals in long or short time frames**, in certain market areas, or for certain critical conditions (like natural catastrophe)
- delegating the **decomposition** of top-level strategies into smaller independently-implementable value-delivery steps (a week each, or 2% of total time or money budget chunks). [URL16]
- To take responsibility for **quality control** (is it intelligible?) of specific parts of the plan; parts they are motivated and competent to check. [URL23]
- To do a **content review** (is it the *right stuff* for higher level needs? Is it complete?)

The possibilities are many, but hopefully this gives some specific ideas of what we might delegate to a diverse planning team.

***Why* might we want to delegate the planning effort and responsibility?**

In simple terms: we want to delegate planning for all the usual reasons we would want to delegate anything else, such as…..

- To get *specialist or expert* knowledge; one planner cannot be expert on all strategies and objectives
- To get plans done *quickly by parallel* work teams
- To *involve key stakeholders* directly, to solicit their needs and knowledge; and finally their commitment
- To *train newcomers*, either by handing them a new responsibility, or letting them work with a more-experienced mentor: 'Pair Planning'.
- To increase motivation, through 'ownership' at several instances (it's *our* plan, *my* plan, I *helped make* that plan)
- To increase the probability of getting other people's *reaction* to bad, unrealistic, or unintelligible planning, done by others
- To involve specific *authorities*, internally at least; HR, CTO, IT Architecture, Trade Union, Marketing, Sales, Finance, Legal, etc.

"If you delegate tasks, you create followers. If you delegate authority, you create leaders."

Craig Groeschel, the founding and senior pastor of LifeChurch.tv, at The Global Leadership Summit, 2012.

http://www.liveintentionally.org/2012/08/09/2012-global-leadership-summit-session-4-craig-groeschel/

Here is a case study of delegation:

Orbit Application Base:
Type: Primary Architecture Option
==== Basic Information ==========

Version: Nov. 30 20xx 16:49, updated 2.Dec by telephone and in meeting. 14:34
Status: Draft (PUBLIC EXAMPLE EDIT)
Owner: Brent Barclays
Expert: Raj Shell, London
Authority: for differentiating business environment characteristics, Raj Shell, Brent Barclays (for overview).
Source: <Source references for the information in this specification. Could include people>. Various, can be done later BB.
Gist: risk and P/L aggregation service,
which also provides work flow/adjustment and outbound and inbound feed support. Currently used by Rates Extra Business, Front Office and Middle Office, USA & UK.

….. below this, is the detailed decomposition of the strategy, and the Risk specification (Issues, Assumptions, Risks, Dependencies)

Planguage Example and Case 8.8 A The slightly 'doctored' real header for a bank project. To illustrate specialist functions. This example was delegation of one major strategy draft to the man considered the expert on it. Other strategies were developed in parallel the same day by other members of the team. This was done on the second day of the Startup Week [URL5], the 'Strategy' day, after the 'Objectives' day.

Case 8.8 London Bank Case, including example above.

- In this real London Bank case
- We had a team of about 10 people working out the strategies in parallel, out of one meeting room. This was the second day of our start-up planning week [URL5].
- I (Tom) stepped out of the meeting, to phone the subject expert for this particular strategy (Brent), who was unable to come to the office that day, due to snow conditions, affecting rail travel.
- I copied the full 'design strategy template with hints', from the Competitive Engineering book [1], and I used it to 'interview Brent, regarding his knowledge of the strategy.
- He accepted the role of specification Owner. The only one authorized to make official changes to it.
- He was able to name a subject expert (Raj), a person who had deep understanding of the strategy, who could be asked questions, and do content review.
- He was able to name two Authorities (people who could make decisions about plan content): Himself and Raj.
- Looking back, I think we should have added a 'Stakeholder' list, for this strategy (see 8.9 below); though we definitely had them listed for all the objectives, which this was designed to satisfy; so by inference, they *were* identified.
- The entire telephone interview, which produced a very full page of specification (this is 20% of it) took one hour.
- I could then return to the strategy team meeting room and present our joint effort. I was really just a structured scribe.
    - Partly as a 'good example', since Kai and I were coaching them.
    - Partly to complete the set of strategies for the days effort.
    - Ready for Impact Estimation, the next day.

"Too many companies believe people are interchangeable. Truly gifted people never are. They have unique talents. Such people cannot be forced into roles they are not suited for, nor should they be. Effective leaders allow great people to do the work they were born to do."

*— **Warren G. Bennis** (March 8, 1925 – July 31, 2014 (less than a month before I decided to quote his wisdom), **Organizing Genius: The Secrets of Creative Collaboration***
http://en.wikipedia.org/wiki/Warren_Bennis

Practical Tips 8.8

- make sure every team member has responsibility for something all alone.
- pair up more-experienced people with novices.
- ask for volunteers for doing parts of the planning that people are motivated, interested, and competent to do.
- assign responsibilities visibly, with names in the specifications, as well as source credits for details.

Policy 8.8 Distribute Planning

- **We will consciously and intelligently delegate specific planning tasks to individuals and teams. It will be a clear 'team effort'.**

- Why?
  - To get the best possible plans early
  - To get the best communication between co-planners and plan readers.
  - To get commitment and motivation.

Version 22 Dec 2015 Kolbotn 23:55

Section 8.9 Managing Stakeholder Awareness

Principle 8.9 Keep Stakeholders Connected To Plans

Keep a written list of top stakeholders integrated with every objective and every strategy

One weakness in most peoples planning methods, that I see, is that stakeholder analysis is not done *consciously* or *systematically*.

Planners might focus on a user, client, customer, or market segment. But the stakeholder perspective is *much* broader.

We have discussed stakeholders fairly thoroughly, thus far in the book [Chapter 3, 4 and elsewhere]. But here are additional stakeholder ideas.

Figure 8.9 A Various stakeholders for a Downs Syndrome (DS) project. Dr. Lindsey Brodie, 2014, PhD Study [URL75 , URL38].

Here is what we are suggesting you can do in your stakeholder planning:

- Make sure you have a *relatively thorough* list of relevant, and critical, stakeholders, concerned with your project.

- The list should probably be, order-of-magnitude 40, stakeholders, even for medium-sized projects: otherwise I suspect you do not understand the concept, and are oversimplifying

- For each major objective, use the Stakeholder statement to clearly document which stakeholders are *primarily* interested in it (usually 3 to 6 stakeholders is fine, the major ones)
    - **Stakeholders**: New Users, Potential Users, EU Law, Marketing.

- In *objectives specification*, use the 'Stakeholder' parameter, or equivalent (User Type, Customer) to indicate specific levels of performance on a Scale:
    - **Goal** [Next year, Stakeholder = Teenagers] 30%

- In *Strategies*: Indicate which stakeholders have some interest in the strategy.
    - **Stakeholder**: Maintenance Technician

- In *Evo Step Planning, specify exactly which stakeholders will receive the step, and others concerned with that step, for example: implementers.*

> **Value Delivery Step 23:**
> **Recipients:** People who make complaints.
> **Implementers:** IT Systems, Database Managers, Service Office.
> **Other Stakeholders:** Consumer Protection Institute. Consumer Laws.

- In Corporate Level IT (or systems engineering), **keep a database over your stakeholders** in general, their general needs to be considered, and any other potentially useful information about them.

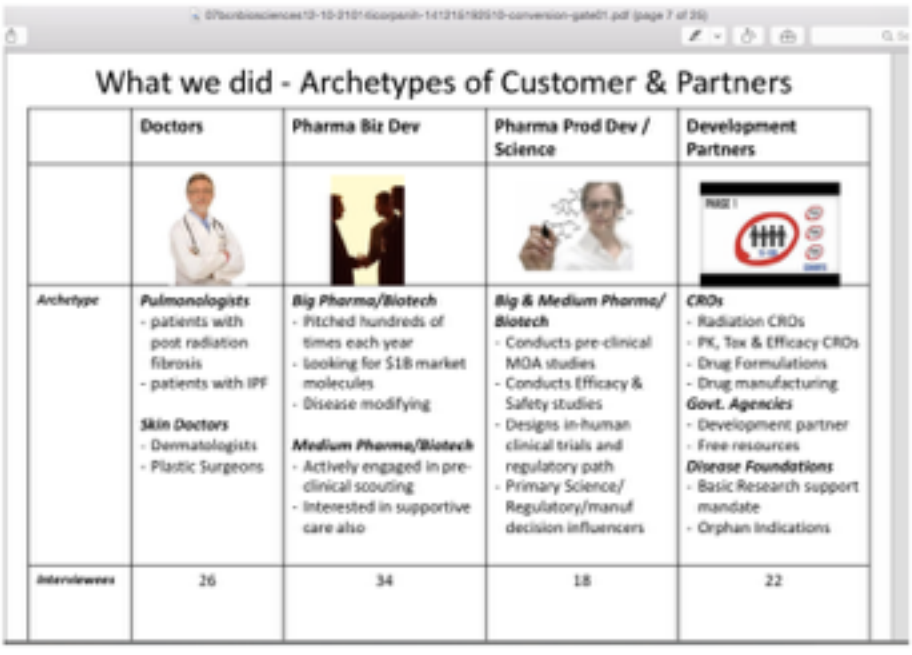
- By building *cumulative knowledge* about your stakeholders, there is less effort needed for the single *new* project to get it right. And there is less chance that you will forget a critical stakeholder on a specific project.
- Such systematic stakeholder databases are not found in my experience. But then the rigorous practice of stakeholder analysis is rarely found either, so perhaps my suggestions are a few decades ahead of real practice here.
- But, at best we seem to be in a mode of 'reinventing the wheel', for each project regarding stakeholders. Rather than any systematic development of stakeholder knowledge. Let me know if you are the exception to prove the rule.

- Include in your 'standards for specification' (your 'Rules') strong best-practice guidance, which will be applied when doing Specification Quality Control.

Rule 'Goal Stakeholders': The 1-5 most critical stakeholders will be specified for specific elements of an objective; like specific Goals.

Rule 'Generic Objective Stakeholders': All objectives will list the 1-5 most critical stakeholders, in general, for that objective, before we get into the detail of specific targets and constraints.

- These Rules, if applied to find defects, in a regular Spec QC process [URL23], will remind planners of the necessity of specifying stakeholders. Because their plans will incur a specification 'defect' if they do not specify stakeholders. This defect will be enough to prevent the plan from exiting the planning process.



What we did - Archetypes of Customer & Partners

	Doctors	Pharma Biz Dev	Pharma Prod Dev / Science	Development Partners
Archetype	***Pulmonologists*** - patients with post radiation fibrosis - patients with IPF ***Skin Doctors*** - Dermatologists - Plastic Surgeons	***Big Pharma/Biotech*** - Pitched hundreds of times each year - looking for $1B market molecules - Disease modifying ***Medium Pharma/Biotech*** - Actively engaged in pre-clinical scouting - Interested in supportive care also	***Big & Medium Pharma/ Biotech*** - Conducts pre-clinical MOA studies - Conducts Efficacy & Safety studies - Designs in-human clinical trials and regulatory path - Primary Science/ Regulatory/manuf decision influencers	***CROs*** - Radiation CROs - PK, Tox & Efficacy CROs - Drug Formulations - Drug manufacturing ***Govt. Agencies*** - Development partner - Free resources ***Disease Foundations*** - Basic Research support mandate - Orphan Indications
Interviewees	26	34	18	22

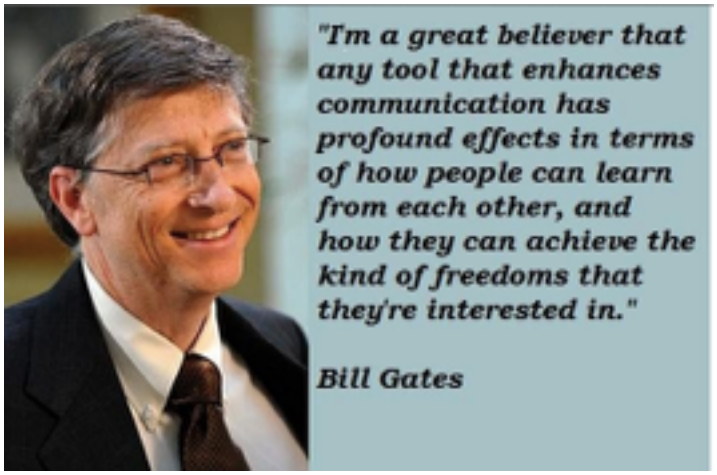
Figure 8.9 B. Stakeholder perspectives from a medical project. Notice the 'Archetype' characteristic (giving a hint as to values), and the number of stakeholder representatives actually interviewed.

Here is *why* you should keep stakeholders visible:

- by definition, the stakeholders we are really concerned with are the *critical* stakeholders. The ones that can make or break your project.
- If a critical stakeholder is forgotten in planning, then there is a big danger (unless you get lucky, and the forgotten stakeholder's needs are accidentally covered by *another* type of stakeholder) that you will be *unaware* of their needs.
    - You will *not include* them in your objectives or constraints.
    - You will *fail to deliver* them.
    - The critical stakeholder *will react*, and *threaten* your project. Like the potential buyer who does not buy.
- If you were to analyse why projects fail so frequently, to some degree or other, I think you will find at root, lack of stakeholder needs being taken seriously in the planning, for one reason (existence, prioritization, etc.) or another.

"Helping convene global stakeholders to establish a set of measurable, actionable and consensus-built goals focused on extreme poverty is invaluable."

Quote 8.9 A Bill Gates (1955-)


"I'm a great believer that any tool that enhances communication has profound effects in terms of how people can learn from each other, and how they can achieve the kind of freedoms that they're interested in."
Bill Gates

Quote 8.9 B

Practical Tip 8.9

- *at least for every Goal statement, at least 1 Stakeholder for that Goal should be specified*
- *make a list of your top 10 most-critical stakeholders, and challenge people to find another stakeholder more-critical than one other on the list.*

Policy 8.9 Keep Critical Stakeholders Visible

- **We will make our critical stakeholders visible in writing, in near proximity to the specifications that concern them.**

- Why?
  - To make sure we develop and review the related specifications, with regard to those stakeholders
  - So that those stakeholders can be invited to comment, judge, review, and accept the specifications as correctly relevant to real stakeholders.
  - to make better plans, and get fewer delays and failures.

Version 23 Dec. 2015 Kolbotn 00:17

Section 8.10 Making sure you get 'value for money'

Principle 8.10 Don't Freeze Unproven Stuff

**Keep strategy options open-ended;
do not commit or contract for 'strategies'
until they are proven by measurable *results* delivered.**

**Make main payments, bonuses, rewards dependent on
timely success in delivering quantified objectives**

The planning decision to *try out* a promising strategy, should not, in practice, be a decision to *commit* to that strategy.

Approval of a trial, cannot be taken as an approval even when the trial consequently fails.

Commitment to continue using a sub-strategy should follow, only when results deserve it, and continue to deserve it.

Diagram 8.10 A. The Process of Commitment to Strategies. All of these are planning sub-processes.

Comment on the Commitment Diagram above:

There is a *logical process* of *gradual* commitment to a strategy URL90]. (above diagram 8.10)

1. You cannot finally commit to a strategy, unless you have clear and committed objectives for the strategy to satisfy, and to be compared to.
2. You need to make *multidimensional estimates* (of all critical objectives, and resource budgets) for a strategy; to ensure that the strategy is a reasonable overall bet of 'good value for money'.
3. But a *big* strategy idea, can *unfortunately but unavoidably* contain a mixture of far-better-than-average sub-strategies, and possibly some really 'deadly' sub-strategy possibilities. So you need to decompose the top-level strategy ideas, and estimate the impacts again, at that decomposed level; in order to pick some sub-strategy 'winners'; to try out (and see if they really *are* winners!)
4. Then you need to estimate the impacts of the sub-strategies. You can use that estimation data, to decide which sub-strategy should be implemented, on a trial basis, first.
5. You *try out* a promising sub-strategy, and see what it is *really* good for.
6. If unsatisfactory, dump the sub-strategy. Try something else that looks better. The main point is that your cannot safely commit to a sub-strategy, let alone its parent strategy umbrella concept, until you have reached this stage and found the implementation to be good enough to keep.

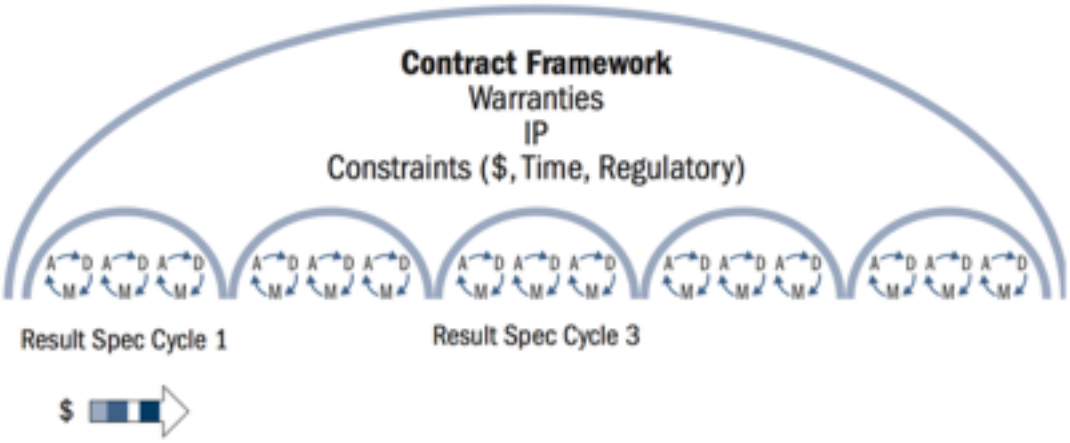


Diagram 8.10 B The concept of [flexiblecontracts.com](). Each stage of value delivery (Result Spec Cycle) is a separate contractual unit, within the overarching legal contractual framework.

Why you need to _avoid_ pre-mature commitment in the planning process:

- You can get *locked in* to long-term contractual without benefits
- The *motivation* for a supplier, to make sure that things really work as advertised, or expected, *might disappear*
- Your *own* staff would be *less motivated* to make sure the values are *really* derived in practice. The game becomes 'implement the strategy', not 'make sure we really get the necessary results', especially lagging measures.
- You might destroy motivation to make serious impact estimates. The attitude would be; 'why bother?', 'we are going to implement it in any case, so let's get on with it'.
- You need to make sure the initial satisfactory, estimated impacts are not a 'flash in the pan'. Make sure results continue in the long term.
    - Value will not continue if, for example, key support experts are withdrawn by a supplier, or the motivation to do things properly by staff disappears in the longer term.
    - There needs to be a long-term measure of what is expected, so we can prove that it is still there, or has disappeared.
    - This planning process is not about 'the strategy is in place'; it is about 'the results are still there'; *even after* motivation is gone, and after many additional sub-strategies are incremented on top of it, into the same system – possibly destroying the initial good results.

Traditional Contract Model	Result Contract Model (Agile)
Requirements are contractual and specified up-front in the main contract.	Requirements are specified at the start of each result cycle.
Changes are managed by means of the change control mechanism.	Requirements are more resistant to change than traditional output requirements. Target outcomes are only specified at the start of each result cycle, are operational for shorter periods of time, and therefore are exposed to less change.
Analysis, design, development, and testing occur sequentially. Big Bang or Waterfall.	Each cycle must deliver value, so design and development occur concurrently. A systems view must be taken, providing real results in real life.
An all or nothing solution.	The solution evolves as a serious of result deliveries.
Constituent modules of software are worked on independently until integration takes place.	There is continuously working and stable software and hardware system.
Testing is used as a contractual tool at the end of the development process.	Testing occurs throughout the development process, providing feedback for improvements.
Success is measured by reference to conformance with the change-controlled contract.	Successs is measuered, cycle by cycle, by requirements delivered, driving value to the customer.

Table 8.10 Source: flexiblecontracts.com. A comparison of traditional and value step contracting.

***How* to avoid commitment in the planning process:**

- Contract for, and pay for, specified numeric results [URL6]
- Agree with suppliers that you will be at liberty to publish the results of their work, with honor for results delivery, and infamy for bad results. Maybe just internally, maybe not.
    - They can add comment to any publication, to defend their position, or tell others they can do even better now.
- Use relevant motivation for your staff and teams to achieve and maintain the results.
- Include a specification, in your 'No Cure No Pay' contract [URL6], that states that their Strategy, will not be paid for just because it gets the results specified (the main objective to be impacted by this strategy).
    - A strategy *must also **not** be* the direct cause of any *serious side effects*, especially on the other Critical Objectives, in the long term, or have unplanned bad effects on resources of any kind (such as maintenance costs, or operational costs).
- Be explicitly clear, in writing, to all stakeholders and parties concerned, that the strategy or sub-strategy is *NOT finally a commitment until* it has proven the results expected, and done so without disqualifying side effects.

<u>Chocolate Cookie Motivation</u>:
Type: Sub-strategy of Food Motivation.

Stakeholders: Cookie Suppliers, Employees, Lunchroom Workers, HR Director.

Description: Free Chocolate Chip Cookies will be given to Teams and individuals who complete at least 80% of scheduled tasks on time.

Impacts: Goal = Timeliness Motivation [Next Year, Office Staff] = 40%.

Cost Estimation: €40/annually per employee.

Assumptions:

A1: no firm commitment of any kind, to any Stakeholder, to keep this sub-strategy in place, is made until after a successful one year trial in at least one office location, with at least 50% of planned impact at ±20% of expected costs.

A2. And even after the successful trial the sub-strategy must maintain approximately the same results, to be kept in place. This will be reviewed annually by HR.

Planguage example 8.10 Example of making the conditionality of a sub-strategy clear, when planning

"The value of an idea lies in the using of it."

Thomas Alva Edison 1847-1931 Quote 8.10

Edison with early prototype of iPhone.

Practical Tip 8.10

- *avoid all standard supplier contracts, unless they are based on payment for measurable planned value delivered*
- *do not buy any products or services where you get locked in to high fixed costs, even if expected value does not materialize, or does not continue.*
    - *I find you can usually rent, borrow, trial, or contract for value if you want to be sure; and to not waste money (even when people try to tell you that is impossible)*
    - *you will motivate the supplier to help you figure out how to get value, so they get their money.*

Policy 8.10 Don't get married, just because you got engaged

- **We will plan carefully, openly, document, and publicize that the initial choice of strategies for integration and trials, does not constitute a long-term commitment to them. They must show the results expected, to survive.**
- Why?
    - So that we are flexible enough to continuously improve the real results
    - So that we shift focus from 'do the strategy' to 'get the results'

Version 23 Dec. 2015 Kolbotn 01:29

Chapter 9. Communication

"In business, words are words; explanations are explanations, promises are promises, but only performance is reality."

Harold S. Geneen
(January 22, 1910 — November 21, 1997), was an American businessman most famous for serving as President of the ITT Corporation. Quote 9.0

Planning is partly communication. Good planning requires good communication.

Our planning language, Planguage, has evolved to improve your ability to both write plans, and read plans.

Planning is not exactly 'poetry' or 'fiction': but planning can still stimulate your imagination and creativity [URL53].

A challenging Goal can lead to invention of imaginative new strategies. But the *intent* of a plan should leave nothing to your imagination.

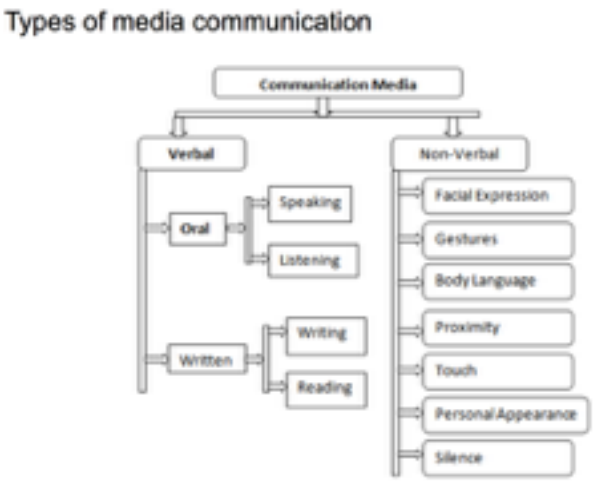


Figure 9. Communication's rich repertoire. http://thebusinesscommunication.com/types-of-media-communication/

This chapter will get in to some interesting aspects of using Planguage to communicate a plan better.

Planguage is a large toolbox of specific methods of communicating between people, without wasting their time by misunderstanding the planner's intent.

As the Harold Geneen quotation above points out, plans are just plans, but only the results of those plans are the interesting reality; the value delivered itself is king.

Section 9.1 Written Communication

Principle 9.1 Written Words Persist

Written plans beat oral plans, in the long run

There is a myth about communication, that some people believe in, and teach.

It is that *oral* communication is superior to *written* communication [URL54]. So they want to virtually eliminate written communication, and mainly use oral dialogue: meetings, telephone, videophones, video meetings etc. for planning.

Our position is that oral communication is great, not least for *collecting information* that goes into a written plan. It is also great for *presenting* some aspects of a plan, for some purposes.

If a plan is simple, and must be made in very short time frames, then 'oral' may be the best and only way. Think 'team games' in sports. Think military action.

We also understand very well, people's negative reaction to 'bad' plans (irrelevant, doomed to fail), to bureaucratic plan cultures, to unclear plans, to unnecessarily voluminous plans, and to theoretical plans without root in reality.

We understand the consequent knee-jerk reaction to these evils. Dump the written plans, *simplify*, by going over to oral planning.

Our assumption, in this book, is that we can radically reduce the above problems that written planning often has been guilty of.

We can make written plans simple, powerful, useful, intelligible.

- *For example:*
  - *the entire project is predominantly run using the top ten critical objectives, on a single page, but quantified; and planned in a single day. [URL1, 2, 3, 4]. That* is simple, written, and powerful.

We can make written plans, very usefully, for large systems and projects.

We believe that we can make useful written plans for both:

- **complex** systems (compound in structure; consisting of interconnected parts),
- for **compound** systems (contain two or more elements), and
- for **complicated** systems (Anything, that is difficult to understand, estimate or predict the actions, reactions, and attributes of.

We can have the best of both the oral and written worlds. We can try to find a balance between the use of oral and written tools.

But we cannot eliminate either, oral or written, communication devices, just because it is too often done badly, by people not trained, led, coached, or motivated to do better.

We need to improve **both** oral and written communication.

One belief we have, is that by improving the *written* planning language, we will also *simultaneously* improve all oral interaction with the written plan.

- We will provide a **much better structure** for *collecting* planning information. For example:
  - 'Which scales of measure are relevant for your 'adaptability' objective?"
  - 'What is the minimum tolerable level of Adaptability we can accept for Teenagers, in China, Next Year?'
- We will provide a much ***better basis for all related meetings***, with oral discussion and analysis of the plans, such as getting buy in, getting additional or improved information, quality control, and review of the plan. For example:
  - 'let's drill down to the detail for this objective'
  - 'lets look at the credibility of the strategies on the Impact Estimation Table'
- We will provide a much ***better basis for any oral presentations*** to audiences, about the plan content, intent and risks.
- The oral presenter, whatever time constraints, whatever audience reception constraints, can rely on the more-detailed structure and background information of written plans, available to both the presenter, and to the listeners.

- We can use presentation slides, which *directly point to the detail* of the plan, so it can if necessary, be clarified better after, or during, an oral presentation.
- The oral presentation will be an attempt to transmit interesting aspects of a plan, to an interested audience: but it will NOT replace the official, details, well-balanced plan itself.

Diagram 9.1 A. The relationship of the written plan, to a variety of oral activities, in connection with it.

We need to find the best, and for *us appropriate*, mix of written and oral communication: the mix that gives us the best return on our communication-and-planning investment. If it does not clearly pay off, we should not do it. If it does, we should.

The answers to 'what is *exactly* the right thing for us on this project' will, of course, vary, depending on the project environment, circumstances and economics.

The reader needs to view all ideas, in this book, and from other sources, as *potential tools*.

Readers will hopefully understand the proper uses of the tools, and make intelligent; often on-the-fly choices, to use, or not use, particular tools in the planning process.

Know your tool set, and use them like a master craftsperson would, is the basic idea.

Here are some reasons that we need to take the *written form* seriously, in general, compared to oral forms of planning and plan retention.

Because the *written* form :

- is *stable*: not subject to fading memory
- is *comprehensive*: includes all necessary detail.
- can be *systematically* improved and changed
- can be *reviewed* and quality-controlled against high enough standards.
- can be *referenced* in any discussion, meeting, presentation
- can be read and reread at the individual's *own pace*
- provides a *legal record*, and can be referenced in contracts, and legal cases.
- allows detailed *interrelationships* between planning elements to be mapped rigorously.
- can easily be communicated widely geographically, independently of people's schedules

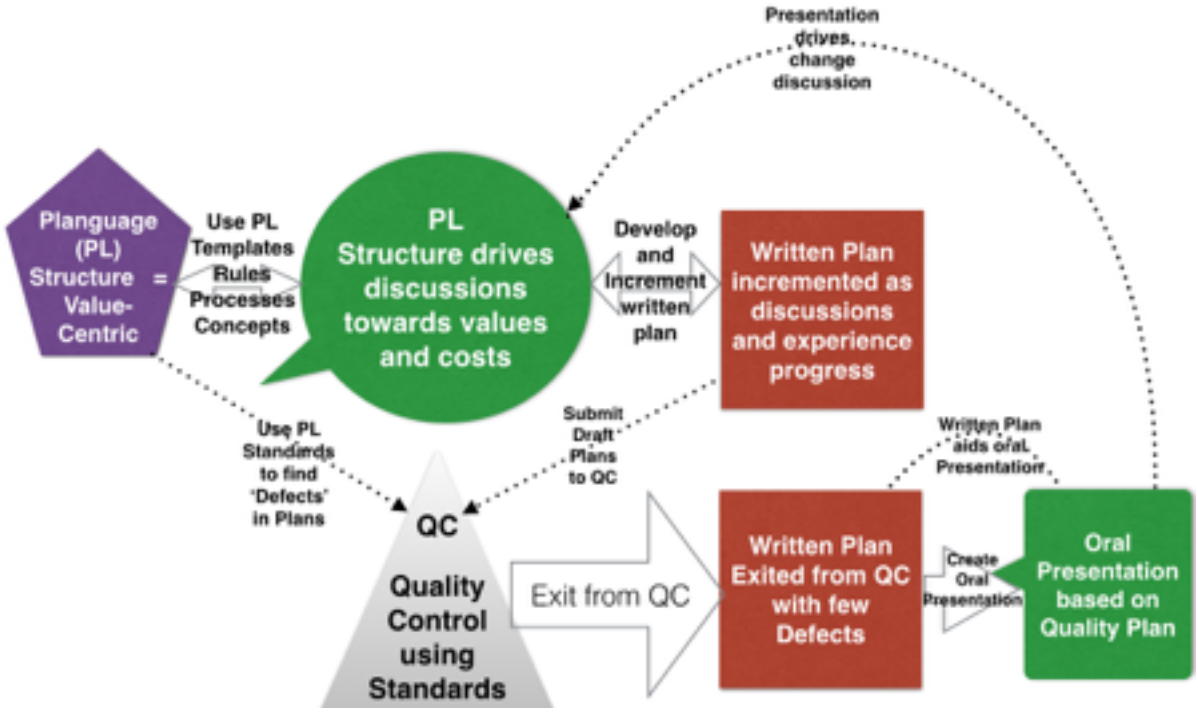


Diagram 9.1 B. ***The symbiotic relationship between oral planning processes (discussion, presentation, Quality Control) and written plans.*** *Planguage as a tool for planning helps to structure the discussion (example 'What are the top ten values quantified?'). The discussion can be structured around a planning process such as Startup Planning Week [URL5]. All draft plans should be quality-controlled before serious presentation, review or approval. The oral meeting process for presentation, review and approval can be simplified, clarified, and more trustworthy - if the understanding is that it is the 'written QC-exited plan' that is the subject of discussion:* ***not*** *any oral misunderstanding, bad unclear draft plan, or presentation slide oversimplification.*

Here are specific Planguage specialities that make the written form far better than conventional planning is usually done*:*

- Planguage is in early, frequent *direct contact with real world* plan implementations, via Evo (Evolutionary) delivery of value and feedback.
- This also means that much detailed *planning itself,* can and should be done, *on a continuous basis*, as we implement and get feedback.
- Planguage is based on a large number of smart 'Rules' [1] for planning, which give *strong guidance* on how to make things extremely clear. *Example: "Give evidence for your estimates".*
- Planguage is particularly designed to *map the large number of relations* between plan elements, in a simple and direct way.
    - Example: *Stakeholders: A, B, and C*
- Planguage is extremely conscious of the *necessity to quantify all critical objectives and all types of costs*; especially qualities.
- Planguage has a large number of built in mechanisms for dealing with threats, attacks, damage, mitigation and risk in general.

"By failing to prepare, you are preparing to fail."

Benjamin Franklin (1706-1790)

Practical Tips 9.1 Getting things written

• *Write down critical oral remarks, and feed them back to the person who said them, to make sure you understood them correctly.*

• *Integrate clarified remarks, into the plans, with source annotation.*

• *When you do not understand what people say or write, then mark it with <fuzzy brackets> to remind yourself and others that it needs clarification.*

• Make your personal standard for written plans 'unambiguous to the intended readers, and clear enough to test'.

• Written does not mean 'verbose'. Keep it short and simple, and non redundant (reuse specifications (Tag and refer), do not repeat them).

Policy 9.1 Written is official: oral is a support process.

- **Our primary planning instrument is the written plan. Nothing less, is official, or useful, as a plan, or even as a proper basis for oral discussion or presentation.**

- **Why?**
- Written plans can be stable, detailed, precise and can be brief.
- Oral decisions, cannot be trusted to be understood, and cannot be systematically quality-controlled or reviewed.
- Oral discussion should lead to the written plan.
- Written plans can be used to back up oral presentations.

Version 24 December 2015 (I just turned 75, in Norway at least, 00:22

Section 9.2 Capturing Background Information in your Plan

Principle 9.2 Write Remarks In Plans

When questions are asked,
***document* the question and answer,**
***integrated* into your plan:**
don't just hear it orally,
don't let it exist only in another document.

I have a strong personal habit when sitting in meetings, or discussing plans. I write notes. So do other people. But I incorporate my notes *directly into the official plans. During the meeting.*

Here are some of the things I do:

- If someone adds to the information, I "quote them", and cite them as the Source (<-).
- If someone answers a meeting question, I note the answer in the plan; somebody *else* might like to know.
- If someone raises problems, I note the problem. Probably under *Issues, Dependencies, Assumptions or Risks*. [Planguage Examples 2.2A, 2.2B, 3.10, 9.2]
- If a new stakeholder is mentioned for any element of the plan, I try to make a note of that stakeholder, in the appropriate place.
- This list has quite a few more specific things I do, but this is enough to get the picture.

What I am noting, is in my category of 'Background' specification. That is, most of these things are not the primary objective or strategy (the 'Core' specification). Background specifications are 'additional related stuff' to the core.

Figure 9.2A (Copy of Fig. 2.2) Some examples of Planguage Background statements worth capturing in relation to a core specification strategy.

But you never know how useful background specs might be, until it is available, over time.

Background information is relatively cheap to collect. It is a kind of insurance. It beats doodling at a boring meeting. And plan readers can easily ignore it, if it does not serve their immediate needs.

But a single background specification element might save a project, or reduce losses downstream. You never know. It is a kind of 'Lean' (upstream, preventative) tactic. Proactive, rather than reactive.

Planguage statements contain very clear headings, which show us where the Core specification is (Scale, Goal), and equally clear headings (Includes, Type, Ambition) and other devices (like '<-', and *<italics as comments>*) which allow us to find, or scan by, the background, as appropriate, just like newspaper headlines help you skim the news.

The extra specification does not have to disturb us, or weigh us down, which it might well do in less-structured planning languages.

Automated Planguage tools can help find or hide background details [URL73]. The Planguage structure is specifically designed to enable information to be automatically selected or ignored.

These special tools permit simultaneous updating from several different people, as do simpler tools like for example Google Docs, and Apple Pages (Keynote and Numbers).

We can easily focus on the plans Core information, and drill down to the Background as needed.

Figure 9.2 B. Stages of incrementing a fully fleshed-out planning specification.

Here is why I feel *motivated* to capture these secondary things, because:

- There may be people *outside* of the meeting who *need* to know, and would *like* to know.
- There may be *future* people, who join the team, who will *need* to know.
- There may be people in the meeting who missed it then, but need to know it *later.*
- It might help reviewers, and QC processes, to correlate specifications for completeness, correctness, inconsistency, and lacking specifications.
- If *someone* asked a question at all, then others may have the same need; and that is a strong indicator of *value* of writing the answer in the plan itself: not *just* letting the question get answered in a meeting alone.
- It gives a list of things that someone can be assigned to look into after the meeting, or after a rough draft plan (‘resolve all issues’, ‘mitigate all threats and risks’, define all <fuzzy terms>).
- It keeps me awake and conscious at the meeting; I am consciously listening for valuable tidbits I can capture.
- Many of these things should ideally have been included in the initial draft of the plan, but the single planner, or Plan Owner, cannot be expected to know all this detail. Although others at the meeting *do* know, and *do* bring new things up at meetings.
- The primary author of the plan, can use this as a tactic, to enrich the plan, with potentially useful detail.
- When people realize their remarks and answers are being quoted, they will take their own remarks, and perhaps those of others being quoted, more seriously.
- Sometimes, in order to make a note, I *just then* recognize that I need to ask *supplementary* questions, which then brings out even *more useful detail*.

Development Capacity:
Version: 3 Sept 20xx 16:26
Type: Main <Compound/Elementary> Objective for a project.
Ambition Level: radically increase the capacity for developers to do defined tasks. <-Tsg

Scale: the Calendar Time for defined [Developers] to Successfully carry out defined [Tasks].

Calendar Time: defined as: full working days within the start to delivery time frame.

Owner: Tim Fxxx.

Past [at 20xx, Site = {Bxx, Lxx, Gxx}, If QA Approved Processes used, Developer = Architect, Task = Draft Architecture] 15 days ±4 ?? <- Rob

Goal [By 20xx, Site = { Bxx, Lxx, Gxx }, If QA Approved Processes used, Developer = Architect, Task = Draft Architecture] 1.5 days ± 0.4 ?? <- Rob

Justification: Really good architects are very scarce so we need to optimize their use.

Risks: we use effort that should be directed to really high volume or even more critical areas (like Main Objective).

Planguage Example 9.2: Actual capture by me at a meeting, in 20 minutes time, with departure point, a badly written objective, 'Improved Development Environment'. I am listening in to remarks of 4 client people, and asking some structured Planguage questions. 'xx' = real data obliterated in this edit.

"Plan for what it is difficult while it is easy, do what is great while it is small."

— ***Sun Tzu, The Art of War***

Practical Tip 9.2 Capture Background Personally

- *at a forthcoming meeting keep a personal log, or notes, of issues, answers, implied stakeholders, implied values, and risks.*
- *ask yourself: if I don't put these in the plan, what happens?*

Policy 9.2 Grab Background while it flows past you.

- All meeting participants, or at least one person *assigned* to the task (**'The Listener'**), will capture a wide variety of potentially useful, or necessary Background information, directly into the planning specification.
- And submit them in writing to the Plan Owner.

- Why?
  - Better to invest now, than get problems later.
  - Impossible for the initial planner to know it all initially.
  - **"the pen is mightier than the sword"** <- Edward G. Bulwer-Lytton (1803 –1873).

Version 24 12 2015, 00:48 CET, TsG

Section 9.3 Traceability: Document Relationships

Principle 9.3 Document The Impacts

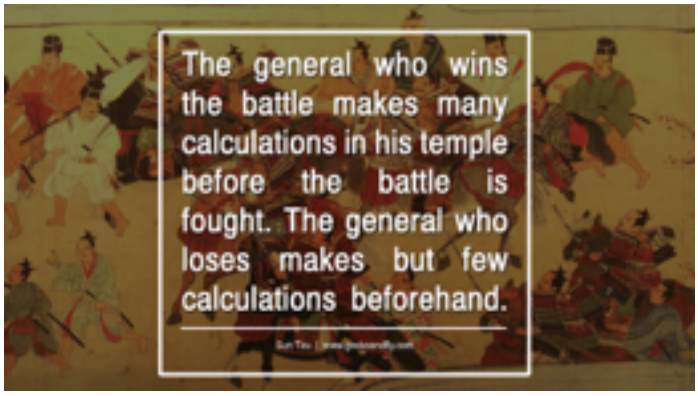
Document the main relationships upwards, downwards and sideways, between objectives, strategies, and other ideas

'Everything' *potentially* impacts 'everything else'.

In planning it is, of course, important to be aware of many plan element relationships. Relationship insights are important because: planning is *early*, and bad relationships and impacts are *later and costly*, perhaps even 'deadly' for your project.

It is very inexpensive to capture information about relationships, when planning. The knowledge is there during the thinking, the discussions, the meetings, the reviews. Just grab the information, flying through the (hot) air, and make a quick note. Too many, rather than too few.

If 1 of 100 10-second relationship notes, saves your project; that is a 100,000 to 1 payoff at least. Just do it. Little to lose, much to gain.



Quotation 9.3 A. It is ancient wisdom, Sun Tzu, to plan thoroughly, if you want to win.

Here is a diagram reminding us of some of the interesting relationship types:

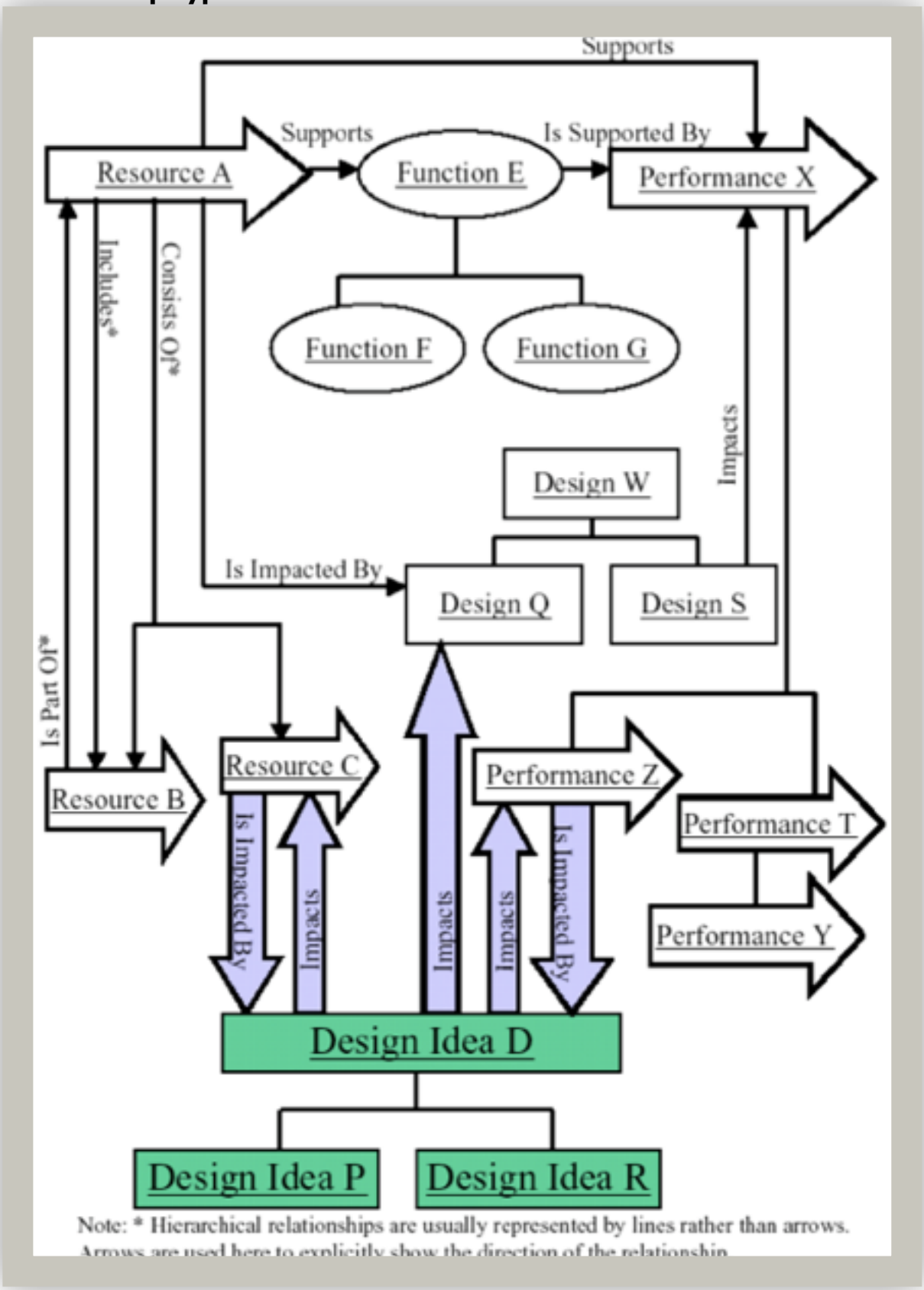


Figure 9.3 A. Some graphical relationships between planning elements. Design = Strategy, Performance = Objectives.

Here is a list of some Planguage-defined relationship parameters. You can extend it yourself as needed:

1. Ambition
2. Authority
3. Source
4. Object
5. Owner
6. Author
7. Consists Of
8. Dependency
9. Depends On
10. Downstream
11. Expert
12. Goal
13. Hierarchical
14. High-Level
15. Implementer
16. Impacted By
17. Impacts
18. Includes
19. Interface
20. Kid
21. Kin
22. Meter
23. Owner
24. Parent
25. Is Part of
26. Supports
27. Supported By
28. Version
29. Set
30. Sibling
31. Derived From
32. Scale
33. Stakeholders
34. Sub-component of
35. Sub-components {list}
36. Supported By
37. Supports
38. Supra
39. Type
40. Dependencies
41. Contract
42. Test Case
43. Upstream
44. Scenario
45. Subset
46. Model
47. And more!

Figure 9.3 B. Some formally defined Planguage 'relationship' parameters. There are more.

Almost all Planguage parameters express some kind of relationship to something else.

The advantage of a relationship parameter (**Parameter: <thing pointed to>.**) is that it simply points to one or more other things in the entire planning specification. A connection is implied, to the artifact, like objective or strategy, being defined just *above* it.

No arrows, no tables. Yet the statement is unambiguous enough to be tracked by a computer, which can chart all relationships of any selected type. For example, track all hierarchies or decompositions.

Part of why this works is that the specification parameter statements are standardized and defined; at all other planning entities have a unique tag, Capitalized.

Another strong Planguage type of 'relationship indicator' is the Impact Statement, and the Impact Estimation Table.

The impact statement:

Strategy B	**Impacts**	**Objective Y**
Strategy A	**->**	**Objective X, Objective Z**

'Impacts' indicates a relationship between a *means* and an *end*.

But that just raises the next, and far more interesting question: 'how *much* is the impact?'

Strategy A	**->**	**Goal X**	**by**	**10 minutes**
Strategy B	**->**	**Goal Y**	**by**	**30%**

The first statement estimates the saved minutes as defined in Goal X, if Strategy A is implemented.

> 'Strategy X impacts Goal X by 10 minutes'.

The second statement estimates that Strategy B will move us from a defined Past level to a defined Goal Y, by Y's deadline, 30% of the way to the defined Goal level.

These are important *relationship* ideas, and expressions, because 'success' is defined as 'getting to 100% of a defined Goal level on time'.

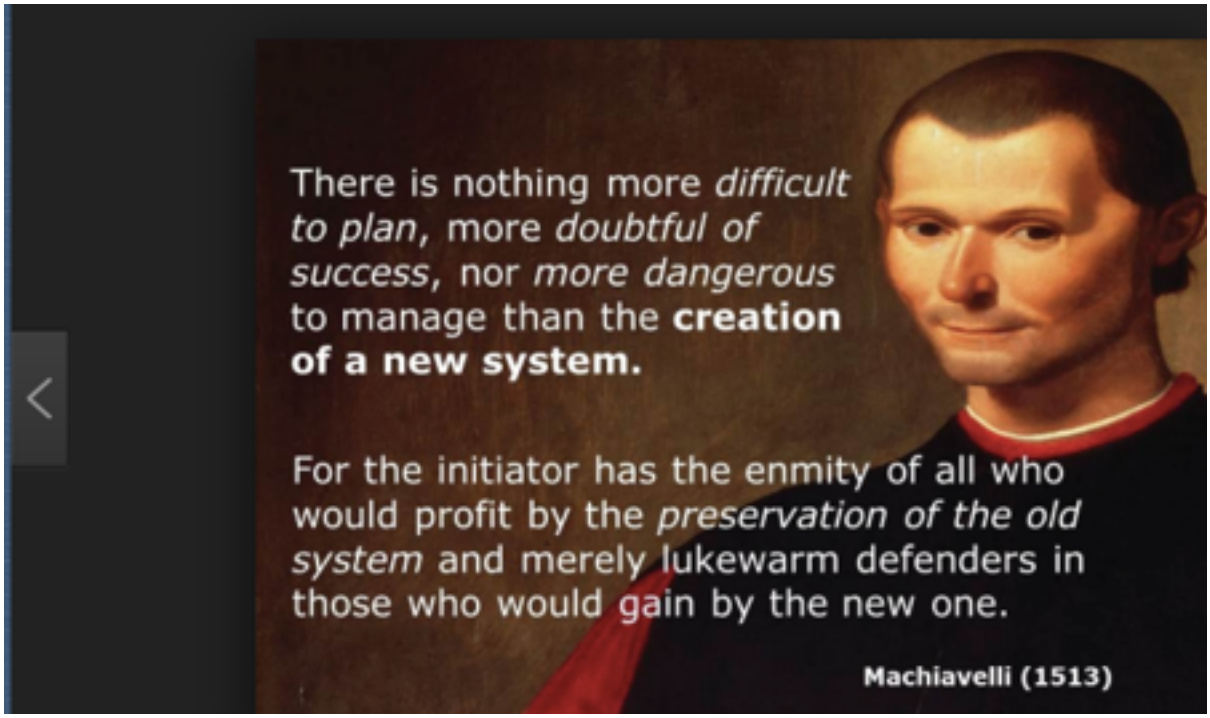
We have to have a way of knowing, or at least hoping, that we will succeed.

Finally, we can use the Impact Estimation **table**, as discussed earlier [URL21, URL75, Part 4, Sections 6.5] to look at *multiple* relationships: many strategies relating to many objectives.

Objectives

Strategies / Goals	Identify Binding Compliance Requirements Strategy	System Control Strategy	System Implementation Strategy	Find Services That Meet Our Goals Strategy	Use The Lowest Cost Provider Strategy
		Strategies			
Security Administration Compliance 25% → 90%	100%	100%	100%	50%	0%
Security Administration Performance 24 hrs → 4 hrs	75%	100%	100%	100%	0%
Security Administration Availability 10 hrs → 24 hrs	0%	Impacts		00%	0%
Security Administration Cost 100% → 60%	50%	100%	100%	100%	100%
Total Percentage Impact	225%	300%	300%	350%	100%
Evidence	ISAG Gap Analysis Oct-03	John Collins	John Collins	John Collins	John Collins
Cost to Implement Strategy	15 man days (US$ 5,550)	15 man days (US$ 5,550)	15 man days (US$ 5,550)	15 man days (US$ 5,550)	1man day (US$ 1,110)
Credibility	0.9	0.6	0.6	0.75	0.9
Cost Adjusted Percentage Impact	202.5%	180%	180%	262.5%	90%

Table 9.3 An Impact Estimation table shows many types of relationships, some illustrated here. The main idea is to show the numeric impact of each strategy, on all objectives, and costs. Real table by John Collins, Citigroup.



Machiavelli 1513 Quotation 9.3 B

Practical Tip 9.3

- *Every statement in planning is related to at least one other thing than its obvious main relationship. For example related to a stakeholder, a strategy, a source. As an exercise, take a page of planning, and add at least one relationship to all specification statements, for example a Source (<-).*
- *At meetings, ask relationship questions where it seems potentially interesting, and request that the answer be documented in the plan 'to help me/us recall the answer' .*
- *For all projects you should at least have a one-page Impact Estimation table showing the relationship of your top ten strategies for your top ten objectives [URL8 A]. This is a primary project overview tool. The IE Table is a map of your primary project relationships. It can be sketched in an hour, and done fairly well in a day.*

Policy 9.3 Chart Critical Relationships

- **We will analyse, and document, all *potentially critical* relationships: while planning.**

- Why?
    - So as to reduce problems and damage later that would be caused by not understanding these relationships, and planning for them.

Version Dec 24 2015 01:07 Tsg

Section 9.4 Manage the Degree of Value Improvement

Principle 9.4 Estimate Strategy Impacts

**Quickly estimate the impacts your strategies have,
on all next-level objectives,
and on all resource types.**

Before we can make decisions, as to what strategies to deliver first, next week and on, we need to try to ensure that we are *probably* going to deliver high value at low cost.

- Make sure your strategy choices give 'high-value for resources-used'
- Then, choose the best 'values/resources' strategies, to do first
- Until your Goals are satisfied
- If you have some left over-resources, consider raising your Goal level ambitions; if not, release the resources for better payoff elsewhere.

***Elon Reeve Musk* - Chairman & Chief Executive Officer, Tesla**

"Yeah, absolutely. We certainly believe in capital efficiency and getting better with that over time.

But we also believe that companies build value by doing hard things.

Not outsourcing those hard things to other people, because then they deserve the value."

Quote 9.4 A. Source: 3 Nov. 2015
http://seekingalpha.com/article/3642146-tesla-motors-tsla-elon-reeve-musk-on-q3-2015-results-earnings-call-transcript?page=2&p=qanda&l=last

There is no point in choosing the best of a 'bad' set of strategies (a low or negative values/resources ratio).

To understand the value/resources ('ROI' is too *narrow a term*) we have to *numerically* estimate the impact of the strategies, in terms of moving us towards our Goal levels.

'Opinions' of strategy goodness (as in Balanced Scorecard and Quality Function Deployment [URL35 QFD, URL62 BSC] like 'high', 'medium', 'low', or '*****', '***', '*' are perhaps more dangerous than useful. Especially if they are subjective, anonymous, and not backed up with facts; and not compared to quantified objectives.

We also need to have *some* idea of resources needed to deploy and operate a strategy. Like, 'can we actually implement a sub-strategy next week', or next month.

"It is difficult to estimate correctly, especially about the future."

Quote 9.4 B. [F9, Gilb with a nod to Niels Bohr]

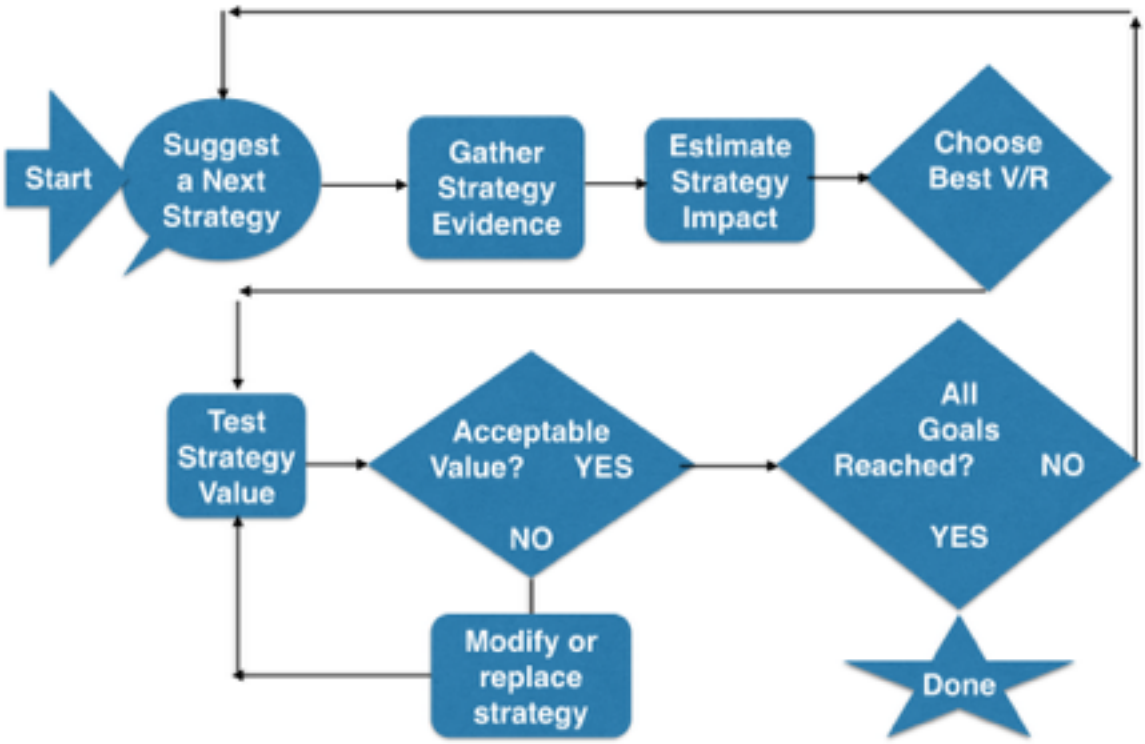
Estimation certainly has its uncertainties. But it is not without its uses. And is usually much better than nothing [URL11, Estimation paper].

The important points about estimation of strategies on Goals, in planning are these:

- It makes people think more clearly, than if they do not try to estimate
- It makes people discuss, and perhaps agree, or come to consensus (and consensus may be more important than being 'right')
- It makes planners realize that the strategies are often so badly defined as to not be understood in the same way, by any two people
- It makes planners aware that the strategies are so badly, or generally and broadly, defined that the range of strategy-impact estimates is too broad, to be useful as an estimate
- It makes people who estimate, feel *responsible* for the estimate, and sometimes, those same people have the job, and power, to make it come true [URL12, Quinnan]
- You do not have to be extremely accurate, if the numbers are just used to *prioritize* what you do. You just have to be *consistently inaccurate* (i.e. roughly order-of-magnitude accurate) to get a pretty good decision about sequencing strategy implementations
- Even if you accidentally get slightly wrong answers, which cause a slightly imperfect priority sequence of strategy implementation, little damage is done. Most of the best strategies still get done early. It would not be worth delaying all implementations for a month or so, to get perfect sequencing, of weekly sub-strategy implementations.
- Since we are going to quickly be able to get real measurement back, next week, we will quickly learn if we are 'crazy' in our estimates, each and every week.
- Since we get real strategy implementation *measurements*, to compare to our earlier *estimates* every week, we will quickly learn how wrong we were, in which direction, to what degree, and for what type of estimates. We can consequently hope to learn better estimation, for the long run (by delivery cycles 20 to 100 for example).

- Estimates are a tool to help us optimize value-for-money overall. But even if we did not estimate at all, just picked random sub-strategies, and then cumulatively tracked value-for-cost delivered, we would still look pretty good, compared to the usual failed project. We would for example dump or modify low impact strategies [URL41, URL12 Quinnan]. Just not quite as good, as if we optimize by trying to estimate.
- If you cannot generally make reasonable estimates for the strategies you have chosen, if you are constantly estimating very badly: then that is a sure sign you are a 'loser' and your team hasn't a clue as to what they are doing. One ought to have a clue! Experts can estimate pretty well, *that* also defines the non-expert (they cannot).

The process of making an estimate:



Process Diagram 9.4. Estimation and validation by measurement are the basis for strategy selection and in-system approval. V/R: Values for Resources. This is one view of the Evo project management process.

If you fail to gather *evidence* for your strategy estimates, then the numbers estimated are of no value. You might as well pick a random strategy and try it out. But you will waste far more effort to reach your goals than if you try to select based on evidence first, before trying ideas out in practice.

Here is a text view of the strategy impact estimation process.

1. Focus attention on the full detailed written description of the strategy, the 'meat', *not* on its title, the Gist, the tag, or what you 'remember'.

2. Ask if you, your company, your industry, anyone on the web, has ever done this, or similar, before?

3. If yes, see if there is any data, any experience, any memory of the attribute (like a quality level, or a cost aspect), that you can dig up.

4. If not, make a very conditional guess, give a very large ± uncertainty, and then your 'lack of evidence' should give you low credibility. See Section 9.5 for 'evidence' and 'credibility' details.

5. But explain in writing what has happened, so that nobody thinks that this is a *serious* estimate!

6. Ask yourself if the strategy needs better definition, so as to enhance the chance that your estimates can be made meaningfully. Vague definitions mean 'many possible variations' and thus a 'very wide range of possible impacts'.

7. If you have collected any evidence from any sources, on the estimate; document that evidence, together with the estimate. Other people will need to see the evidence and sources, and you will need to refer to them.

8. Be honest. Do not try to pretend you have knowledge you do not have. You *will* get caught out, and your project will *suffer* if you pretend you have better knowledge than you really have!

9. If a better estimate can make a critical difference, then consider using more time, and more probing, to get better answers.

10. But sometimes the fastest way to get facts is to *try the idea out* on a small scale. Just do an Evolutionary Value Delivery Cycle; a week or so [Part 4, URL18, URL31, 1C]. Call it an 'experiment', a 'pilot': but hopefully it is a *real* change to a *real* system, so that you in fact get a *realistic* number, and at the same time gain *real* ground on the march to your Goal levels [URL11, Estimation paper].

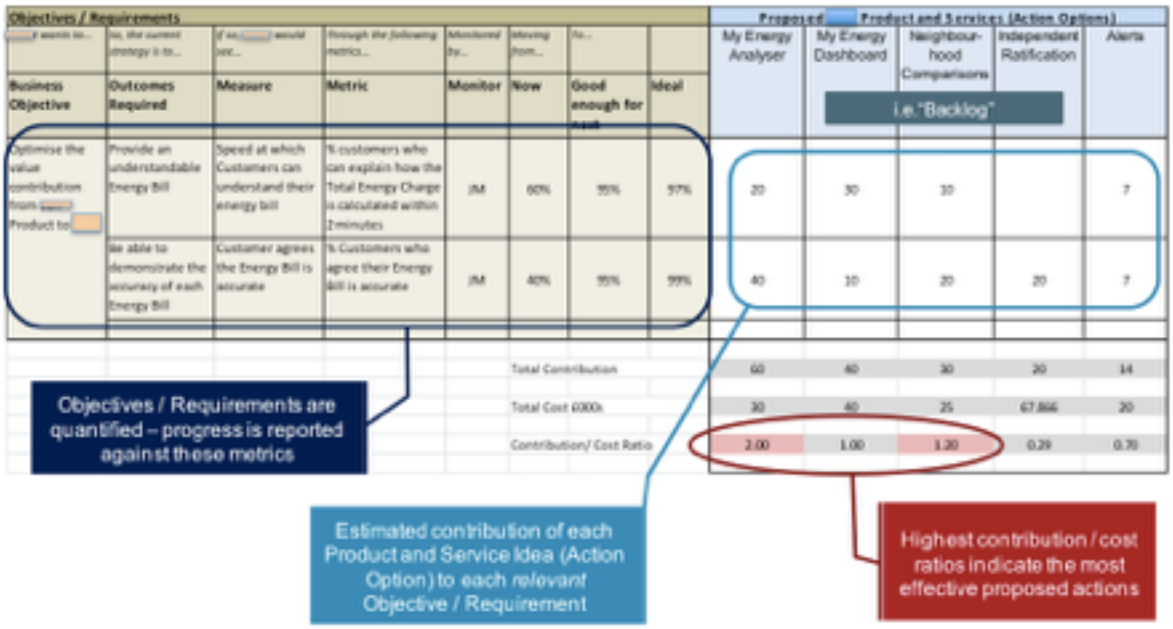


Diagram 9.4 A. A real (anonymous client, by a professional consultant , student of the author), 2014, example of using Planguage to evaluate sub-strategy options in relation to Goals. At left the Objectives (see 9.4 B below for enlargement), and on top right the sub-strategies, with estimates for their impact below. (see 9.4 C below for enlargement) The tables are designed to be presented to managers, who are initially unfamiliar with Planguage. This is an extract of a very detailed plan extract, with commentary of the main points.

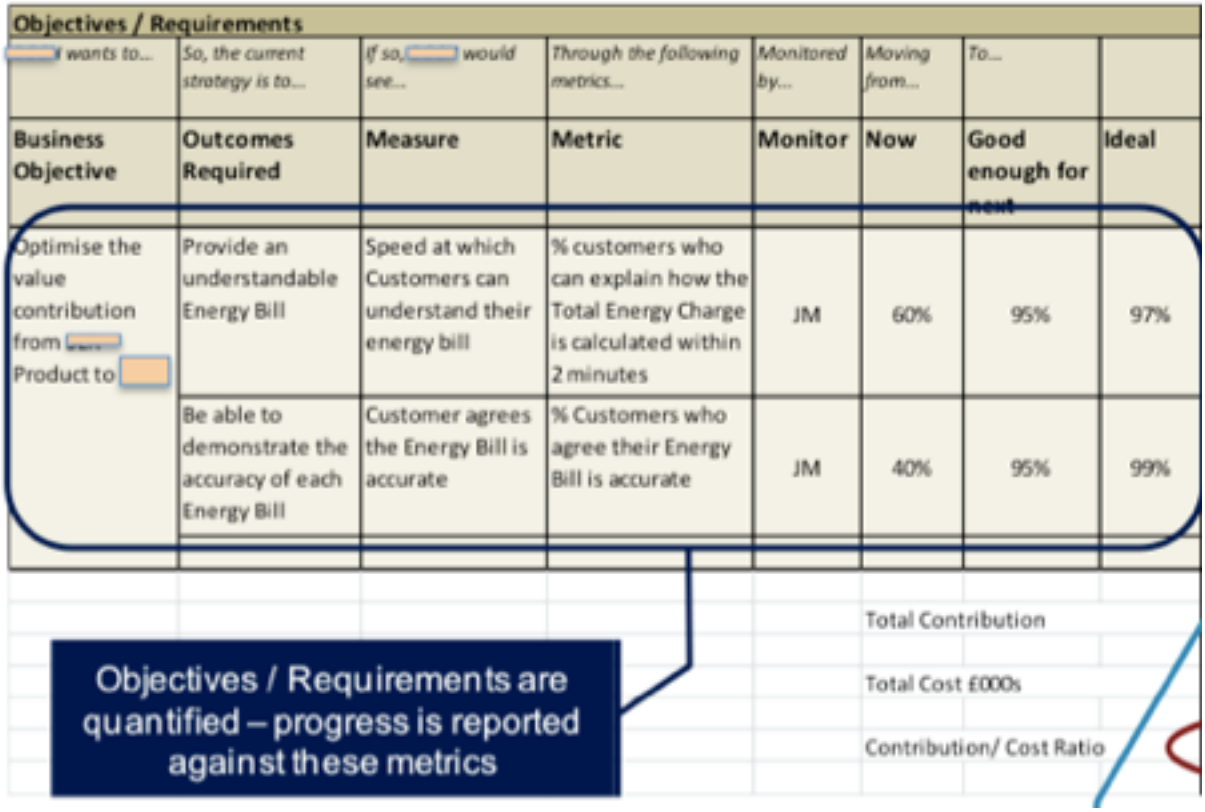


Diagram 9.4 B (subset of 9.4 A). The Objectives, Presented step by step with annotations.

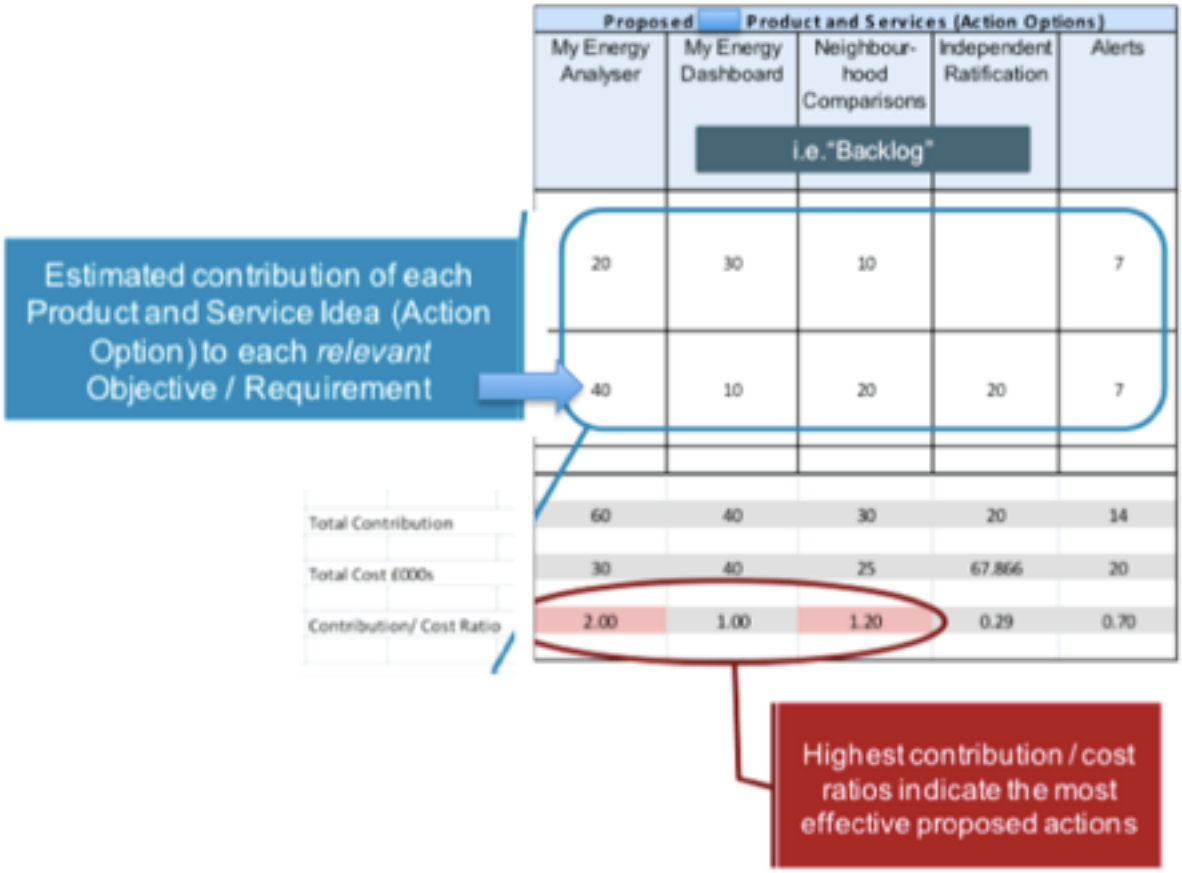


Diagram 9.4 C (subset of 9.4 A). An Impact Estimation table, presenting the estimates of 5 Action Options, with a contribution to cost ratio, to help us see which ones should be done first.

The rectangle patches in Diagrams 9.4 A, B and C are to hide identification information, so we can use a realistic example.

> ***"Human beings, like plans, prove fallible in the presence of those ingredients that are missing in maneuvers - danger, death, and live ammunition."***

— Barbara W. Tuchman (1912-1989), The Guns of August, Quote 9.4C.

Practical Tip 9.4 Probe the lack of real facts immediately

- *At the very least, **admit your ignorance** of 'critical impact facts' by appending estimates to their strategies. Like 'Impact on Goal X = ?? <- <your initials>'.*
  - *That is a beginning of exploring the impact level.*
  - *It does mean you alone, are wise enough and first, to recognize the need to have an estimate, and you challenge others to supply better information (if they are 'experts' they can do that).*
- *When people suggest a strategy, **play '12 Tough Questions**' [1.5, URL15] with them.*
  - *'Thats Interesting Jane, what do you think the impact on Goal X will be, roughly?….. Thanks. Can I cite you on that ?'*
    - *'Of course, Jane, it is not a* **commitment** from you**:** *just a starting point in our decision making.'*
  - *'By the way, can you tell me, roughly, how you arrived at your estimate? …. Thanks, sounds impressive, can you mail me a link to that evidence after the meeting, that I can cite and explore? Great! Thanks again!'*

Policy 9.4 Estimate Impacts Sensibly

- **We will estimate the benefits and costs for our strategies, on our critical objectives.**
- **We will do so to the degree it pays off.**
- **We do not have to be perfect; just to have a reasonable idea of which strategies might pay off to implement first.**

- Why?
  - To give ourselves guidance as to the probably most profitable priority for implementing frequent value delivery steps.

Version 24 December 2015 02:04 CET, TsG

Section 9.5 I'm from Missouri, you gotta *show* me!

Principle 9.5 Prove It

**Evidence and sources for estimates,
reduce risks of bad decisions.**

Even if you happen to know the exact correct Strategy-Impact estimate, which is very unlikely, you cannot just write your estimate down alone. You must make *some* note of how you arrived at the number; even if you arrogantly write 'divine gift'. *Other* people need to understand how you arrived at your estimate, for all kinds of reasons. And you need to show respect for those needs.

Sometimes, quite frequently in fact, people have no actual evidence whatsoever for some of the estimates they make. Yet we *feel* we can make an 'intuitive' estimate, based on our experience with similar things. Fine! Better than nothing.

Let us just be honest and forthright with what we decide to do. No shame in not knowing such details, and no shame saying so openly. But give other people a chance to say, *wait a minute, I want better hard evidence than that, before I support this strategy, and before I need to make a decision for or against the strategy.*

By 'Evidence' we mean the same concept as in law courts, and science labs. Evidence quality varies, and no evidence is perfect. But we need to specify it in our plans, formally and clearly. In law and science we are required to do so.

In value management planning we *should be required* to give evidence (*'Rule: evidence will be given for all estimates'*), but we are not usually mature enough, in our knowledge base, enough to make this the rule, rather than the exception.

Evidence should be as 'hard' (factual from reliable sources, and from reliable measurement processes) as we can find. 'Planning' is not usually for a Nobel Prize in Sciences, nor is it for a 'war crimes case', so we do not need the same academic standards of evidence as they do.

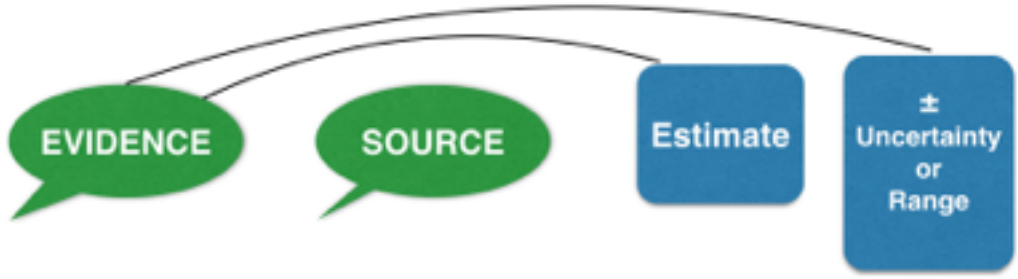


Figure 9.5 A. The basics for evaluating the impacts of a strategy. 'Evidence' is the knowledge, experience, or facts. 'Source' is 'who' or 'what' we got the evidence from.

We base our estimate of the strategy impact, and the potential ± range of the impact, on the evidence.

We do need *good enough evidence* for the purpose at hand. Like everything else in Planguage, the investment, in gathering and checking evidence, should be *justified by the costs*.

But, the better the evidence, the better the *credibility* of your estimates. And better credibility estimates, when *other things are otherwise equal*, should, assuming management is logical, be the winner in management decision making.

It would be a sad day if management claimed they preferred to make decisions based on '*lack'* of credibility.

However, it is easily observable that managers constantly make *big* and *bad* decisions, *without* asking for evidence, and *without* looking at the quality of any evidence given, at all.

For example:

Strategy A Impacts Goal X by 30% <- Tom

The above is a simple assertion, estimating an impact, but *without* evidence. The source (Tom) is *not* evidence.

Strategy A Impacts Goal X by 30% <- Tom

Evidence: All our 100 uses of strategy A last year attained at least average 30%±10% of the Goal level, or sometimes much better **<-** Corp.com/RandD

The 'Evidence' statement informs the reader about the reason we estimated '30%'. The Source (**<-**) tells us where this evidence can be found. If we want to check it, or just get some idea of its credibility.

Credibility (of Evidence and Source!)
Rating Scale (CE p.274, fig. 93.)

Credibility Rating	Meaning
0.0	Wild guess, no credibility
0.1	We know it has been done somewhere
0.2	We have one measurement somewhere
0.3	There are several measurements in the estimated range
0.4	The measurements are relevant to our case
0.5	The method of measurement is considered reliable
0.6	We have used the method in-house
0.7	We have reliable measurements in-house
0.8	Reliable in-house measurements correlate to independent external measurements
0.9	We have used the idea on this project and measured it
1.0	Perfect credibility, we have rock solid, contract-guaranteed, long-term, credible experience with this idea on this project and, the results are unlikely to disappear

Table 9.5 The Credibility rating for both evidence, and the quality of the source of the evidence [1, CE Book].

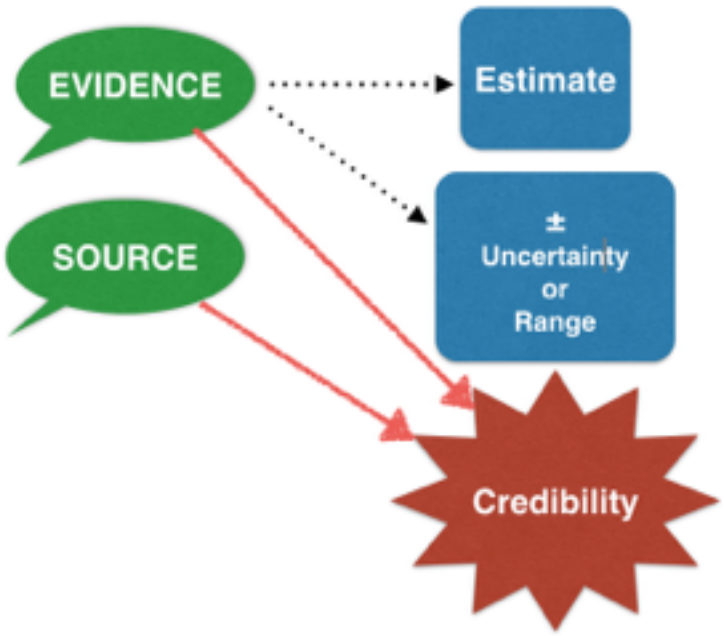


Figure 9.5 B Both the evidence and the source ***combined*** *give us the information we need to stipulate the Credibility of our estimate. We use Table 9.5 above to do that*

Case 9.5: Medtronic in Minnesota: The Invention of the Credibility Index

<- A Pacemaker.

- Kai and I were working with the heart 'pacemaker' company in Minnesota, 'Medtronic'. We were working with the group that develops their pacemaker 'Programmer', which tailors a pacemaker to an individual wearer. The Programmer Group was planning its' coming year's Research and Development, using Planguage.
- When it came time to summarize our strategies and objectives in an Impact Estimation table, we were asked how such a big (10 by 10 approximately) table could credibly be presented, in the mere 15 minutes the Programmer Group was going to get, for their management presentation, to get a decision on their next year's R&D Budget.
- The usual situation, elsewhere in our experience, had been that managers would probe the table on maybe one (strategy:goal) intersection, that they had some detailed knowledge of. Then, if that sounded credible to them, they tended to accept the table as credible enough.
- But, Medtronic, especially as they were in the Medical business, and had to get FDA (Federal Drug Administration) approvals for actually letting equipment go to market, had a much higher standard. So, on the spot, we 'invented' the Credibility Rating Index (Table 9.5 above). We rate the level of 'credibility' for each estimate.
- This means that the *planners themselves* have to evaluate all their estimates.
- Then, based on specified evidence, and the evidence *source*, assign a 'Credibility Level': '0.0= none, to 1.0= perfect' (see Table 9.5 above).
- This is a quite reasonable delegation of analysis; to the engineers making the estimates. Management has to trust and understand the process. But they do not have to do the credibility ratings, personally.
- Management does not even have to look at the credibility rating detail at all, which might mean looking at 100 individual credibility ratings per 10x10 IE Table.
- The reason they can ignore the detailed 100-or-so Credibility ratings is because we use those 100-or-so credibility ratings to reduce their 100-or-so Impact estimates to a suitably conservative size.
- And even those 100-or-so estimates get totalled up in various summaries, and can be reflected in a *single* 'total value for cost with respect to credibility ratings' ! (Fig. 9.5 D and E)
- This is in fact done in Kai Gilb's Value Decision Table Tool [URL68] automatically. And in Richard Smith's tool [URL73]
- To finish the Medtronic story: a few months later the Medtronic Team invited us for dinner 'at their expense', and said that they had interesting news for us.

- "We got our proposal approved. And your methods are worth a million bucks", they said.
- We asked what *that* meant, and they said they got the whole $2,000,000 they asked for, and no other proposal, by any other group, got more than *half* of what *they* asked for. So, we cheerfully assumed their presentation 'credibility', was very credible to management.
- Management can *see*, via the Table, what values they would *fail* to achieve, if they wished to save money' by deleting certain strategies. So I guess they did not dare to give our team less than asked for.

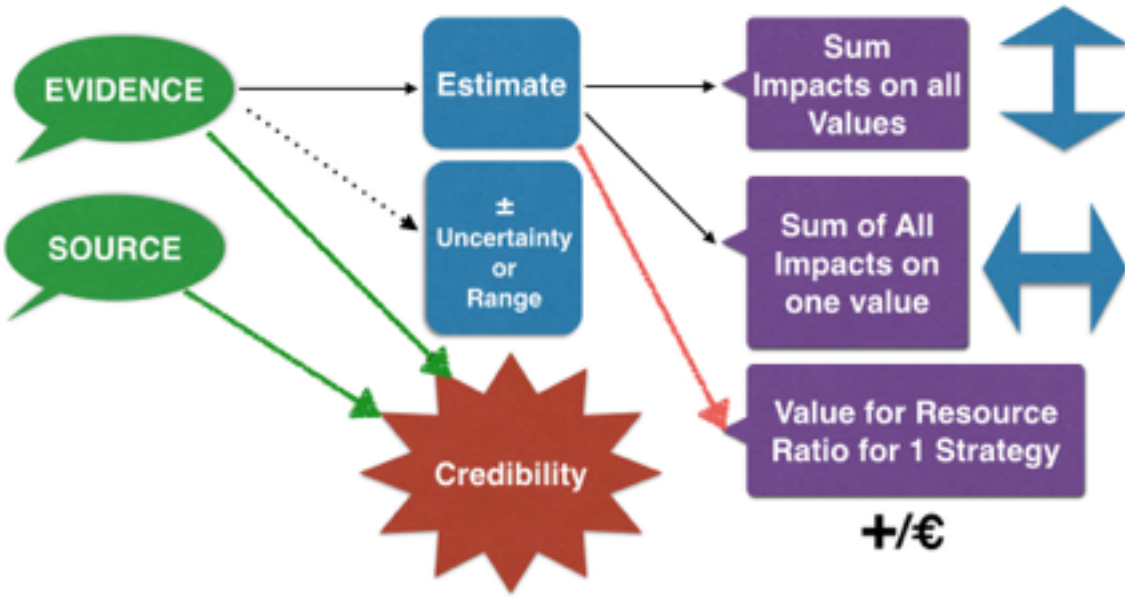


*Figure 9.5 C. The estimates, and optionally the ± range, can be used to sum up vertically the power of one single strategy, on all planned values. The resource estimates can also give us a total impact on all constrained or budgeted resources. We can similarly add a horizontal row of different strategy estimates (**<->**) to get some idea of the power of a set of strategies; to answer the question 'do we have enough strategies?'. And also 'do we have enough Safety Margin' ?*
And we can summarize the efficiency (or ROI if you prefer, or +/€) of a strategy, in meeting our objectives within *our constrained resources.*

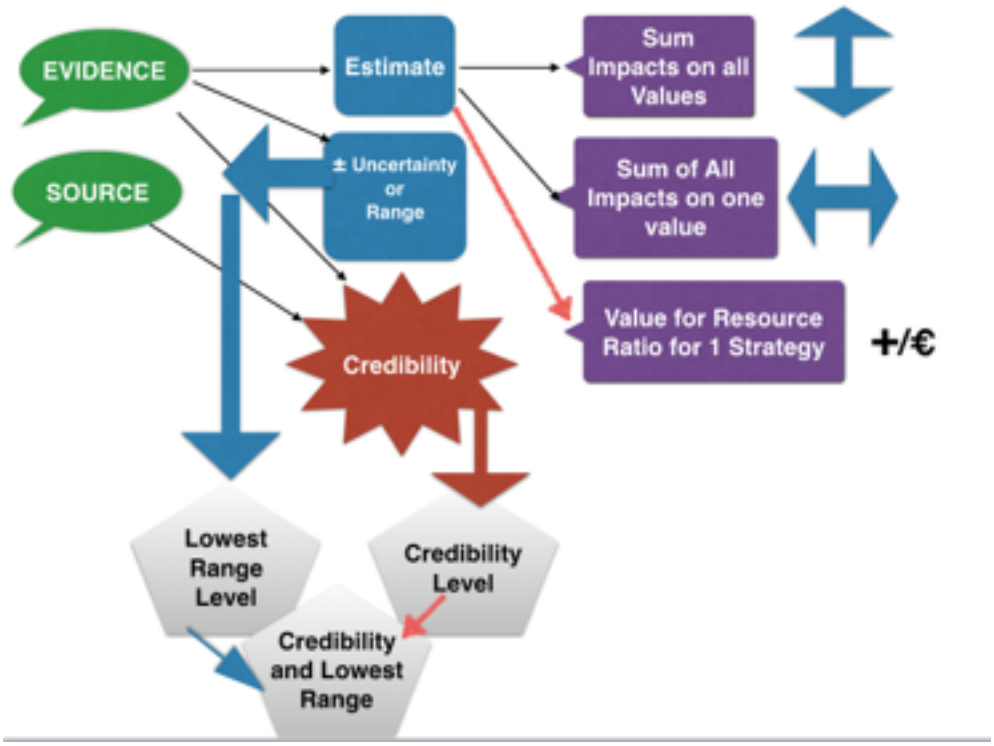


Figure 9.5 D. ***Finally we can calculate 3 notions of the riskiness of our numbers.***

1. *The **lowest range level**. For example 70±20 = 50. This is the 'worst level' experienced by anyone according to the evidence. (see Section 9.6)*
2. *The **Credibility level**. For Example 70 x 0.5 = 35. An arbitrary but useful impact reduction, for less perfect evidence and sources.*
3. *The combination: the **'Worst Worst Case'**. For example 70±20 = 50 x 0.5 = 25 (See Section 9.6)*
4. *and we can do this for individual strategies, and for sums of strategies (the 3 right hand sums in the diagram)*

These calculations are usually done automatically in a spreadsheet or a tool [UR68, URL73].

These riskiness calculations can be used to directly and automatically select winning strategies according to your decision-making policy; like 'pick the strategy which is best in the most conservative case'.

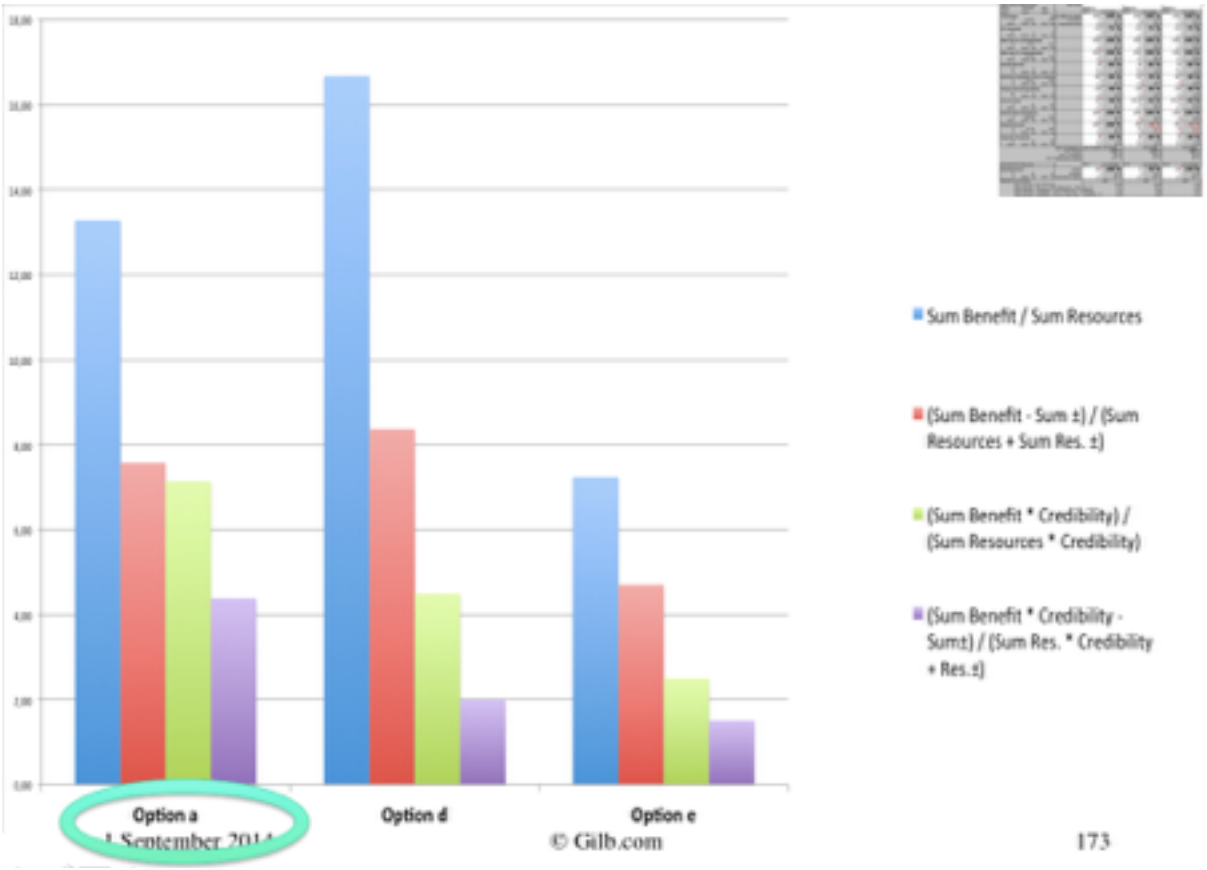


Figure 9.5 E. ***Real banking case. London****.*

Comment on Figure 9.5 E
3 Major choices. Option 'd' looked best initially, Value for money (left column).

But when we used the ±uncertainty pessimistic level, option 'd' still looked best, but option 'a' was much closer to it.

When we looked at a Credibility alone, option 'a' was the winner.

When we combined the evaluation of ±uncertainty pessimism, *and* credibility, then option 'a' was a very clear winner (right hand column of the 3 groups).

So the question was to the management: "Do you feel lucky punk?"
[Dirty Harry, Clint Eastwood, 1971].

The upper right corner is a miniature of Kai's Excel Tool [URL68] for calculating this 'bottom line'.

Quote 9.5

"The fact that an opinion has been widely held is no evidence whatever that it is not utterly absurd;

indeed in view of the silliness of the majority of mankind,

a widely spread belief is more likely to be foolish than sensible."

— Bertrand Russell (1872-1970), Marriage and Morals

Practical Tip 9.5

- *after any estimate feel free to ask politely*
    - *'can you help me build a case for your estimate, that would stand up in front of my boss?'*

Policy 9.5 Earn Your 'Street Cred'

- **We will, when we are concerned with the credibility of our estimates,**
  document the *evidence and sources* for them,
  and use that, to rate the *level* for each estimate,
  and do this for the entire package of strategies for all goals, in a management summary.

- Why?
    - To force ourselves, as planners, to come up with better, more-credible plans, before wasting management time, which would be frivolously wasted evaluating our, dubious, presentations and estimates
    - To give management evaluators and decision takers, ability to make quick decisions with confidence, and to help their understanding of the level-of-risk they are accepting.

Version 24 Dec 2015 02:28 CET, TsG

Section 9.6 Dealing with the range of uncertainty pessimistically.

Principle 9.6 Worst Case

**Estimate the ± range
for best and worst case impacts
of your strategies**

It is natural, when we *propose* a strategy, that we *believe* it is a 'good thing'.

We also are tempted to put it in a good light, so that we look good for proposing it, and to increase its saleability to decision makers and colleagues. In short, we tend to be optimistic, defend our suggestions, and hope for the best.

But, our *optimistic* impact estimate is *not* what decision makers need most. They need to know what *might happen* when Murphy's Law kicks in (If anything can go wrong, it will; at the worst possible moment. Case 9.6 below).

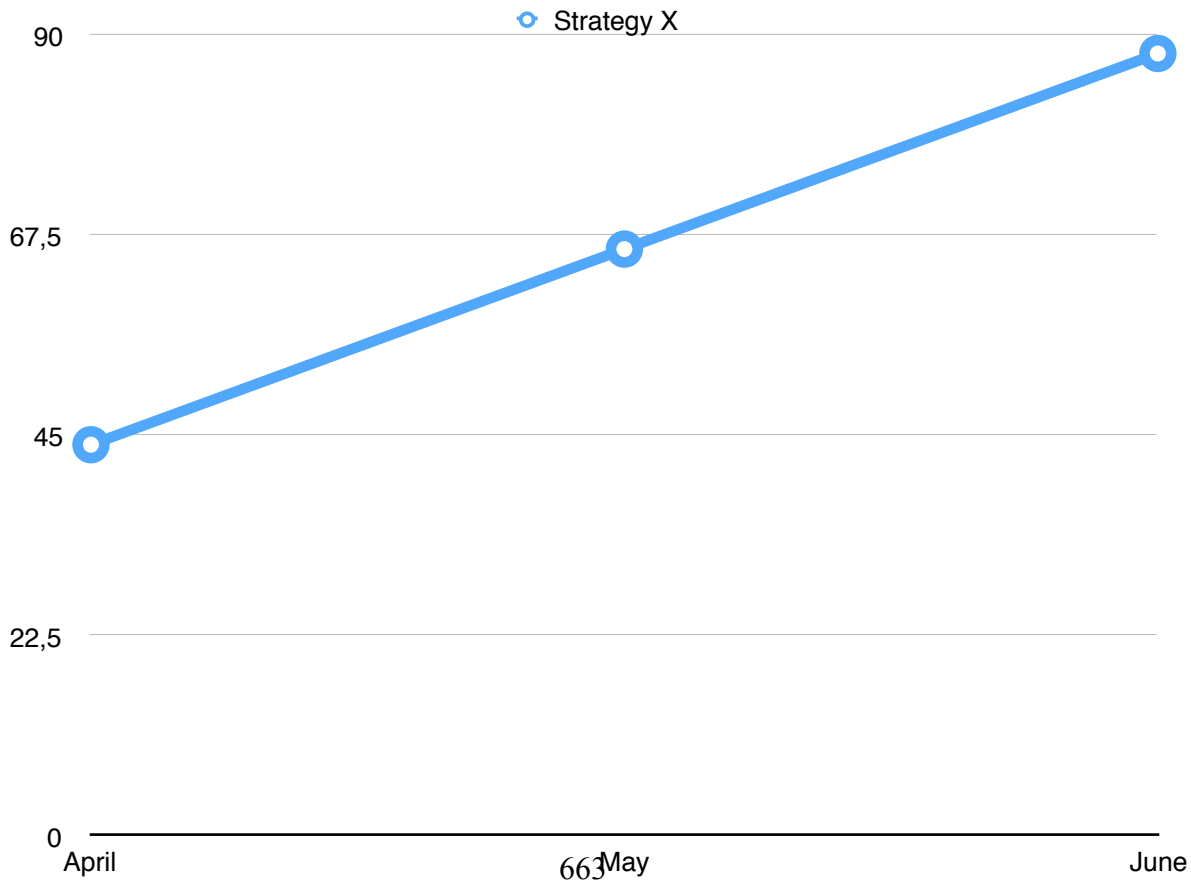


Figure 9.6 A. The evidence here, in a simplified example, is that the impact of Strategy X is 66 ±22. Or, it ranges from 44 to 88 if you prefer. In both cases 44 is the pessimistic level of experience.

Case 9.6 Murphy's Law

At McDonnell Douglas Aircraft, El Segundo California (about 1989), I once mentioned Murphy's Law [URL55, a complex history of Sods Law]. They replied, "*Tom*, Murphy *works* here."

"Yeah sure! You mean he exists, he's alive, and *really working* at your company?" They nodded.

I said. "Prove it!" They did. They provided a recent In-Company Magazine with a photo of the Ed Murphy [Photo 9.6 B] (a Safety Engineer) at their Apache Helicopter Plant.

Well, there are a lot of very complex systems in aircraft engineering and aircraft production, I learned.

Once some young aircraft wiring engineers I was working with, were almost in tears, they told me, when they realized the potentially fatal consequences of their engineering planning, if it went wrong. The wiring for a single aircraft, if stretched out, would go around the earth (or something to that effect) they told me.

After that, I never merely referred to Murphy's Law. I just said; "Remember guys, Murphy works ***here***!"

Col. John Paul Stapp aboard the "Gee Whiz" rocket sled at Edwards Air Force Base.
Photo courtesy Edwards Air Force Base

Photo 9.6 A. John Stapp, who popularized Murphy's Law in action, when it happened [URL55 A4]

Engineer Edward A. Murphy, Jr. spent his career finding ways to make failure impossible in aircraft cockpits and beyond. Image courtesy of Nick T. Spark.

*Photo 9.6 B "**If there's more than one way to do a job, and one of those ways will result in disaster, then somebody will do it that way."** This was Ed Murphy's own wording according to his son (People magazine, January 1983). [URL55 A5]*

When you make an estimate of any kind, in Planguage planning, you should habitually write a '±' (alt & '+' on a Mac, or +/- if you like) after it, and think about the lowest most-pessimistic value that could happen in the real world of Murphy's Law.

If you want, you can indicate the most *optimistic* value too.

It is the range of possible values we are after, and the most *useful* value is the 'pessimistic' number.

Ways of expressing uncertainty in Planguage:

30±20 balanced deviation
30 +20 -30 unbalanced edges
0 to 30 a range
0-30 a range
30? signalling doubt
30?? signalling great doubt
30±30 <- Published data source citation. *(perhaps suspect)*
<30-60> a range using Fuzzy Brackets

Planguage 9.6 A. Ways to express the idea of a worst possible case, and still look knowledgeable when doing it.

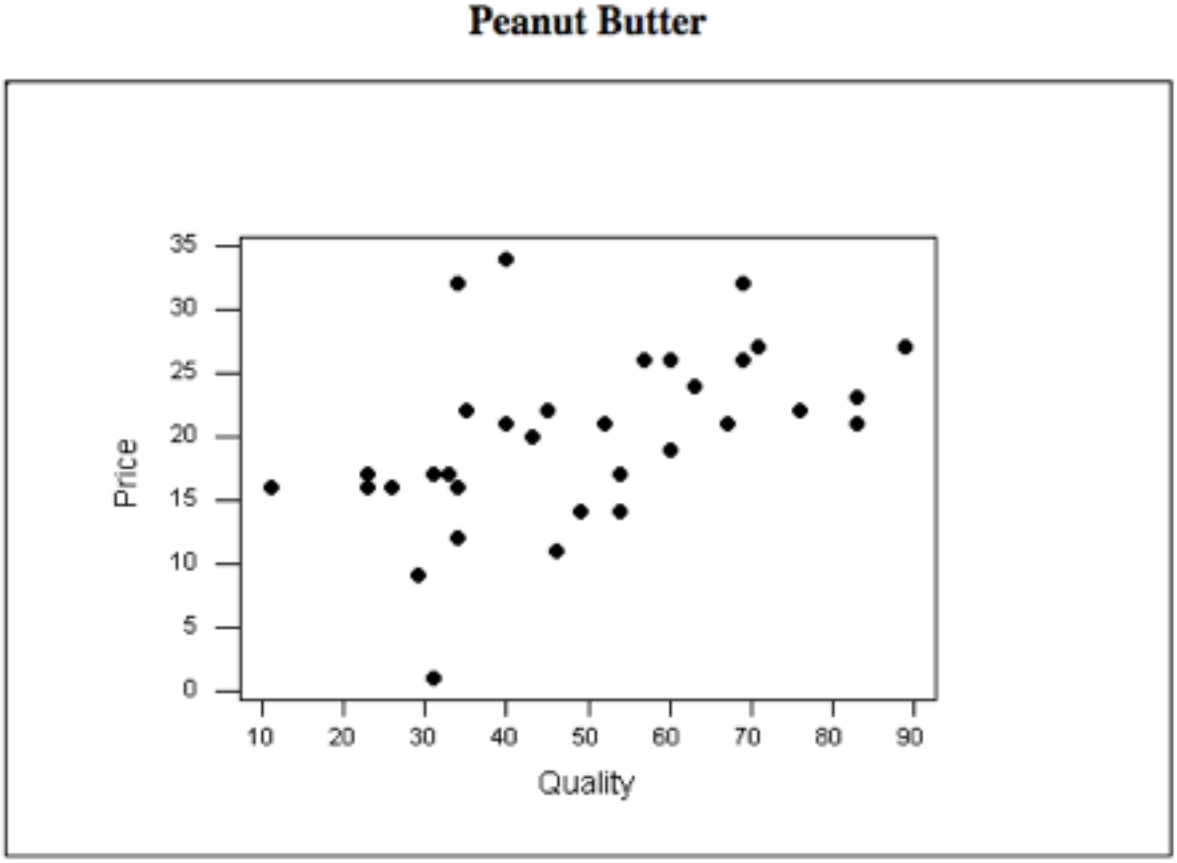


Diagram 9.6 B [URL56]. If my strategy for getting a quality level of at least 60 (of 100 quality rating points) was to buy peanut butter. I would, using this experience data, estimate it would cost me at least 17, as much as 33, and about average 23-27 . We could express this as:

Estimated Cost [Peanut Butter, Quality: Tolerable = 60] 25±8 ?

In fact there needs to be a Rule, like R7 below, to support this '±' activity, to legitimize it, because people are shy, and wrongly think that 'uncertainty specification' admits to a personal incompetence, of knowing the 'right answer'.

The Uncertainty Rule.

<u>R7: Uncertainty</u>: The Uncertainty (based on the evidence experience borders) of the Scale Impact estimate shall normally be specified.

Percentage Uncertainty values are then calculated in a similar way to the Percentage Impacts.

Example: 60%±20%.

Usually, the uncertainty values are calculated individually for each cell.

An exception to this occurs when some overall uncertainty (such as ±50%) is declared for the *whole table* or specified parts of it.

Another more fundamental exception can be when a decision is made to defer dealing with uncertainty data.

*Planguage 9.6 B: The **formal Rule** recommended in the Impact Estimation Chapter of the handbook for Planguage, Competitive Engineering [1]. This is grouped with other Rules for Impact Estimation. [1 E, Rules Download]*

In everyday Planguage use, we normally simply use the balanced '±' format;

'Estimate with Uncertainty' structure example:

Impact [Strategy X on Goal Y, First Release] 60%±30% ? <- TG.

An *unbalanced* (60+30/-15) estimate is often dependent on having more data about a strategy than we have actually got.

Think of the '±' statement as us giving a *range* of possibilities, or think of it as reflecting a scatter diagram of experience data upper and lower experience [Diagram 9.6 B].

Figure 9.6 B. The ± uncertainty specification can be used to systematically provide very conservative estimates (the worst levels actually experienced).

Think of the main number, in your estimate, as a result that is *possible*, and 'as likely as any', and maybe as a *frequent result*. That is good enough for our purposes. Statistical formality is not necessary for most planning purposes: though it is advisable for some!

Our main purpose, in planning, is to get a somewhat more realistic idea of the pessimistic case, so as to enhance our ability to prioritize more realistically, with regard to 'risk', of bad things happening.

Of course **'knowing you don't know'** is a higher form of expertise than foolishly imagining that you do know something that you *cannot* know, because of all the 'many future unknown forces', that will determine the *actual* result.

In fact the long-term result itself is no constant number, and will usually vary within some limits.

Inspired by Murphy's Laws I wrote something similar in 1975, which seems to be quoted on the internet or twitter, on a frequent basis to this day! [URL55 B]

> ***"Computers are unreliable, but humans are even more unreliable."***
> *And*
> ***"Any system which depends on human reliability is unreliable."***
> *Plus a few more 'Laws'.*

I once emailed someone who had quoted this, but not credited me (Gilb's Laws). His reply was

"OMG I thought you were dead!"

I secretly fear that when all else is forgotten, including this book, Gilb's Laws will be my only remaining '15 minutes of fame'.

You, dear reader, can prevent that happening! I have tried my very best to give you some interesting and practical sound bites: please share if you deem some ideas worthy of spreading! Steal them without credit if you must! The important thing is sharing useful ideas. Credit is just an academic game.

"True wisdom is knowing what you don't know"

"By three methods we may learn wisdom:
First, by reflection, which is noblest;
Second, by imitation, which is easiest; and
third by experience, which is the bitterest."

Confucius, Died 479 BC (aged 71–72) Sorry, no photo ☺

Practical Tip 9.6 Write '±', after estimates, as a *habit*.

- *Make it a habit, for any estimate, to write '± ?' after an estimated number number, initially*
- *Later, you or someone more-knowledgable can attempt an approximation of the uncertainty degree.*
- *It would be great to get the right order of magnitude! ±0.1, ±1.0, ±10, ±100*
- *or at least have an interesting discussion of*
    - *why you cannot agree on a ± approximation*
    - *what it implies in practice if the real ± deviation from your estimate is very large*
- *No '±' at all* ***implies*** *±0. But that is very improbable: so bring up the uncertainty as a certainty.*

Policy 9.6 Be honest about the bad news, and give it early

- **When making an estimate, we will always clarify the *pessimistic* level, so that nobody will be misled by false optimism.**
- Why?

- If you cannot clarify the pessimistic level, you are probably incompetent to estimate anything.
- This improves the practical value of the plan
- Your project is more likely to be perceived as 'successfully' delivering within these pessimistic expectations, or better.
- Management decision-makers can choose to go with conservative, worst case, scenarios. Take responsibility for such outcomes.
- To avoid *you*, the planner, getting *blamed* for unrealistic planning (overoptimistic planning)
- To be 'noble and wise, like Confucius advise'.

Version 24 Dec. 2015 02:39 CET, TsG

Section 9.7 Connecting Stakeholder Levels

Principle 9.7 Multi-Level Impact Tables

Connect Impact Tables
at different related levels of planning.

Impact Tables connect *ends* to their *means*.

But this opens up another planning possibility. We can directly, and numerically, connect Impact Tables with each other. We can connect related stakeholder levels of planning with each other.

This opens a unique opportunity in management and technical planning. The CEO can 'see' what is going on, at the most detailed level of their support, for corporate plans.

In fact our experience is that CEO's are very excited to do this. They never had such a good chance to make the inner workings of their organization transparent to themselves.

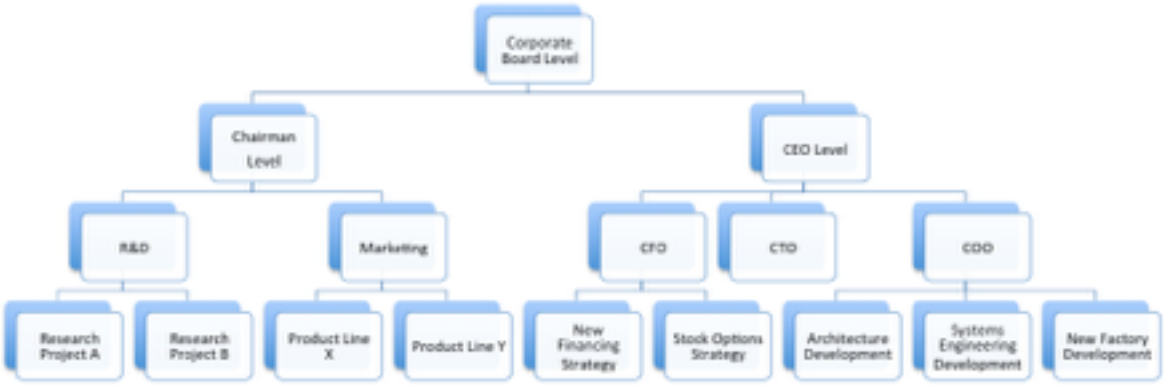


Diagram 9.7 A conceptual example of different levels of stakeholders, that can have Impact Estimation Tables connected to adjoining levels of concern.

Here is a real example:

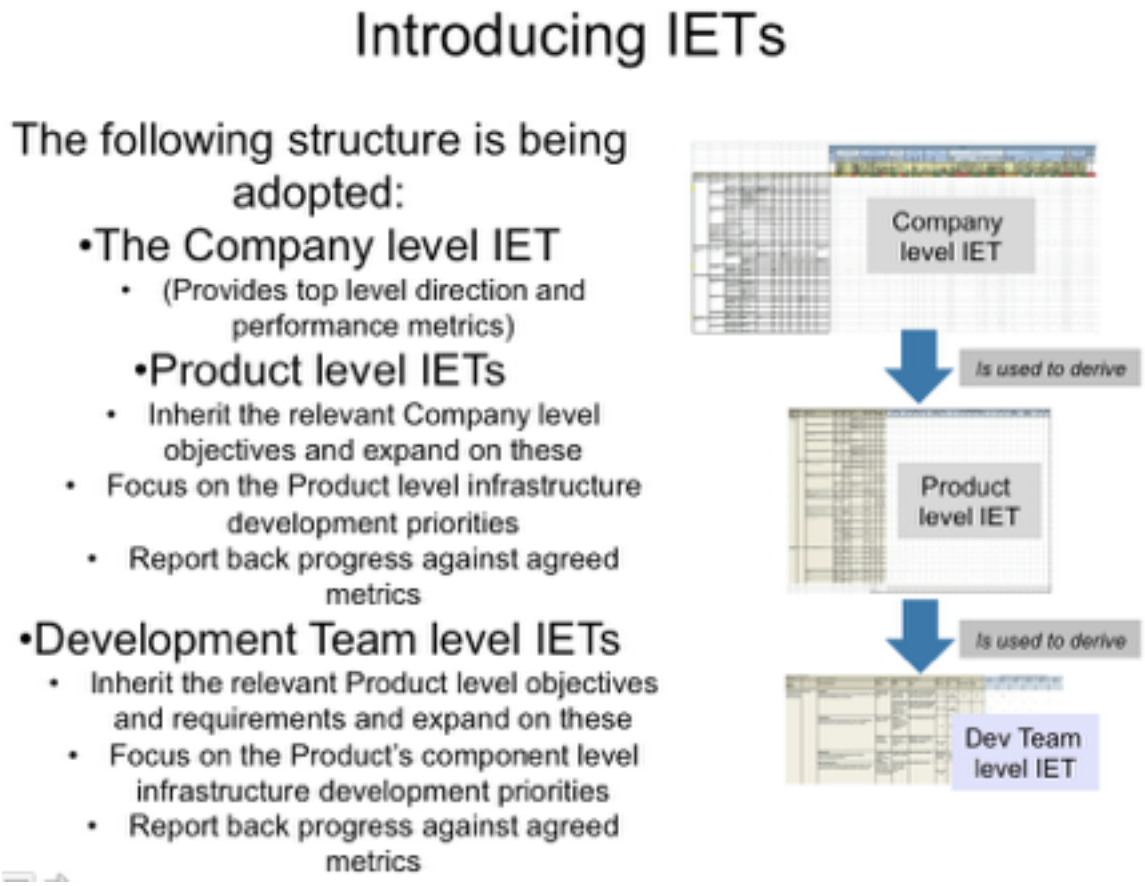


Diagram 9.7 B Slide from actual management presentation, 2014 by a consultant trained in our methods, to a major Utility in UK.

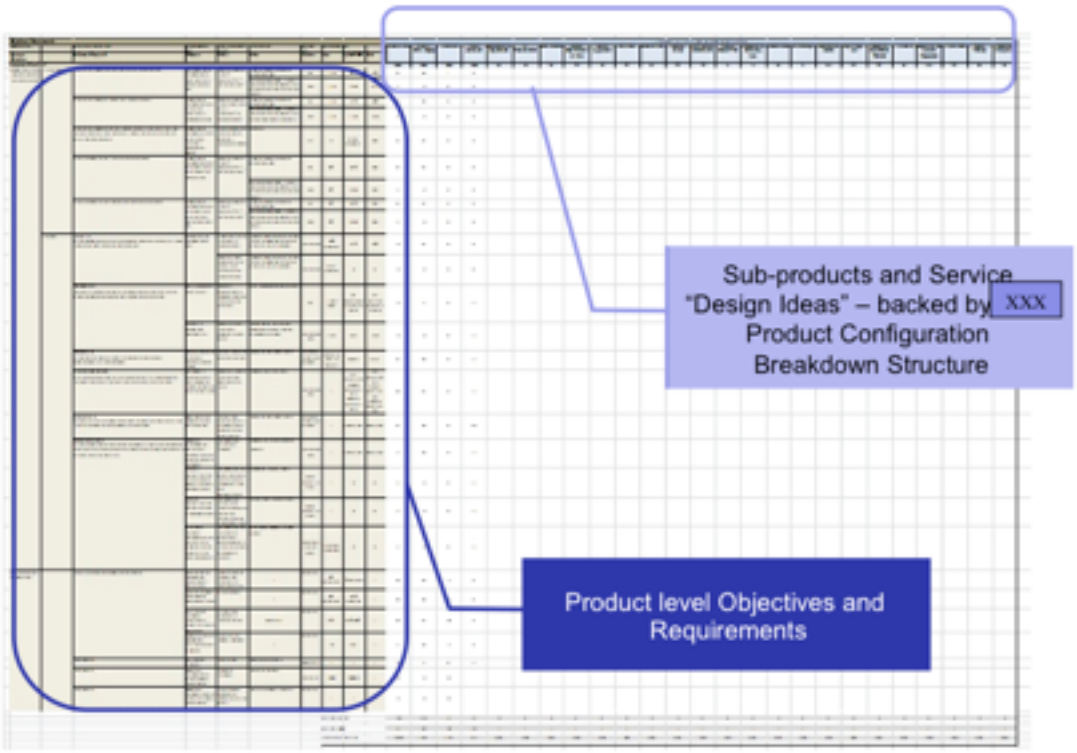


Diagram 9.7 C Here from the same management presentation is a miniaturization of the Objectives/Strategies spread sheets used to model the planning for a major new product line. Some detail from this was shown in Diagram 9.4 A B C. No, the text is not supposed to be readable.

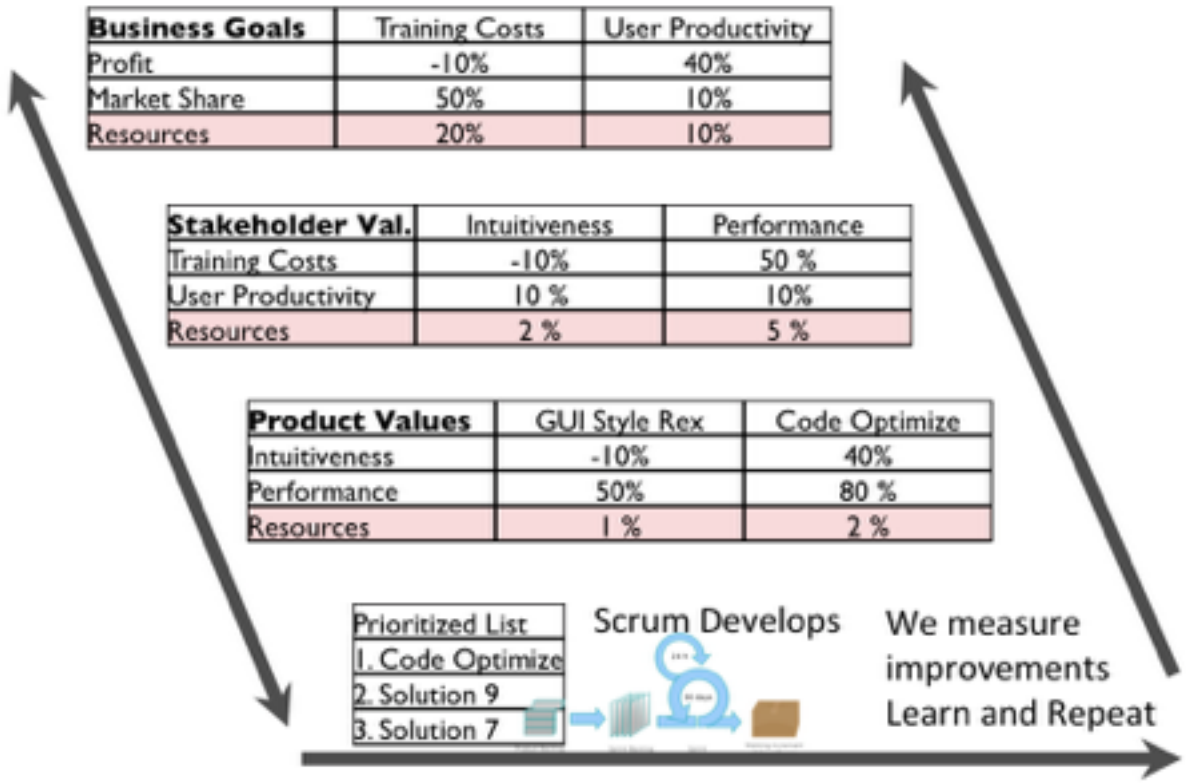


Diagram 9.7 D. A simplified extract from a much-larger real case, The 'Bring' Transport Company Case [URL36] by Kai Gilb. Notice how the strategies at the higher level become the Means Objectives at the next level down. The 4th level in this case is an Agile 'Scrum' backlog of things to do. When 'done' they should contribute to the Product Values, and so on up the hierarchy. So we decompose top-level objectives to get to a lower level, and when actual changes are made the values flow upwards towards the top level.

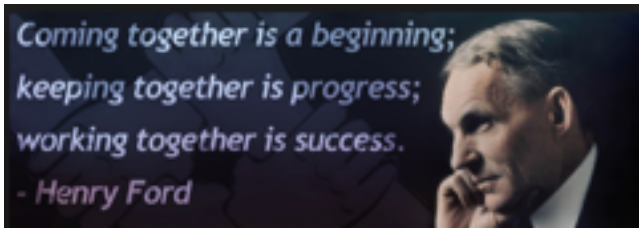
Kai Gilb has developed a spread sheet-based IET tool [URL68], which we use with clients and students. It does 3 levels of concern simultaneously, in real time using Google Docs. It 'models' 3 levels of a corporation in real time planning processes, and even feeds in results, from the Evolutionary project management Value Delivery steps. It is not for sale, but users are allowed to keep a copy. Most of our clients build their own spread sheet models to do these things.

The Commercially available Richard Smith Planguage tool also covers multiple levels [URL73].

Quote 9.7 A

"If everyone is moving forward together, then success takes care of itself."

Henry Ford



Quote 9.7 B

Practical Tip 9.7

- Connect your planning objectives to your next level of management upwards.
- Use a simple Impact Table to model the relationship.
- How many levels of interest can you outline, that might be interesting to map?

Policy 9.7 Connect Stakeholder Levels Directly

- **We will model each level of concern, using Impact Estimation tables, and then *connect* these levels automatically, and numerically, to give better communication and planning overview.**

- Why?
  - To communicate better about any and all organizational efforts
  - To ensure far better **alignment** to higher level objectives
  - To help management see the meaning for them, of what is being done by us, in support of their objectives

Version 24 Dec. 2015 02:47 CET, TsG

Section 9.8 The incremental value delivery of a series of strategies; stepwise learning.

Principle 9.8 Big Picture

Summarize overall strategy impacts on all objectives and constraints, with respect to riskiness

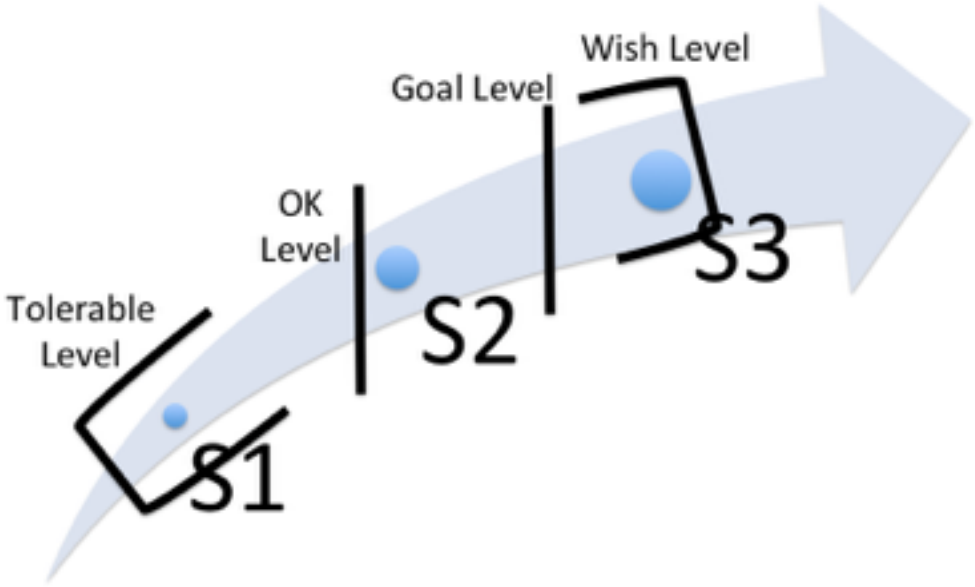


Diagram 9.8 A. The cumulative effect of 3 strategies with respect to constraints and targets for one objective. Strategies add up. You can estimate it, and you can measure it. And you can measure it and its changes for the system lifetime.

It would be quite a surprise if one strategy alone fulfilled all 'Competitive' Goals. The norm is that several strategies are needed.

So, we need to consider how to 'add up' the impacts of strategies.

They don't simply add up, but they do have a cumulative effect: sometimes 'synergic' (2+2 > 4), sometimes 'thrashing' (2+2 < 4) with each other.

Some bad strategy-evaluation practices:

1. Simply specifying the 'names' of a set of strategies, with no attempt to estimate, or indeed to measure, their impacts, on objectives (where the Goal and Scale are probably not even defined, in any case).
    - This is the most common practice. It is doomed to fail

2. Implementing all strategies in a big-bang build, and *hoping for the best* when the new system has gone live.
    - You are doomed to unpleasant surprises, and difficulty of retreat
    - You will have no idea what actually works, in the strategy mix, and what does not. You cannot selectively retreat.
    - This is how Generals, ran the World Wars: bloody defeat for some. Too much blood spilled for all. Everybody a loser anyway.

Some *better* strategy evaluation practices:

You can 'roughly estimate' and 'add up' the individual impacts of, multiple strategies, on multiple Goals. See Planguage Table 9.8 for example.

- This *cannot* give *accurate* predictions of effects.
- But it can give a *reasonable initial assurance* that you have, for the moment, until further feedback is available, identified enough strategies, to allow you to continue
    - (continue planning with doing decomposition, partial implementation, measurement and learning).
- You can decompose the best strategies.
    - 'Best' according to your priority policies, for example 'value-for-money'.
    - You then *try out* the most-promising sub-strategies in practice.
    - You can then learn what really works, what it costs, and what does *not* work. See Diagram 9.8 B Cycle.

- This experience can be 'fed up', or 'fed back', to the top levels of planning, and can result in:
    - *dropping entire strategies*, because the promising decompositions did not work well enough,
    - realizing that *some* marginal (less value-for-money than others) *strategies will never be needed*, because you can attain your goals with strategies that have been already applied successfully.

In other words our method is one of **evolutionary** (incremental with feedback, Plan Do Study Act [URL42]) strategy planning, and strategy evaluation, in real-life practice, by adding strategies to existing systems, with the option of retreating when strategies fail.

Big Failure is structurally impossible using this approach. We just do not place big risky bets! And we bet on 'sure things' first. This is why people using our methods never fail big, and usually succeed quite early, and totally.

Diagram 9.8 B. © Kai Gilb. The 'Evo' Value Management Process.

Comment on the Value Management Process (Diagram 9.8 B above)**.** This is an extension of the Deming PDSA Cycle [URL42].

The bottom oval is a 'construction process' (build the new system, or evolve it from an older system).

The rest of the cycle is about consciously managing the quantified stakeholder values, and ensuring that the focus is on *value*, not building.

This is the management cycle that actually 'cumulates the effects' of the strategies, one by one, until Target levels are reached, or until there is no competitive need to continue getting better.

Dr. W. E. Deming told me, about 1983 in London, that the PDSA cycle goes on forever '**as long as there is competition**'. Interesting point!

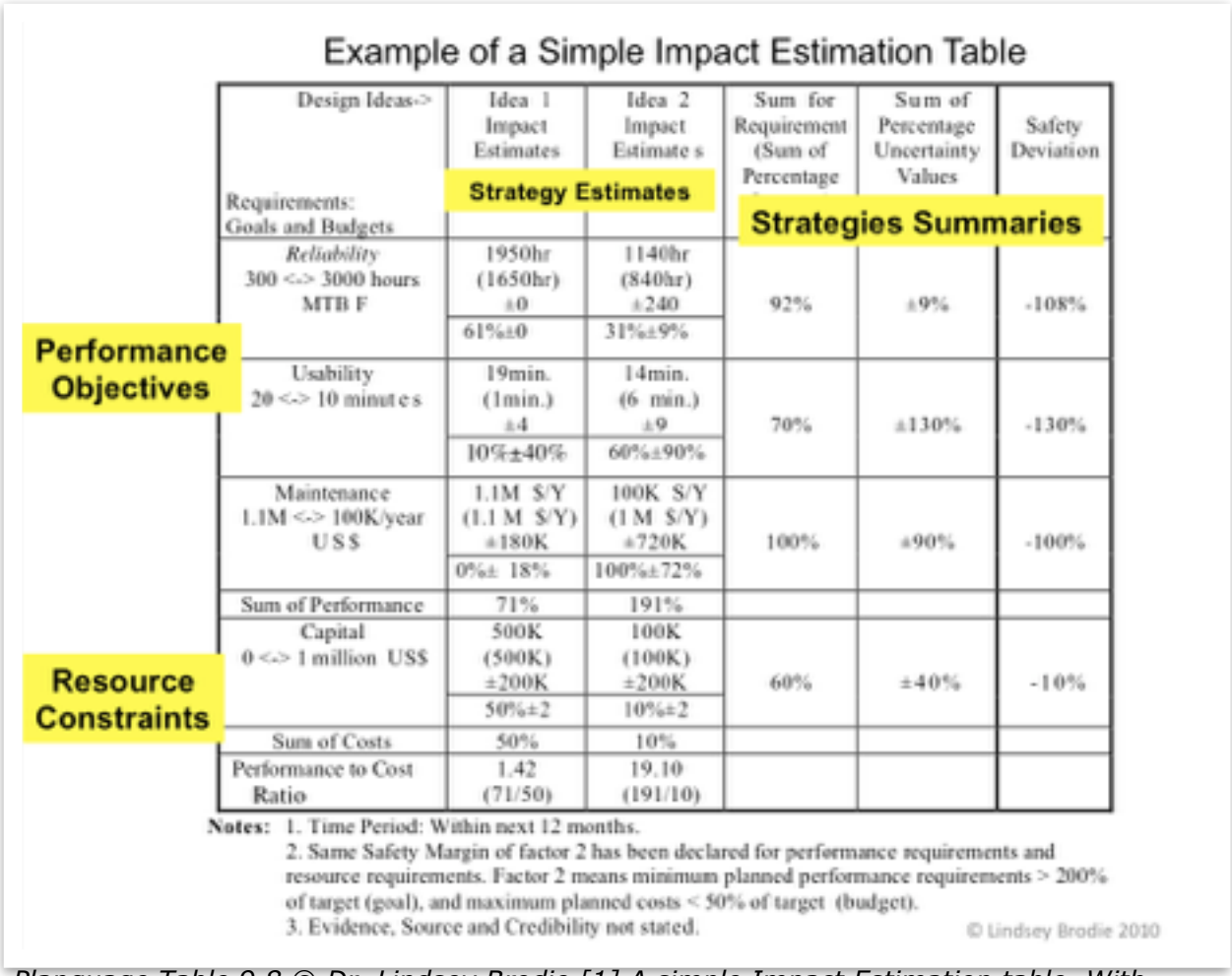
So there is a subtle point here:

- Conventional notions like the 'end of a project' do not apply: this is a competitive **process**.
- If the value delivery cycle, an *experimental* cycle, is short (we prefer 'one week') then *it does not really matter where you enter or begin the cycle*. Any convenient point will be OK. The notion of *beginning* with requirements, *might* be less valuable than beginning with getting experience *first*, *then* setting requirements! But, it doesn't much matter, '**Just Do It**'.

Some subtle points and practical considerations:

- The impact of any strategy, on any objective depends on the strategies *already in place*.
  - So the estimate you made for Strategy Y, when Strategy X was NOT in place, is probably invalid when X *is* in place. And in any case the initial estimate must be reconsidered now that we have feedback from Y. Sort of like cooking or chemistry.
  - Estimates of the effects of adding the next strategy to the current mix are very difficult because:
    - There is probably no previous experience of that exact combination of strategies and environment
    - The slightest practical detail, in this and previous strategy implementations, can change the effect substantially; Butterfly Effects.

- The initially *estimated* impact is rarely the same as the 'real impact', experienced and measurable, once a sub-strategy has been really implemented. We have little accuracy in our estimates.
    - So we need to update our Impact Estimation tables, with the real observed impacts [as Confirmit did, URL28 esp. C and D], as we go along. We cannot keep the older estimates; they are invalid, and they would give us a false idea of our future planning needs.
- The initial 'real impact' may itself be a misleading signal, and require reconsideration:
    - Because the measurement methods were too quick and dirty (like too-small an unrepresentative sample).
    - Because we used leading indicators, which did not reflect the later lagging indicator measures.
    - Because the system, under later heavier load (more people, more customers, more product etc.), in time may display quite different measurements, and even 'break down' under the pressure, or wear (on people and machines).
    - And so *initially acceptable* sub-strategies might have to be improved or replaced.
    - Not to mention, that in a competitive environment, an initially satisfactory level of performance, may become unsatisfactory, when competitors respond or improve. Yesterdays successful strategy, and Goals, may be tomorrows failure.



Example of a Simple Impact Estimation Table

Design Ideas-> / Requirements: Goals and Budgets	Idea 1 Impact Estimates (Strategy Estimates)	Idea 2 Impact Estimates (Strategy Estimates)	Sum for Requirement (Sum of Percentage) (Strategies Summaries)	Sum of Percentage Uncertainty Values (Strategies Summaries)	Safety Deviation (Strategies Summaries)
Reliability 300 <-> 3000 hours MTB F	1950hr (1650hr) ±0	1140hr (840hr) ±240	92%	±9%	-108%
	61%±0	31%±9%			
Usability 20 <-> 10 minutes	19min. (1min.) ±4	14min. (6 min.) ±9	70%	±130%	-130%
	10%±40%	60%±90%			
Maintenance 1.1M <-> 100K/year US$	1.1M $/Y (1.1 M $/Y) ±180K	100K $/Y (1 M $/Y) ±720K	100%	±90%	-100%
	0%± 18%	100%±72%			
Sum of Performance	71%	191%			
Capital 0 <-> 1 million US$	500K (500K) ±200K	100K (100K) ±200K	60%	±40%	-10%
	50%±2	10%±2			
Sum of Costs	50%	10%			
Performance to Cost Ratio	1.42 (71/50)	19.10 (191/10)			

Performance Objectives

Resource Constraints

Notes: 1. Time Period: Within next 12 months.
2. Same Safety Margin of factor 2 has been declared for performance requirements and resource requirements. Factor 2 means minimum planned performance requirements > 200% of target (goal), and maximum planned costs < 50% of target (budget).
3. Evidence, Source and Credibility not stated.

© Lindsey Brodie 2010

Planguage Table 9.8 © Dr. Lindsey Brodie [1] A simple Impact Estimation table. With summaries in all directions, including ± uncertainty, and safety margin. The Credibility Factor, which would give additional risk analysis, is not included in this example.

- ***"It takes as much energy to wish as it does to plan."***
- ***"Learn from the mistakes of others. You can't live long enough to make them all yourself. "***
- ***"Great minds discuss ideas. Average minds discuss events. Small minds discuss people."***
- ***"The purpose of life is to live it, to taste experience to the utmost, to reach out eagerly and without fear for newer and richer experience."***

— Eleanor Roosevelt (1884-1962)

Photo 9.8 Eleanor Roosevelt, Anna Roosevelt, and John Boettiger, Jr [F10]

Practical Tip 9.8

- ***It is simple:***
  - ***real systems are very complicated and dynamically intertwined.***

Policy 9.8 Continuously CALCULATE THE OVERALL PICTURE

- **We will use Impact Estimation tables, in a continuous mode of discovery and summary, to get the emerging big picture of all strategies, all goals, and all scalar constraints.**
- **We will also consider including estimation of the risk factors, uncertainty and credibility, both individually and by simple addition, when it would seem a good reasonable and useful investment.**
  - Keep in mind, in some systems, the properties are not simply cumulative: they emerge in difficult-to-predict combinations of strategies.
  - We can at least systematically measure and observe the emerging and surprising properties of our system, and hopefully exploit the surprises to our advantage
  - some surprises will be positive, and we can quickly act to get rid of the negative surprises.
- Why?
  - The combinations and details of strategies that exist are innumerable, and their influence on the objectives we desire unfathomable in advance.
  - So we can make initial guesses, as best we can; humbled by the impossibility of accurately knowing.
  - And we can quickly test, promising hypotheses, and find out what is *really* emerging from our experiment, like the good 'Management Scientists' we need to be, in the face of the complexity [URL87] of our task, and our local planning universe.

Version 24 Dec. 2015, 03:02 CET TsG

Section 9.9 How to play safe, with a Safety Factor.

Principles 9.9 Play Safe

Build up your planned set of strategies until you have a safety factor of at least two.

- If we could accurately estimate the individual impacts of our strategies on all our objectives and resources, and
- if strategies were simply arithmetically additive, and
- if the strategy implementers did their job properly, without misinterpreting our plans, or cutting corners, then
- in theory, we would need a set of strategies, that would add up to exactly 100% (on the % of Impact scale).

But, since we are nowhere near fulfilling any of the above conditions, we need to plan extra 'back up' strategies, just in case.

This subchapter is slightly ambiguous. On the one hand we will look at the attribute of safety itself (keeping people and things safe from harm, and bad risks) in systems. But the real theme is the tool of the 'safety margin' (as expressed by a numeric safety factor, like 2X), when planning, for *all attributes*, in our planning, and our execution of plans.

How important is a safety margin in safety planning?

"Safety is not an intellectual exercise to keep us in work. It is a matter of life and death. It is the sum of our contributions to safety management that determines whether the people we work with live or die" –

Sir Brian Appleton after Piper Alpha (Oil Platform Disaster, 1988).
(Kai and I use the methods, in this book, to help plan Oil Platforms these days.)

Here is some safety advice for your next walk in our Norwegian mountains. The mountains, as a tourist experience, are also a life and death matter, every year.

The Mountain Code

1. ***Be prepared. Be sufficiently fit and experienced for your intended trek.***
2. ***Leave word of your route. This can mean life or death in case a search is necessary.***
3. ***Be weather wise. Check the weather forecasts, but don't always trust reports of good weather. Good weather can turn bad in an instant.***
4. ***Be equipped for bad weather and frost, even on short walks. Always carry a backpack and proper mountain gear.***
5. ***Learn from the locals. Experienced local trekkers can inform you of safe routes, weather conditions and things to look out for.***
6. ***Use a map and compass. GPSes are handy too, but don't rely on them. A flat battery and poor reception can cause problems.***
7. ***Don't go solo. Being all alone in the mountains can be a magnificent experience, but in case of an accident it's good to have someone who can give first aid or get help.***
8. ***Turn back in time; sensible retreat is no disgrace. If you are not sure if you can reach your destination because of weather or conditions, turn around! Others might have to risk their lives trying to rescue you. Also try to notify anyone that may have been expecting you.***
9. ***Conserve energy and build a snow shelter if necessary. Eat and drink frequently, and try not to work up a sweat. If you need to build a shelter, do so before you are exhausted.***

Quotation 9.9 A. The Norwegian Mountain Rules [URL58]

The Mountain Rules are good planning rules for many situations. Rule 4 is of interest here.

"4. Be equipped for bad weather and frost, even on short walks. Always carry a backpack and proper mountain gear."

- In my interpretation, the backpack should have food for about 3 days, in case you get stranded in a 3 day storm. If you don't hit the storm, fine. You don't *have to* eat the food either. Save it for your next trip.

We think the strategic planner needs to use that same rule. We need to simply plan more strategies, than we initially think we will need, in order to meet our goals.

We do not have to pay for, or implement, the strategies we do ***not*** *actually need, once the Goals are achieved. But it should be a comfort that they are there, 'in the back pack', in case of a 'sudden mountain storm'.*

Engineers are conservative in the face of risks with complex systems, so they can make systems work as planned, on time, and hopefully within budget

Quotation 9.9 B.

"The Engineering Method is the use of heuristics to cause the best change in a poorly understood situation within the available resources"

"Engineering is a risk-taking activity. To control these risks, engineers have many heuristics:

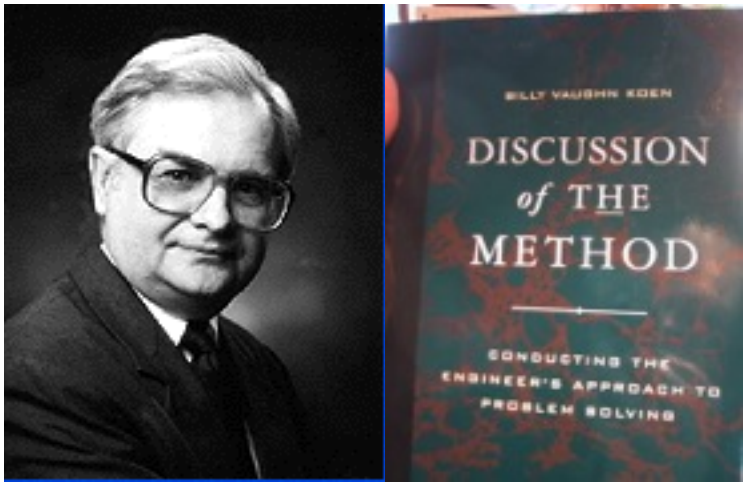
1. They make only small changes in what has worked in the past, make small changes in the Sota.
Sota' = Engineering State Of The Art Heuristics

2. They try to arrange matters so that, if they are wrong, they can retreat, and

3. They feed back past results in order to improve future performance."

"Engineers cannot simply work their way down a list of steps, ... but ...they must circulate freely within the proposed plan ...".

Prof. Billy V. Koen, Retired from Univ. of Austin, Texas [12, 20]



Quotation 9.9 B Koen

I love this definition of engineering from Billy Koen. His examples of Heuristics above (which I would call 'Principles') are useful reminders of the principles in this book.

But there is one fundamental engineering, and financial heuristic I would like to bring up just here. It is the Safety Factor.[URL61]

Some of my Dutch engineering friends once taught me that metal structures need a traditional safety factor of about 1.7 to 1.8 (70 % more load bearing capacity than theoretically required), more or less depending on which metal they are using, iron or aluminium.

And most of us know about the safety factor concept of 'distance to the car ahead being larger, depending on your driving speed'.

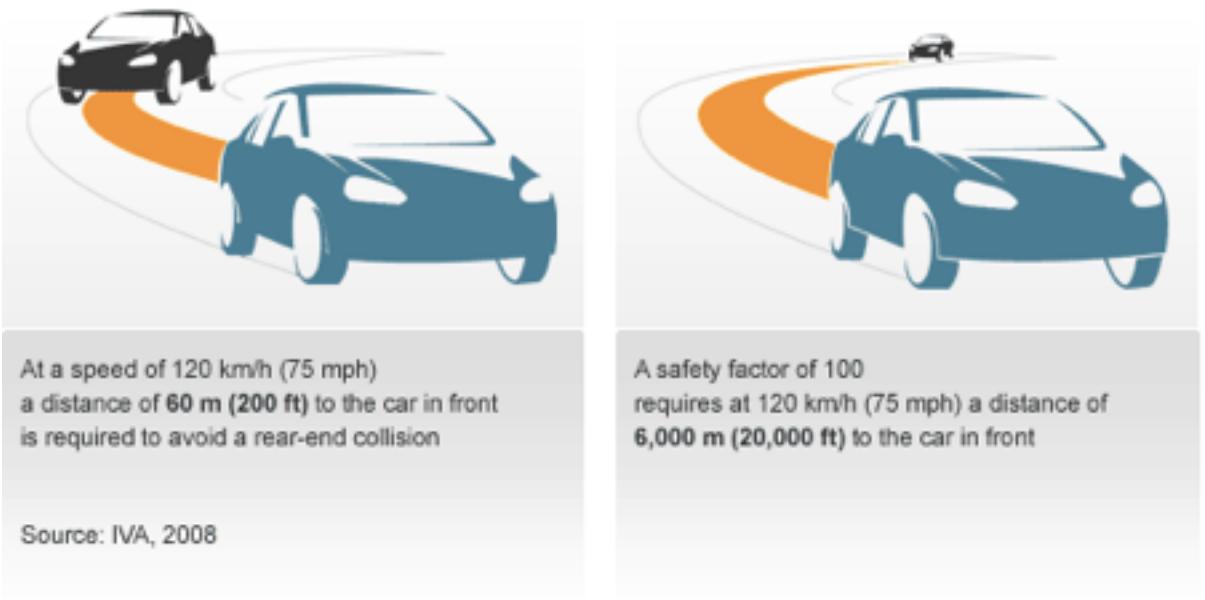


Illustration 9.9 A Car safety factor. [URL59]. I wonder which factor is programmed into collision avoidance computers? Maybe you can adjust the factor yourself ?

And in business economics

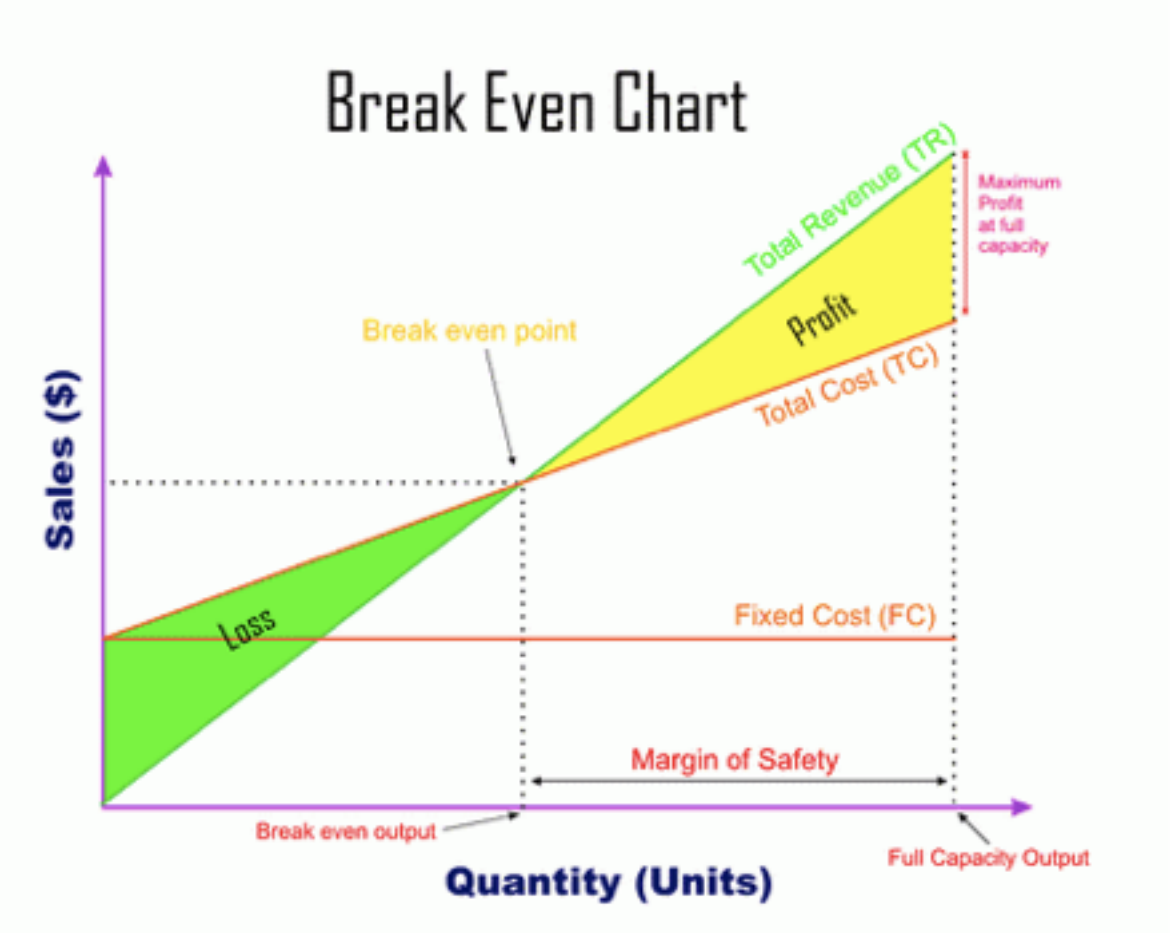


Illustration 9.9 B An economic margin of safety. [URL60]

You get the idea. This is an old idea, used in many disciplines, in nature, and your daily life.

So, long ago, we asked managers, who were going to get plans from us, using Impact Estimation Tables, what safety factor they would require for each Objective, before they would look at our planned set of strategies.

The answer is 'at least factor 2', and sometimes 4x (400% when all strategies are added up horizontally).

So we make this a general and formal Rule. That means when we do Quality Control on the Impact Estimation table, and the safety factor is not 2 (sum must be 200% or more, for factor 2), then we have identified a 'Major Defect', in the planning to standard. Enough to prevent release of the plans.

Impact Estimation Rules [URL57]

"R . Calculations:……..By default, a standard safety margin of factor 2 (200% for performance requirements, 50% for budgets) will be used.

For example, if the required safety margin is 200% and Sum for Requirement for a performance requirement is 120%, then ''–80%'' is the deviation to be displayed. (A 'horizontal' sum.)"

Planguage Example 9.9 in short, a 'default safety factor' of 2x is required.
This example is from Competitive Engineering Rules [1]. See application example [Planguage Table 9.8 above].

It is sufficient to just add the estimates left to right, and total them at the right hand [Planguage Table 9.8 above] for each objective or resource..

One could also add up the corresponding ± uncertainty factors, to get a cumulative view of uncertainty.

And one could subtract the pessimistic uncertainty border before adding up, giving the 'worst case' estimate.

It does not much matter how you do this, as long as you have clarity in your safety factor, and your planners, and management *plan readers* agree to the standard you are working to.

A simplification of this rule could be:

Rule [Safety Factor] Plan strategies to be twice as effective as your Goal actually needs, and a half the cost of your budgeted resources. But do not commit to actually implementing more strategy than you *actually* need.

It is a reasonable argument that:

- **If** you are not going to *commit* to a *complete* set of strategies at the outset, *(For example by a binding contract)*
- **and** if you are going to build value, as we consistently recommend, by implementing small inexpensive deployments of decomposed sub-strategies,
- **and** you believe that there is a sufficient 'supply' of additional strategies that are cost effective for you (even if you have not formally planned them yet),
- **then**
- you do **not need any safety factor** in your initial planning.

You can, in principle, just start immediately with the first small strategy that comes to mind, see if it works, and continue the process until all Goals are reached.

Possible 'Painting Yourself Into A Corner'.
The problem is, that if you are in hot international competition, the Goals are set high – state-of-the-art, and beyond (see Koen above on 'Sota', a very important concept (State Of The Art limits) for engineers).

Maybe you would need to venture *outside* the state of the art, into very risky science and research. And there may be, that there are *no* existing or available cost-effective strategies to get you to your Goals on-time and cost-effectively.

So, you risk starting, 'lift off', and not being able to *land*, where you wanted to cost-effectively. If you had known *that*, you might have waited, or made alternative plans.

Consequently, since we have experience, in our Project Start-up weeks [URL3&4] that we *can* usually get a safety-factor-2 impact estimation table, by the third day of the process, why not give yourself that luxury, and comfort of a safety margin?

If it took 10 weeks, or 100 weeks, to plan a good safety margin, *then* we could talk about dropping the safety margin planning with extra strategies, and go to simply 'learning by doing'.

Quote 9.9 C

> ***It's a very sobering feeling to be up in space and realize that one's safety factor was determined by the lowest bidder on a government contract.***

Alan Bartlett Shepard ***(1923-1998)*** *[F11]*

Practical Tip 9.9

Use a safety factor of 2 *(200% impact theoretically for your planned strategies)*

Policy 9.9 Doubly Cautious: Redundant Planning

- **We will normally collect enough strategies, to give us a safety factor of two.**
- Why?
    - For all you can know, that kind of 'over planning' might be 'just right', the real minimum, to get to your Goals.
    - It should cost little, and normally have far greater value: as a 'reserve tank', when fuel runs out unexpectedly.

Version 24 Dec. 2015 =3:20 CET, TsG

Section 9.10 The 'Project Startup Week': Necessary Minimum Planning, then into battle.

Principle 9.10 Plan A Little, Act Early

Build a reasonable Impact Estimation table in a day or so, and start testing the most value-promising strategies by the next week: don't fall for analysis paralysis: act.

When people see a real 10 x 10 impact estimation table and they hear me saying 'we did this in a 3 day workshop', they are quite taken aback.

They feel they would have trouble enough with making any single strategy impact estimate, let alone 100. They have trouble enough with quantifying a single Goal, let alone ten.

They certainly do not think they can make an Impact Estimation Table for *their* project. Their project has been dragging along in meetings for months and has not achieved this level of clarity. Even the idea of doing this alien act of 'quantifying and estimating quality levels' strikes fear of failure in their hearts.

But, as experience tells us, if you just get started. Take it one simple step at a time. Set clear daily deadlines. Time box it. Then you get this quantified result, a 10X10 Impact Table, every time. There is no failure, ever, if you just work at it.

The numeric results are far from perfect, and far from perfectly correct. But months of *planning* would not make them significantly better.

At least we are not 'up to our ears' in hundreds and thousands of details, before committing to 'practical value delivery action'.

We are initially very focused at the *top level* of our concern.

- The top ten critical objectives, quantified on a single page, on Monday.
- The identification of the maximum top ten 'power' strategies, on a single page, by end of Tuesday.
- And to summarize, the 10x10 Impact Estimation Table at end of Wednesday.

Maybe we need quite a bit of guessing initially.

- But 'expert opinions' are now at least *clear (documented with numbers)*, if often somewhat wrong, at the moment.
- We have a project startup structure which can be used to make decisions about priorities, including priorities of *getting better data* on both goals and strategies; which includes the short-term notion of strategies which 'we should try out quickly', next week.
- We can start the process of value delivery. The process of really getting to know the real system, the real organization, the real marketplace. It is the process of 'building our project credibility', as we actually deliver real tangible value.

We gave an overview and a few examples of this project startup week in Chapter 5.8. [URL3, URL4, URL5].

Now I'd like to revisit it, and try to go deeper. Exploring it as the Power 'Communication Tool' it is.

Figure 9.10 A. The startup Week, day by day. [URL4 A, the standard]

Assemble a team, of 3 to 10 people. Certainly include the project manager. Include anybody else who is a strong proponent of change in this area.

Get responsible management to address the team in the first hour, and tell you what they expect, and what they will support.

Hopefully: *"I want a clear quantified structure for the objectives and strategies for this project. And I want quick convincing results. I will back you fully in getting these results."*

If people have not been trained or experienced in quantifying all objectives, you will need a coach in that process too. But the team does not have to go on a course to learn to do this. Learning on the job is the best way.

Figure 9.10 B. The Startup week: Purposes, day by day.

The critical essence of your outputs for each day can easily be *summarized* on a single page, one page each day.

During the daily work process it is natural to divide into a few parallel work teams. Each team will work on a *few* of the 10 objectives, and a *few* of the strategies. Their output of each objective and strategy can be as little as few lines of text, up to a full page of text.

It is a very good idea to have a daily briefing for the management. This gives people a deadline motivation to work towards. It gives the manager some feedback that things are progressing satisfactorily.

It gives the manager

- time to absorb what is going on step by step. No big surprises at the Friday 'Management Presentation'.
- a fair chance to question the wisdom of what is happening, to suggest ideas THEY (or other powers) prefer (if they dare to share them!), and to warn of cultural problems in the organization, that the team might not be aware of.
- an early opportunity, to open additional communication lines to other parts of the organization, that need to give inputs to the process: maybe quickly bring in additional team members for a day or the rest of the week

The Startup Week, Day by Day.

Figure 9.10 C. Monday of the Startup week: Clear Quantified Critical Objectives.

The agenda, as a template, simplified, is, for *each* of the objectives: specify

- **Tag**:
- Type: Top Level Critical Objective.
- Ambition
- Scale
- Past
- Goal/Wish

Planguage 9.10 A. A basic target statement for an objective

But more-detailed templates might be used, with an opportunity to collect more-detailed information, such as justifications, sources, experts, value of reaching the Goal, issues, risks, dependencies, constraints, special conditions, and other interesting [qualifier] variations.

Just be done the same day, with pretty good drafts, one hour before finish of day. Use the last hour to present, review and discuss, so everybody sees what the others are doing. Then they can perhaps get inspired to work to a higher standard. They they can get feedback from other people than their initial sub-team on their content.

It is very nice to get this, and all later work put into a spreadsheet tool, or specialized Planguage tool like Kai Gilb's Tool [URL68]. If this is in Google Docs, then everybody has full real time access to the buildup of the objectives.
Or Richard Smith's Tool [URL73] which has built-in simultaneous access to the ongoing plan buildup for everyone.

If this is a group of people who are not trained and experienced in doing this, then some form of coaching is necessary, to make sure people don't get stuck, and if they do, they can get unstuck rapidly.

Guidelines in this book for this work, and examples, are found many places. Particularly Chapter 1.

Day 2 Strategies: how to get to our Goals

Figure 9.10 D. Tuesday of the Startup Week: identify and flesh out the most powerful strategies needed to help us reach all our goals, within our budgets for time and money. 'S1 -> O3' is symbolic for 'Strategy 1 impacts Objective 3'

If you have people like System/IT Architects, CTO, Technical experts, it might be the right day to invite them in to this strategy 'design' party.

Use the first workshop hour of the day (after perhaps a manager briefing, and perhaps with the manager still present at the brainstorming) to brainstorm the *most powerful* strategies.

The input to this session, the drivers of today's strategy planning activity is primarily the top-level critical objectives, with specific numeric Goal (Wish) levels, with deadlines for the Goal, and perhaps with other qualifiers, like - to *which* stakeholders, markets, tasks, working conditions etc.

If there has been some kind of strategy list previously, or one derived from analysis of earlier meetings and presentations, then this raw list can be made available as another input, so people do not forget to consider potentially promising strategies.

One way to approach this is :

1. Focus initially on any interesting Goal. Finally run through all other top critical objectives in turn.
2. Ask then '**what is the single most powerful strategy for getting to the goal?'**
    - If there are several strategy suggestions, note them.
    - this is *brainstorming*; do NOT argue or criticize: an evaluation be done Day 3, with Impact Estimation.
    - But do not allow an opinionated person to intimidate or scare other people off, even if they are 'right'.
3. If you like, you can ask *quick* questions, and make a *quick* note, in the planning, about people's opinion, regarding
    a. the **rough degree of impact** (total, partly, a little bit)
    b. the **rough costs**: (far too much, possible to swallow, not too bad, trivial, totally unknown)
4. Avoid discussion. It takes too long, and is not necessary here.
5. If necessary, ask people who want to argue, to feed their ideas in writing to a person (perhaps the Specification Owner for that Strategy, if appointed) who will collect them, and integrate them, with credit, into the rough strategy specification. All voices will be heard and considered later. None will be forgotten. But this day is not about debate.

This initial strategy brainstorming session should be concluded after a single hour, with a planned break. You should end up with a page of about maximum 10 powerful strategy ideas. The list does not have to be complete or correct. We can change anything later. It should contain a lot of very powerful ideas, and at least one strategy 'worth trying out' in the short term.

The rest of the day, before the final hour, should be spent in small (2-3 people) parallel teams, each team working on 2 or 3 strategies, using about one hour for each strategy.

They need to work with a template, that minimally looks something like this:

Strategy Tag:
Type: Top level Powerful Candidate Strategy.
Version: <Date and Time>
Owner: one or more people take responsibility for the specs here
Short Description: <10 to 20 words>
Detailed Breakdown: S1 S2 S3 etc
Main Objective Impacted: <give tag of the objective>
Other Objectives Impacted: <give tags of the objectives>

Capital Costs: <name them, rough estimate if you can>
Maintenance Costs:
Issues:
Dependencies:
Risks:
Assumptions:

Planguage 9.10 B. A simple Startup-Week Strategy data collection template. This can also immediately be put into a spreadsheet tool, integrated with the objectives, and later serve as a basis for impact estimation. But it can also simply be a basis for later putting some of the information into such a tool.

Guidelines in this book for this task, and examples of specification, are found many places. Particularly Chapters 2, 3 and 5

Day 3: Impact Estimation: how good are the strategies for our purposes

Figure 9.10 E. Putting best-estimate numbers on the impacts of our top strategies on our Goals and costs: Do we have a fair chance to get to our goals, or are they more ambitious than our reality - of strategies, that we can afford?

The Wednesday, 3rd Day of Project Startup Planning, is focused on getting a first estimate of how good the suggested strategies are. We insist on a very *systematic* approach here. We consider it silly to just decide on strategies based on opinions, fashions, powerful supporters etc. If we are *really* going to deliver value, then we have to manage that value numerically,

Anybody can 'estimate', but they then 'owe us' their *evidence*, and perhaps their *source*, so we can understand the estimate's *credibility*. In principle the people who *suggest*, and *champion* a strategy, should be the first to estimate its benefits and costs.

Once an estimate, with evidence, ± uncertainty, sources, and credibility is drafted, those who have different opinions are welcome to add their evidence etc.

The important thing to keep in mind is the purposes we have in making the estimates:

- The estimates are NOT
  - A binding commitment by anybody, or
  - the accurate truth of what will happen, or
  - information that is static, and cannot be changed to better estimates later.

- The estimates ARE
  - A way to **communicate opinions** about what we should **do**, and what we should **prioritize.**
  - A way to **communicate the multiple side-effects** of a strategy, so we don't get to narrowly focused.
  - A way to **ask for better information**, sooner or later, from those who might know better.
  - A basis for **learning**, by getting corresponding measurement and feedback from actual strategy implementations.
  - A basis for deciding to swap out 'bad' strategies with much smarter ones (dynamic **design to cost**, [URL11, URL12]).

Guidelines in this book for this Impact Estimation task, and examples are found many places. Particularly Chapter 9, this Chapter, earlier.

Day 4: Finding something useful to do next week.

*Figure 9.10 F. Time to get very practical. **What can we do 'next week'**, to try out the most promising strategies, in our real environment, and see if we are capable of doing anything useful at all, even the simplest thing?*

This is Thursday's work , or day 4. Lets keep it real simple,

Figure out something, anything we can really do next week, to deliver some value on our Goal-specified value agenda. [URL16]

It never ceases to amaze us, that so many people we work with are in immediate denial about the possibility of doing anything realistic 'next week'.

Yet every time we work our way towards this idea, they *themselves* find solutions. The solutions to doing something practical next week are many, and the strategies are always surprisingly simple and obvious.

There is a big cultural and educational barrier here. But we can dissolve it by practical experience. Too bad the business schools are not into this yet! At least, practical and successful business people are into it, long ago [8, and many other of our book references].

If our teams find *several* 'next week', possibilities, and you have enough capacity; do several. Next week. In parallel.

Guidelines in this book for this task of finding a value delivery possibility next week, and examples, are found many places. Particularly Chapter 5 (Decomposition) and 6 (Priority)

The Management Presentation, Friday Day 5

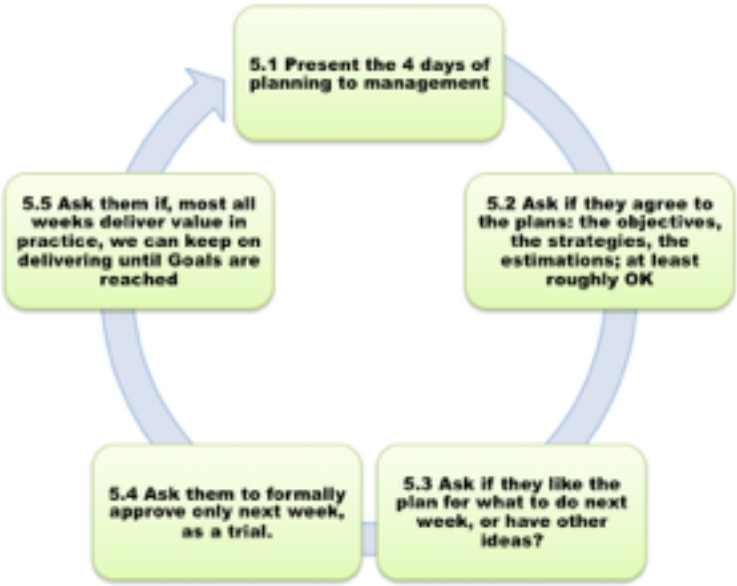


Figure 9.10 G. Getting management support for the launch into at least the first week of Evo value delivery. They cannot resist seeing how this will work. But they might well have some advice on **where** *to try* ***what*** *next week.*

As mentioned earlier, if you can involve your management, in small time-slices the first 4 days, they come far better prepared, to the Friday meeting. They can then get rid of anything that would otherwise cause them to say 'no', days earlier. Early management briefings can help us get 'we need some changes before I can yes' remarks, early.

At McDonnell Douglas Aircraft we got managers to sign a form authorizing the next week. They always did, 25 aircraft projects. And they largely had at least a half hour every day to interface with the planning team. Often early evening for 30 minutes.

"Being busy does not always mean real work.

The object of all work is production or accomplishment and to either of these ends there must be forethought, system, planning, intelligence, and honest purpose, as well as perspiration.

Seeming to do is not doing."

~ Thomas Alva Edison (1847-1931)

Quotation 9.10 Edison

Practical Tip 9.10 Try the Startup

- *Try the Startup Week once.*
- *Make up your own mind on whether it is a good way for you to initiate projects.*
- *Allow at least 1 to 3 following weeks of 'actual value delivery' to happen, before you draw conclusions.*

Policy 9.10 A Week Planning, followed by weeks of doing

- **We will start all projects with a Startup week, followed by weekly increments of value delivery, until all Goals are satisfied.**

- Why ?
  - Gives far better definition of project objectives and strategies than you normally will get

  - Puts us quickly, in a *doing and learning* cycle; necessary for complicated (difficult to understand, predict) systems, where we really do not understand much about the emergent properties, before they happen.

Version 24 Dec. 2015 03:43 CET TsG

Chapter 10. Quality Management.

Quote 10.

- ***"The first guarantee of quality in design is in well-informed, well-educated, and well-motivated designers.***
- ***Quality must be built into designs, and cannot be inspected in or tested in.***
- ***Nevertheless, any prudent development process verifies quality through inspection and testing.***
- ***Inspection by peers in design, by users or surrogates, by other financial specialists concerned with cost, reliability, or maintainability not only increases confidence in the design at hand, but also provides designers with valuable lessons and insights to be applied to future designs.***
- ***The very fact that designs face inspections motivates even the most conscientious designers to greater care, deeper simplicities, and more precision in their work."***

Harlan Mills IBM [URL41 A]

This Chapter will focus on planning processes for planning quality.

Most ‘values’ can be classified as qualities.

Qualities are ‘how well’ the thing works - the ‘-ilities’.

The first sections will focus on quality control of planning specifications.

Then we will look at quality control of the *implementations* of the plans.

Section 10.1 Defining which qualities any plan should have; the 'rule of rules'.

Principle 10.1 Define The Plan Quality You Need

Adopt a short set of powerful rules enforcing the necessary quality, for each different type of plan.

If planners have no clear, written, enforced rules to guide them, then the Major Defect rate (using such lack of rules) will be, according to our measurements, at least 100 defects per page. Common sense, good intent, or high education is not enough to stop this typical flood of unintelligible planning specification.

'Rules' are our short term for the 'plan-writing standards' we choose, which guide us on how to specify, and then to quality-control our plans. '*Specification Rules'* is *the longer form of 'Rules'.*

Simply allowing a planner to write as they normally would, is quite insufficient. Our long-term wide-spread measurements tell us that unmanaged freedom to write plans does not work well at all. Planners are incredibly sloppy in their planning language. They communicate badly to each other.

Most people are so accustomed to plans written to this everyday unintelligible standard, that they hardly see the magnitude of the problem, until they actually participate in the *measurement* of a plan with which they have some personal familiarity.

When a person finds 15 major defects in 15 minutes, in 100 words, they begin to suspect the quality is bad. But when the review leader's calculation, based in initial defect findings, is that there are *actually* 100 Major defects in the page, of which the team found 30 and you found 15 defects; and 70 more are provably there too, but not yet identified by the review team; then we begin to wonder if anybody at all can understand any plan correctly!

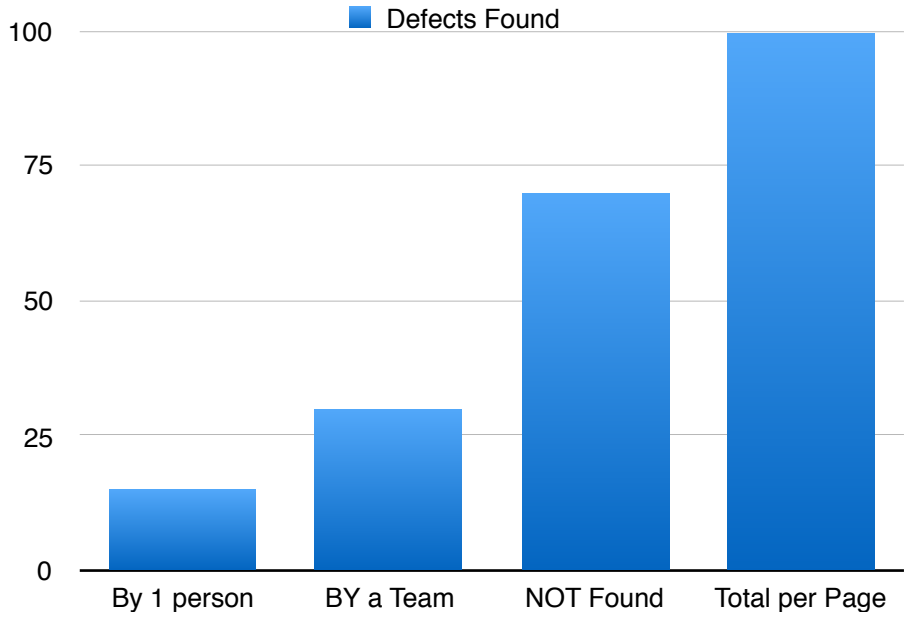


*Figure 10.1 A. One person can find about 1/6 of the defects on a page. A team can find 1/3. but 2/3 will be missed initially. So one person doing QC is not very good at it. Even a Team is not much better. So correcting all defects found by a team still leaves far too many to accept economically. The only solution, we know of, is to dramatically reduce the initial defect-injection rate from 100-per-page, down to more like maximum 1.0 per page [DPP, URL40]. As Mills said (Quote 10 above) "**Quality must be built into designs, and cannot be inspected in or tested in."***

There is a difference between:

- I can read it: and **My supplier can read it the same way**
- I can read it: and **I really understand it**
- I think I understand it: and **I provably do understand it as intended by the writer**
- I read it and understand it: and **But I cannot know what vital specs are missing.**

Suitable 'Rules' will help quality checkers to see the difference and find these kinds of defects.

There is an analogy: 'one way to eliminate crime, is to eliminate laws'.

The planning environments we see, are a Wild West, with little or no Rules, of even the most basic kind.

Let us start with the simplest basics:

1. do you think that your plans should be *unambiguous* to the plan readership?
2. Do you think your plans should be *clear* enough so that we can devise tests to ascertain that the plan is carried out successfully?

I never met a manager who would argue in public that their plans *should* be *ambiguous and unclear*. But, all their plans *are* ambiguous and unclear. The plans they write, and the plans they read. 100 words are typically defective per page of 300 words!

What happens if a plan is ambiguous?
It will get misunderstood, and misapplied.

What happens if a plan is unclear, and cannot be properly tested for intended delivery?
We cannot prove that it was carried out properly.

There is something we can do about this 'planning pollution', but few have been informed as to *how*, and fewer have actually done anything about it.

Here are our recommendations

1. **Legislate**: write, and agree to, some basic rules for planning specification
2. **Police**: review your plans, and find the defects: i. e. 'specs which fail to follow your rules'

3. **Judge**: decide how many defects are 'permissible' before you hand over the plan to someone who is going to use it.

   This is known as the 'Exit level, from the planning process'.

Figure 10.1 B. Specification Quality Control basics.

Case 10.1 Banking System Planning Quality Improvement

An international bank client of ours, in London, measured its plans, IT Requirements from the banking business to the IT developers, using the following rules.

R1. The requirement is uniquely identifiable
R2. All stakeholders are identified.
R3. The content of the requirement is 'clear and unambiguous'
R4. A practical test can be applied to validate its delivery.

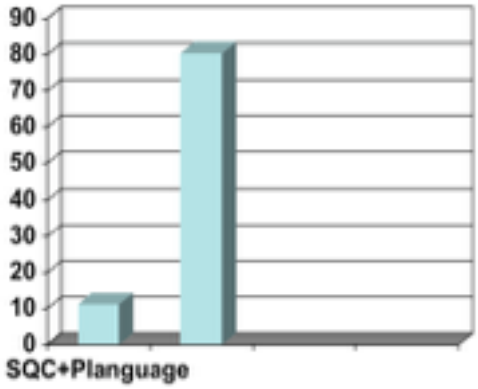


Figure 10.1 C. Initially 80.4 defects per page was normal. Six months later only 10 defects/page was the new norm. This is defect injection reduction using process improvement. It is not due to any 'detection and removal' process.

Case Study 10.1 Continued.

Six current project plans, written like most others, most always, showed a Major Defect rate of 80.4 Major defects/300 words. Figure 10.1 C
That is actually pretty good, compared to others who have far more (100 to 300 defects is normal).

But the IT Director, realized this was unacceptable. The 'Garbage In' to his people was causing so much delay, and low quality systems. And his IT division was getting the blame, but it was really due to the poor work done by the banking 'Business Analysts' poor quality of work.

At first he decided we should train the Business Analysts in clear planning, using Planguage. We did. But that alone did not help, because they did not report to the IT Division, and were under other pressures. They continued to produce bad plans.

If you would like to see another bank's example of highly polluted top level objectives, and then a detailed example of how one part of it "improved development environment' was rewritten to clearer Planguage, see case 2.10 above.

So, finally the Director (Vince Chandler, an innovative director) decided to *completely rewrite* all Business Analyst specs, using his own staff in the IT sector.

But first he decided to train his own people, in the practice of *clear plan specification,* using Planguage, a 3-day course. And then 'motivated' them, with a low numeric Exit-level of defects, before their work was accepted.

We decided on two Goals for planning-quality levels.
1. In the long term, the specs released to build IT systems, needed to get down to something like less than 1.0 major defect(s) per page.
2. In the short term, that year, we suggested it was realistic to aim for about maximum 10 Majors per Page. Figure 10.1 C

This is a big cultural change, and it takes time. How long time would it take you to improve anything *else* by eight to one? Like wealth, health, knowledge, power, strength, endurance?

By using the rules above (small and simple, right?) and using our Simplified Spec QC method [1 D] they reported, after 6 months, the result of the 'new requirements method' (Planguage).

"Across 18 DV (DeVelopment) Projects using the new requirements method, the average major defect rate on first inspection is 11.2" .

"4 of the 18 DV projects were re-inspected after failing to meet the Exit Criteria of 10 major defects per page". *Figure 10.1 C*

An 8X improvement in specification quality for real projects, in six months is a pretty good accomplishment. This translated, we assume, into better quality and faster delivery of their development projects. But we never managed to capture that evidence. Clients are under no obligation to report back to us, about results they consider competitive.

But the Intel case [3.8] is some evidence of what we know happens when planning defects are radically reduced. In this case we got lucky and got some evidence which was published, and cleared by their legal staff.

Getting to 10 Majors/page using Planguage, is about the same level as initially achieved by Intel (10 majors per 600 words, or 5/page), as written up here earlier [Case 3.8], for initial drafts. By re-working hard, Intel reduced the 'Garbage in' further to 0.22 majors/600 words, before Exit, and claimed a 200 to 300% project productivity result for doing so. [URL22]. They also reported 50% better logic quality.

"Raise your quality standards as high as you can live with, avoid wasting your time on routine problems, and always try to work as closely as possible at the boundary of your abilities.

Do this, because it is the only way of discovering how that boundary should be moved forward."

— Edsger W. Dijkstra (1930-2002) Computer Scientist.

Practical Tip 10.1 'Clarity' is the minimum basis for being able to judge plan 'relevance'.

- *At the very least you need the following rule for plans, to make sure they are 'intelligible'.*
- ***Golden Plan Rule**: **All plans shall be clear enough to test, and unambiguous to the intended readership***
- *That rule alone will help you to reduce the plan unintelligibility by 100 defects per page.*
- *Now you are ready to hunt for more-significant defects. like 'is it a* **an effective** *plan?'*
- *If you have 100 unintelligible words per page, you have no way to judge if it is a good (effective, cost-effective) or bad plan. It is simply 'unintelligible'.*

Policy 10.1 Set Planning Standards and get serious about them

We will clearly establish simple powerful rules [URL57] **for planning specification, so that we have a clear basis for measuring whether people are taking plan specification seriously, and for rejecting their work until they do meet our 'Exit' standard.**

Why?

- It works. People radically change technical or planning writing habits for the better.

Version 24 Dec 2015 16:32, TsG, now off to Xmas eve party with 4 grandchildren!

Section 10.2 Spec QC: Measure how bad your planning communication quality is.

10.2 Police It

All serious plans
need to be measured against
your specification rules,
immediately:
to motivate people to learn and to practice the rules.

You have 2 choices. You can accept your plans as they are, with probably at least 100 major defects per page; and accept the consequent project delays and quality problems to your projects.

Or, you can use Spec QC, suitable specification Rules, and Exit levels; to get vastly much better intelligibility of plans, and consequent time saving and quality. Most managers don't seem to know about the second option.

Now that **you** do know about it, what are you going to do about it?

OK, I'll wait until you read the rest of this, and you try it out once.

But if you do not adopt Specification QC then, would you *please* let me know a better method you know about? In over 50 years, I have never seen anything as effective as this, in changing real behaviour in planning.

The question you want to ask yourself is,

'how many major defects per page do you want to live with?

100 majors per page?

Zero sounds nicer.

The theoretical answer is that the Exit-level limit should be at least low enough to *force you to clean up the plans now*, while the costs are still very low compared to later.

Something like maximum 1 major defects per page is a good approximation.

Yes, that is probably 100x better than you probably are today.

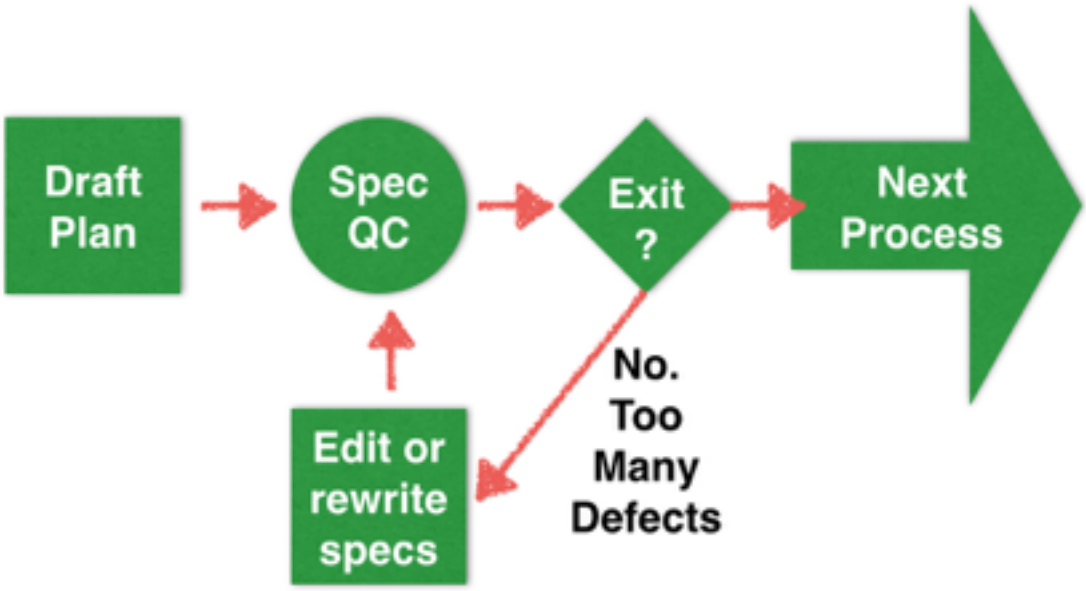


Fig 10.2 A. A plan needs to be quality controlled against its rules. The Spec QC process 'result' is an 'estimated total number of defects per page', based on a small sample. An Exit process can refuse exit to other processes, if the defect level per page exceeds a standard, such as 0.2, 1.0 or maybe 10.0. The Exit level is where it 'pays off' to release, and then fight remaining defects later in future SQC or in tests. It is not a matter of correcting the defects found. It is a matter of cleaning the whole document, so you can survive the next SQC, which can sample anywhere in the document. Sampling means that the measurement is quick (less than 1 hour), cheap (2 people) and can be done early in planning, and done frequently

The short-term answer is that *we should at least plan to get 10 times lower defects this year.* Then, see if that experience makes you want to get even better.

Major defects tend to cost 10 to 100 times or more, on average, later in your project [14]. So getting a clear clean defect-free specification pays off nicely.

One rare problem is measurement of defect density, when you are in fact near the ideal Exit level (say near 1.0 defects per page). Your sample might detect nothing, when you in fact have a very few defects.

Simple and cheap measurements, like Spec QC will find you about 30% of the defects actually in the document. So, if you have very few defects, like 0.5 per page, your QC process might say 'We found none', but you might have a few hiding anyway.

This is no big problem for most applications, but can be a problem for very high-quality systems like at Intel, which are, in published cases, aiming at maximum 0.10 defects/300 words for exit, about the same as NASA. But Intel persist, wisely, in using these methods, after many years of experience. [URL22, URL46]

The initial *advantage* of setting an Exit limit is that the unfiltered garbage can't get through. You have a guard at the gate. The next advantage is that planners *will try to meet* the Exit level. They are *motivated*.

It turns out that you cannot easily meet the exit level by simply correcting just the 'found' defects, as actually *found* by a review team.

They only find one *third*, and maybe they are only sampling a *small fraction (1% for example) of the entire planning document.*

Case 2.10 A. A-Sim at Ericsson.

In one real case we sampled as little as 3 random, but representative, pages of 40,000 pages of requirements for the Ericsson A-Sim (Air Flight Control Training Simulator) project, to get a same-day by-top-executive measure. There were 20 major defects per page. That estimate was accepted by the Division Director, when he checked the details, as a valid indicator of the larger planning-document defectiveness.

This is not at all about 'correcting defects *found'*. A defect density measure is just a 'signal' from your peers ("You have 50 Majors/page, the acceptable exit is maximum 10", or 1.0 or 0.1).

It is ultimately (this month, not next decade) about *learning your planning trade*, to a high, acceptable, quality level, as defined by the Rules, which *your* culture has chosen, as their *necessary* best practice.

So, our experience is that planners are 'forced' to learn by peer pressure.

"We have all achieved the exit level, and you can too, if you focus on learning how". Your peers tell you.

Planners are forced to learn what the rules *actually* are; and *how to apply them* in practice. They *have to* try to *'do it right the first time'.*

This is our experience [Boeing, Ericsson, and on training courses].

On the our public training courses, which are a good simulation of 'real life personal learning curves' [Diagram 10.2 (Ericsson), Case 7.7 A (Boeing)] we have done 4 measurements and corrective attempts, in a week-long course.
We confidently find that they learn at a *highly predictable* personal and individual rate of 50% reduction in *their injection* of defects, each time they are measured by an SQC review process. We tell them in advance, after the first day's QC of their specification. Here are some details.

Case 10.2 B. Our Training experience.

We can predict how many defect people will find, and will NOT find.

Instructor: *OK you have about 100 defects per page. The agreed exit level is 10 (which is actually pretty bad, but it is 10x better than 100).*

Your classmates found about 35 major defects/page today, the best individual found 17 defects/page, and they all missed about 65 defects/ page (so they are not perfect either), but 100 defects total are *there. And you will all help us prove that, this week.*

We know you do not believe that the defect density is so high. We know that you think you are 'so capable' that you will deliver near zero defects tomorrow. But you will not.

But, do your best, to correct all defects in the entire document, not just the sample we did the QC on. We will measure again tomorrow.

We, your colleagues, will find that you still have about 50 defects/page remaining tomorrow. We, the QC team, being imperfect, will only find about 15 defects/pages altogether. And so on, day after day.

Sooner or later, you will begin to believe that you <u>really do</u> have such a high defect density."

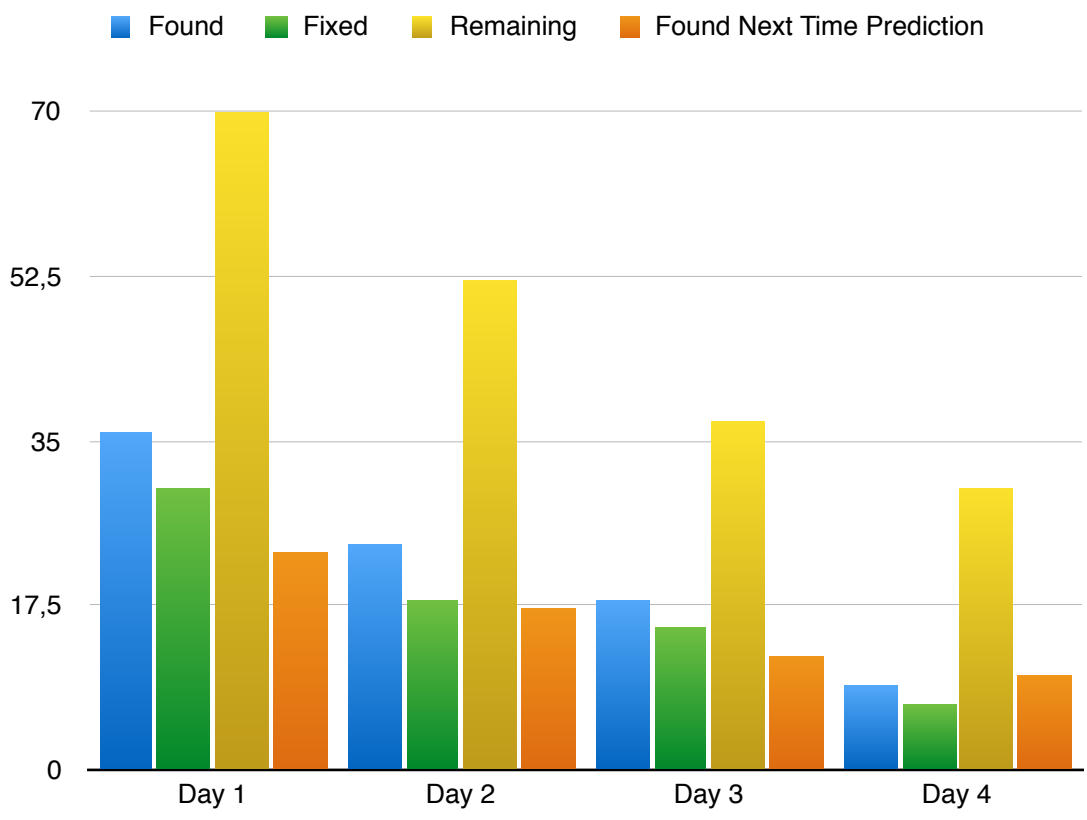


Figure 10.2 B. The daily classroom pattern. The remaining defect density is what we need to estimate in order to determine exit. The remaining density is about 2/3 of what the team found, plus the 1/6 of the found defects that we failed to fix. We can predict that a team will find about 1/3 of that remainder on the next Spec QC. What is found is truly only the tip of the iceberg! But people do not believe that until they see we can predict 'how many they will find' at the next Spec QC correctly. This pattern is related to the personal learning curve (Case 10.2 C below) and the assumption is that because of learning to follow the rules, we do not have to start at a polluted level of defects every time. People learn to do it right. This diagram relates to Case 10.2 B Above and below.

CASE 10.2 B (CONTINUED)

Instructor: "Do not be ashamed of this result. It just means you have the normal '50% at a time' human learning-rate. It just means you, like all others, started, and for years persisted, with some pretty bad planning specification habits. Blame your education and your company training for that. You are the unwilling victim of lack of knowledge. In the old days you would get access to a 'Master' coach, while you were an 'apprentice'.

Now you do not.

The important thing is that you will very quickly, this week, become ten times better, 'forever' ! (as long as you remember what you have learned).

You will 'master the rules', which are a replacement for the advice of the Master Craftsman. And the Master would be standing guard, as the Exit level does, to make sure amateur products did not get out your door.

Hopefully, you will want to convey this practice to your colleagues, and to your organization, when you see the dramatic improvement, in the space of a few days, that it can give every individual.

Maybe you will become a true Master, for someone else?"

Photo 10.2 A. *Kai Gilb in a 2014 Training Class we held together at British Computer Society, London. One of 10 per year, for several years, sponsored by the BCS Specialist Group on Quality, led by Professor Margaret Ross MBE, FBCS.*

They cannot learn much faster, but mere days or weeks are what it takes; about 5 to 7 cycles of feedback, and learning, for the individual.

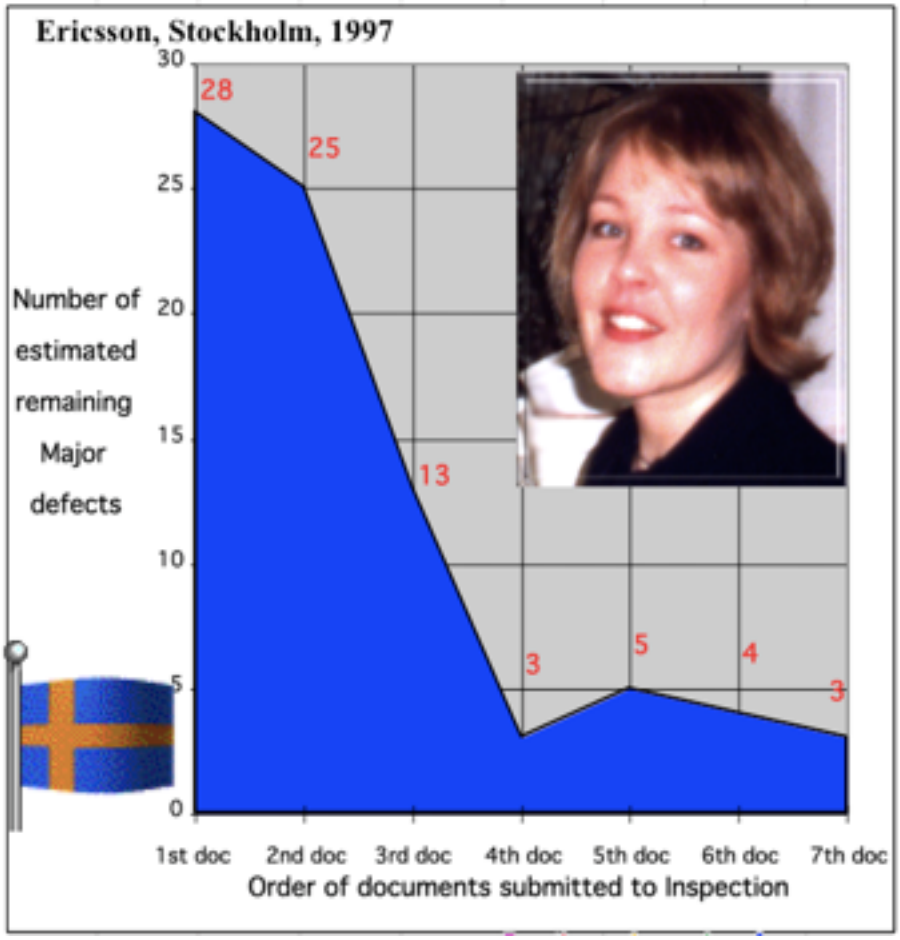
Any large number of defects divided by 2, and then that result divided by 2, repetitively, quickly becomes a very small number. 100, 50, 25, 12, 6, 3, 1.

Then the knowledge of how to *get it right the first time*, sits there, in the long term.

This is called '*defect prevention* ***by individual learning***'. Other kinds of defect prevention remove the root systemic cause of the defect [URL40].

The individual learning is driven by a clearly agreed, management supported, 'Exit' level, from the planning process.

This is much easier than the well-known '10,000 hours' practice to master a sport, or music discipline [Malcolm Gladwell, TED talk]



Case/Diagram 10.2 C. Ericsson, Stockholm.

Case 10.2 C. Ericsson

Marie Lambertsson was championing quality processes, and for fun, she shared with us all, her personal learning curve.

Maybe so that others would not be afraid to admit to high defect levels, to begin with (Swedes are very shy about being imperfect).

She got good fast, and stayed there.

She was saying *"This is what we expect from you all.*

Get more-perfect quickly. Set a goal, Exit level, and learn to achieve it, for the long run.

This is the quality level Ericsson expects of us all."

And now some words from a great successful explorer: Quote 10.2.

"Adventure is just bad planning."

And, advice for complicated exploration expeditions:

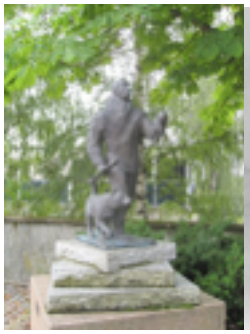
"I may say that this is the greatest factor — the way in which the expedition is equipped — the way in which every difficulty is foreseen, and precautions taken for meeting or avoiding it.

Victory awaits him who has everything in order — luck, people call it.

Defeat is certain for him who has neglected to take the necessary precautions in time; this is called bad luck".

— from 'The South Pole', by Roald Amundsen
Roald Amundsen (1872-1928) He died, in a not-too-well planned (no time available, emergency airplane rescue) attempt, to save an Italian fellow explorer, Nobile, lost in the Arctic. A 'Heroic Adventure' [F12].

Photos 10.2 B Roald Amundsen



Kolbotn Statue.
Tom's 'neighbor'.

'N-25', 1925
Build your own runway.

'Latham 47', 1928 Tromsø
Fateful heroics, no time to plan

Practical tip 10.2: Simple Check

- *Do a simple check, on a single page of planning, counting all ambiguous and unclear words. Ask why there are so many. Ask if they could make your planning useless.*
- *Remember, you personally will probably only find 1/6 of the defects that are really there now!*

Policy 10.2 Numeric Exit Gate from Planning

- **We will have a profitable numeric exit limit to our planning processes, maybe maximum 1 major/page remaining at exit.**
- Why?
  - It pays off about 10 to 1 [14]
  - It is really not smart to allow high levels of defects in planning. Your plan is then worthless.

Version 28 December 2015 17:17 CET, TsG

SECTION 10.3. Prevent Planning Viruses Going Downstream Contagiously

Principle 10.3. No G.I.G.O.

Massive rule violation
(over 10 to 300+ per page)
is common if not checked,
and must be declared intolerable;
bad work products cannot be handed over as valid.

No Garbage In, and thus No Garbage Out (No GIGO)

We have already discussed above, these QC *related* subjects:

1. the idea of having *clear rules* to set the **standard** for writing plans (Section 10.1)

2. the idea of having a maximum allowed quality defects (defects/page) before allowing release of a plan: a process **Exit Level** (Section 10.2)

The next, related, Quality Control idea is how to get some understanding of the **calculable consequences** of allowing defects past a planning process.

In simple terms we can say that:

A THIRD OF THREATS - BECOME ATTACKS.

Maybe only one third of your Major defects ('threats'), will *actually* cause a real problem. The other 2/3 will be okay, because an expert reads the plan, or asks an expert before acting, or some other good fortune, 'if you are feeling lucky today'.

- The problem is that we do not know in advance, *which third* is going to hurt us. So, we have to *act* as though *all of them* might hurt us.
- Of course *if* we get to zero defects (in your dreams) it does not matter. But if we only have 3 defects, then one might ruin our whole project.

"One single nuclear bomb, could ruin your whole day"

Quote: 10.3 A. Bumper sticker seen by the author in Putney London, British understatement at its' best.

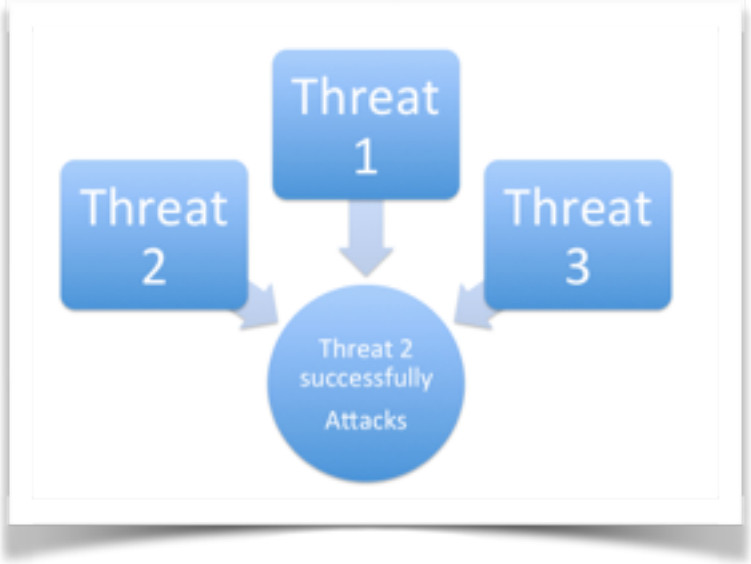


Diagram 10.3 A. Of 3 threats, such as Major defects in a plan, maybe 1 threat on average, will in fact cause downstream damage to the post-planning parts of your project.

Downstream costs can kill your project
For every major planning defect that triggers a successful attack, downstream (after planning) the *cost* to mitigate it downstream will be roughly, average 'ten times', maybe far more in a given case, the cost of dealing with it *during* the planning stage (upstream).

We are talking about 'threat mitigation costs' (stage 2. in diagram below) before the threat becomes an attack.

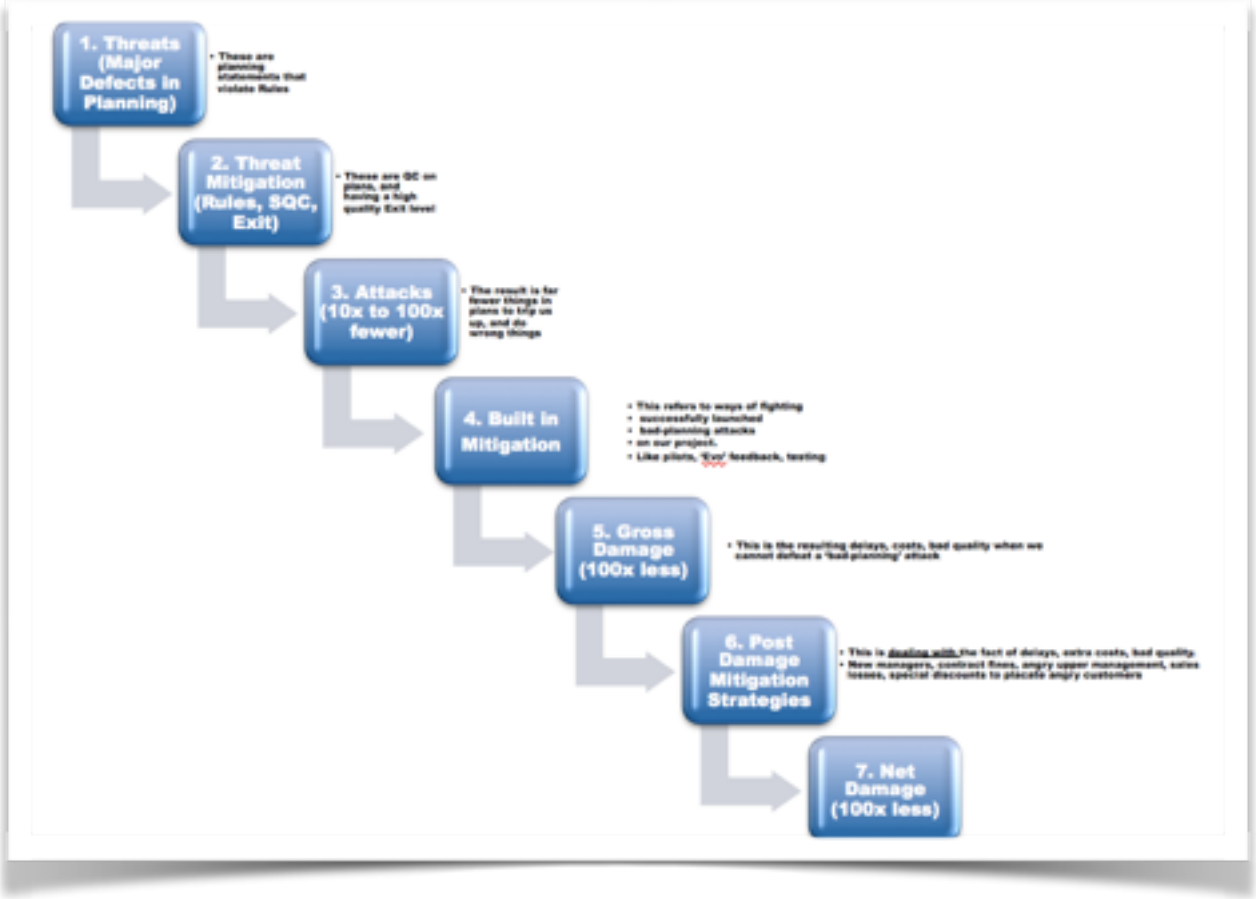


Diagram 10.3 B. an overview of the nature of the problem of 'bad planning'.

Comment on Diagram 10.3 B. above.

The Threat Cost Stream

1. Our 'sloppy' planning language, sets traps, **threats**, for downstream plan readers to misunderstand, miss the point, or fail to see vital connections.

2. Using Rules, checked by Spec QC, with an Exit-level gatekeeper, we can reduce the effect of defects by about 10 to 1. **Threat Mitigation**

3. The defects we *fail* to prevent, or to detect and correct, are still 'free' to **attack** the plan readers.

4. Some of those defects might get caught somewhere in the plan use process, by Spec QC (checking specs *derived* from the upstream specs with defects, by reviews, by tests , by pilot applications, by feedback from Evo delivery cycles. **Attack Mitigation**

5. What is not caught, and some of what is caught late (in final tests) causes disproportionate damage, to time schedules, budgets, credibility, qualities. **Gross Damage**

6. Once damage is done, we can plan what to do about it, like compensate injured parties (money or services), pay contract fines, allow insurance to cover damage. **Damage Mitigation**

7. This rest, which cannot be compensated, + the costs of compensation (6.), is the total net damage caused by your bad planning culture. **Net Damage**

Case Study 10.3 Thorn EMI UK, by Trevor Reeve [Source F13]

Thorn EMI at the time, was 1,500 staff, working in defence electronics systems markets, near Gatwick Airport.

Thorn EMI is an interesting case study for two main reasons:

1. They used Spec QC (Inspection of Specifications) in the entire factory for all kinds of industrial planning, not merely software projects.

2. They took the trouble to analyze the downstream alternative costs for 2,000 defects found in Spec QC process: which is our main concern here. Boeing was pretty good at *that* too!

Trevor Reeve was responsible for the implementation, and Tom Gilb was initial trainer and consultant.
They reported costs of "£50,000 for the first 2 years of start up" (finding and fixing plan defects) of SQC.

They reported "savings of £500,000 per year" (by avoiding rework costs), and in addition "Quality increased, and development time has been reduced significantly." [14, page 314].

We see here a frequently reported 10 to 1 return on investment (20:1 in this case at least).

After 2,000 major defects had been found in Spec QC reviews, there were some voices who said, 'What is the point? We will catch these defects *later* anyway.'

The CEO (in 'British', the MD, Managing Director) ordered an analysis of all 2,000 defects, asking the downstream potential defect-recipient staff, to estimate the time they would need normally, to deal with each particular defect.

The estimated time lost varied, but the median was 9.3 hours to deal with escaped defects, lost downstream (Chart 10.3]. Preventable losses.

The average cost to find and fix them upstream, using SQC,was about 1 hour each.

The CEO then sent a letter to the staff, which Trevor showed me, basically saying. 'Shut up. Get on with it. This is a very clear proven winner.'

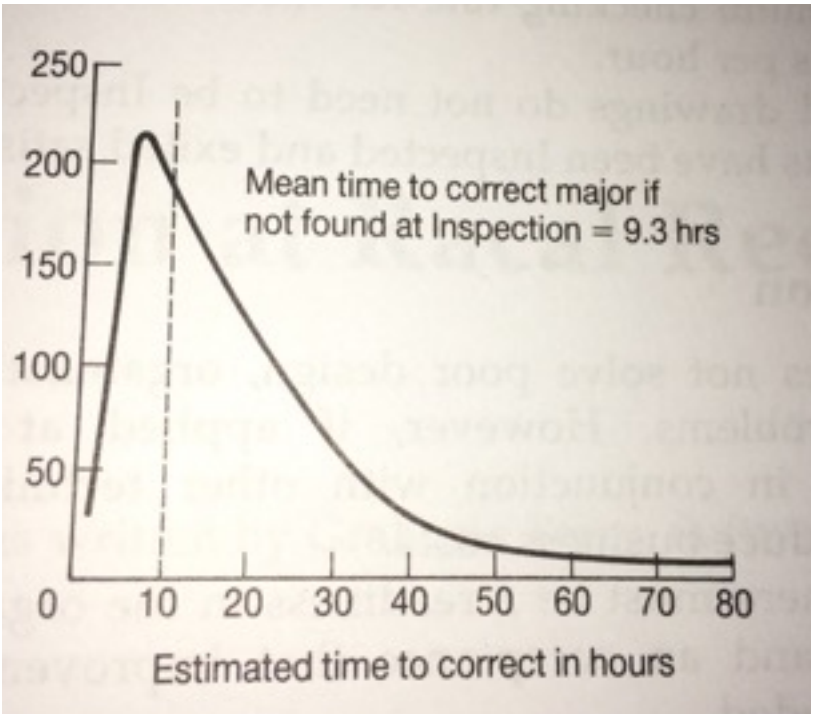


Figure 10.3 C The vertical axis, 0 to 250 is the number of the 2,000 studied defects that had a certain time to correct. The mean downstream loss (alternative to being caught early) from 2,000 Major Defects at Thorn EMI was 9.3 hours. The mean cost to mitigate these threats in specs, using SQC) was 1 hour. 'Estimated time' means that knowledgeable people were asked, for each defect, 'how much time it would have cost them, according to their experience, if the defect had slipped through?'. The payback is about 9.3 to 1, for finding defects earlier than test. Source [F13].

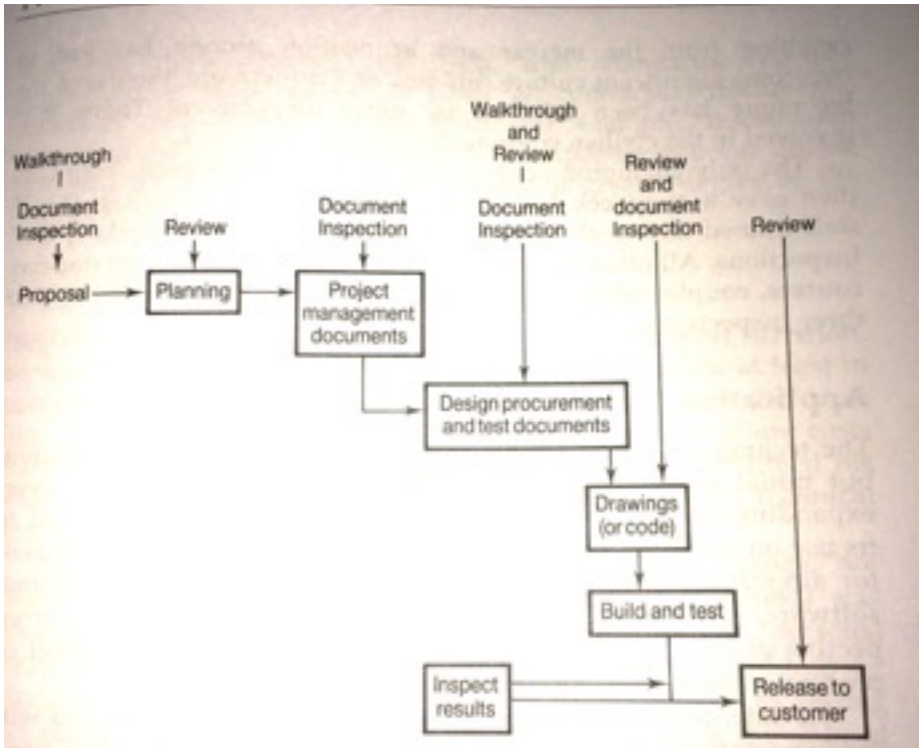


Figure 10.3 D. The Thorn EMI QC Process. We can see the comprehensive variety of reviews and SQC ('Inspections') as well as the many planning document types. Source [F13].

Quote 10.3 B.

"The hardest part of the software task is arriving at a complete and consistent specification, and much of the essence of building a program is in fact the debugging of the specification."

- *Frederick P. Brooks* ***, Jr. (1931-) 'No silver bullet', 1987.***

Practical Tip 10.3. Do something simple, rather than nothing

- *Use at least these minimum rules for all planning: **'Clear and unambiguous'.***
- *Have at least a minimum exit standard: like 'no more than 10 Majors/300 words'.*

Policy 10.3 KILL OFF BAD-PLANNING THREATS IMMEDIATELY

- **We will make ourselves aware of the cost differential between our upstream costs of bad-planning mitigation, and our downstream costs for escaped defects.**
  - Like the 9.3 to 1 ratio in the Case above. Figure 10.3 C
- Why?
  - We will use this cost information to decide *where* to mitigate bad planning defects.
  - The answer of course is usually, to fix things 'upstream'.
  - 'Fix' does not mean 'correct the defects'. It means fix the faulty process producing the defects. Prevent them!
  - But, it can be useful, as in the Case above, know *why* this policy, and to be able to defend your policy decision.

Version 28 December 2015 17:52 CET, TsG

Section 10.4 Keep Quality Control Simple and Inexpensive.

Note: you might like to skip this Section. It mainly explains the historical background of how the Spec QC process in Section 10.5 evolved.

Principle 10.4 AVOID LARGE-SCALE REVIEWS AND INSPECTIONS

Keep
plan-quality measurement
simple, cheap, motivated, and direct.

Planning Quality Control should not use large-scale reviews and Inspections, *after* the planning documents are fully written.

We should measure small increments of planning against a standard, early and frequently; we should use these to motivate planners to keep to a high planning standard.

More important, we should look at the improved system's measurable values-delivered in small increments, to know that the plans are 'working'.

Judge plans on the actual value-delivery results.

Planning 'Inspections' (what we have redefined and re-titled 'Specification Quality Control', SQC) are something we did a lot of, after we learned about IBM 'Inspections' from Mike Fagan, IBM, in early 1970's.

We finally wrote a book about it [14, 1993] 20 years later. 'Software Inspection', with Dorothy Graham.

'Inspections' worked very well for industrial clients, such as IBM, HP, Ericsson, and, Boeing, etc. as documented in this book, and its references.

But we gradually realized that there was an even smarter way to review plans.

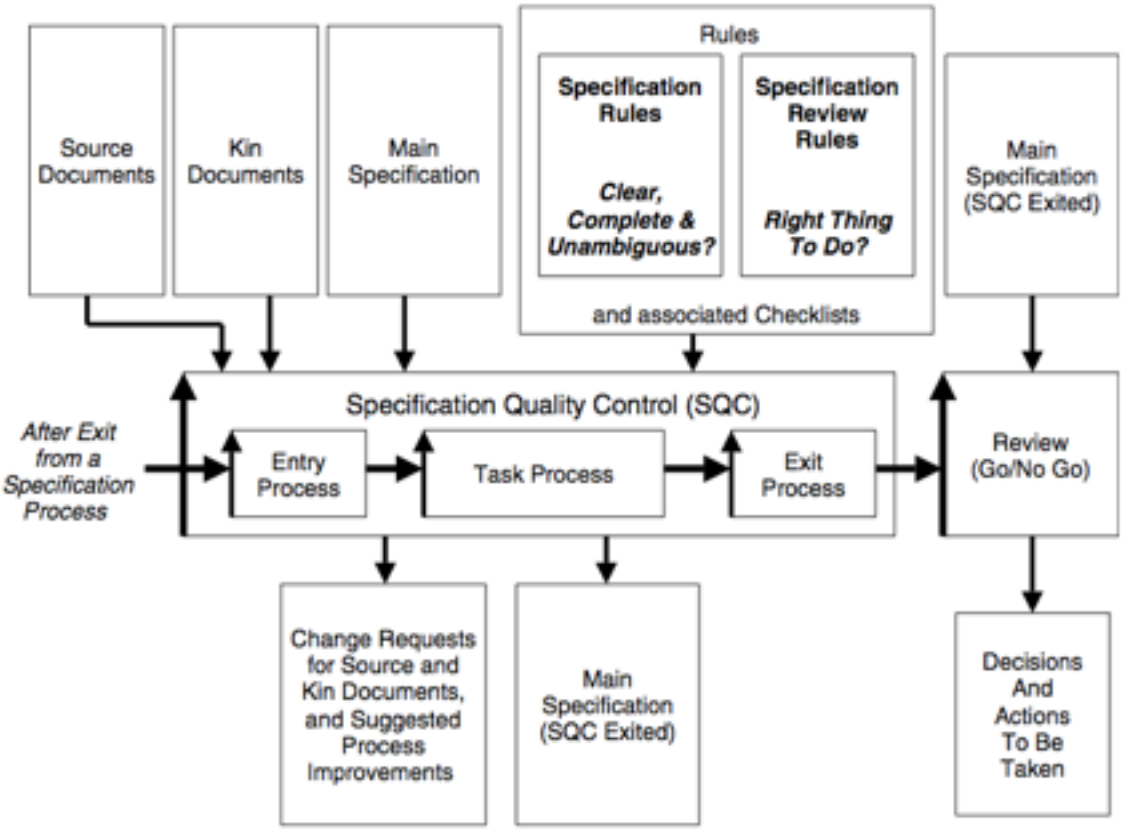


*Figure 10.4 A. Full blast Spec QC, based on the older Inspection method. This is optimized to maximize the **finding** of all interesting types of defects, both clarity and content [URL23], using relations to other planning documents and standards. It is not limited to a simple **measure** of defects. Source [1 D]*

The simplification of 'Document Inspection' started when we were 'forced' to take small samples of very large random and representative plans. For example 3 of 40,000 pages for example at Ericsson A-Sim [Case 2.10 A], and 2 of 82 pages at GE Aero Engines [URL91].

If we found too many Major Defects per page (100-300 defects/Page), and concluded that all other pages were just as bad; then we saw no point in doing rigorous Inspections on all other pages.

Previous methods of 'Inspection' assumed that we would examine all pages in a document, even if there were 40,000 pages. The main idea was to clean them all up; not to merely measure the defect density.

The cost of doing this clan up properly, was about four (±2) qualified people using one hour per page, just to read and check. Then these expensive people got involved again, using more time for reporting what they found at meetings.

The net result of such a huge cost would still be that we found only *a third* of the defects, and even when correcting that third, we would still let *two thirds* of the defects (the unfound) *remain*.

There was hardly any point, even though we would make savings for the one third we found, by correcting them early, and ten times cheaper. We still had the bigger problem of the 2/3 we did not detect.

It became obvious that, for non trivial defect densities (10 and up per page) it would be even cheaper to 'burn' the specifications, and write them all over again, but next time using people who could write to the standard expected in the Rules.

We drew some conclusions from this experience:

1. If we found almost no defects from the representative *sample*, we could Exit the *whole* plan (normal industrial job-lot sampling practice) to the next processes.

2. If we found many defects, we should 'burn' (discard) the plan, and start over (drastic, but still cheaper, faster, better) *properly*.

3. We had to institute, for large planning efforts (over 100 pages), the idea of *sampling* the planning work at very *early* stages, like after 10 or 100 pages, at least, after a week's work.

4. If these early samples had unacceptable major defect densities (10 or more/page), then we could not continue producing more 'garbage', without getting the planners to go back, and learn to follow the standards, and get successful Exit, in small batches.

5. High accuracy in measuring defect density was not critical, as long as we were near-the-truth consistently, and could decide Exit-or-not clearly.

The influence of short value delivery cycles on plan 'Inspections'.
At the same time, our clients (like HP, Boeing, Ericsson, Nokia, Confirmit, Citigroup) were moving, on our advice, towards Evolutionary project models. Weekly cycles of delivery of value, instead of several year 'big bang', 'waterfall' projects.

The **rapid** (weekly) and realistic (real value, not just plans) **feedback**, **means** we can find out how well plans are working **without** any *other* form of QC at all.

The realistic weekly feedback is better, cheaper, faster.

Confirmit (confirmit.com, Norway), in 2003 drew this conclusion immediately. *'We don't need your Inspection, because we have the weekly measurable feedback you taught us.'*

We were rapidly moving in the direction that this book reflects: very small quantities of planning up front, and gradual detailing of the plans, on the go, value cycle by value cycle. The 'big volume plan', for us, has disappeared.

One of the other problems was that 'Document Inspection' as we knew it, was totally focused on **finding and removing** defects. We call that the 'cleanup mode'.

But 'Document Inspection' was very bad, almost useless at **finding and removing** defects.

One third of the defects detected, was the *best* it could do.

We needed to get to better than 99% *reduction* of 'normal' defects injected.

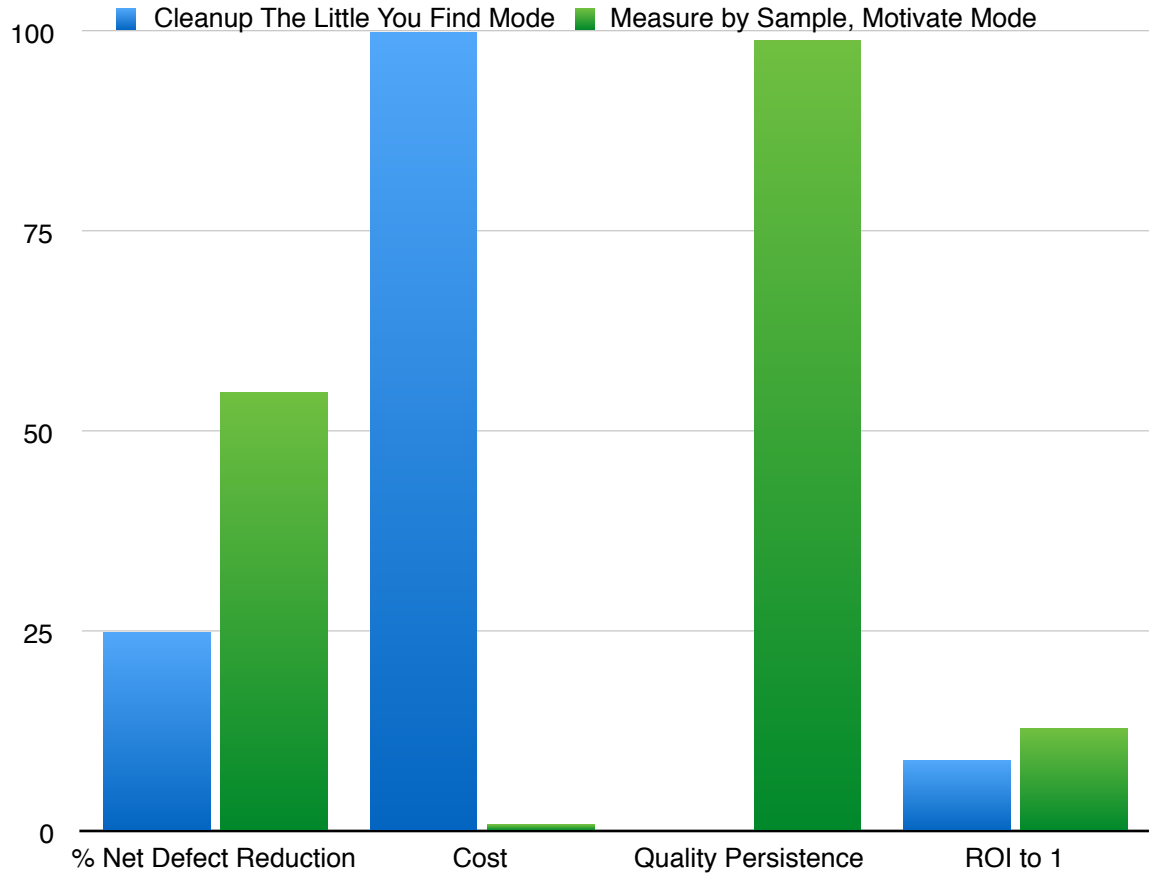


Figure 10.4 B. Some rough differences between Old style 'Inspection' (Cleanup Mode) in Blue , and Spec QC in Sampling (Section 10.5) mode in Green.

The % defect reduction effect is much better with SQC because it motivates planner to avoid defect insertion, and drives people towards a low Exit level.

The cost of SQC is 100X lower, than traditional Inspections, mainly due to sampling size. 1% rather than 100%.

SQC makes the quality levels persist because people learn how to do it right all the time.

The Return on Investment is about the same but better for SQC.

The ratios are just rough approximations based on the author's experience and observations. The real results will vary a great deal depending on many factors, such as the rigor of the process, the Exit levels set, the checking rates etc. One pretty good source on Inspection data is 'Software Inspection' [14]. The purpose of this diagram is to graphically dramatize some of the expected differences. More explanation of SQC below.

We found that we could make massive defect injection reduction happen with cyclical learning ('Evo', URL4), and by 'improving the working environment'.

For example, we could improve the work environment in many areas by using the Defect Prevention Process (DPP, which came out of IBM 10 years *after* Inspection) [URL40].

We began to notice, first at Boeing (about 1988-1990, Aircraft Engineering, URL44) that if the Exit level was taken seriously, then planners had large, persistent reductions (100 to 1) in personal defects they injected. They achieved this reduction by personally learning to apply the Rules. They got 50% better at applying the Rules for each experience of failing to Exit.

This was very much more effective than *fixing* a mere one-third of injected defects.

And, it was sufficient, for our purposes (Exit level control), to quality-control only small samples, to generate that learning behavior (50% defect injection reduction), and to get 100x better results.

So, for all these reasons, we and our clients, turned away from the 'big bang' Inspections, and focused on the simple Specification Quality Control, which does samples of a larger work, or can do *all* pages of a very small short plan.

Around 2003, Vince Chandler, a Director at Citigroup in London, and a big fan of our Inspection method for many years, came to Kai and me and said:
"One thing I have learned about this big complex bank, is that big complex work processes do not work very well. Give me a much simpler Inspection method".

We formalized, that day, for the bank, what we had been doing experimentally and accidentally for many years (the Defect Measurement Process, see Section 10.5).

It is so essentially different from its 1970s Inspection ancestor in nature, that we used the opportunity to change the name (which had its problems in any case, since it could be confused with final product testing Inspection, as in aircraft).

We called it the Defect Detection Process (DDP), and combined it with the Defect Prevention Process, to get the Specification Quality Control (Spec QC) [URL23].

And that is how it appeared in our 2005 book [1 D].

The variant of SQC which did 'sampling' we called Simple SQC, since a 1% sample could measure pretty well, using extrapolation, the *true density* of major defects, even when we could not find them all – *because we could assume there were 3x as many as we found*.

It was also 100x cheaper than a 100% sample, to take a 1% sample.

What we had done with SQC was to:

- Shift focus from
  '100% checking, and cleanup the little we found (33%) early'

  towards

  Measure, well enough, to
  1. Determine if Exit could be met
  2. Motivate planners to learn, and follow, the rules
  3. Include Defect Prevention Process (DPP).

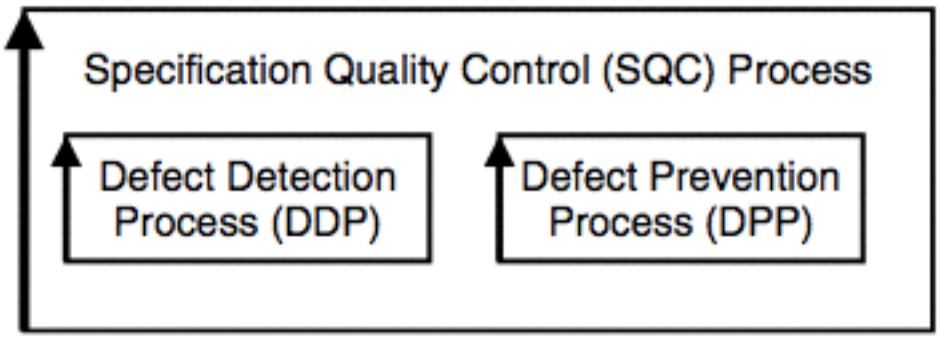


Diagram 10.4 C. Spec QC combines the Defect Detection Process (find defects, either to measure by sampling (perhaps better-named Defect Measurement Process, DMP), or with a view to removing them (DDP)) and the Defect Prevention Process (analyzing real defects at grass roots, with a view to changing the work environment to reduce the flow of defects, caused by a bad process).

In very rough popular terms, we had found a specification quality control variation that was:

1. 100 times more effective at resulting in few defects in plans, at
2. 100 times less cost.

Not too bad. But it took us decades of experiences to figure out something 'so simple'.

That is why we are writing this Section. So you do not have to learn the hard way.

What we did NOT attempt to do with *this* (SQC), was:

To improve 'Planning Quality ***Assurance***'.

- **Quality Assurance** means making sure your planning process *itself* has quality, that your plans *themselves* have quality, and that the *products* of your plans have high competitive qualities**. The subject of this book.**

- **Quality Control:** in a planning context means simply measuring the defect level of a plan. QC is more limited in nature; sort of like measuring speed limits of cars: giving some of them traffic fines; not the same thing as traffic 'safety assurance'.

We have already given several cases on use of SQC in this book, and in a following sub-chapter [10.5] we will dive into the process itself in more detail].

The purpose of this sub-chapter is to give the reader some background on how we *evolved*, SQC, and what it is, and what it is not.

Quote 10.4. Dr. Deming summarizes these ideas nicely

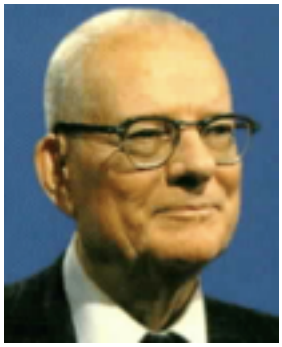
"Point 3. Cease dependence on inspection to achieve quality.

Eliminate the need for inspection on a mass basis by building quality into the product in the first place."

"Rather than use mass inspection of every output product, the output can be statistically sampled in a cause-effect relationship through the process."

Dr. W. Edwards Deming (1900-1993)

This Wikipedia interpretation of Deming, above, is helpful in bringing out the 'sampling' idea of this sub-chapter. He wrote a book on Sampling (Some Notes on a Theory of Sampling). Deming was big on sampling. [F14, F5]. Sampling was our breakthrough idea. IBM's michael Fagan never went there. But his co-inventor of Document Inspection, Ron Radice spoke positively of our sampling idea in his Inspections book [22]. The 'Inspection' Deming refers to in the quote is a different kind of inspection from our document inspection (aka SQC here). He refers to final QC and Testing of final assembled hardware products. Like cars off an assembly line.



Dr. W. Edwards Deming

Practical Tip 10.4: Sample at least.

Sample important plans such as contracts, request for proposal, management vision statements, and ask if they are clear enough for purpose, and if not, why not?

Policy 10.4 Smart Planning QC

1. We will Avoid large, costly, bureaucratic reviews and inspections of planning work.

2. We will measure (using SQC) small samples of large plans, to decide Planning Process Exit, or not.

3. We will measure planning work flows *early*, either by SQC or by measuring the planned value, as delivered Evolutionarily.

- Why?
  - To reduce overhead costs of planning and projects
  - To get high planning quality by motivation to do it right, rather than cleaning up bad planning badly

Version 31 December 2015, 01:22 CET, TsG

Section 10.5 Simple Quantified Plan Quality-Control Process: 'Spec QC'.

Principle 10.5 Exit Focus: No Garbage Out

**Focus on getting a credible estimate of
major defects per page,
quickly,
so you can decide to
Exit a plan to the next process, or not exit.**

Citibank, in 2003, wanted a very simplified process, for quality control. Something **much** simpler than the Gilb/Graham Inspection Process [14].

So, Kai and I formalized the 'Spec QC process' based on our earlier experience. We tried to get down to the basics, focus on prevention, and get rid of the fat. 'Lean' thinking, of course. See history of this in Section 10.4.

Our Spec-QC-process design-objective was:

- to be able to estimate rough Major Defect Density per Page, as quickly as possible, with as little use of costly people, as possible.

The purpose of the defect-level estimation was to determine 'Exit or not', depending on the Maximum Majors/Page, as allowed by any given local Exit-level standard.

As we showed earlier, at Citigroup London, they moved from 80.4 Majors/page to achieving Exit levels of 10 Majors per page (our short term target) six months later, using our methods.[Case 10.1].

Here is a simple summary of the larger process: **Defect Estimation Process (DEP)**:

DEP = Spec QC + Estimations + Exit Decision

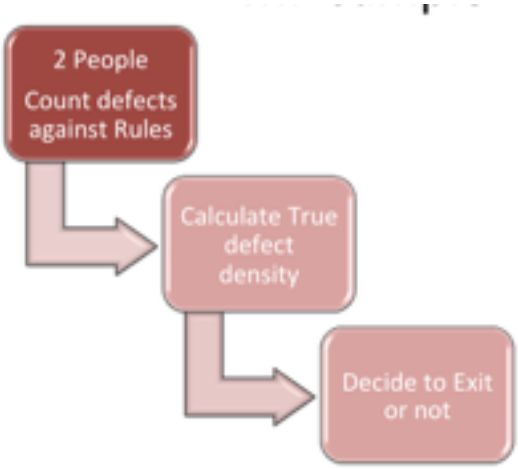


Diagram 10.5 A. The Defect Estimation Process (DEP) Simplified. Notice that there is no attempt to list defects found, in detail, or to 'clean them up' in this process. The focus is on answering a simple question "would it be economic to release the planning document now?" . The second two sub-processes,above, can be totally automated on a spreadsheet or other automated tool.

Here is the Defect Estimation Process, including the SQC sub-process in *detail*:

Entry Conditions:
1. A group of two, or more, suitable people, peers, are needed to do it.
2. You need to schedule 30 to 60 minutes. Irrespective of plan size.
3. You should have a trained (1/2 day) SQC team leader at the meeting to manage the process.
(or if that fails, follow this Process 10.5 and the Template 10.5 instructions, below.

Defect Estimation Procedure:

P1: Identify Checkers: Two people, maybe more, should be identified to carry out the checking. One should be the Owner or Author who drafted or updated the plans to be checked.

P2: Select Rules: The team identifies (or acknowledges a standard) at least three Rules to use for checking the specification. (My favorite Rules are clarity ('clear enough to test'), unambiguous ('to the intended readership'), and completeness ('compared to sources'). For 'requirements', I also use the Rule of 'no design'. Any appropriate set of rules can be used.

P3: Choose Sample(s): The team then selects sample(s) of about one 'page' in length (300 non-commentary words). Choosing a page at random can add credibility – so long as it is reasonably representative of the overall content. The team should decide whether all the checkers should use the same sample, or whether different samples for some people are more appropriate or useful. If 300-600 words or less do it all.

P4: Instruct Checkers: The team leader briefly instructs the checker(s) about how to use or interpret the Rules, the time to be used checking (10 to 30 minutes), and how to document, list, count - any perceived defects, and how to determine if they are major defects (majors).

P5: Check Sample: The checkers use between 10 and 30 minutes to check their sample against the selected rules. Each checker should 'mark up' their copy of the document as they check (underlining suspected defects, and classifying them as 'major' or not). At the end of checking, each checker should count the number of 'possible majors' they have found in their checking scope.

P6: Report Results: The checkers each report to the team leader their personal number of 'possible majors.' The team leader leads a discussion, if necessary, to determine how many of the 'possible defects' are actually likely to be major defects.

P7: Analyze Results: The team leader (using the Template or Spreadsheet tool) extrapolates from the *reported* defect findings, the estimated **total** number of majors in a single page. This will be about 6 times the 'most majors' found by a *single* person. Or alternatively the total majors/page is 3 times the 'unique majors' found by a 2 to 4 person team. This estimates the major defects/page density. If you are using more than one sample, you can average the densities found by the set, of the different pages. The team leader then multiplies this average major defects/page density by the total number of pages to get the *total number of major defects in the specification* (for dramatic effect!).

P8: Decide Action: If the number of majors/page found is a large one (ten majors or more), then there is little point in the group doing anything, except determining how they are going to get someone (Author, Owner) to re-write the specification properly. There is no economic point in looking at the other pages to find 'all the defects', or correcting the **majors already found. There are too many majors *not* found.**

P9: Suggest Cause: Choose any major defect just found, and think for a minute, why it happened. Then give a short sentence, or better still a few words, to capture your verdict on 'why?'; root cause. Document this, and seek practical action, or experiment, yourselves or others, to see if the insight can reduce defect insertion frequency in general.

Exit [From this QC Process]
1. if less than 5 majors/page (or other agreed Exit level standard), extrapolated total defect density was determined, you can Exit the plan to the next processes.

2. You should note the release level in the plan (Quality Level: < 5 Majors/Page)

3. Otherwise, refuse Exit. Expect an upgraded plan to be resubmitted for Spec QC.

Process 10.5. P9 is a simplified form of, and beginning of , the Defect Prevention Process. Process Version 301215 TsG.

Simplified Specification Quality Control Form

SQC Date: **May 29, 200X**. SQC Start Time: _____
SQC Leader: **Tom**.
Author: **Tino**.
Other Checkers: **Artur**.

Specification Reference: **Test Plan**.
Specification Date and/or Version: **V 2** Total Physical Pages: **10**.

Sample Reference within Specification: **Page 3**.
Sample Size (Non commentary words): **approx. 300**.

Rules used for Checking: **Generic Rules, Test Plan Rules.**
Planned Exit Level (Majors/logical page): _____ or less.

Checking Time Planned (Minutes): **30**. Actual: **25**.
Checking Rate Planned (Non commentary pages/hour): **2**.
(Note this rate should be less than 2 logical pages/hour)

Actual Checking Rate (Non commentary words/minute): _____
Number of Defects Identified by each Checker:
Majors: **6, 8, 3**. Total Majors Identified in Sample: **17**.
Minors: **10, 15, 30**.

Estimated Unique Majors Found by Team: **16 ± 5**.
(Note 2 × highest number of Majors found by an individual checker)

Estimated Average Majors/Logical Page: **~16 × 3 = 48**.
(A Logical Page = 300 Non commentary words)
Majors in Relation to Exit Level: **48/1 (47 too many)**.
Estimated Total Majors in entire Specification: **48 × 10 = 480**.
Recommendation for Specification *(Exit/Rework/Rewrite)*: **No exit, redo and resubmit.**

Suggested Causes (of defect level): **Author not familiar with rules**.

Actions suggested to mitigate Causes: **Author studies rules, All authors given training in rules**.

Person responsible for Action: **Project Manager**.
SQC End Time: **18:08**. Total Time taken for SQC: _____

Version: August 15, 2004. Owner: Tom@Gilb.com

Template 10.5 [Source CE book, 1 D] An example of the SQC process, with collected data. Notice the input to the Defect Prevention Process (DPP) at the bottom. The rest of the process is estimating the defect density.

Between the process above, and the Quality Control Form above, you have enough detailed information to try this out, don't you?

Case Study 10.5 Early Intel Experience, via Erik Simmons

Case Study Environment: Sampling

- "Large e-business application with 8 requirements authors, varying experience and skill.
- Each sent the first 8-10 requirements of estimated 100 requirements per author (table format, about 2 requirements per page including all data)
- Initial reviews completed within a few hours of submission
- Authors integrated the suggestions and corrections, then continued to work
- Some authors chose additional reviews; others did not
- Inspection performed on integrated document to assess final quality level". [Erik Simmons, Intel]
- *I will skip 2 of the cases for brevity, and give 1 below. TsG*

Case 3: "Average major defects *per requirement* in initial review: **8.**

Average major defects *per requirement* in completed document: **3.**

Time investment:
12 hours in initial review (1.5 hours per author); about 8 hours in additional reviews

6 hours in final inspection (2 hours, 2 checkers, plus prep and debrief)

Major defects prevented: 5 per requirement in ~750 total requirements

One interesting sidelight:
When the program manager was informed of the quantified quality level of the requirements (Estimated at 5 majors per requirement, assuming 60% effectiveness), panic set in until the context could be explained, and a pathway towards greater levels of quality determined.

This reaction is probably typical of individuals who find out what the true quality level is in a document.

In this case, 5 majors per requirement (or even the roughly 16 original per requirement) was not that bad compared to many reports of first experiences from Tom Gilb, whose clients often find 25-50 majors per page. Some requirements inspections at Intel have generated similar defect levels." [Erik Simmons, Intel], see a later Intel Case [URL22]

Erik's hobby is organ music. It's on iTunes. He's good! (Music and *Planguage)*

This stuff works!

Quote 10.5 From Erik Simmons, in Foreword to Competitive Engineering book [1]. He is referring to Gilb Planguage methods as experienced at Intel.

Practical Tip 10.5. Buddy Up and Check Up

Get a friend to do SQC on one of your own plans, together with you.

Tough. But practice-changing.

Policy 10.5 Use an Hour to do 'Spec QC'

- **We will use at least an hour to quality-control all critical plans, and other critical documents such as contracts, bids, management presentations etc. which have *plans* as background.**
- **There will be a formal Exit level of acceptable defects.**

- Why ?
  - Because it will motivate planners to do a much better job of writing plans.
  - Because if will easily pay back, in terms of time saved downstream.

Version 30 Dec 2015 01:41 CET, TsG

Section 10.6: 'Quality' and other related fundamental concepts.

Principle 10.6

Quality is a quantifiable 'how well'

**Qualities are 'how well' a system functions;
and they are always
a *quantifiable* planning attribute.**

It is fundamental to value planning that we have a clear idea of the distinction between a 'quality' and other related planning attributes such as functions, resources, work capacity and strategies. This is because 'qualities' are such a large part of what we 'value'.

Other related points, principles if you like, about quality right away, in addition to Principle 10.6:

1. Once we define a quality numerically ('Scale'), we can always find *several ways to measure and test* it ('Meter') in practice, that are sufficiently accurate, and cost-effective for our current purposes.
2. There are always *several different* qualities that we need to include in planning for critical values of a system. Quality is never a single dimension in a real system.
3. Qualities make up a large part of what we call 'values' or 'stakeholder values'; about 7 out of 10.
4. If you do not bother to define a critical quality using a Scale of measure, you will lose control of that value delivery; your project will fail.
5. Serious, critical, high (competitive) qualities need to be 'engineered' (a numeric logical process of design, URL90:
   The Logic of Design) into a system; you cannot test, inspect, or review them into a system.
6. Quality levels cost you scarce resources (in both short term, and the entire life cycle), and perfect quality levels cost *infinite* resources. Even *you* cannot afford perfection.
7. So you need to be careful about asking for too-high a quality level, it may not pay off.

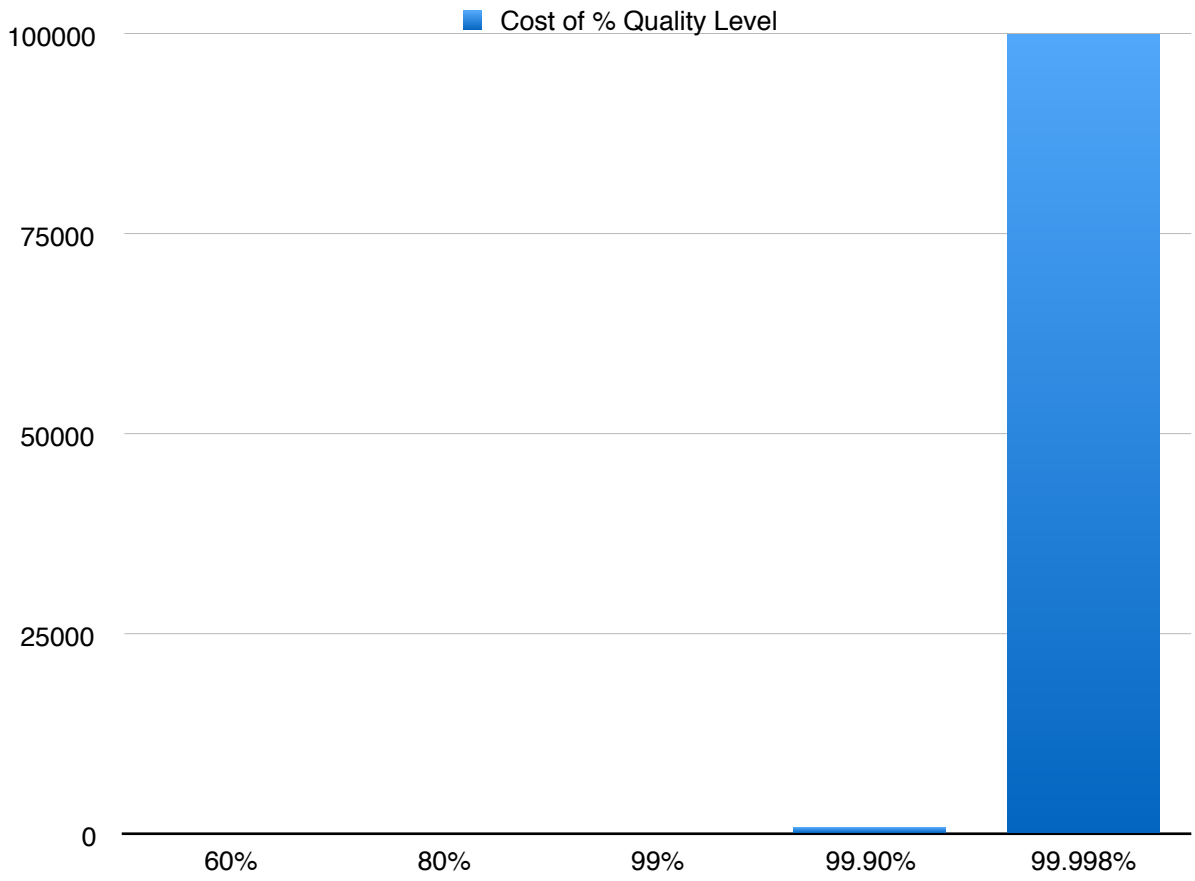


Chart 10.6. ***The impossible cost of perfection****. As quality levels you plan for, increase towards perfection, the expected cost to deliver and maintain that system level, will increase towards infinity; the 'cost of perfection'. Your job as a planner is to plan for a cost-effective level. Any fool can demand perfection. Not even a fool will ever get perfection. But in some systems it might be cost effective to get 'damned close to perfection', for a long time, like 99.998% availability (the highest known level for telephone switching systems, in 2015) for a whole year. The costs are symbolic approximations here, but based on real published cost increments for ATT systems.*

Recall this **Lord Kelvin** quotation from 1.6 (more detail there). Quote 10.6 A.

> *"In physical science*
> *the first **essential step** in the direction of **learning any subject***
> *is to find **principles of numerical reckoning***
>
> *and **practicable methods** for **measuring some quality** connected with it."*

Lord Kelvin

Notice, he said '*some* quality', implying there are several quality attributes involved. And notice the quality is 'numeric'.

We have used the word 'quality' often in this book, to now, but mostly as an adjective (like 'quality control'). I would now like to turn the reader's attention to the *noun* 'quality'. And I would like to try to go deeper into the 'quality' concept than you have probably ever experienced. Though it could also be argued that this entire book does that too

I feel this is central to planning because I believe that 'qualities' are a subject very badly taught and practiced, in planning. With great failure and delay as a direct result.

What is our 'bad' practice with 'quality' planning?:

- we do not **define** the **many qualities** we are concerned with, at all (we just say the *naming* words: like 'better security')
- we, even less frequently, **quantify qualities** so that they can be seriously planned for: (like '99.9% chance to detect hacking')
- we don't even operate with a **decent definition** of the *general* notion of 'quality': but we bandy the word around often. ('improved quality')

- Qualities are not the only *values* we need to manage in our 'value objectives'; but my experience is that they are **among the most critical** for most stakeholders (ask stakeholders what they most want to improve; and you get 70% of the answers are 'qualities')
    - (non-quality values can be, for example: *costs, time, work capacity of systems*; and we are pretty good at quantifying *them*)

It does not surprise me any longer, that when you ask people what they mean, when they use the term 'quality', they tend to narrow in on '*how often system fails'* (reliability = MTBF), or for *how long a system 'breaks down'* (availability = % up time).

We can remind people that they do, in fact, in everyday life, consider and discuss the *multiple* qualities of very many things and services: they normally consider a broad spectrum of qualities.

Qualities in a car for example, safety, beauty, ease of driving, ease of maintenance, resale value, acceleration, braking ability, comfort, task flexibility, winter-driving capability.

In software, when the term *quality* is used, it almost always refers to reliability, and sometimes is a narrow definition, 'bug freeness'. That is all there is to the '*quality* assurance' profession (bug testing) in software, sometimes. Far too narrow!

For example, other software system 'qualities' certainly include *security, safety, usability, and maintainability* etc.

But somehow these obvious categories of 'quality' slip people's mind often, or are treated by themselves, in splendid isolation. But they are not each treated separately, and seriously, and holistically as one of many quality aspects to be planned for, simultaneously.

It helps to remind people that we are talking about '-ilities'. But, many qualities do not have a name ending with 'ility'. However, this is a constructive start.

A basic definition of 'quality':

- A 'quality' describes a '**how well** the system functions'.
- (Focus on 'HOW **WELL'**).

Normal dictionaries are not helpful for our purposes. Dictionaries all cite something like 'a **distinctive characteristic** of something'.

The problem being that there are *many* 'distinctive characteristics', which we would *never* really classify as 'qualities.

For example ('distinctive characteristics', but *not* qualities):

1. Functions (**what** it does)
2. Resources, costs, expenses (**how much resources**)
3. Work Capacity (**how fast** it does it, or how **much it** can do)
4. The Designs, technologies, architectures it uses (**how** it does good stuff at a certain cost)

More on all these ideas in a moment, below.

The above 4 are all potential 'distinctive characteristics' of a system.

If you think I am 'splitting hairs' here, I can assure you I have very good reasons for making these distinctions in *planning*, even though the ordinary 'civilian' has no critical need of being precise. Value planners, you, *do* have a need to be more precise.

Here is a summary of the reasoning, to justify these distinctions:

1. 'Functions' are a set of characteristics of systems, which we define, (and so do dictionaries define 'function' this way too)**, as '*what* the system *does'*.**

- A *set of functions* is a basic 'entry requirement' into a world of competitive products, or competitive services,.
  - *If you do not have functions such as 'lend money', 'acquire money', and 'pay or charge interest', you cannot be a bank.*
  - *But you need **better qualities and lower costs** to be a 'competitive' bank.*
- From a planning point of view, we need to initially 'acknowledge' the set of functionality necessary in our planning domain. What is our 'market', our product or service area?
- But that is only a basic 'entry' ticket to the planning process.
  - **Planning is NOT about 'function'.**
  - **Planning is about 'functioning *better'*.**
    - Improved Cost-Performance, but *functions can be unchanged*.
  - Same functions, but better 'cost-benefits'.

- Functionality is 'binary', 0 or 1, present or absent. There are no degrees of improvement of a function itself. You've got it, or you don't. Pay interest, or do not.
- Many people totally *misuse* the term 'function' to mean quite other things than *our* definition ('what it **does')** [URL81 SAVE, 23 Miles].
- But that is dangerous terminology misuse for the planner, as we argue below.
- We need to translate people's sloppy use of terms, into the values they are *actually* referring to, and to plan for *delivering those values, via suitable strategies*.
  - This is a business analysis skill.
  - What, for example, does *'we need better product and service functionality'* really mean?

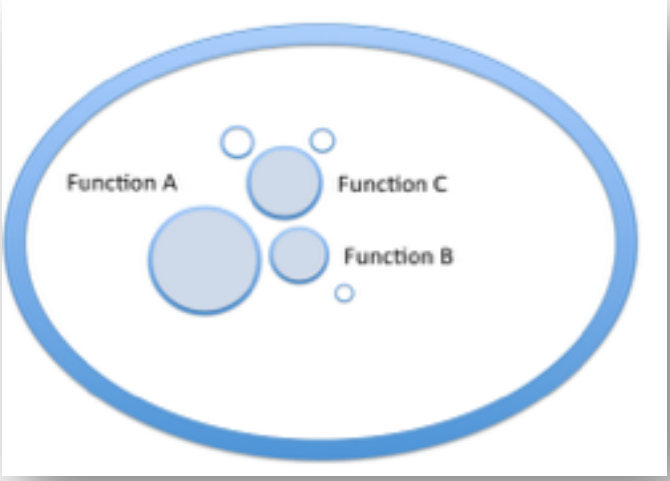


*Figure 10.6 A. In Planguage, we use an oval shape to diagram 'functions', i.e. what the system **does**. Functions can contain nested sub-functions. The highest level of functionality is defined in Planguage as the 'mission' [URL2]. For greater depth on function, see [1, Ch. 3, FUNCTIONS, What systems 'do'].*

2. Resources (costs, expenses, profits, incomes, money, time) is a set of characteristics of systems, which planners usually have a clear quantified relationship to, in projects.

> 'Resources' is already *fundamental* to all forms of planning. Think 'budget', 'deadline', 'skilled employee'.
>
> But there are some other system characteristics quite different from limited resources (though they may *impact* the consumption or generation of resources) that we need to plan for (like: *qualities, work capacity*).

Case Study 10.6 The Balanced Scorecard (BSC[URL62]) at a Bank:

The whole point of the Harvard Business Review 'Balanced Scorecard' was that traditional business planning was too-narrowly focused on money ('financials'), and had lost clear *planning focus* on the stuff that *really* 'makes money' [2] (like 'qualities', ask Apple, Tesla).

Part of the problem was that the non-financials (qualities and other performance characteristics), in BSC, were often *not really* quantified.

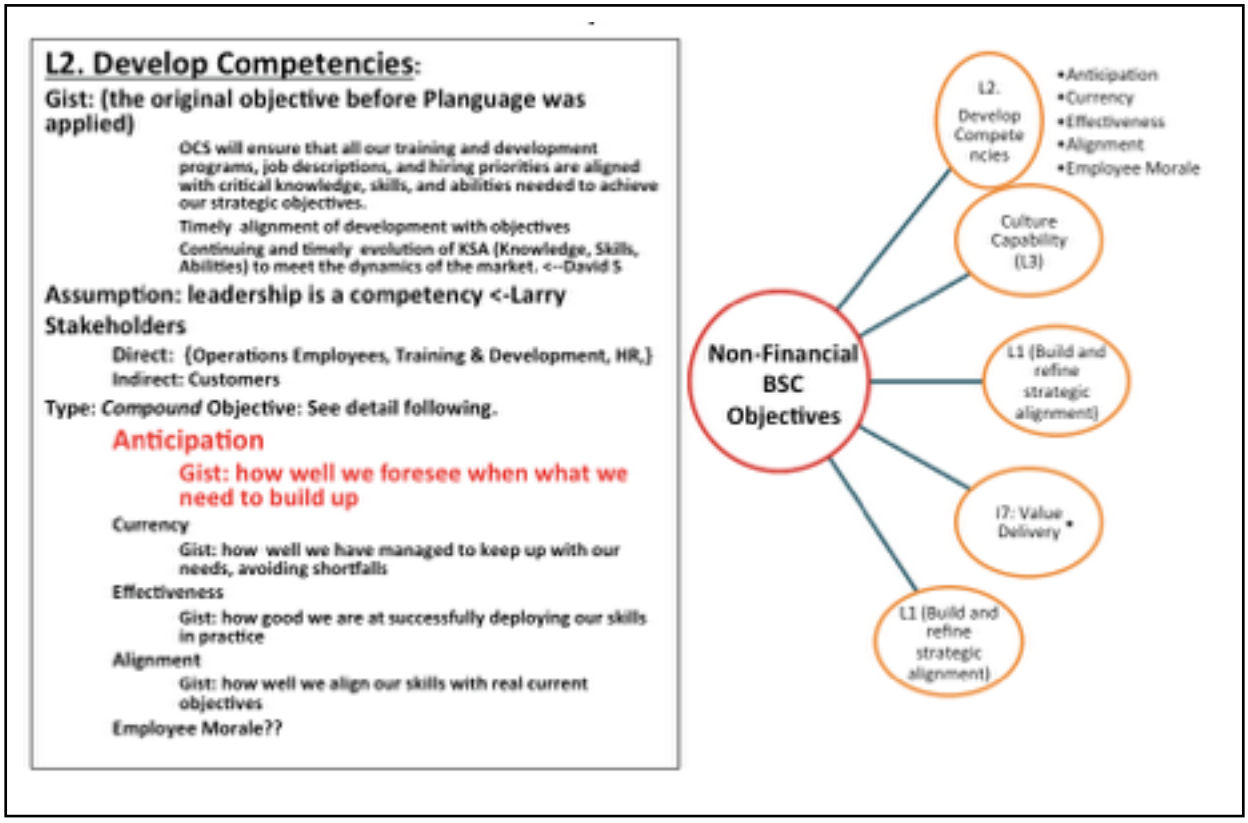
The typical published BSC non-financial's planning statements were just 'words'. Practice varies, measurable goals *were* a BSC stated intent, but there was no business culture to make non-financials numeric (that was the initial problem!), so people ignored the advice to quantify.

Or, worse, outside consultants quantified, and managers ignored that, and felt alienated [URL62A].

We have witnessed large-scale failure, in a very large US bank (a key player in the Financial crisis 2007+), *of Balanced Scorecard planning*, due to the fact that none of the executive planners *could* quantify their organizational qualities; nor were they even aware that that *was* the problem. They were just *very frustrate*d trying to communicate their planning ideas to each other. Like, 'waste of time'.

They, like most managers, and even MBA's simply had no training or knowledge on how to quantify their non-financials.

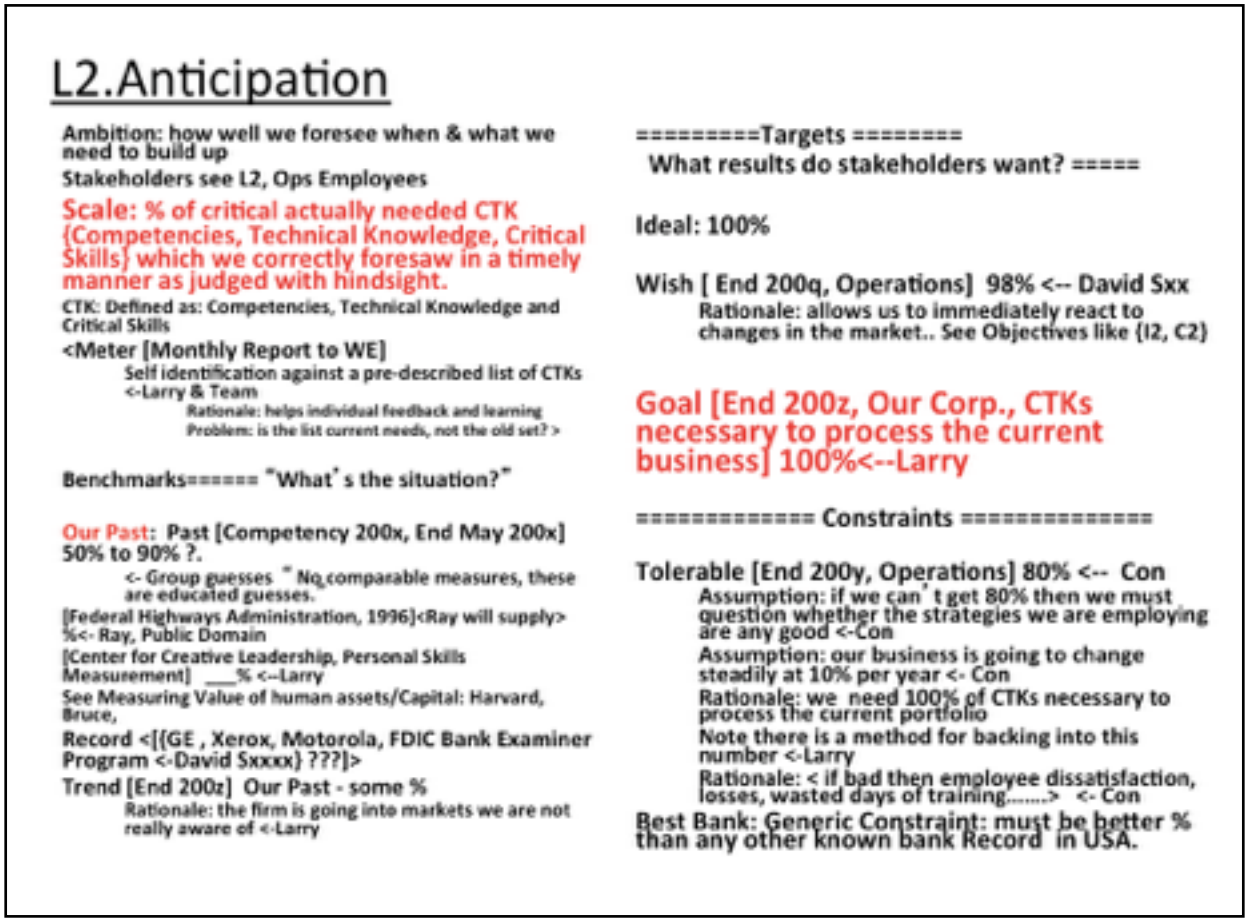
That is 'normal'. They experienced extreme frustration in understanding the BSC, with *these* 'nice sounding' non-financials.



Planguage 10.6 A. The original BSC Objective statement, the managers were using is in the 'Gist' statement above, in the second-level down, decomposition of their non-financial BSC objectives. In order to make it clearer, and measurably operational, our (Kai and I) first step was to decompose this compound objective, into the next level of its parts (Anticipation -> Employee Morale), in a meeting we had with a small team of managers. The next step of clarification is either further decomposition (not presented here) or a next step of quantification. See next Figure 10.6 B.

They agreed with this: that their objectives were not clear, and not quantified, after Kai and I showed them *how*. See Figure A & B here, and URL62 B ("What is wrong with Balanced Scorecards?" talk slides).

Case 10.6 continued



L2.Anticipation

Ambition: how well we foresee when & what we need to build up

Stakeholders see L2, Ops Employees

Scale: % of critical actually needed CTK {Competencies, Technical Knowledge, Critical Skills} which we correctly foresaw in a timely manner as judged with hindsight.

CTK: Defined as: Competencies, Technical Knowledge and Critical Skills

<Meter [Monthly Report to WE]

Self identification against a pre-described list of CTKs <-Larry & Team

Rationale: helps individual feedback and learning

Problem: is the list current needs, not the old set? >

Benchmarks====== "What's the situation?"

Our Past: Past [Competency 200x, End May 200x] 50% to 90% ?.

<- Group guesses " No comparable measures, these are educated guesses.

[Federal Highways Administration, 1996]<Ray will supply> %<- Ray, Public Domain

[Center for Creative Leadership, Personal Skills Measurement] ___% <--Larry

See Measuring Value of human assets/Capital: Harvard, Bruce,

Record <[{GE , Xerox, Motorola, FDIC Bank Examiner Program <-David Sxxxx} ???]>

Trend [End 200z] Our Past - some %

Rationale: the firm is going into markets we are not really aware of <-Larry

=========Targets ========

What results do stakeholders want? =====

Ideal: 100%

Wish [End 200q, Operations] 98% <-- David Sxx

Rationale: allows us to immediately react to changes in the market.. See Objectives like {I2, C2}

Goal [End 200z, Our Corp., CTKs necessary to process the current business] 100%<--Larry

========== Constraints ==========

Tolerable [End 200y, Operations] 80% <-- Con

Assumption: if we can't get 80% then we must question whether the strategies we are employing are any good <-Con

Assumption: our business is going to change steadily at 10% per year <- Con

Rationale: we need 100% of CTKs necessary to process the current portfolio

Note there is a method for backing into this number <-Larry

Rationale: < if bad then employee dissatisfaction, losses, wasted days of training.....> <- Con

Best Bank: Generic Constraint: must be better % than any other known bank Record in USA.

Planguage 10.6 A. Here is the real (slightly confidentiality edited) detail of the first decomposition objective, 'Anticipation'. Others are at URL62 B. Take some time to slowly study the content here, you will get a better picture of real Planguage practice, with many practical hints as to useful statements.

So the BSC had failed to get 'balanced', because the key to the 'balance' was *quantification* of non-financials (like qualities), not spouting more management BS.

End of Case 10.6

Here is more detail on the key concepts, that the planner must carefully distinguish between:

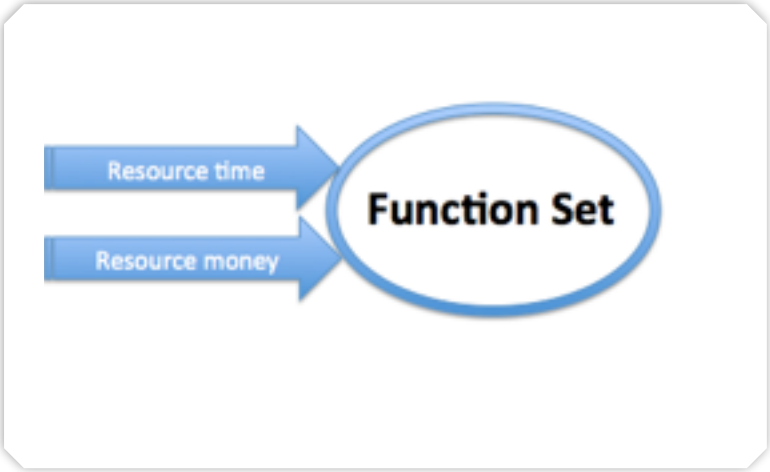


Figure 10.6 B. We use arrows 'pointing in' to the function, to diagram the limited or budgeted resources, needed to create the 'outputs' known as 'performance attributes' (work capacity and quality attributes, Fig. 10.6 C to F). These resource constraint plans are also sometimes known as 'budgets' and 'deadlines'. They are limited resources, possibly 'budgeted', resources, which are necessary to use, in order to deploy strategies, in order to get resulting performance levels (qualities and work capacities).

3. Work Capacity: *are* 'characteristics of a system', but they are *not* what most people would ever describe as 'qualities'. The usual term for work capacity is 'system performance'. This is **about 'how fast', 'how big'**. It amounts to the **capacity of the system to get things done**, and I prefer to use the narrower term 'work capacity'.

The thing about 'work capacity', which distinguishes it from 'qualities' is that we, as planners, are already very good at quantifying it, and managing 'work capacity'.

So I have comparatively little advice to you, on how to do that.

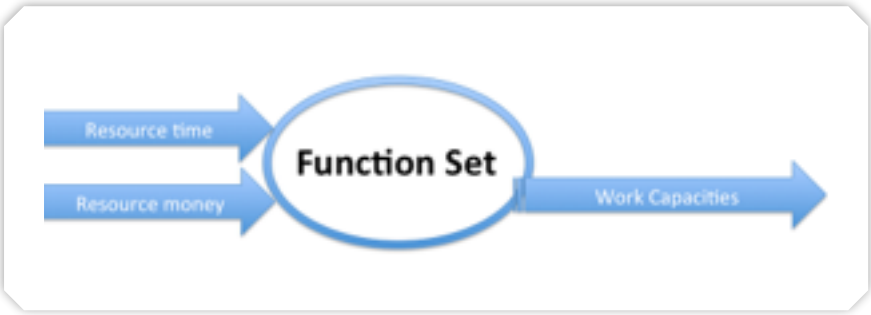


*Figure 10.6 C. An arrow coming out of the function right-side is a Planguage 'drawn icon' representing '**work capacities**'. There can be several of these, depending on how many you want to plan to achieve, or to present. These arrows are the graphical equivalent of the 'Scale' planning specification. There will be defined points on the arrow scale corresponding to concepts like Past, Tolerable, and Goal levels, on that scale.*

4. Designs (means, **strategies**, architecture, solutions etc.) are the 'second stage of planning' (after Objectives), we need them exclusively (only for this reason) to deliver these 'improved values':

1. Work Capacities
2. Qualities
3. Earnings, sales, profit
4. Within budgeted resources

for a defined 'Functionality' (example: Ambulance, Bank)

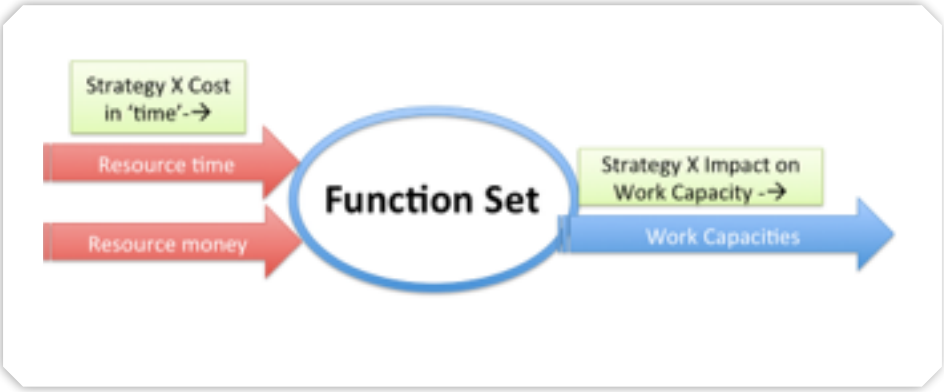


*Figure 10.6 D. 'Strategies' are the means by which we hope to achieve certain levels of performance (work capacity and or qualities). The Planguage icon for strategies (any 'means' to get objectives) is a **rectangle**, whose width is the degree-of-impact or degree-of-cost, the strategy is expected to have. Or actually did have.*

5. Qualities, are NONE of the above. They are *structurally* similar to 'work capacity' (how fast, how much), in that they are 'variable' attributes. They belong to the same joint category of 'Performance characteristics.

But qualities are *the set of objectives* describing '**how well**' the system functions.

- The combination 'Qualities + Work Capacities' is 'how good' the system is, for the stakeholders ('overall system performance characteristics').
- Systems engineers call this combination 'Performance' characteristics. Balanced Scorecard uses 'Performance' too [URL62 A]. So do I, in Planguage.
- The interesting problem with 'qualities', from a planning point of view, is that 'most everybody' (except the reader I assume) is terribly bad at planning for them, in general.
- We don't seem to have a *language*, and a *method* for doing so. So we shout some slogans ("We will achieve state of the art quality"), also known as management BS [URL1, URL2], and then the project manager focuses on costs, tasks, and deadlines; and loses control of even *these*, because we have no effective planning clarity, regarding our 'qualities': *which determine the costs*.

- See Case 10.6 above for good planning clarity; derived from an initial very fuzzy 'Gist'.

- **Quality *quantification* is a fundamental tool for handling complicated systems** [URL10].
  - Qualities are ***dynamically fluctuating***, compared to the more stable and predictable 'functions'.
  - So, qualities are primarily those characteristics we need to manage well, when faced with '**Complicated'** (confusing and unpredictable) systems environments.
  - And qualities must be *numeric* to be useful.
  - The failure to quantify qualities, could help explain why people are frustrated in trying to plan for complicated systems.
  - They do not have a *numeric multidimensional quality framework* to deal with the related feedback in short time frames.

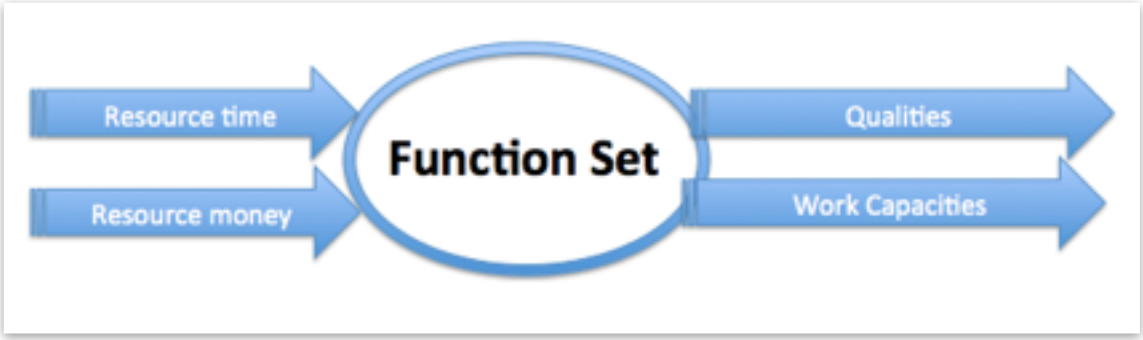


Figure 10.6 E1. Qualities are one class of 'Performance' characteristics (how 'Good' the system is). ***Qualities*** *are* ***'how well'*** *the functions perform. Note that the left-side arrows are resource 'consumption', the debit side of the ledger (in the 'red'). The right hand-side arrows are system performance. They include all manner of resource saving, or resource generation. That means the financials go on the left side; these right arrows are the 'credit' side (in the 'black') of the ledger.*

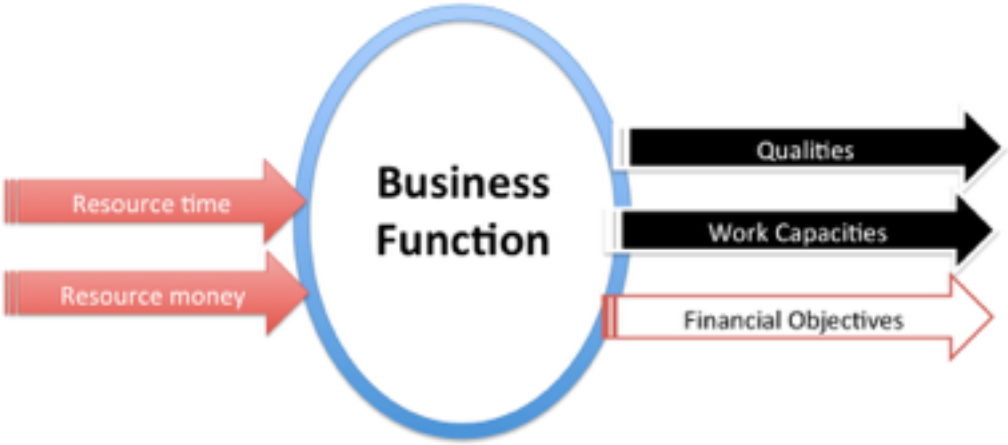


Figure 10.6 E2 In business planning, the Financial objectives, are of course part of the Balanced Scorecard picture. In our way of thinking, the Qualities and Work Capacities are the 'Means Objectives', at a level serving the 'Strategic Level' Financial Objectives. So this diagram oversimplifies, and hides the cause-effect relationship. But it serves to show where the BSC Performance objectives are, in relation to function and resources. (reused as Fig. 1.1 B)

Finally, here is the *impact*, graphically (using Planguage Graphical Icons [URL34]) of a Strategy, on resources, qualities and work capacities

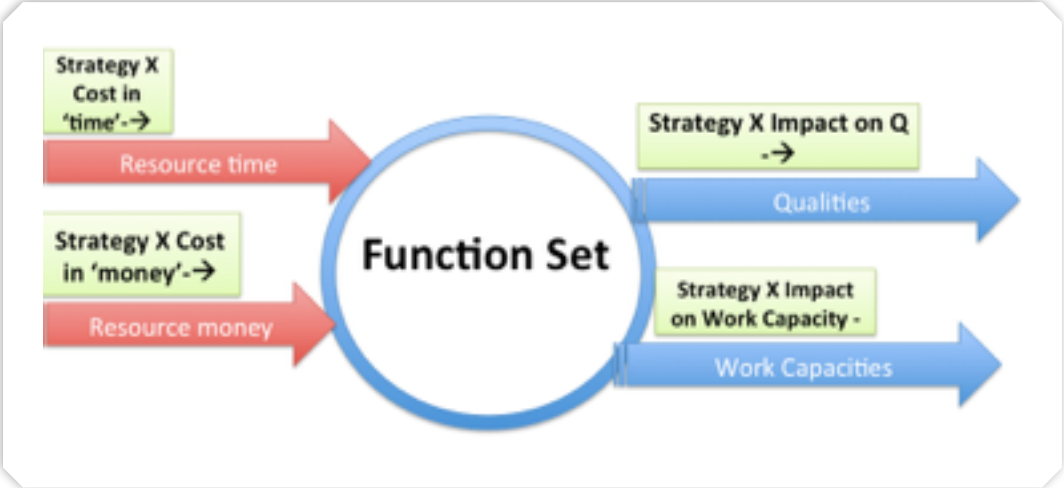


Figure 10.6 F. Here is a graphical diagram showing 4 effects (2 on budgeted or limited resources, and 1-each on a quality and a work capacity. The width of the rectangle is equal to the magnitude of the effect. This of course is a graphical representation of an Impact Estimation Table. [URL21]. If the Performance objectives (qualities, work capacities of a product, service or other system) here are reached, then we hope that the next *level of objectives, which these are 'Means Objectives' for, probably Financials, are also achieved, or at least contributed to.*

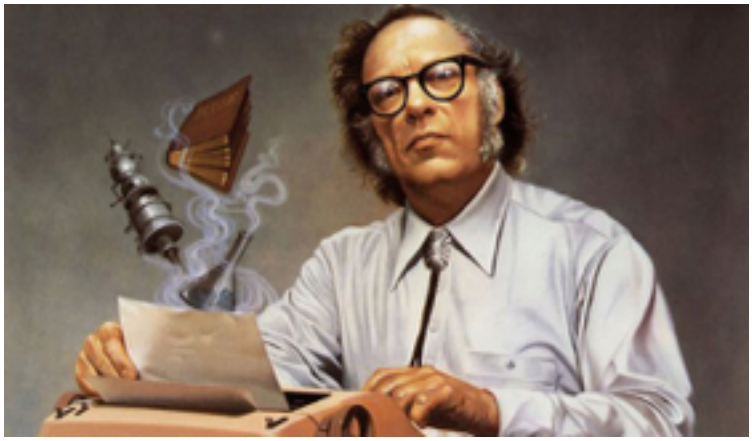
"Intelligence is an extremely subtle concept.

It's a kind of understanding that flourishes if it's combined with a good memory, but exists anyway even in the absence of good memory.

It's the ability to draw consequences from causes, to make correct inferences, to foresee what might be the result, to work out logical problems, to be reasonable, rational, to have the ability to understand the solution from perhaps insufficient information.

You know when a person is intelligent, but you can be easily fooled if you are not yourself intelligent."

— Isaac Asimov (1920-1992) Quote 10.6 B
Source: In Irv Broughton (ed.), The Writer's Mind: Interviews with American Authors (1990), Vol. 2, 57.



Practical Tip 10.6: The Mafia Offer: Make People Acknowledge Their Real Objectives.

- *Focus most of your initial planning attention on collecting the most-critical stakeholder values, most of these are qualities.*
- *Ask stakeholders what they value, but help reformulate their inputs as 'qualities', and do not allow their strategies, for the values, to get mixed in.*
- *The qualities they really value, are often 'only implied' by their strategies.*
- *Give them the Mafia Offer:* ***do you want to get that strategy you mentioned, even if you do not get the quality it implies, that you <u>really</u> want? Or, do you want the quality objective, even if you cannot ever use that strategy you mentioned?***
- *Force them to choose what they 'really want'.*
- *Then you are free, as a planner, to help them find good strategies to get there; while at the same time taking into account all other planning objectives, constraints and stakeholders, not to mention the unpredictable unknowns of confusing systems. Holistic, systems, planning.*
    - *This is a complicated and complex system of planning.*
    - *It would be nice if we could ignore that.*
    - *But if we oversimplify we get pain far greater than the pain of considering these system parts and their relationships during a planning process.*

Policy 10.6 Clear Fundamental Planning Concepts

- **We will make *clear distinctions* between the *concepts*** (whatever we choose to call them by name, consistently) **of functions, resources, work capacity objectives, strategies, and quality objectives.**
- **We will not stumble and fail, in the quicksand of vague concepts, or vague planning content.**

Why?

- Because otherwise we will totally lose control of many levels of critical qualities for our stakeholders.
- And as a consequence lose control of markets and costs.
- And we will lose a necessary framework for rapid competitive response to complicated environments.

Version 30 Dec. 2015, 04:12 CET, TsG

Section 10.7 Enriching the planning specification of quality values.

Principle 10.7 Richly specify your ends, upon which your success depends.

Quality objectives,
and all other performance objectives
benefit from being *enriched* with realistic detail.
and modeled realistically,
rather than oversimplified,
as is too often the case.

Qualities are one small, but vital, element of our planning domain: here we see their place on a map of many classes of system requirements:

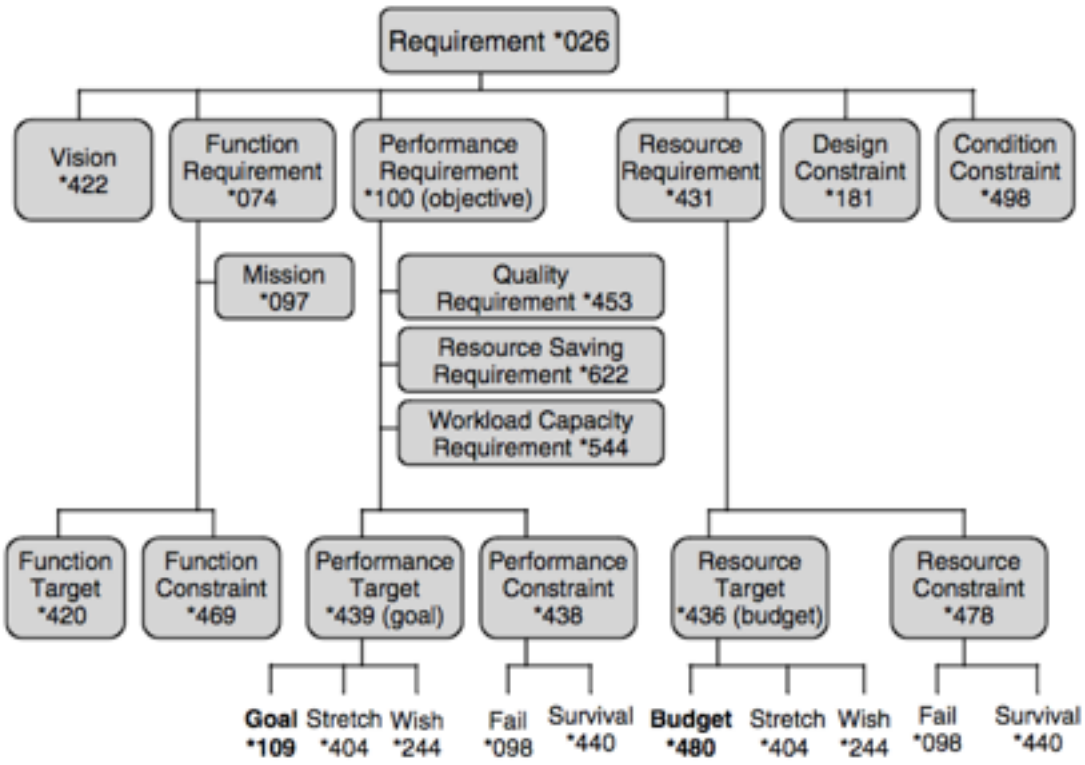


*Diagram 10.7 A [Source [1], Glossary, URL34B]. Can you spot the 'Quality Requirement' (Planguage Concept *453). There are quite a few other planning elements, just on the 'requirements' side of planning work. But quality objectives are very critical, and high priority. But quality objectives are nevertheless very badly treated in conventional planning. So, we are going to present ideas here to help you to treat quality objectives as they deserve. So stakeholders actually have a fair chance of getting what they value most; and your organization can then succeed better in its business.*

Our conventional planning culture has a narrow view of what 'qualities' are. It is, as mentioned in Section 10.6, often *very narrowly* used in daily reference (like 'how often it screws up' or 'fails').

Qualities are far richer than even the highly generalized set of qualities, below.

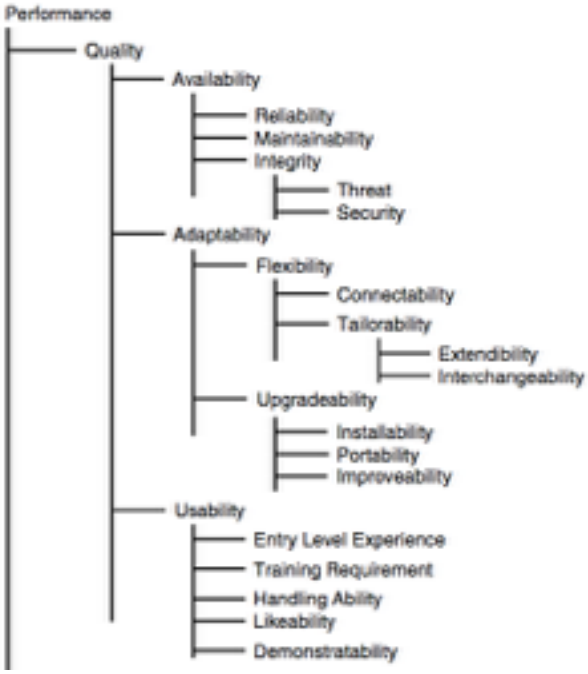
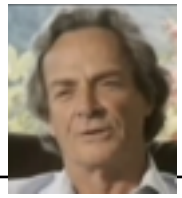


Diagram 10.7 B [1 D]. A general tree diagram of some common quality ideas. There is no standard set of qualities. Even when the 'names are the same' (Feynman, Quote 10.7 A), the actual definition (Scale, Goal etc) you need, may be radically different. However the set of common qualities above is a useful starting template, since these qualities are frequently valued.

- ***"You can know the name of that bird in all the languages of the world, but when you're finished, you'll know absolutely nothing whatever about the bird.***
- ***You'll only know about humans in different places, and what they call the bird.***
- ***So let's look at the bird and see what it's doing—that's what counts."***
- ***(I learned very early the difference between knowing the name of something and knowing something.)" [URL79]***



- ***Richard Feynmann***

Quote 10.7 A. For example words like 'Flexibility,' or 'Likeability' (Diagram 10.7 B) will have very different meanings when they are defined in practice by Scale, Goal and other parameters.

The main points I wish to make in this Section are:

- Qualities are infinite in variety.
- There is no simple 'standard set' of qualities which will be good enough for value planning. You can't cut and paste earlier quality definitions, without thinking deeply about your own business, just now and in the future.
- Planguage has a rich set of practical devices for helping you articulate any kind of specialist quality objective, that is of interest to you. Tailoring your quality objectives is a *necessity*.

Here are the basic Planguage Tools for tailoring quality objectives:

These have been discussed earlier in the book.

- The 'Scale' definition
  - (example) **Scale: the average % responses from customers per year.**
- The [Scale *Parameters*], which 'generalize' the Scale, and make it 'reusable'.
  - (example) **Scale:** the average % defined **[Responses]** from defined **[Customers]** for defined **[Reasons]** per year.

- The Scalar **Levels**
  - **Benchmarks** (Past, Trend, Record),
  - **Constraints** (Tolerable, OK),
  - **Targets** (Wish, Goal, Stretch, Ideal)

  Scalar *Levels* allow us to indicate what degrees of a quality are required and desired, and what *priority* these levels, for specified conditions, have.

  Examples:

Scale: the average % responses from customers per year.

OK [Deadline = Trial Release, Market = Singapore] less than 3%.

Goal [Deadline = First Release, Market = Asia] less than 1%.

The Scalar Levels can have **[Qualifiers]**, allowing us to tailor to '*when*, *where* and *If*' conditions.

For example:

Scale: the average % defined [**Responses**] from defined [**Customers**] for defined [**Reasons**] per year.

Goal [Deadline = First Release, Market = Asia, **Responses** = Complaints, **Customers** = Long Term, **Reasons** = Product Failure, **If** Product is NOT in Beta] less than 1%

As you can see, we can add any other conditions of interest, in addition to the ones defined in the Scale Parameters. We added 'Deadline', 'Market' and 'If', here. This allows us to model our reality, more exactly and more relevantly. We avoid unnecessary generalization. We get realistic.

Here are some additional Planguage tools for tailoring objectives:

- A great portion of the rest of Planguage are potential tools for tailoring, but we will mention a few:
  - Source
  - ± range
  - Notes
  - Justification
  - Value
  - Issues
  - Assumptions
  - Dependencies
  - Constraints (legal rather than Scalar)
- and very many more. [1]

Decomposition of Objectives:

- Decomposing an objective into one or more levels of sub-objectives is another way of tailoring. See previous example in Case 10.6 of decomposition.
- Decomposition allows us more control over details, which is essential:
  - In large systems
  - In competitive systems
  - In complicated systems

Here is another example of a useful, reusable pattern for decomposition of 'Maintainability', which most people are happy to leave at the traditional simple level of 'the Mean Time to Repair'.

Diagram 10.7 Each sub-process of maintainability, above, contributes to the total closure of the 'fault' problem. And each process will need quite different strategies to be effective.

But advanced engineering has long since (1966 and before, actually) seen a practical need for a more-tailored version, like this:

Maintainability:
Type: Complex Quality Requirement.
Includes: {Problem Recognition, Administrative Delay, Tool Collection, Problem Analysis, Change Specification, Quality Control, Modification Implementation, Modification Testing {Unit Testing, Integration Testing, Beta Testing, System Testing}, Recovery}.

Problem Recognition:
Scale: Clock hours from defined [Fault Occurrence: Default: Bug occurs in any use or test of system] until fault officially recognized by defined [Recognition Act: Default: Fault is logged electronically].
Administrative Delay:
Scale: Clock hours from defined [Recognition Act] until defined [Correction Action] initiated and assigned to a defined [Maintenance Instance].
Tool Collection:
Scale: Clock hours for defined [Maintenance Instance: Default: Whoever is assigned] to acquire all defined [Tools: Default: all systems and information necessary to analyze, correct and quality control the correction].
Problem Analysis:
Scale: Clock time for the assigned defined [Maintenance Instance] to analyze the fault symptoms and be able to begin to formulate a correction hypothesis.
Change Specification:
Scale: Clock hours needed by defined [Maintenance Instance] to fully and correctly describe the necessary correction actions, according to current applicable standards for this.
Note: This includes any additional time for corrections after quality control and tests.
Quality Control:
Scale: Clock hours for quality control of the correction hypothesis (against relevant standards).
Modification Implementation:
Scale: Clock hours to carry out the correction activity as planned. "Includes any necessary corrections as a result of quality control or testing."
Modification Testing:
Unit Testing:
Scale: Clock hours to carry out defined [Unit Test] for the fault correction.
Integration Testing:
Scale: Clock hours to carry out defined [Integration Test] for the fault correction.
Beta Testing:
Scale: Clock hours to carry out defined [Beta Test] for the fault correction before official release of the correction is permitted.
System Testing:
Scale: Clock hours to carry out defined [System Test] for the fault correction.
Recovery:
Scale: Clock hours for defined [User Type] to return system to the state it was in prior to the fault and, to a state ready to continue with work.

Source: The above is an extension of some basic ideas from Ireson, Editor, Reliability Handbook, *McGraw Hill, 1966* (Ireson 1966).

Template 10.7 A [1 D, CE Chapter 5]. Maintainability, decomposed in very useful ways for complicated systems management.

Comment on the 'Maintainability' Template above: I consider this Maintainability Template to be a *general* template. Almost every decomposition in my books are general, and even most of my case studies are general too. 'General' in the sense they apply to *any class of system*.

Because, although you initially look at this template (10.7 A) and think, 'that is for hardware maintenance', or 'that is for software maintenance' (in this case I tailored a hardware template to be used for software), it is in fact a very useful *general pattern* for 'fixing' any 'fault' (bad thing); or more-broadly *changing any system*, such as an organization, a project, a manager, or a process.

I have found that when I encounter 'new' problems, like 'can we usefully decompose the objective of *fixing a failing project*?', I start solving the problem by homing-in on other known, or discoverable-on-the-internet, 'fixing' patterns, like this one.

Then I can 'solve the problem of decomposition', using those patterns as a *guide* and a *checklist*.

It requires a little bit of ability to *generalize*, and to *tailor*. But *not* as much as if you have to 'invent the concept from scratch'. Templates help us be creative, so we solve our problems more realistically.

I wonder if my readers could tailor '**project** maintainability' using this template?

Well, maybe not *all* of you could.

So, to amuse myself, I'll have a go – I never did it before, either !

Project Maintainability:

Type: Compound Quality Requirement.

Includes: {Problem Recognition, Administrative Delay, Tool Collection, Problem Analysis, Change Specification, Quality Control, Modification Implementation, Modification Testing {Unit Testing, Integration Testing, Beta Testing, System Testing}, Recovery}.

Problem Recognition:
Scale: Average Days from Serious Project Problems Occur, until Program Management have it on their Dashboard, for observation of recurrence.

Administrative Delay:
Scale: Average Days from Serious Project Threat Level Occurs until Management decides to step in with First Meeting about situation.

Tool Collection:
Scale: Average Days from First Meeting Scheduled, until all necessary (defined in Problem Analysis process Entry Conditions), data about the project, is available to the Manager.

Problem Analysis:
Scale: Average Office Calendar Days needed to analyze root causes of Project Problems, so that the Objectives for Change, can be quantified.

Change Specification:
Scale: Work Days needed from Project Change Objective is formulated, until we formulate an Approved Response Strategy for The Project, and possibly all other Projects (Process Change).

Quality Control:
Scale: Clock hours for quality control of the Approved response Strategy (against relevant standards, including Impact Estimation of effects and costs).

Modification Implementation:
Scale: Clock hours to carry out the Approved response Strategy.

Modification Testing Activities:

Unit Testing:
Scale: Clock hours for a simple test of how the Approved Response Strategy works under realistic conditions.

Beta Testing:
Scale: Clock hours to carry out defined [Beta Test], for limited parts of one project for the fault correction, before official release of the correction, to the entire project, is permitted.

System Testing:
Scale: Clock hours to carry out defined [System Test] for the fault correction, including all impacted Projects.

Recovery:
Scale: Average Days required to recover any losses of time, budget or required performance characteristics (the Project Faults) after the Approved response Strategy is successfully in place.

*Template 10.7 B. An attempt to use the Technical maintainability template (10.7 A above) for **organizational** planning purposes. It works for me ☺. I am trying to demonstrate how you might adapt any template to your specific purposes.*

Comments on the Template for Project Maintainability:
Notice that only after the **Recovery** phase, (above) are we back to where we *should* have been, *before* the faults occurred.

It is not enough to 'fix the faults', you have to 'fix the *damage*' too.

There is no point in merely 'fixing damage' for a *recurrent fault* ('Attack' type) before you stop it recurring.

You will find that a decomposed objective like this, really makes you think, far more realistically and deeply, about the planning problem you are facing.

It makes you plan more *thoroughly*, like Roald Amundsen, the Polar Explorer, so that we 'get lucky' and don't 'have bad luck'. [10.2 Quote]

One consequence of this quality decomposition, you might be able to understand better now, is that we are going to need to plan *quite different strategies* for *each* of the different sub-objectives.

If we had not decomposed, we would be less likely to recognize this need for different strategies. We might think that *one*, or a *few*, strategy decisions were sufficient.

Another point that becomes obvious with the decomposition is that the problem has a much broader *time* frame than most people would imagine.

I think most of us would be focused on the 'fault fix' (**Modification Implementation**) phase *alone*, and forget to plan for the many other activities, before and after that, which are necessary to handle this problem.

This template offers us better 'systems thinking', and offers us less sub-optimization.

Decomposing objectives, like this one, opens many more 'channels' to get feedback, and therefore is a tactic for better managing complicated systems.

You are; with so many more measurable feedback channels, more likely to sense changes, and to sense the *need* for changes in shorter time horizons.

"This is what customers pay us for - to sweat all these details so it's easy and pleasant for them to use our computers.

We're supposed to be really good at this.

That doesn't mean we don't listen to customers, but it's hard for them to tell you what they want when they've never seen anything remotely like it.

Steve Jobs (1955-2011) . Quote 10.7 B.

Thanks for the products Steve, I am enjoying my Apple Watch at the moment. I could not have imagined it as well as Apple did.

Practical Tip 10.7: Quality Decomposition helps deal with Complicated Problems.
If you have a challenging project, large, complex, complicated, new stuff; then consider decomposing (multiple scale definitions, a Compound Objective Type) at least one of your most critical top objectives; just to see if you feel there may be benefit on continuing down that path.

- *Compound Objectives help fight Confusing systems.*

Policy 10.7 Model that nasty reality, before it gets nasty with you.

- **We will normally decompose into Compound Objectives most all of our top-ten critical objectives; at least for non-trivial projects.**

- Why?
  - To help you recognize problems *upstream*, instead of 'too late'
  - To enrich your recognition of all the *different strategies* you need to plan, before it is too late
  - It is fairly cheap insurance, and you can always *simplify afterwards*, if you feel that is a better tactic.

Version 30 December 2015 04:32 AM! CET TsG

Section 10.8 The attributes of a planning language.

Principle 10.8 Qualities of Planning

**All planning languages
and planning processes
can be characterized and evaluated
by their multiple qualities.**

This book has presented a planning language, 'Planguage', and associated processes, such as Spec QC, decomposition, and Evo.

This is the 'Quality' Chapter, 10. So I wanted to make the point that any *language* or *process* can also be understood better, in terms of their quality characteristics.

Any language or process ('methods') could be picked as a strategy to help meet an objective. We can use Impact Estimation tables to illuminate the characteristics, and compare the methods.

What is this useful for?

- We can compare everything in this book, with our old methods, other methods,
- We can select methods more intelligently for purposes and environments
- We can present our decisions about methods to other people in a more holistic way.

What are some of the quality dimensions a **planning language** has, that might be on your agenda? *Without trying to define these quantitatively*:

- **Intelligibility**
  - The degree that target readership correctly understands specifications or plans.
- **Comprehensiveness**
  - The degree of the planning problem that can be covered.
- **Adaptability [URL26 A, Tailoring using PL]**
  - The degree we can modify the planning language for new purposes

        - **Global Adaptability**: change capability for all users in a defined environment.
        - **Local Adaptability**: change capability for a single user, on the spot.
- **Extendibility**
    - The degree we can add to the language for any given purpose.
- **Portability**
    - The ease of adapting the language to various human languages.
- **Generality**
    - Applicability to various problem domains.
- **Automatic Interpretability**
    - Ease of computer-support tool building.
- **Consistency Checking Ability**
    - Ease of correlating statements with other related statements for consistency checking, or error finding.
- **Ease of Learning**
    - How quickly people can learn to write it, read it, exploit it at an advanced or comprehensive level.
- **Audibility**: Ease of doing quality control, of plans, in relation to standards.
- **Risk Management Capability**: ability to deal with risks.
- **Dynamic Reaction Capability**: ability to sense problems and react with necessary change.

Here is an example of defining one planning language quality, for a given organization's purpose:

Comprehensiveness:
Type: Planning Language Objective.
Version: 9 September 2014 16:41.
Stakeholders: Planners, Planning Director, CxO**s.**
Ambition Level: One Planning language to cover all our aspects of planning.
Plan Element Owner: Tom Gilb.

Scale: The % of a defined Planning [**Scope**] that can be Best covered by a defined Planning Language [**Content**] for defined types of [**Planners**].

Meter: manual evaluation by planning expert, in house. Should take less than 1 hour.

Goal [Deadline = Within 1 Year, Scope = Top Management Plans, Content = {Objectives, Qualities, Costs, Strategies, Estimates, Feedback}, Planners = CxO level planners and their direct planning staff.] > 95% ? <- TG strawman.

Planguage 10.8 A. This objective is for one short term Goal, for one type of planner and planning. You can easily see how other goals using other qualifiers can be formulated.

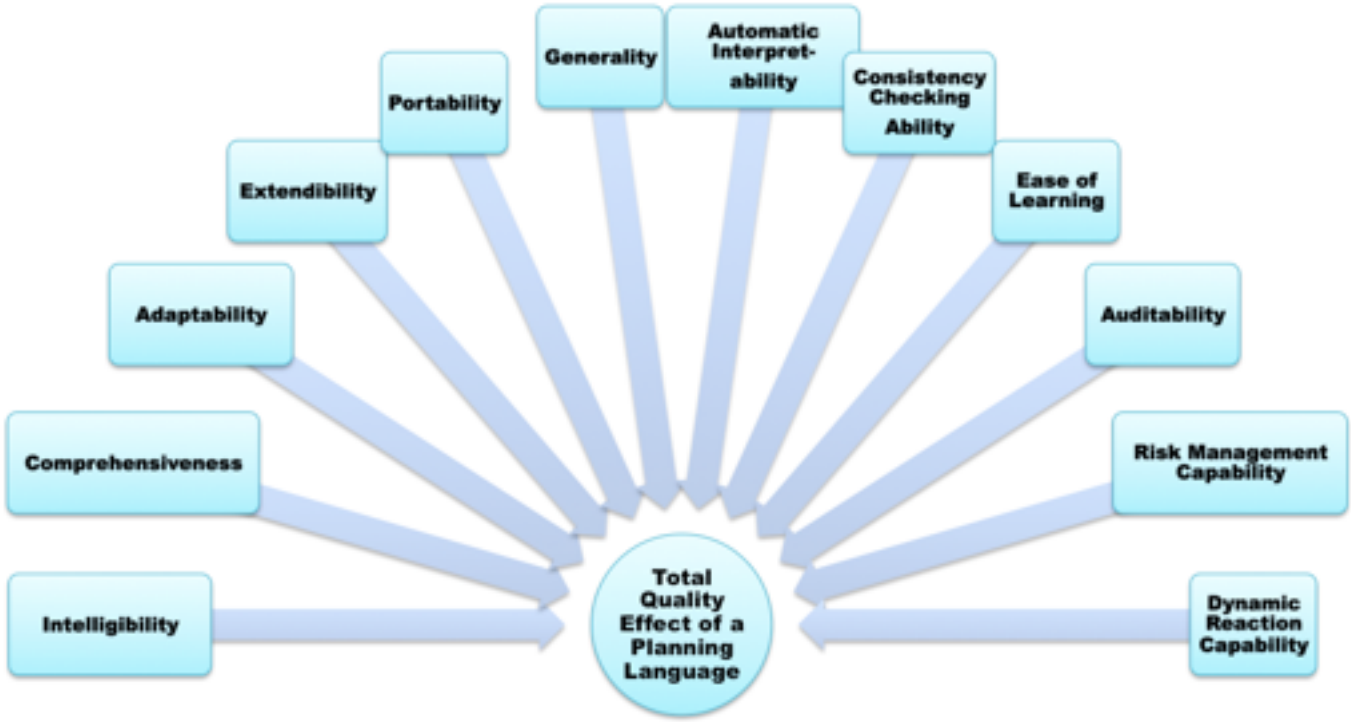


Diagram 10.8 A. Some measurable quality characteristics of a planning Language.

Planguage Examples 10.8
Draft of Scales for all planning-language quality characteristics.

Intelligibility
Gist: The degree that target readership correctly understands specifications.

Scale: The % of defined [**Planning Elements**] that defined planning [**Readers**] understand Correctly, with defined [**Training**], upon first reading, within minutes, alone and Unaided.

Comprehensiveness
Gist: The degree of the planning problem that can be covered

Scale: The % of a defined Planning [**Scope**] that can be Best covered by a defined Planning Language [**Content**] for defined types of [**Planners**].

Adaptability
Gist: The degree we can modify it for new purposes.

Note: we were able to devise such a general scale with scale parameters including 'Types, that this scale will allow us to cover several of the other quality attributes listed initially.

Scale: the defined [**Ease**, default Work Hours] with which defined [**Adapters**] can Successfully make defined adaptation [**Types**]**.**

Types: 'of Planning Language Adaptability'

- **Global Adaptability:** change for all users in a defined environment.
- **Local Adaptability:** change for a single user, on the spot.
- **Extendibility**
  - The degree we can add new parameters, grammar or constructions, to the language for any given purpose.
- **Portability [Human Language]**
  - The ease of adapting the language to various human languages.

Generality:
Gist: Applicability to various problem domains

Scale: the % of defined Problem [**Domains**] that can be adequately handled by a defined Planning [**Language**].

Automatic Interpretability
Gist: Ease of computer-support tool building**.**

Scale: [Resources: default = Work Hours] necessary to build or acquire automated tools with a defined level of [Support] for defined Planning Language [**Aspects**, Default = All].

Aspects: of potential automation, for a planning language.

- Selective Presentation**.**
- Graphical Presentation
- Consistency Checking
- Rule Violation Measurement, or hint at possibility.
- Intelligent Choices
- Automatic Partial fill in
- Learning from Experience.
- Integration: ability to make use of other related specifications. Example: Strategy impacts from a Strategy Spec , in an IE table.
- Missing specifications (for sure, potentially, possibly)

Consistency Checking Ability
Gist: Ease of correlating statements with other related statements for consistency, or error finding.

Scale: The % ability to detect and report defined [**Inconsistency**

Types] using defined Checking [**Means**] with respect to defined Consistency [**Rules**] with a view towards defined checking [**Purposes**].

Ease of Learning
Gist: How quickly people can learn to write it, read it, exploit it at an advanced or comprehensive level.

Scale: The average speed in minutes, using defined Learning [**Methods**] for defined types of [**People**] to provably and repeatedly Understand defined planning [**Expressions**].

Auditability:
Gist: Ease of doing quality control in relation to standards

Scale: average % of defined [**Violation**] types found by defined [**People or Devices**].

Risk Management Capability:
ability to deal with risks.

Note:*This is a key characteristic for dealing with 'Complicated' systems. We have to understand threats and risks that exist, and pre-plan as far as possible or economical, mitigation. This will not save us from risks and threats we cannot foresee. Dynamic Reaction Capability, below, is our answer to the 'unknowable' in advance. But we will be able to decrease the attack and damage levels considerably if we choose to invest in it.*

Scale: the % capability of identifying and reporting defined [**Threats**] according to defined Threat ID trigger [**Threat Rules**].

- **Threats**
  - **T1:** Identified Threat of any type, with no corresponding planned mitigation.
  - **T2:** A Goal, with less than 100% impact estimation total.
- **Threat Rules**
  - **R1:** All Goal levels will have at least safety factor 2.
  - **R2:** All Type: Critical Objectives shall have at least 1 Constraint (OK, Tolerable] level.

Dynamic Reaction Capability:
Gist: Ability to sense problems and react with necessary change.

Note: *This is a key characteristic for dealing with 'Complicated' systems. We have to rely on feedback, in order to sense emerging, but not well-predicted, (and not easily predictable) attributes*

Scale: the average defined reaction % to defined [**Update Data**] types, according to defined Reaction [**Rules**] for defined Planning Language [**Scope**].

Update Data:
U1: Measures from Value delivery cycle.
U2: Feedback from Critical Stakeholders.
U3: External Environment Changes.

Rules:

	R1: warn when actual cumulative value is more than 20% disproportionate with time used to date % of time to deadline.

So, now you have a basis for building your own model to describe your planning language objectives, and then to evaluate Planguage and all other planning languages for the capabilities you want.

What are some of the quality dimensions that a **planning process** (such as SQC or Evo) might have, on your agenda?

- **Adaptability**: ability to change the process for other purposes than designed for.
- **Automated Supportability**: the degree to which we can get computer support for the process.
- **Comprehensiveness**: the range of applications possible.
- **Ease of Learning**: time for people to learn to use the processes.
- **Result Predictability**: the degree to which various expected results can be consistently relied on.
- **Language Integration**: the degree to which the processes can exploit a given planning language component.

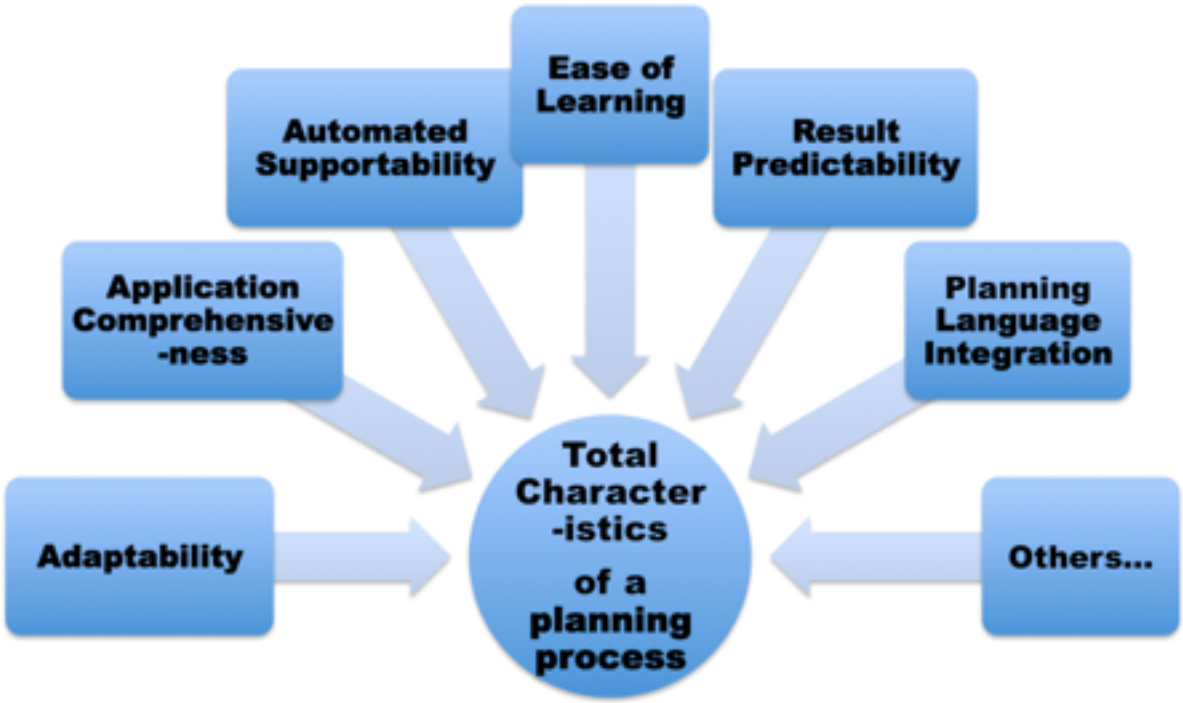


Diagram 10.8 B. Some generally important qualities of planning processes, which can be used to judge their suitability for your purposes. Perhaps using an Impact Estimation table for comparison.

> ***"It's not hard to make decisions***
> ***when you know what your values are."***

— Roy Disney (1930-2009)

Practical Tip 10.8 Consider Planning Qualities

When choosing, or arguing for, a planning language; you should evaluate multiple dimensions of it, for example some of the ones suggested above.

Policy 10.8 Choose planning methods intelligently: plan for them!

- **We will choose our planning language and tools to fit a planned profile of our planning needs.**
-
- Why?
    - If you do not use rational thinking to select your planning methods then
        - You will end up using whatever you do now, and it will be as bad as it is now
        - In frustration you will try out the latest management fashion in planning, which will not work well for your needs either.

Version 30 Dec. 2015, 04:58 CET, TsG

Section 10.9 What Professors Should Really Teach: Lasting Wisdom.

Principle 10.9 A. Learn 'Efficient' Knowledge

You should focus your energy on acquiring knowledge with a long half-life.

Principles of Knowledge, from a Business Graduate Point of View [URL43, complete paper].

Summary

There are some very basic things that planners should be taught. These things are both *fundamental* and *classic*.

They are *fundamental* because we can reuse them in a very wide variety of planning situations. They are *classic* in the sense that they have a very long 'usefulness' half-life. They are probably useful for at least a career lifetime.

When I was in my Twenties, I decided to collect, to learn and to develop these planning basics [F7]. Now, in my Seventies (75), I am more than ever convinced that these fundamentals should be shared with students, maybe even open minded teachers!

The fundamentals are: Principles (heuristics, laws), Measures (ways to quantify critical factors), Concepts (really useful definitions of fundamental planning ideas), and Processes (really useful planning processes).

I have published these fundamentals in several books [1] and papers already. I would like to argue here, why they need teaching, in undergraduate studies, for business schools and any other planning domain.

I *believe* that the usefulness and longevity of these fundamentals, is demonstrated in my own decades of work with them. Their durability is acknowledged by many professional colleagues and some academics, and is hopefully self-evident upon examination. Time will tell. Hello from me if you are reading this in one hundred years or more.

Hopefully this Section (and the next Section, 10.10) can stimulate others to adopt at least the general ideas, if not my exact artefacts.

My 'Principles of Useful Knowledge'. (Source URL43, © Gilb, 2007)

UNIVERSALITY:

1. Knowledge is more useful, when it *applies to more circumstances*.

ETERNALITY:

2. Knowledge is more worth learning, if it can be *applied for a long time*, after learning it.

VALUE:

3. Knowledge is more useful, if there is a high *value* from applying it.

SHARING:

4. Knowledge is more useful, if it can *easily be shared* with others.

PROOF:

5. Knowledge is useful, when early *feedback can prove its usefulness,* in practice.

SYNCHRONOUS:

6. Knowledge is more useful, when it can be *used together with a larger body of knowledge.*

MEASURABILITY:

7. Knowledge is more useful, when the *results of its application can be measured.*

ACCEPTANCE:

8. Knowledge is more useful, when it is *widely accepted in your culture.*

COST:

9. Knowledge is more useful, when the *cost of applying it is low*.

GENERATION:

10. Knowledge is more useful, when is can be used *to generate even more-useful knowledge.*

I hope these principles are self-evident observations. I hope that these principles characterize the class of knowledge that our 'institutes of higher learning' should be focussing on.

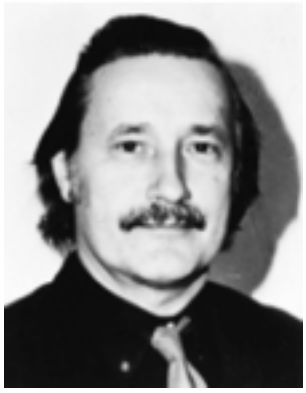
Here is some of my personal inspiration for suggesting what should be taught at undergraduate level: See complete quote [F15], from Prof. Christopher Strachey, Oxford

Quote 10.9 A. An academic plea for sanity.

…. "The premise is that it is clearly wrong to teach undergraduates the state of the art; one should teach them things which will still be valid in 20 years time: the fundamental concepts and underlying principles. Anything else is dishonest.
…
But at the moment we are in a very primitive state of development; we don't know the basic principles yet and we must learn them first.

If universities spend their time teaching the state of the art, they will not discover these principles and that, surely, is what academics should be doing.
…
But, before teaching students we must get our basic principles right."

[Strachey 1969] First Oxford Computer Science Professor



Christopher Strachey (1916 Hampstead-1975 Oxford)

Here is a discussion of my 'Principles of Principles' [URL43, © 2007]**, in the light of the 'Principles of Knowledge', above**.

1. The Notion of *Usefulness* of Principles:

A 'principle' is a short statement that guides people to take certain decisions or action. It is condensed wisdom. Principles are useful, if they remind us, or teach us, to act in a better way, than we otherwise would do.

For example: ***The Risk Principle. (Principle 10.9 B)***

> ***"There is lots of uncertainty and risk of deviation from plans in any project.***
>
> ***You cannot eliminate risk.***
>
> ***But, you can document it, plan and design for it, accept it, measure it, and reduce it to acceptable levels.***
>
> ***You may want to avoid risk, but it doesn't want to avoid you."***

Source (Gilb [1] page 23) 2005 and earlier books such as 'The Risk Principle' in PoSEM, 1988 [5 F].

This Risk principle tries to warn about the *inevitability* of risk. It also is specific about what you can *do* about risk. It teaches that you cannot eliminate risk, but you can try to manage it in various ways.

From the departure point of this principle, the teacher can then be more specific on how to identify, specify and mitigate risks. See this book, Chapter 7 on Risk, for lots of practical advice and tools.

2. The Notion Of Half-Life of Principles

If a principle were to become obsolete in a few years – perhaps because of new technology or new economics, then it would be less valuable to learn, and might even be dangerous to continue to practice, beyond its true lifetime.

So I prefer principles that we can imagine 'always were true', and we can so no clear reason why they 'will not be true for the foreseeable future".

I use this ("always were true …") as a thought test, for all principles that I publish. [URL5, URL20]

It takes decades or more, from when a principle is first stated or published, until it becomes taught in any substantial way. The student has decades of their future in which to apply a principle. So it makes good sense that the principle is something we can rely on using in the long term.

3. The Notion of Fundamentality of Principles

Principles should be *fundamental*. They should be basic tools for everyday use in planning, engineering, discussing, decision-making, and reasoning.

We should be able to use them as the basis for all of our more-detailed actions and thinking processes.

For example: (of 'fundamental')

The Principle of 'Quality In, From the Beginning'. (Principle 10.9 C)

Quality needs to be designed into processes and products.

Cleaning up bad work is a loser, but cleaning early is better than late.

A stitch in time, still 'saves nine',

But an ounce of prevention, is still worth 'a pound of cure'. "

Source © Gilb [1] CE, page 24.

P.S. for Metrics System friends. 1 Pound = 16 ounces weight.

The above principle applies to all planning work. We humans seem to have a strong natural tendency to clean up our faulty work, *when* faults are discovered, rather than to consciously discover how we can *prevent* the faults, from getting into our work, in the first place.

This 'plan quality in' principle is fundamental. It is at the basis of all improvement efforts in a planning process.

It is the basis for a paradigm shift for many professionals I deal with; the shift from 'fix problems', to 'prevent problems'.

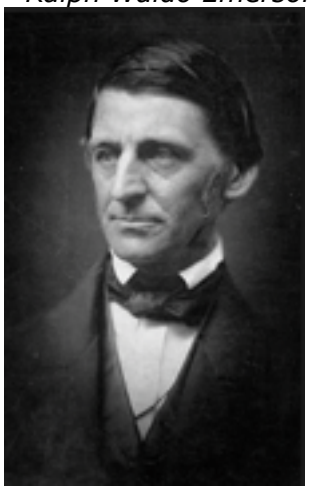
Students should be taught such profound principles, before they waste years discovering them, if they ever do.

Quote 10.9 B

"As to methods, there may be a million and then some, but principles are few.

The man who grasps principles can successfully select his own methods."

- Ralph Waldo Emerson (1803-1882) Photo ca. 1857



Thanks Rolf Goetz for this quote!

I reuse this quote (from 6.3) here. It is so apt.

In 'Competitive Engineering' I have offered 100 such principles [Gilb, [1], URL20]. I have 'brainstormed' many more, in other books and papers, including this one. I am sure many other people will also continue to develop principles, that deserve to be taught formally.

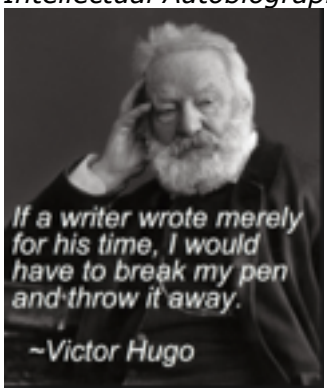
My concern is that we place far too little emphasis on acquiring and teaching any real principles at all. My concern is that students do not even get a dozen good principles, to base their professional work on.

I think we need a course, called something like "The Most Important Fundamental Planning Principles".

Quote 10.9 C

> ***"Change your opinions,***
> ***keep to your principles;***
> ***change your leaves,***
> ***keep intact your roots."***

— Victor Hugo (1802-1885),
Intellectual Autobiography: Ideas on Literature, Philosophy and Religion



I agree Victor!

Practical Tip 10.9. Priority to Long-Lasting ideas.

- *If you are going to learn a new idea, ask yourself whether it would have worked for the last hundred years, and might well work the next hundred.*

- *If not, there may be better things to spend your time on.*

Policy 10.9 Teach The Classics

- **In university education, and in Business Corporate Training, we will try to teach powerful, fundamental, eternal principles.**

- Why?
  - Better value for teaching effort
  - More effective for the planner to know if they can apply them

Version 31 Dec. 2015, 02:43 CET, TsG

Section 10.10 What Professors Should *Really* Teach: Clear, Deep, Useful Sets Of Professional Concepts.

Principle 10.10 Clarity is a Tool

Clear concepts,
are powerful tools for reasoning.
Unclear concepts are a waste of everybody's time.

Quote 10.10 A

"Mystification is simple; clarity is the hardest thing of all."

— Julian Barnes (1946-), Flaubert's Parrot

Concepts versus Terms
Most planners we meet seem to use a personal, and seemingly accidental, definition of their planning concept words, like 'strategy'.

They seem to *not* be aware, that almost *all other* project participants would *define*, and thus 'interpret', that same term *differently*.

They seem to *not* have reasonably clear (in their own minds) and useful (for their work) concepts, of even the most fundamental planning concepts, such as 'requirement', 'design', 'architecture', 'measure', and 'quantify'.

And they rarely seem to have few identically-defined concepts in *common*.

Here is a starting point about *clarity*: Global and Local concept definition and clarity.

1. **Clear Global Concepts**.
It is an intellectual necessity, for a professional domain like planning,or decision-making, to have **clearly defined 'Global'** domain **concepts**:

This is a higher-level education *teaching* necessity, in our opinion. (Global: *375 [URL63 D])

2. **Precise Local Definitions**:
It is a practical necessity, in addition to Clear Global Concepts, to have in a given planning document ***precise pointers*** of one kind or another, **to *local definitions,*** of potentially unclear or ambiguous terms. (Local: *376 [URL63 D])

Both these notions, Global and Local definitions, are necessary for ease of writing, and for ease of precise shared understanding of a planning document.

'Clear Global Concepts', has the following *advantages*:

- ***Define once, re-use thousands.*** A 'culture' (like 'those who use Planguage', or 'those educated for a given MBA degree') can pre-define the several hundred, most-frequently needed, concepts in their domain.

- *This is a one-time effort. It can be instant and free if you adopt a known standard, such as Planguage. Or it can be carefully done in your domain or organization by mature experts, or teams.* This Global Concept Glossary can be improved over time.
- W*ith the global concepts already pre-defined, the individual planner does not have to re*-invent the famous wheel badly. And the 'plan reader' does not have to guess badly, each time they read or hear a term, either.
- With frequent long-term use of a pre-defined concept, and fairly standard pointers (like '**S**tretch **L**evel', written with Capital letters, or bold, or underlined etc. as a formal signal that it is a formally defined term) the plan reader, and writer, can be confident of understanding terms, *without* even looking terms up. Only the novice or outsider will need to look up the term's concept definition.

Unfortunately, there is 'no such thing', as far as I can observe, as a *'global concept definition set'* in use, for most all management planning we encounter in practice. I am sure some associations have had a stab at making one, but I do not see them being used in the planning world I wander in.

I am sure there are local exceptions, including Planguage, but the normal planning document we see internationally has no such Global Concept glossary in practical use. But I do not see more than a very small fraction of all real planning. But I do work worldwide in many disciplines, so I see a lot of the practice, which is a pretty good sample.

People 'just use' the same widely-used terms like 'alignment', 'mission', 'strategic' for example, with no known reference to any standard definitions, even locally in the corporation.

Nobody can know and be sure what those terms mean. And I suspect that even the plan specification writers, do not know *either*, until proven otherwise. They are just 'parroting' conventional terms. Cut and paste terminology. No deeper responsibility for the precise shared meaning.

The second notion, '**Local Concept definitions**, is a logical necessity for tailoring a plan to current local needs.

Planguage is heavily oriented towards extensive local tailoring of concepts in a plan, exploiting the initial starting basis of about 700 Global Concepts already-defined. [URL64, 'Agile Planguage' paper].

Practical Tip: 10.10 A. Steal My Glossary, Please.

You can, freely, without license or permission, take into use our Planguage Concept Glossary, now, and modify it to your needs when you like. [URL63, especially A].

Academics and Corporations, be my guest!

Nice, if you credit the Source!

Nice if you share <u>your</u> version, or translation, with others, as freely as I do.

Practical Tip: 10.10 B. The 'Ambiguity Test'.

As a consultant, I need to dramatically demonstrate to my clients, at meetings, that they really do ***not*** understand key terms in plans and presentations, *in exactly the same way.*

So, I will pick a term, like 'extremely productive'; hopefully one on which some discussion of meaning has just happened, at the meeting.

I will also judge, or ask participants, if it is critical, to have a clear agreed definition, of the term. Usually they will acknowledge that it is critical.

I then ask all participants to write down *their* interpretation of the term.

It is important that this is in the context of that particular plan, and that particular organization. Some definitions are very local!

I then ask people to read their interpretations aloud, and I key them in, showing them on a screen, for all to see.

What is 'guaranteed' in advance, is that no two definitions are exactly alike.

What is also a pretty sure expectation, is that there is no 'official' written, and agreed, definition of 'extremely', or 'productive' in the plan, or elsewhere.

There is often a local plan-document glossary; but it fails to cover most all of the ambiguous terms!

No one can be bothered to define *all* words used in a plan. But we need to avoid costly misunderstanding, by using more up-front effort to make definitions available, Global and Local.

The measured 'density of ambiguous terms' is still about 30% of all words used, according to our frequent (SQC) measurements of real plans internationally. This is the agreed opinion of top managers, who we often use to decide what is 'unclear for purpose' ! [URL1, URL44, URL45, URL46]

Another piece of fun, is when the client's or director's suggested definition is very different from everybody else's.

It always is. So when collecting opinions from the group, I suggest you can save that surprise for last. Ask the boss last.

By the way, do not be surprised if the 'boss' decides they like someone else's definition, better then their own, once they see it.

A useful follow-up exercise, if you have time (30 more minutes) is to try to reach agreement, there and then, on one common critical definition.

A good start to this, agreement exercise, is to vote for the 'best draft'. Then take that definition, and add to it some of the ideas from most of the other suggestions by other people.

When everybody gets their favourite idea in the official definition, they are generally happy to accept the revised definition.

The most important thing, in these definitions, is NOT to get the 'right' definition. There is no 'right' or universally correct definition.

The main point is to get a stable, agreed definition. One that people will respect. And which they can change systematically, if it is later shown to be defective.

This 'ambiguity test' can hopefully result in management agreement to 'get more formal' about defining critical terms much better.

I find that it works dramatically to that effect.

Practical 'clarity' tactics, such as Rules about clarity, Specification QC, Defect Exit levels, as discussed in this book, might hopefully follow. They are practical answers to the definition problem.

I have a backup plan for the unlikely event that two people ever give me *exactly* ('not one word different') the same definition suggestion.

I will in that case ask them to define one-or-more words in their 'identical words' definition. I do not *then* expect to get identical answers. But if that *ever* does happen, maybe you can guess my 'Plan B'?

The point emerges. *We all have different interpretations.* But we can organize things, so can we work together, using common agreed definitions of critical terms.

If the term is critical to a plan, there is only one reasonable solution. We need to have a *formal written definition*, in the plan, or a link to it.

That will be the 'right' definition for practical purposes. Everything can be improved later, but we will then improve the common definition itself, for the benefit of the team.

This has legal and contractual implications, of course. But the worst scenario is the cost of lawyers, billing you at great cost, discussing the interpretation of words, which you could have clarified using 3 minutes of work, a year ago.

Maybe if you had to pay the lawyer's fees out of your *own* pocket, you might be motivated to take the trouble to define plan concepts more formally?

Figure 10.10 A . The defects report from the 4 managers on a single page of project plan, in Case 10.10 below.

Case 10.10 at Major USA Multinational in Cincinnati Ohio. [URL91]

This is a case where we were asked to convince sceptical managers that they should do quality control of their project requirements or other plans.

We used 4 Program managers, checking one page for 30 minutes. And another 4 managers checking another 1-page sample of an 82-page document.

The counted ambiguous and unclear concepts in a page of Project objectives.

One 'team' of 4 found about 60 'unique' Major defects in the page, which (we have learned from experience) is about one-third of the defects *actually there* 'right now'. So there are quite probably about 180 major defects (3 x 60) in the page of about 300 words.

It does not matter much if there were 30 or 230 defects, since one major defect is *too many*: enough to cause a disaster in the project. We are talking about things like 'security' here. How many knife-stabs to your heart are 'too many'?

The page in the example (82 of 82, defects reported as in Figure 10.10 A) was about 'quality requirements', including 'security' (example of what was actually there was the *strategy 'use a password'*).

The project was, as a consequence, predictably (we did the prediction correctly before learning the actual result) *two-years delayed*.

This is equivalent to (40,000 hours), of which a 1-year delay, half of that, had *already been experienced,* the managers informed me.

Here is how we arrived at this prediction of project delay.

The average major defects found in the 2 pages sampled was 150.

150 average majors/Page x 82 pages total = 12,300 major defects total in the project objectives plan.

Roughly every third Major Defect will (our experience shows) cause a downstream time loss of about 10 hours [F13].

So 12,300 total defects / 3 = 4,100 loss-causing defects..

So 4,100 x 10 hours lost each = 41,000 hours downstream loss.

That is about 2 years delay, with the number of people working on the project.

They believed me *only* because they had *already* lost the first year (20,000 hours of work), fighting downstream problems, and knew they were only halfway through the process of getting ready for acceptable product release.

Planners cannot even seem to get 'conceptual clarity' about *ends* and *means*, since they constantly declare some 'means' to be their 'ends'.

- The 'use a password' requirement, in case 10.10 above, instead of a 'Security Objective' definition [URL1B, URL66] is a practical example of this mix-up.

Planners normally mix strategies, directly, into their statement of objectives.

- *"We are going to get much better productivity through the Magic Method X".*

And sometimes, they even forget to mention which objective they are addressing, as in the example about 'password' above.

I have personally given up hoping, that people can agree on the meaning of ordinary words.

It is not the *words themselves* that are critical. Any agreed 'pointer symbol' will do, to find an agreed definition.

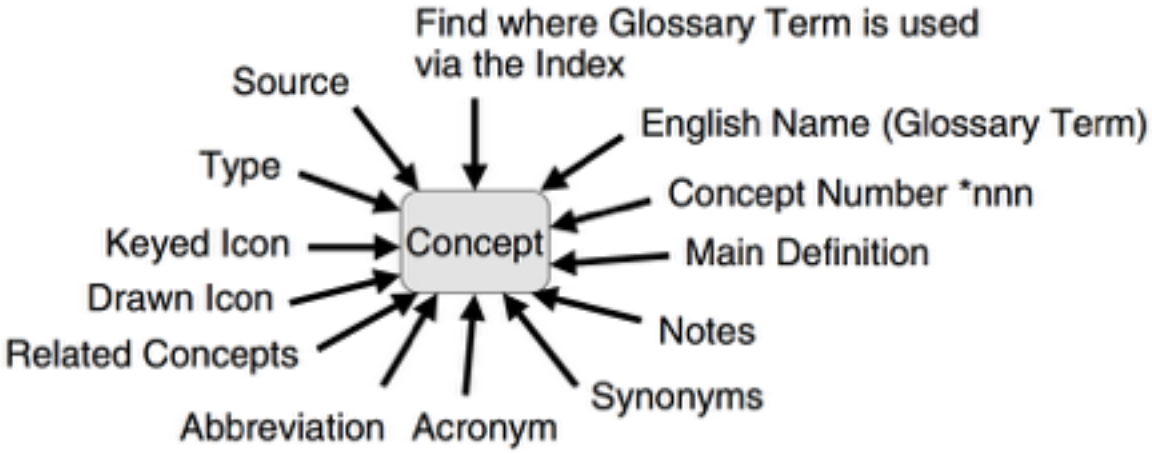
Warring about words is fruitless time-wasting.

It is the '*concept* definitions' that we need to learn, not the 'word' or 'name' pointer definitions.

We can then *assign* words or symbols to the concepts, in order to reference them (see Fig. 10.10 B below).

We can specify anytime and multiple times, the particular set of terms we choose to use, to reference these concept definitions.

*Figure 10.10 B. I advocate **detaching the concept definition** itself, from a wide variety of devices (concept pointers) which can be used to **reference** the concept [1, CE page 323]. These concept pointers can be words (in any and all languages), synonyms, numbers, and visual symbols (Icons) [URL34].*



My primary concerns are that:

- *We do not have a **rich enough** set of planning concepts.*
- *For example: We need to distinguish between **many types** of requirements, **many types** of designs, **many types** of constraints – and much more. See Figure 10.10 C below, for the various types of requirements.*
- *We use 'words' in plans and about the planning process with no **agreed** meaning, as if we believed that other people will know exactly what we mean. They will not.*

Goal **Planguage Concept: *109**

A Goal is a primary numeric Target-level of Performance.

An implication of a Goal specification is that there is, or will be, a *commitment* to deliver the Goal level (something not true of a Stretch or Wish target specification).

Any commitment is based on a trade-off process, against other targets, and considering any constraints.

The specified Goal level may need to go through a series of changes, as circumstances alter and are taken into consideration.

A specified Goal level will *reasonably satisfy* stakeholders.

Going *beyond* a Goal level, at the cost of additional resources, is not considered necessary or profitable – even though it may have *some* value to do so.

A Goal parameter is used to specify a performance target for a scalar attribute.

*Planguage 10.10 An example of a concept definition: Planguage Term *109, also called 'Goal' in the CE book [1]. But, called other synonyms in other cultures.[URL63 D]*

Defining concept number '*109'.

The term 'Goal' is assigned, in order to cross-reference concept *109, for the purposes of the CE book, *only* [1, CE, page 366, URL63 D].

Anyone else can assign any other term or symbol they want to, in order to reference concept *109. For example, we can assign the terms "Commitment', 'Main Objective', 'Promised Level', 'Dead-lined Target' to *109.

But they should somehow 'clearly connect' their 'synonyms' for *109 to the *109 concept definition.

One Planguage tool we use to connect to a concept Glossary, is to write the terms (all words of it) in **C**apital **L**etters, so as to declare or signal, that it is a *formally defined* concept. So, look it up.

If you want to be certain of the exact meaning of Capitalized Words, you need to look at either the Concept Glossary or somewhere 'local' in your plan.

It follows logically, that in one plan, there should not be more than one unique such-defined term.

Hierarchical Tags, and other tagging practices for clarity.

It is a Planguage rule that if we need to distinguish such potentially ambiguous terms (same Tag, more-than-one concept definition), we can prefix the term with a 'Hierarchical Tag' (Planguage concept *146). We can for example refer to

Productivity. Goal [Asia]

Or

CE. Planguage or **Intel.**Planguage

We can also directly add a Tag to any element of a Planguage specification, to give it a Unique cross reference.

Unique Tag 123: **Stretch** [Asia, 2025] 60%.

I personally prefer to **bold** and underline the tag of any primary specification (**Unique Tag 123**), so the eye quickly catches the main definition site.

One client of ours writes all tags in a wiki format (for example WordOne, WordTwo) so that they are automatically made into a cross-referenced link everywhere else in the plan.

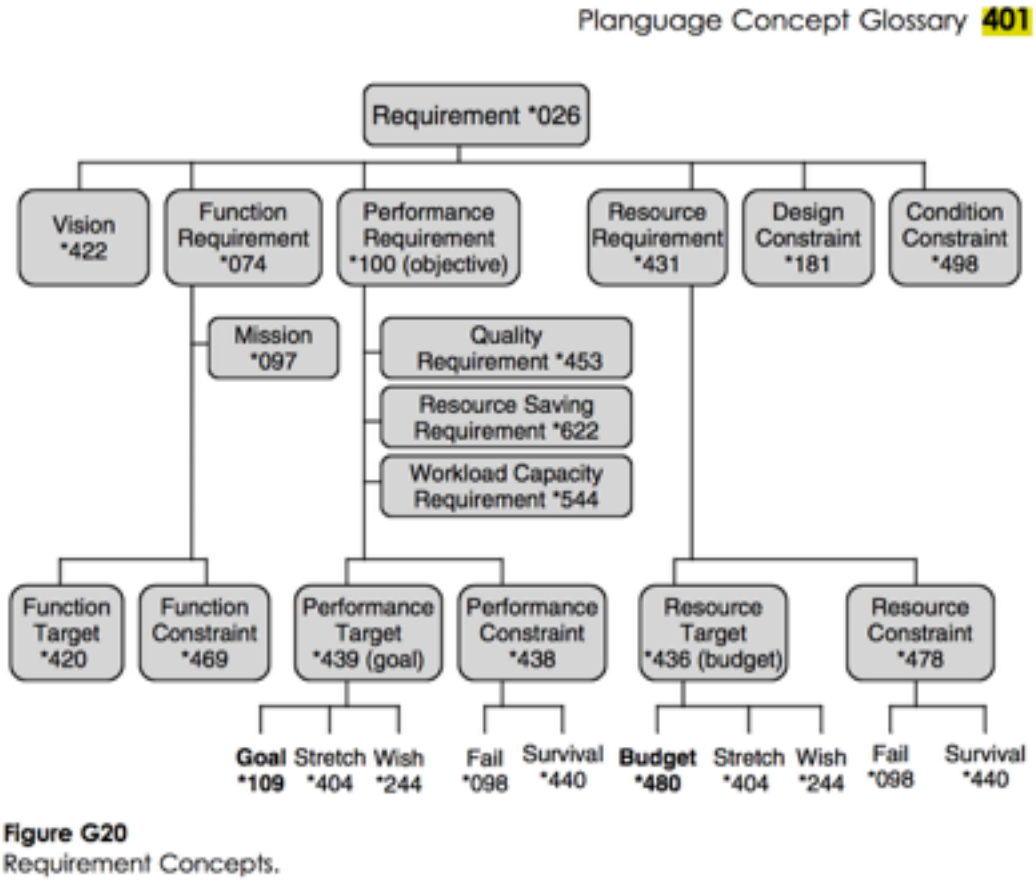


Figure 10.10 C. A hierarchy of related concepts in Planguage. [Figure G20 in URL63 D]. There are many very different types of requirements.

Suggestions to planners and planning teachers and managers.
I would suggest that we need to define our key concepts much better.

Then, we need to teach a decent set of the concepts systematically.

We need to teach them in the context of practical processes for planning.

We also need to teach people to know, *when* they need to define their local use of terms, in their specifications.

- Hint: the first time anyone asks what it means. *Write* your answer! Later: re-use it!

Quote 10.10 B.

> ***"It's up to the artist to use language that can be understood, not hide it in some private code.***
>
> ***Most of these jokers don't even want to use language you and I know or can learn . . . they would rather sneer at us and be smug, because we 'fail' to see what they are driving at.***
>
> ***If indeed they are driving at anything--obscurity is usually the refuge of incompetence."***

— Robert A. Heinlein (1907-1988) Science Writer, Stranger in a Strange Land

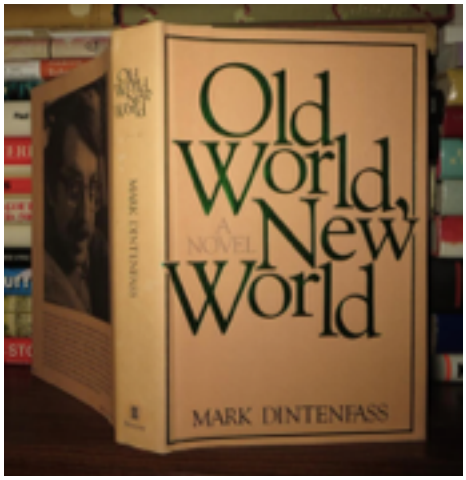
Practical Tip 10.10 C. Insist on written definitions

- *If you are not given written formal definitions, for words, in plans, consider asking for them: and if you can, consider adding definitions to the plan, yourself.*

Quote 10.10 C

> ***"...it is precisely because the world appears to us to be multiple, ambiguous, and paradoxical, that we must strive to speak and write clearly."***

— Mark Dintenfass, Author



Policy 10.10 Avoid Any Critical Plan Misunderstanding

We will make sure that all terms, critical to a plan, will have formal definitions.

We will either use a corporate planning-term glossary, a Planguage Glossary *[URL34 B]* ***and/or specify definitions in the plan itself.***

Why?

- *Because it is cheap to do, and expensive to have your plans misunderstood.*
- *A misunderstood plan, is a worthless plan.*
- *Do you really want people to 'guess' what you mean?*

Section Version 31 December 2015 03:20 CET, TsG
Wow I managed to complete my editing of the 2014 Manuscript during 2015!
Now there is all the other hard work to do on the book! Tom

———————————————————

This is the end of the 100 Sections, and their 10 Chapters Appendix.

If you got this far please let me know, and what you think

at

Tom@Gilb.com

Literature References:

1. Tom **Gilb**: "**Competitive Engineering**" **2005**

- A. The book
- Gilb, Tom, Competitive Engineering, A Handbook For Systems Engineering, Requirements Engineering, and Software Engineering Using Planguage, ISBN 0750665076, 2005, Publisher: Elsevier Butterworth-Heinemann. Sample chapters will be found at Gilb.com.

B. Chapter 5: Scales of Measure:
http://www.gilb.com/tiki-download_file.php?fileId=26

C. Chapter 10: Evolutionary Project Management:
http://www.gilb.com//tiki-download_file.php?fileId=77

D. Chapter 8, Specification Quality Control
For a summary see URL23.

E. For Extensive technical detail see Gilb& Graham, Software Inspection, 1993, Book [14]

F. Planguage Rules Collection from all chapters
http://www.gilb.com/dl829

2. William and Ken **Hopper**: 'The Puritan Gift'
A book that documents the history of American management practices, and shows the bad influence of Business Schools.

3. Ralph **Keeney**

: **Value-Focussed Decision Making**

A. Keeney, Ralph L. 1992. **Value-focused Thinking: A Path to Creative Decision-making**. Cambridge, MA and London: Harvard University Press. ISBN 0-674-93197-1. http://www.fuqua.duke.edu/faculty/alpha/keeney.htm/

B. A Summary of Keeney's Book

- http://lesswrong.com/lw/gdq/valuefocussed_thinking_a_chapterbychapter_summary/

C. Keeney Paper of Chapter
Developing Objectives and Attributes
http://www.usc.edu/dept/create/assets/001/50847.pdf

D. After some debate amongst friends, we arrive at the conclusion that Keeney was inconsistent and unclear about his use of terms Fundamental Objectives, and Strategic Objectives. We are going with the terminology we understood from his book, rather than D. link above. Even direct email with Keeney did not result in explanations that we we sure of interpretation. But the basic structure, we have understood, or misunderstood, seems useful anyway. Referenced in 1.3 and other places.

4. Jack Welch, Jack, **Straight from the Gut**, Warner Business Books, 2001.

5. T. Gilb, **Principles of Software Engineering Managemen**t,

- 1988
  - Internet Chapters (text only, no illustrations):
  - A. pdf 'Ch 14 POSEM Productivity' gilb.com/dl560
  - B. pdf 'Ch 15 POSEM Deeper Perspectives on Evolutionary delivery gilb.com/dl561
  - C. Chapter 13.4 (page 237-241) Open Ended Architecture
  - D. Chapter 21 ICL Inspection Experiences (cited F8.2)
  - E. 282-3, 20, 168 The Aspect Engine and Goal automated tools for Planguage developed by Lech Krzanik. lech.Krzanik@oulu.fi.(cited 1.1)
  - F. Risk Principles (ref. 10.10) pages 72-76, for example
    - "If you don't actively attack the risks, they will actively attack you." (p. 73)

6. **Dan Ariely, The Upside of Irrationality. The unexpected benefits of defying logic at work and at home**
ISBN 978 0 00 735478-8, UK £8.99, 298 pages, Harper 2010

7. Gelb, Michael: **How to Think Like Leonardo Da Vinci**

8. Tom Peters
Reinventing Work, the project 50.
Quoted in 5.3
Alfred A. Knopf, New York, 2000, ISBN 0-375-40773-1. See Peters' website www.tompeters.com, $15.95
See also his book 'the Quick Prototype50'.
See especially his emphasis on 'quick prototyping

9. **W Edwards Deming, Out of the Crisis**

http://mitpress.mit.edu/books/out-crisis

10. **Tom Peters,**
'In Search of Excellence', and other books
tompeters.com/
early books referred to in 5.5, see [8]

11. **Michael Gelb, How to Think Like Leonardo da Vinci:**
http://michaelgelb.com/products/

12: **Simon Ramo** (quote in 6.5)
Simon Ramo and Robin K. St.Clair, The Systems Approach: Fresh Solutions to Complex Civil Problems Through Combining Science and Practical Common Sense, 1998, 150pp, Ó TRW, Inc., Manufactured in USA, KNI Incorporated, Anaheim CA. Free copy at TRW Stand at INCOSE conference 2002.
Free Download: www.incose.org/ProductsPubs/DOCSystems Approach.pdf

13. **Robin Olds, FIGHTER PILOT: The Memoirs of the Legendary Ace.**
386 PAGES Amazon $22.02
ISBN 978-0-312-56951-8
St Martins Griffin, NY, Stmartins.com
Quoted in F3, and 7.3 on grass roots delegation.

14. Gilb& Graham, **Software Inspection**, 1993, Book

15. **Joseph M. Juran**
- **Juran's Quality Handbook: The Complete Guide to Performance Excellence 6/e** Hardcover – June 9, 2010
- by Joseph Defeo (Author), J.M. Juran (Author)
- Cited 7.8?

16. **Walter A. Shewhart**
- A. Economic Control of Quality of Manufactured Product, 1931
- B. Statistical Method from the Viewpoint of Quality Control, 1939
    - http://books.google.no/books?id=5utN7ulkCCAC

17. **Craig Kaplan et al Case IBM STL**
- Secrets of Software Quality by Kaplan, Clark, & Tang (McGraw-Hill 1995)
- See URL48

18. **Jevons, The Principles of Science**, 1874

http://ia600406.us.archive.org/27/items/principlesofscie00jevoiala/principlesofscie00jevoiala.pdf

- Free download, not in copyright.

http://en.wikipedia.org/wiki/William_Stanley_Jevons

- The machine on the left side of the cover photo is Jevon's Piano, or Jevon's Logic machine. (Sounds like a computer to me!)

http://history-computer.com/ModernComputer/thinkers/Jevons.html

19. **Peter Drucker**

"Management: Tasks, Responsibilities, Practices."

Quote 6.10

http://www.druckerinstitute.com/2013/07/measurement-myopia/

20. Billy Koen

- Koen, Billy V. 1984. **Toward a Definition of the Engineering Method.** Proceedings of the ASEE-IEEE Frontiers in Education. 14th Annual Conference, Philadelphia, PA. 3-5. October 1984. Pages 544–549. The paper also appeared in Engineering Education. December 1984. Pages 150–155. Also in Spring 1985 in The Bent of Tau Beta Pi. Pages 28–33. A full page extract is in Gilb (1988, Principles of Software Engineering Management). An extended and very interesting comment on the paper's ideas is in Koen (2003).

  Profession Website; (2014)
  http://faculty.engr.utexas.edu/koen

-
- Koen, Billy Vaughn. January 2003. **Discussion of the Method: Conducting the Engineer's Approach to Problem Solving**. Oxford University Press. ISBN 0-195-15599-8. Pages 260. http://www.me.utexas.edu/faculty/people/ koen.shtml/.

*

- Video Presentations 2007 Illinois Lecture

http://faculty.engr.utexas.edu/koen/etc-lecture

  - (2014 STABLE SITE)

21. Richard Rumelt, **Good Strategy Bad Strategy**: The Difference and Why It Matters, 2011

22. Ron Radice, **High Quality Low Cost Inspections**, 2001. Ron's Website www.stt.com.

Historical Note: Ron Radice invented the Capability Maturity Model at IBM (see IBM Systems Journal) and credited Tom Gilb with " much of what is" 'Level 4' (quantifying quality). He was my client at IBM while inventing CMM under his boss Watts Humphrey (who wrote the Foreword to our Software Inspection book, 1993), who took CMM to Software Engineering Institute (SEI at CMU) and made it a US DoD Standard from there. Level 3 of CMM was initially modelled on IBM Software Inspections. Interestingly Robert Mays worked for Radice and invented the Defect Prevention Process (DPP) which was the basis for Level 5. Level 2 was based on the Deming (Quote 10.4) SPC statistically stable state of process before improvement.

23. Larry Miles, ***Techniques for Value Analysis and Engineering,*** *1993, 3rd Edition. Google books epub $1.99*

See many Value Analysis Downloads at
http://www.valuefoundation.org/Innovate-Research.htm

See Miles Value Foundation created 1977:
facebook.com/valuefoundation

see URL81 (SAVE) an organization based on Miles ideas.

See especially Chapter 17-1 FAST Function Analysis System Technique.Using the determination logic questions 'How?', 'Why?' and 'When?'. A useful hierarchy of concerns method.

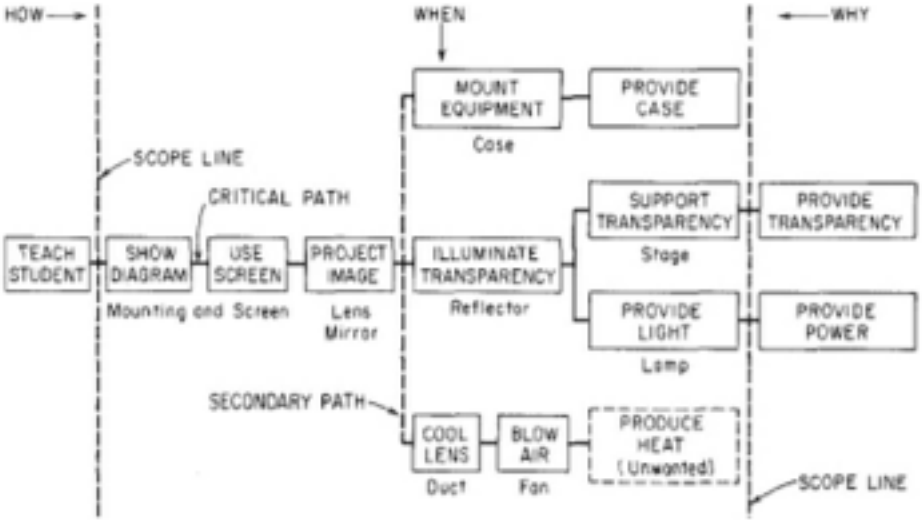


Diagram from Chapter 17-1 of Miles book.

Also 17-2 ‘Quantitative Evaluation of ideas?’
which recognizes the engineering limitations of non-numeric methods such as FAST.

24. **Built To Last**: Successful Habits of Visionary Companies,
By James C. Collins and Jerry I. Porras. Harper Business Essentials, 1994
ISBN 0-06-051640-2, $10.10, paperback, amazon.com

24. Alice Calaprice, **The Expanded Quotable Einstein.**
A. The statement “Things should be as simple as possible but no simpler”
is in the sections on ‘things people think Einstein said, but there is no proof or documentation from the research libraries to prove it’. I made the mistake of crediting AE in my PoSEM book [5]. I have since done quite a bit of personal research on this quote.
B. A good analysis of this quote is quoteinvestigator.com , May 13, 2011 (you can search by month and year. It is there August 2015.
C. I went as far as an email exchange with Marvin Minsky, who knew AE personally and took long conversational walks with him. He could not prove it either. But he said ‘I can not be sure because of his heavy german accent’ !

25. Drucker, P. F., **Management: Tasks, Responsibilities, Practices**, Harper & Roe, New York, 1973. Quotation Chapter 5 Beginning.
Picked up via J. Kasser Holistic Thinking: Creating innovative solutions to complex problems. Second Edition.

URL1: **Quantifying Management Bullshit**
http://www.coremag.eu/fileadmin/Papers/Quantifying_Management_TGilb_core3.pdf

version 18 april 2011
Quantifying Management Bullshit: forcing IT Stakeholders to reveal the value they really want from your IT Project.
By Tom Gilb

RQNG 25 May 2011
Quantifying Management Bull: Forcing IT stakeholders to reveal the value they really want from your IT project

http://www.gilb.com/dl465
CORE Version

URL2: "**Vision Engineering". Paper for top management planners.**
http://www.gilb.com/tiki-download_file.php?fileId=237
This has been substantially rewritten and added to this book as Part 0 to 5.

URL3: **An Agile Project Week: 'Evo Start'**
Our Column in AgileRecord.com, as published 7 March 2013
http://www.gilb.com/dl568

URL4:

A. The Agile Evo Project Startup Week Standard
http://www.gilb.com/dl562

This is a detailed standard for conducting an
'Evo' (Evolutionary Project Management, Gilb's Agile Method)
as described in my book Competitive Engineering, Chapter 10
http://www.gilb.com//tiki-download_file.php?fileId=77

B. Evo Project Management Standard, Jan 12 2013
http://www.gilb.com/dl563

See also [URL5]

URL5: **ONE WEEK STARTUP PLANNING** FOR PROJECTS; FRONT END TO EVO

- **A: An Agile Project Startup Week: Papers and slides**
  - Talk **slides** pdf from ACCU Conference, Bristol UK, April 9 2014

- 90 minutes talk. Includes Startup Planning for Business Startups, Confirmit, US DoD case, 2 Bank cases, Detailed Startup week outlines and links to sources.

http://www.gilb.com/dl812

-
- see also [URL4]

- B. See Persinscom Case (cited in URL16 too)
  - **"111111 Unity Method of Decomposition into weekly increments of value delivery"**. (10 min. talk slides)

http://www.gilb.com/tiki-download_file.php?fileId=451

- Includes Persinscom case US DoD.
-

- C. **Software Plans in Less Than a Week**
-

http://www.dtic.mil/ndia/2004cmmi/CMMIT7Tue/MelissaOlson.pdf

- Melissa Olson
  Raytheon Company McKinney, TX
  972-952-4502 Melissa_olson@raytheon.com Abstract #1196

D. An Agile Project Startup Week, same as URL3

Gilb's Mythodology Column
www.gilb.com/dl568

URL6: **No Cure No Pay Contracting**

A. Agile Contracting for Results The Next Level of Agile Project Management: Gilb's Mythodology Column, in Agile Record August 2013.
http://www.gilb.com//dl581
see in that respect
www.flexiblecontracts.com

B. No Cure Slides
http://www.gilb.com/tiki-download_file.php?fileId=85

C. NO Cure Paper
http://www.gilb.com/tiki-download_file.php?fileId=38

D. Contracting For Value. Slides 2015
http://www.gilb.com/dl864

URL7: **Robustness decomposition**

2013 Lightning Talk 15 min. At ACCU Conference 2013
Robustness, Softcrafters, User Stories
http://tinyurl.com/bnurqec

URL8: **Top Ten Critical Objectives and Vision Engineering**

A. Gilb's Mythodology Column

The **Top 10 Critical Requirements** are the Most Agile Way to Run Agile Projects
http://www.gilb.com/dl554

B. **Vision Engineering** paper
http://www.gilb.com/tiki-download_file.php?fileId=237
This is also URL2
Rewritten as Chapter 11 of this book.

URL9. **Quantifying the Unquantifiable** Trondheim,

- 17.5 minutes. "Quantifying the Unquantifiable", TEDx Talk, Video, 2013
  http://tinyurl.com/GilbTedx
  http://www.youtube.com/watch?v=kOfK6rSLVTA&list=PLylSVHkZybrxemH8D1D7zdq2Dmn06nh7m&index=5

Sept 22 2011 Primary Control Dashboard slides
http://www.gilb.com/tiki-download_file.php?fileId=483

Setting and Tracking Project Objectives (BCS Dec 2010 Talk)
http://www.gilb.com/tiki-download_file.php?fileId=455

Top level Critical Project Objectives ROOTS Version.pptx
http://www.gilb.com/tiki-download_file.php?fileId=180

Dashboard Talk JPM Feb 2010
http://www.gilb.com/tiki-download_file.php?fileId=385
(obsolete see
http://www.gilb.com/tiki-download_file.php?fileId=483)

URL10: **Quality Quantification.**

A. TEDx Talk, **Quantifying the Unquantifiable**
Trondheim 2013 Tom Gilb 17.5 minutes
http://www.youtube.com/watch?v=kOfK6rSLVTA&list=PLylSVHkZybrxemH8D1D7zdq2Dmn06nh7m&index=5
http://tinyurl.com/GilbTedx

B. **Quality Quantification** Paper
http://www.gilb.com/tiki-download_file.php?fileId=124

C. SLIDES: **Quantifying Quality: How to Tackle Quantification of the Critical Quality aspects for Projects for Both Requirements and Designs**, 2011 version
http://www.gilb.com/tiki-download_file.php?fileId=486

D. Handbook: Competitive Engineering, Chapter 5 **Scales of Measure.**
http://www.gilb.com/tiki-download_file.php?fileId=26

URL11: **Estimation**
T. Gilb, 'Estimation: A Paradigm Shift Toward Dynamic Design-to-Cost and Radical Management. http://www.gilb.com/tiki-download_file.php?fileId=460
Software Quality Professional, VOL. 13, NO. 2/© 2011, ASQ

- Slides made for BCS SPA June 1 2011
'Estimation, a Waste of Time'
http://www.gilb.com/tiki-download_file.php?fileId=470

URL12: **QUINNAN AND MILLS CLEANROOM**
http://www.gilb.com/dl821
is contained in these slides.

URL13: **DEEPER PERSPECTIVES ON EVO DELIVERY**
Chapter 15 in (1988) Principles of Software Engineering management
www.gilb.com/dl561
"Deeper Perspectives on Evolutionary Delivery"
plus a page extra of quotations from Agile Gurus crediting it as inspiration for them, and it being first.

URL14: "**POWER TO THE PROGRAMMERS"** TALK SLIDES AND VIDEO
Note Paul Klipp has transcribed this and published my talk in a book. 21.7.14 mail hpps://leanpub.com/ACE2014

http://www.gilb.com/dl821
(Slides)
Power to The Programmers, as held Krakow ACE Conference June 2014
Video: http://vimeo.com/98733453
EDIT NOTE; POLAND HAS ASKED FOR A PAPER ON THIS, NOT WRITTEN AS OF 3 AUG

URL15: **Twelve Tough Questions**

12 tough questions paper
http://www.gilb.com/tiki-download_file.php?fileId=24
A summary of 12 Tough Questions was published in 2005 in the Competitive Engineering book.
See Section 1.5 for a 1 page articulation.

URL16: **Decomposition of strategies by Value**

A. "Decomposition of Projects - How to design small incremental result steps", 2008 Paper
http://www.gilb.com/tiki-download_file.php?fileId=41

The 20 decomposition principles alone, from CE [1) Ch. 10 on Evo are in the book at Figure 5.6 A

B. "The Unity Method of Decomposition"
Column 2 of Gilb's Mythodology
in Agile Record
http://www.gilb.com/dl826

C. "111111 Unity Method of Decomposition into weekly increments of value delivery". (10 min. talk slides)
http://www.gilb.com/tiki-download_file.php?fileId=451
Includes Persinscom case US DoD.

URL17: Competitive Engineering, Chapter 5 **Scales of measure**
http://www.gilb.com/tiki-download_file.php?fileId=26

URL18: Competitive Engineering, Chapter 10: **Evolutionary Project Management:**
http://www.gilb.com//tiki-download_file.php?fileId=77

URL19: **Keeney**
A **Summary** of Keeney's Book
http://lesswrong.com/lw/gdq/valuefocussed_thinking_a_chapterbychapter_summary/
B. **Slides** for Kenney
Gilb: Productivity Slides incl Ericsson, and Keeney
http://www.gilb.com/dl559

URL20: **Gilb Principles collection**
http://www.gilb.com/tiki-download_file.php?fileId=352

Competitive Engineering book Principles Gilb and Others 8MB.

Extended Collection, **Undergraduate Basics (Paper)** Principles, Gilb's Datamation **Laws of Unreliability**, ICL **Bill of Rights**, Demarco and Gilb's Law of Measurability, Risk Principles, Clinical Risks Slides, GILB'S INTERPRETATION OF **ERICSSONS CORPORATE QUALITY POLICY**, **Decomposition Principles** (from CE 10), **12 Tough Questions**,

URL21: **Impact Estimation**

Impact Estimation Table MASTER.ppt (8.49 Mb)
You can download this file using: http://www.gilb.com/tiki-download_file.php?fileId=146

Design Evaluation Paper
http://www.gilb.com/tiki-download_file.php?fileId=58

See **IE Table Chapter in CE Book** [1]

Impact Estimation Tutorial MASTER 2012_compressed.pdf
http://www.gilb.com/dl553

Impact Estimation Tables
Understanding Complex Technology Quantitatively
http://www.crosstalkonline.org/storage/issue-archives/1998/199812/199812-Gilb.pdf
Crosstalk, US DoD, December 1998.

URL22. **Intel Specification Quality Control Experience + Planguage**

"The Impact of a Requirements Specification on Software Defects and Other Quality Indicators"
John Terzakis
Intel Corporation
August 31, 2011 RE 11 Conference Trento, Italy
http://selab.fbk.eu/re11_download/industry/Terzakis.pdf

URL23: **Specification Quality Control**

"Agile Specification Quality Control:
Shifting emphasis from cleanup to sampling defects", in Testing Experience, March 2009
http://www.gilb.com/tiki-download_file.php?fileId=264

See also Chapter on decomposition in CE [1] book.
See also URL44 (McDonnell-Douglas, Boeing, Philips cases)
See Intel case, URL45

- The Impact of a Requirements Specification on Software Defects and Other Quality Indicators by john.terzakis@intel.com

URL24: **Human Values**

- hayyalee.com/wp-content/uploads/2010/04/List-of-Personal Values

URL25: http://www.thinkatheist.com/profile/PeytonDracco

URL26: **Modelling and Tailoring Using Planguage**

A. **Agile Aspects of Planguage for Cost-Effective Engineering**
By Tom Gilb and Lindsey Brodie
Agile Record Issue 05
January 2011
www.Gilb.comdl39
www.agilerecord.com

Modelling:

B. **"What is drastically wrong with most software engineering modelling languages and approaches, and 10 necessary principles for a really good modelling language"**
http://www.gilb.com//dl795

45 minute lecture, pdf format, 62 slides
Quality Days Conference, Vienna
10:45 to 11:30 , Wednesday January 15 2014

C. Link to 13January 2014 Vienna Quality Days workshop slides on Modelling
http://tinyurl.com/QWGILB

"How to Model Qualitative aspects of software requirements, software architecture, and project progress – quantitatively" (3.5 hours)

URL27 **Raytheon Case.**

A. Ray Dion et al
"Raytheon Electronic Systems Experience in Software Process Improvement"
TR-017
MAY 2014 new URL site
http://resources.sei.cmu.edu/asset_files/TechnicalReport/1995_005_001_16415.pdf
See Appendix B here for a dozen related papers.
Such as
"Process Improvement and the Corporate Balance Sheet," IEEE Software, July 1993

and
"Measuring the ROI of Software Process Improvement," DACS Fourth Annual Software Quality Workshop, August 3, 1992

- **B. Experiences in Root Cause Analysis and Defect Prevention Methods, slides**
    - Kelly L. Lanier Raytheon Network Centric Systems klanier@raytheon.com www.dtic.mil/ndia/2004cmmi/CMMIT5Tue/ **Root_CauseAnalysis**.pdf (slides, from which ill 7.3 made)
      Presented at http://www.dtic.mil/ndia/2004cmmi/2004cmmi.html

URL28: **Confirmit** Case Study.

A: The Green Week Slides
http://www.gilb.com/dl660
Smidig/Agile Conference 2013 Oslo
Nov 5 2013

B: The GREEN WEEK- Agile Technical Debt Engineering beats Refactoring.
http://vimeo.com/78635151
Agile Oslo 2013, Video 10 minutes.
IN NORWEGIAN, English slides

C: Gilb, co author Trond Johansen **Confirmit case** paper
http://www.gilb.com/tiki-download_file.php?fileId=32

D: Value Driven Project Management
17.5MB *slides* 2008
Includes Firm Case
http://www.gilb.com/tiki-download_file.php?fileId=152

E. 'What's Wrong With Agile Methods? Some Principles And Values To Encourage Quantification' with Confirmit Case.
http://www.gilb.com/dl50

URL29: **Lean Startup Compared to the Evo method**

- **Lean Startup** March 24 2014 Krakow ALE Slides Dec 6 2011.pptx.pdf

http://www.gilb.com/dl805

URL30: **Tse and Kahlon: Planguage Case NHS**

Source Oct 2013 Paper

How Planguage Measurement Metrics Shapes System Quality

Man-Chie Tse[1,2] **& Ravinder Singh Kahlon** [1,2]
{Man-Chie, Ravi}@dkode.co

Title of 2013 SQM paper:
the principles and application of Planguage for Managing System Innovation
http://www.gilb.com/dl582

Great real pharmaceutical hospital case with real improvements.
Slides http://www.gilb.com/dl583

[1] dkode Limited, London, United Kingdom.
[2] University of Ulster, Ulster Business School, Northern Ireland, United Kingdom.

Abstract: It is known for innovative IT projects in the public sector healthcare within the UK to fail and disappoint. The announcement of National Programme for IT (NPfIT) is an example, at a cost of £12.7 billion that ended. The problem of IT projects failing have a destructive impact on wasting resources and at the socio-economic cost to the tax payers.
The aim of this paper is to improve the understanding and inter-relationship qualities between people, process and technology. A quality healthcare innovation project was to consign new improved services and processes to be more efficient and to deliver value to the stakeholders in a competitive sector.

I emailed them 2 aug 2014 for updated references. THEY REPLIED THEYWILL GET BACK WITH UPDATES

•

URL31 **HP Evo**

A. The Evolutionary Development Model for Software
by Elaine L. May and Barbara A. Zimmer
August 1996 Hewlett-Packard Journal
http://www.gilb.com/tiki-download_file.php?fileId=67

B. Evolutionary Fusion: A Customer- Oriented Incremental Life Cycle for Fusion
by Todd A
http://www.gilb.com/tiki-download_file.php?fileId=35

August 1996 Hewlett-Packard Journal

C. RAPID AND FLEXIBLE PRODUCT DEVELOPMENT: AN ANALYSIS OF SOFTWARE PROJECTS AT HEWLETT PACKARD AND AGILENT (2001)
by
Sharma Upadhyayula
http://www.gilb.com/tiki-download_file.php?fileId=65

M.S., Computer Engineering University of South Carolina, 1991
And
Massachusetts Institute of Technology
January 2001

D. Best Practices for Evolutionary Software Development
by
Darren Bronson
http://www.gilb.com/dl825

57 pages., 1999.

URI: http//hdl.handle.net/1721.1/80490

URL32: **The Loo Watt Startup Case**
http://www.gilb.com/dl792
The Evo method with cases Loowatt, Confirmit, US DoD
For 1 hour talk #BuildstuffLT Dec 10 2013
pdf version

Project Website
www.Loowatt.com

URL33 **PRIORITIZATION (SEE CHAPTER 6)**

A. Choice and Priority Using Planguage:
A wide variety of specification devices and analytical tools.
Copyright © 2006 by Tom Gilb.
http://www.gilb.com/tiki-download_file.php?fileId=48

B. Managing Priorities:
A Key to Systematic Decision-Making
Tom Gilb Tom@Gilb.com Mark W. Maier Mark.w.maier@aero.org

Copyright © 2005 Tom Gilb and Mark Maier. Used by Permission of the authors by INCOSE.
http://www.gilb.com/tiki-download_file.php?fileId=60

C. Dynamic Design Prioritization in the 'Evo' Agile Framework for Scrum or other Iterative Methods
http://www.gilb.com/dl602
Slides up pdf 122 slides, for London Software Architect Conference Oct 9 2013

URL34: **Planguage Icons**

A. Plicons: A Graphic Planning Language for Systems Engineering
Copyright © 2006 by Tom Gilb. .
http://www.gilb.com/tiki-download_file.php?fileId=37

See also most chapters of Competitive Engineering Book [1]

B. For information about Planguage Icons. See the Glossary in the book and more extensively the extended Planguage Glossary
CE Full Glossary
http://www.gilb.com/tiki-download_file.php?fileId=386

A private, unpublished, even on internet, file of all proposed keyed icons, is available from the author on request.

URL35: **QFD** , see also Brodie research PhD [URL75)
Gilb & Brodie (2008) **"How problems with Quality Function Deployment's (QFD's) House of Quality (HoQ) can be addressed by applying some concepts of Impact Estimation (IE)"**
http://www.gilb.com/tiki-download_file.php?fileId=119

URL36: **Bring Case** and more: Hierarchical Impact Estimation Tables
http://www.gilb.com/dl500

"Value-Driven Development: Principles and Values."
Slides for , 50 minute talk, Software Passion Conference 20 March 2012 Gothenburg, Sweden

Value Management (Evo) with Scrum development, March 2010 English Version , Kai Gilb
www.gilb.com/tiki-download_file.php?fileId=277

The Inmates are running the asylum, Construx Summit talk Oct 25 2011 Seattle
Contains considerable Bring Case slides
www.gilb.com/tiki-download_file.php?fileId=488

Norwegian Version, Bring Case
www.gilb.com/tiki-download_file.php?fileId=279

URL37: **How Good Is A Process?**
Evaluating Engineering Processes' Efficiency.
http://www.gilb.com/tiki-download_file.php?fileId=51
INCOSE Conference 2006. Tom Gilb

URL38: Lindsey Brodie, Research Paper

- **"Towards Understanding Multi-dimensional Stakeholder Value Using Absolute Scale Metrics"**
- Lindsey Brodie and Mark Woodman School of Engineering and Information Sciences
- Middlesex University London, UK L.Brodie@mdx.ac.uk
- PUT URL HERE, request sent 14.8.15

URL38: **Ramo.** same as reference to book [12] since it is both a book and a download.

- Simon Ramo (quote in 6.5)
- Simon Ramo and Robin K. St.Clair, The Systems Approach: Fresh Solutions to Complex Civil Problems Through Combining Science and Practical Common Sense, 1998, 150pp, Ó TRW, Inc., Manufactured in USA, KNI Incorporated, Anaheim CA. Free copy at TRW Stand at INCOSE conference 2002.
- Free Download: www.incose.org/ProductsPubs/DOCSystems Approach.pdf

URL39: **Ericsson Management Policy**

- Including Quality Policy and Risk Policy (CP 7.1 and earlier)

http://www.ericsson.com/res/thecompany/docs/comp_facts/how-we-manage-our%20business.pdf

URL40: **Defect Prevention Process**

R Mays et al, IBM Systems Journal 1/1990, **"Experiences with Defect Prevention Process"**, DPP

http://www.gilb.com/tiki-download_file.php?fileId=457

http://agileconsortium.pbworks.com/f/Mays1990ExperiencesDefectPreventionIBMSysJ.pdf

SEI CMM Level 5
http://www.sei.cmu.edu/reports/93tr025.pdf

Also see Chapter 7 and 17 in Tom Gilb, Software Inspection, Book (1993) which are on the DPP method. 17 is by Mays, as above.

See also URL77 (KAN IBM)

URL41 **Cleanroom**

A. Mills, H. 1980. The management of software engineering: part 1: principles of software engineering. *IBM Systems Journal* 19, issue 4 (Dec.):414-420.
Direct Copy
http://trace.tennessee.edu/cgi/viewcontent.cgi?article=1004&context=utk_harlan
Library header
http://trace.tennessee.edu/utk_harlan/5/

B. Mills, Harlan D.; Dyer, M.; and Linger, R. C., "Cleanroom Software Engineering" (1987). The Harlan D. Mills Collection. http://trace.tennessee.edu/utk_harlan/18

C. Mills Generally
http://trace.tennessee.edu/utk_harlan

D. see URL12 for Gilb slides on this.

URL42: **PDSA Plan Do Study Act The Deming Cycle**

Original Deming Lecture to Top Management 1950 Japan:
http://deming-network.org/deming_1950.htm

Evolution of the PDSA Cycle:
http://www.cologic.nu/files/evolution_of_the_pdsa_cycle.pdf

URL43: **Undergraduate Basics paper. See 10.9 and 10.10**

A. Gilb, Tom, Undergraduate Basics for Systems Engineering (SE), using The Principles, Measures, Concepts and Processes of Planguage. (INCOSE Conference 2007 Presentation)
http://www.gilb.com/tiki-download_file.php?fileId=98

My thesis here is that we need to educate in basics, such as principles, measurement, and concepts: to prepare people for unknown and rapid change decision making.

B. Paper presented at
Academic Forum
INCOSE International Symposium 2007, San Diego
Monday, 25 June 2007
www.incose.org/symp2007/documents/AcademicForum.doc.
Otherwise never published.

C. A talk June 2015 Based on this paper: KSSE Conference Kongsberg Norway
http://ksee.no/wp-content/uploads/2015/06/Some-Principles-of-Useful-Knowledge-1738-4-June-GILB.pdf

URL44: **The McDonnell-Douglas Case Study** of **SQC and Engineering Improvement**
Case DAC Inspection 1988-89 . ROI 4.5:1
Contains also Philips (UK, detail in [14] as case study) and Boeing SQC client cases. We did Boeing (Renton) after McDonnell-Douglas success and recommendation.
www.gilb.com/dl254
See also [URL23, 1 D]

URL45: **Intel Report on SQC** (Gilb methods used here <- E Simmons)
The Impact of a Requirements Specification on Software Defects and Other Quality Indicators by john.terzakis@intel.com
http://selab.fbk.eu/re11_download/industry/Terzakis.pdf

URL46: **Intel Experience with Planguage and SQC 2011**
- Erik Simmons, Intel, 2011, 21st -Century Requirements Engineering: A Pragmatic Guide to Best Practices, Erik Simmons PNSQC 2011 (Pacific Northwest Software Quality Conference) http://www.uploads.pnsqc.org/2011/slides/Simmons_21st_Century_Requirements_slides.pdf

URL47: **Statistical Process Control SPC**
A. Florac, SPC for Software, 1999 SEI, PDF).
http://www.sei.cmu.edu/str/descriptions/spc_body.html
CHECK THIS LINK AS MAYBE OLD, SEI HAS CHANGED

URL48 **Secrets of Software Quality, IBM STL Study**
Slides

Secrets of Software Quality
http:\\www.iqco.com **exists2014**

Software Quality Week
Craig Kaplan, Ph.D.
ckaplan@iqco.com **(I emailed aug 2014)**
I.Q. Company
SeeBook citation [17]
Cited in Chapter 7.8

URL49: **ICL Case in 8.2**

A. ICL as Wilmot took over, near collapse
http://www.chilton-computing.org.uk/acl/associates/companies/wilmot.htm

B. Wilmott
http://www.zoominfo.com/p/Robb-Wilmot/247803694

C. Gilb, ICL Case. BCS June 12 Lecture 2015
Slides are at
http://www.gilb.com/dl846

URL50: **Jussi Aromäki**

Excellent Paper , quoted from in 8.3
Tips for successful communication with suppliers
by Jussi Aromäki on Apr 06, 2011
https://www.ictstandard.org/article/2011-04-06/Tips-for-successful-communication-with-suppliers
and his linkedin
https://www.linkedin.com/pub/jussi-aromäki/1/a13/8b0

URL51: **ICL - Group Quality**

- **INTRODUCTION TO QUALITY METRICS (Issue 2)**
- Pamphlet
  - Largely authored by Tom Gilb
  - Official Corporate wide use of Planguage at ICL
  - Referenced in 8.6

http://www.gilb.com/dl828

URL52: 25 Ways to Reward Employees (Without Spending a Dime)

- By **Dan Tynan**

http://www.hrworld.com/features/25-employee-rewards/
Referenced in 8.5

URL53: A. **Practical Purposeful Creativity**. Paper, T Gilb

http://www.gilb.com/tiki-download_file.php?fileId=22
This paper is written as an invited contribution to a book "Creativity, Innovation and Cooperation" (Springer) and a special issue of "AI & Society: the Journal of Human-Centred Systems and machine Intelligence". The editor is Robert C. Muller Published around 1992.

B. **PLANGUAGE AND INNOVATION: How Metrics Drive Ideas**
http://gilb.com/dl847
Published 26 June 2015 as slides at Gilb Seminar, London
Based on the paper above (A.)-

URL54: **User Stories**

A. User Stories paper by Tom and Kai Gilb
In Gilbs' Mythodology Column, Agilerecord.com March 2011
http://www.gilb.com/tiki-download_file.php?fileId=461
"User stories and the conversations provoked by them comprise *verbal communication*, which is clearer than written communication." (Mike Cohn via Denning)
B. Stephen. Denning, The Leaders Guide to Radical Management.
http://stevedenning.typepad.com

URL55: **Murphy's Laws, Gilb's Laws, Gilb's Competitive Engineering 100 Principles**

A. **Murphy's Laws, History**
and big informal collection
http://www.murphys-laws.com

A2. http://en.wikipedia.org/wiki/Murphy's_law is better and focussed
Both give the history.
A3. http://en.wikipedia.org/wiki/Edward_A._Murphy,_Jr.
is the personal history of Ed Murphy, Jr.
A4. http://people.howstuffworks.com/murphys-law1.htm
is the best condensed story of the origin as far as Murphy is concerned.
A5. http://www.daytoninnovationlegacy.org/murphy.html
is another with a photo of Ed Murphy jr, and
●**"If there's more than one way to do a job, and one of those ways will result in disaster, then somebody will do it that way."** This was Ed Murphy's own wording according to his son (*People* magazine, January 1983).

B. **Gilb's Laws and Principles**:
by the way, at http://www.murphys-laws.com/murphy/murphy-technology.html

the Law
"Computers are unreliable, but humans are even more unreliable."
And "Any system which depends on human reliability is unreliable."
Are two of Gilb's Laws from March 1975 Datamation, paper, "The Laws of Unreliability".
A complete set, Gilb's Laws will be found at
http://www.gilb.com/tiki-download_file.php?fileId=336
and a larger summary of my principles will be found at
Gilb 100 Planguage Principles and friends (Gilb's Laws and 1st photo page of original Datamation paper)
http://www.gilb.com/dl352

•

URL56. Peanut Butter Statistics used in 9.6
http://jwilson.coe.uga.edu/emt668/EMAT6680.Folders/Brooks/6690stuff/Statistics/Statistics.html

URL57: Planguage Rules from CE Book [1]
Planguage Rules Collection from all chapters
http://www.gilb.com/dl829

URL58: Mountain Rules
http://mylittlenorway.com/2011/05/the-mountain-code/

URL59: Car Safety Factor
http://www.agro.basf.com/agr/AP-Internet/en/content/competences/health_and_nature/safety_first/index

URL60: Economic Safety Factor
http://www.dineshbakshi.com/images/break%20even%20chart.gif

URL61: Safety Factor Theory
http://en.wikipedia.org/wiki/Factor_of_safety

URL62: Balanced Scorecard
A. http://en.wikipedia.org/wiki/Balanced_scorecard
B. Gilb Slides 'What is Wrong with Balanced Scorecard'. slides
http://www.gilb.com/tiki-download_file.php?fileId=135

URL63: **Planguage Glossary**
A. the extended Planguage Glossary, Tom Gilb's personal version **'CE Full Glossary',** This is updated at Gilb.com periodically
http://www.gilb.com/tiki-download_file.php?fileId=386
B. The **Competitive Engineering** book Glossary, and the whole book outside the glossary as a source of defining Planguage terms.
http://www.gilb.com//dl540

- edit note I am not sure I want to keep this full copy here, but I'll leave it for the moment. In any case if I do, it might be reference (1). But it is one good way to get access to the detailed discussion of the concepts! It may even stimulate paper sales ? A discussion with LB and Publisher. But I have the rights to it all now. TG 130914
- See the CE Handbook keyword index, for treatment and examples in the entire handbook

Paper copies available via Amazon etc.

C. **Gilb.com, Resources, Concept Glossary**

This is a voluntary team project, led by Kai, and supported byour BCS course participants, to extract the basic definitions and make them available on the live web. Easy to access in a meeting!

D. **Planguage Concept Glossary as edited in Competitive Engineering book** 2005 (10% of Full Glossary)
http://www.gilb.com/tiki-download_file.php?fileId=387

E**. 'A Conceptual Glossary for Systems Engineering: Define the Concept, don't quibble about the terms.**'
Paper on the Concept Glossary: 2003-2006, Held at INCOSE San Diego 2007
http://www.gilb.com/tiki-download_file.php?fileId=46

URL64: **Agile Planguage**. Referenced in 10.10

'Agile Aspects of Planguage for Cost-Effective Engineering', by T. Gilb & L. Brodie
Published in Agile Record magazine, January 2011
http://www.gilb.com/tiki-download_file.php?fileId=39
A description of the adaptability of the Planning Language to various circumstances and needs.

URL65 **Planguage Philosophy. Not referenced yet 13 sept 2014**

Planguage [Philosophy]: Background depth for teachers, consultants and the curious (from 1995)
http://www.gilb.com/dl822

URL66: **Security Definition**

- **'How to Quantify Security'** by Tom Gilb. Ref. 10.10

http://www.gilb.com/tiki-download_file.php?fileId=40

URL67: CONTINUOUS IMPROVEMENT edit, I need to insert this somewhere

- A. SEE ALSO URL40 DPP
- B. Masaaki Imai - Definition of KAIZEN
- https://www.youtube.com/watch?v=jRdTFis4-3Q
  - Courtesy of Venkatesh Krishnamurthy

http://agileworld.blogspot.com

o

URL68: Kai's Tool for Planguage

A. Kai Gilb's Google Docs, Excel Spreadsheet tool

Reference to be supplied

In the meantime send info requests to Kai at Gilb . com

B. Kai Gilb's New Tool developed for Statoil 2014.

Reference to be supplied

URL69: Helmuth Karl Bernhard Graf[1] von Moltke

http://en.wikipedia.org/wiki/Helmuth_von_Moltke_the_Elder

I found a detailed read of his history and thinking to be well worth the time invested.

URL70: Planguage.

'Planguage: an 'engineering' language and process for real software and Systems engineering - not 'programming'

http://www.gilb.com/dl831

Oct 18 2014 for

Book Chapter: for "Software Engineering in the Systems Context" Lawson/Jacobson eds.

URL71: Managing the Realization of Business Benefits From IT investments

Joe Peppard

John Ward Cranfield University

Elizabeth Daniel Open University

http://www.som.cranfield.ac.uk/som/dinamic-content/research/documents/peppardwarddaniel07.pdf

- TO DO: **REFERENCE THIS AND USE THE PRINCIPLES AND DEPENDENCY DIAGRAM NOV 6 2014**

URL72: Einstein Quotation

http://books.google.no/books?id=dlYEAAAAMBAJ&pg=PA62&source=gbs_toc_r&redir_esc=y#v=onepage&q&f=false

Quoted by William Miller in Life Magazine, May 2, 1955

The 'Time' photo is also published at this link, a 1947 Favorite of Eienstein.

URL73: Richard Smith's Planguage Tool:

http://needsandmeans.com

There have been several tools supporting Planguage. This tool by Richard Smith

rsmith@rsbatechnology.co.uk

is emerging in 2015. We have used it on training courses in 2015. I am impressed by it capabilities and ease of use. It is very helpful on the *dynamic prioritization* methods described in this book in Section 6.1 and on.

See also Smith's Case Studies URL 82.

URL74: Priority Papers by Gilb
'**Managing Priority**' by Tom Gilb and Mark Maier
http://www.gilb.com/dl60

Choice and Priority Using Planguage:
A wide variety of specification devices and analytical tools.
By Tom Gilb
http://www.gilb.com/dl48

URL75:
Lindsey Brodie, PhD, 2015
Title: "**Impact Estimation: IT Priority Decisions**"
Middlesex University online publications.

Temporary site until available at Middlesex U.
https://www.dropbox.com/s/iik9wafmzwpttsq/Lindsey%20Brodie%20PhD%20Thesis%20
FINAL%20V7.pdf?dl=0

URL76: Designing Maintainability and Adaptability into Systems

A. Gilb ACCU Designing Maintainability, slides. 90 minutes
http://www.gilb.com/tiki-download_file.php?fileId=171

B. Maintainability Paper: 2008 INCOSE Conference.

'Designing Maintainability in Software Engineering: a Quantified Approach.'

By Tom Gilb

http://www.gilb.com/tiki-download_file.php?fileId=138

C. Chapter 5 CE Book: Scales of Measure:
http://www.gilb.com/tiki-download_file.php?fileId=26
see the chapter for specific scales of measure suggestions.

D. se also URL28 Green Week at Confirmit.

URL77: "Software Quality: An overview from the perspective of total quality management"
Steve Kan, Victor Basili, L. N. Shaprio, IBM SJ, 1994, Vol. 33, No. 1
REFERENCED IN 7.3 the 2167 changes (this might be in the below ref. it is not found here)

52. **S. H.** Kan, "Modeling and Software Development Quality," ***IBM Systems Journal*** **30,** No. 3, 351-362 (1991).

2162 DPP Actions implemented

- between Dec. 91 and May 1993 (30 months)<-Kan
- See URL40 on DPP.

need a download link
I have a copy in a file 'Kan.pdf' filed in Lean QA DPP course documentation
check maybe this quote is in Mays paper??

URL78 Leading and Lagging Indicators illustrations and concepts.
Source: http://www.risktec.co.uk/media/62859/fig%201%20-%20high%20res%20compressed_529x401.jpg

Another very interesting diagram with rich commentary on safety considerations needing leading indicators is found at
https://www.energyinst.org/technical/human-and-organisational-factors/human-factors-performance-indicators
"Following the Texas City refinery accident in 2005, increasing attention has been directed at performance indicators for process safety, with particular emphasis placed on the need for organisations to adopt 'leading' indicators. In contrast to 'lagging' indicators (which are used to monitor incidents and accidents themselves) leading indicators are used to monitor the precursors to potential incidents and accidents, and provide an early warning for management to act before problems become serious."

URL79 Richard Feynman
http://www.haveabit.com/feynman/2
Names

URL80: **Alberto Savoia, Pretotype IT**
The Book in Pdf. 2nd Edition.

http://www.pretotyping.org/uploads/1/4/0/9/14099067/pretotype_it_2nd_pretotype_edition-2.pdf

pretotyping.org

asavoia@gmail.com, author email.

URL81: SAVE International. Value Engineering.
http://www.value-eng.org

Referenced in 10.6 Function.

SAVE defines Function as “what the product or service is supposed to do”.

So does Planguage.

SAVE has as a core idea that the Function should be described by a 2 word statement, and should be the real core function without any inadvertent constraints. For example a digital projector ‘projects image’. Their theory being that if you get the core function, you can more easily creatively find alternative, perhaps cheaper technologies, for doing that core function.

On the other hand they define

“Value = Function / Cost”

they state this in more detail as:
“ Value is the reliable performance of functions to meet customer needs at the lowest overall cost..”

and here they clearly have included all value giving things, like qualities (PL: how well it functions) in the concept of Function. We feel this is a bit messy.

Is ‘reliable performance’ the same as in Planguage ‘meeting the Goals of the critical set of Performance Attributes (Values)’ ?

In addition, ‘Value’ is not a function of ‘cost’. Costs might be hidden from some or all stakeholders, who will have a view of the values of the product or service anyway.

Of course we need a better definition of cost (manufacture, acquire, use, money/time/other resources ?). But even then the stakeholder perception of value is initially independent of costs. Though at some point some costs may be factored in to get some kind of ‘net value’ or ‘profitability’ or ‘ROI’.

‘Value Analysis ‘came from GE 1940s as a response to wartime shortages. Book: Larry Miles, Techniques for Value Analysis and Engineering [23]. Google books epub $1.99
See Miles Value Foundation created 1977:
facebook.com/valuefoundation.
and see many Value Analysis Downloads at
See many Value Analysis Downloads at
http://www.valuefoundation.org/Innovate-Research.htm

Especially
http://www.valuefoundation.org/paper01.pdf
Integrating Lean with Value Engineering Donald E. Parker, PE, CCE, CVS, FSAVE

Which defines ‘Lean’ like this:
1. Specify value
2. Identify the value stream
3. Flow
4. Pull
5. Perfection

Answers to the interpretation of the SAVE ideas are thoroughly discussed in the Miles book [23]. See especially Chapter 17-1 FAST Function Analysis System Technique. Using the determination logic questions ‘How?’, ‘Why?’ and ‘When?’. A useful hierarchy of concerns method.

URL82: Richard Smith’s Case Studies:

A. Citigroup: Mainly CE book review followed by a short case.

http://rsbatechnology.co.uk/blog:8

Referenced in Section 1.2: notice the Objectives did not change in the 14 month project.

B. Japanese Bank, ‘Price Sentinel’ Case.

http://rsbatechnology.wdfiles.com/local--files/site-files/Price%20Sentinel%20Planguage%20Case%20Study%20v1.pdf

There are BCS Presentation slides for this: (ask RS)

C. uk.linkedin.com/pub/richard-smith/

D. Gilb slides on Smith’s Citigroup Experiences A above:

“Experience in Multinational Banks

with Advanced Stakeholder Value Requirements: Quantified for Testability and Tracking

Our talk May 16 2012 London Testing and Finance Conference, London

http://www.gilb.com/dl532

Smith is slide 8 to 15.

E. See also Smith's Tool, URL73

URL83: **EIFFEL TOWER:** (Used in Section 2.2)

https://en.wikipedia.org/wiki/List_of_the_72_names_on_the_Eiffel_Tower

Gustave Eiffel engraved on the tower seventy-two names of French scientists, engineers, and mathematicians in recognition of their contributions. Eiffel chose this "invocation of science" because of his concern over the artists' protests against the tower. This engraving was painted over at the beginning of the twentieth century but restored in 1986–1987 by the Société Nouvelle d'exploitation de la Tour Eiffel, a company contracted to operate business related to the Tower.

https://en.wikipedia.org/wiki/Eiffel_Tower

"Eiffel had a permit for the tower to stand for 20 years; it was to be dismantled in 1909, when its ownership would revert to the City of Paris. The City had planned to tear it down (part of the **original contest rules for designing a tower was that it should be easy to demolish**) but as the tower **proved valuable for communication purposes it was allowed to remain after the expiry of the permit.** Eiffel made use of his apartment at the top level of the tower to carry out meteorological observations, and also made use of the tower to perform experiments on the action of air resistance on falling bodies." [Case 2.4]

"Maintenance of the tower includes applying 50 to 60 tonnes (49 to 59 long tons; 55 to 66 short tons) of paint every seven years to protect it from rust. The height of the Eiffel Tower varies by 15 cm (5.9 in) due to temperature." [Case 2.4]

1967: According to interviews, Montreal Mayor Jean Drapeau negotiated a secret agreement with Charles de Gaulle for the tower to be dismantled and temporarily relocated to Montreal to serve as a landmark and tourist attraction during Expo 67. The plan was allegedly vetoed by the company which operated the tower out of fear that the French government could refuse permission for the tower to be restored to its original location. [Case 2.4]

324 metres (1,063 ft) tall,[2] about the same height as an 81-storey building

URL84: [Picture 2.2 C] **This is the guarantee by Raigad Fort's Chief Architect / Engineer "Hiroji Indalkar"** engraved on the wall about the strength of the Raigad Fort.

http://raigad-fort.blogspot.no/2010/10/guaranty-by-hiroji-indulkar-raigad-fort.html

URL85: The Demise of the Iron Triangle
by Patrick Weaver
Nov 23, 2012
http://network.projectmanagers.net/m/blogpost?id=1606472%3ABlogPost%3A250097

URL86: Peter Drucker slides and photos and Ch. 5 quote
http://www.slideshare.net/nagz428/peter-drucker-252714

URL87: Making Complications Simple: using Planguage
http://www.gilb.com/dl854
30 Sept 2015

URL88: IBM LAMPS FSD Memoirs of a Defense Contractor/ Surviving a Career in the Defense Industry - T.H. Henning - Google .pdf
https://books.google.no/books?isbn=0595807623

URL89: The Space shuttle primary computer system
Communication the ACM, September 1984.
http://klabs.org/DEI/Processor/shuttle/shuttle_primary_computer_system.pdf

URL90:
The Logic of Design: Design Process Principles.
http://www.gilb.com/dl857
T Gilb
14 Oct 2015

URL91:
AGILE INSPECTIONS: Reviews by sampling and measuring defects.
Extreme inspection and reviews based on objective and quantitative review methods.

http://www.gilb.com/dl239

JUSE (Japanese Union of Scientists and Engineers, Tokyo, 2008, Invited Keynote, Slides.
Slides 3 to 21 is the GE Aero Engine case (identified there only as a major corporation). Cincinnati Ohio, referred to in Section 10.4, and Case 10.10

In this case, a selection of 8 managers sampling 2 of 82 pages of requirements in about 30 minutes, allowed me to estimate, correctly, that the project would lose 41,000 hours (2 years with 10 people) as a result of polluted requirements.

URL92:

F1. R e the Philips Evo delivery Illustrations

F2. RE Tesla Dealership Fight

- http://www.forbes.com/sites/jeffreydorfman/2014/03/22/free-markets-tesla-battles-car-dealers-over-right-to-sell-cars/
    - The GxxLine PXX Optimizer EVO team proudly presents the success of the Timing Prediction Improvement EVO steps.
    -
    - Shown are the results of the test set used to monitor the improvement process.
    -
    - The size of the test set has grown, as can be seen in the first column. (In the second column the week number is shown.)
    -
    - We measured the quality of the timing prediction in percentages, in which –5% means that the prediction by the optimizer is 5% too optimistic.
    -
    - Excellent quality (–5% to +10%) is given the color green, very good quality is yellow, good quality is orange, & the rest is red.
    - Eindhoven 2001
    - The results are for the ToXXXz X(i) and EXXX X(i), and are accomplished by thorough analysis of the machines, and appropriate adaptation of the software.
    -
    - The GXXline Optimiser Team presented the word document below to the Business Creation Process review team.
    -
    - The results were received with great applause. The graphics are based on the timing accuracy scale of measure that was defined with Jan verbakel. Classification: Unclassified
    -
    -
    -

F3: Fuller quote from Robin Olds, in book Fighter Pilot

- "Here's what I learned over the years.
- Know the mission, what is expected of you and your people.
- Get to know those people, their attitudes and expectations.
- Visit all the shops and sections.
- Ask questions. Don't be shy.
- Learn what each does, how the parts fit into the whole.
- Find out what supplies and equipment are lacking, what the workers need.

- To whom does each shop chief report? Does that officer really know the people under him, is he aware of their needs and training?
- Does that NCO supervise or just make out reports without checking the facts? Remember, those reports eventually come to you.
- Don't try to bullshit the troops, but make sure they know the buck stops with you, that you'll shoulder the blame when things go wrong.
- Correct without revenge or anger.
- Recognize accomplishment.
- Reward accordingly.
- Foster spirit through self –pride, not slogans, and never at the expense of another unit.
- It won't take you long, but only your genuine interest and concern, plus follow-up on your promises, will earn you respect. Out of that you gain loyalty and obedience. Your outfit will be a standout. But for God's sake don't ever try to be popular! That weakens your position, makes you vulnerable. Don't have favorites. That breeds resentment. Respect the talents of your people. Have the courage to delegate responsibility and give authority to go with it. Again, make clear to the troops you are the one who'll take the heat."
- There is a passage just above this
- " …respect, leaders like Dooey Spatz, Jimmy Doolittle, and many more. They all connected with their troops on a personal level and learned everything they could about every part of the organization"
-

F4: More from my source of Gruehl chart, NASA in 2.2 and 7.7
SOURCE IVY HOOKS AUG 2002, original SOURCE NASA

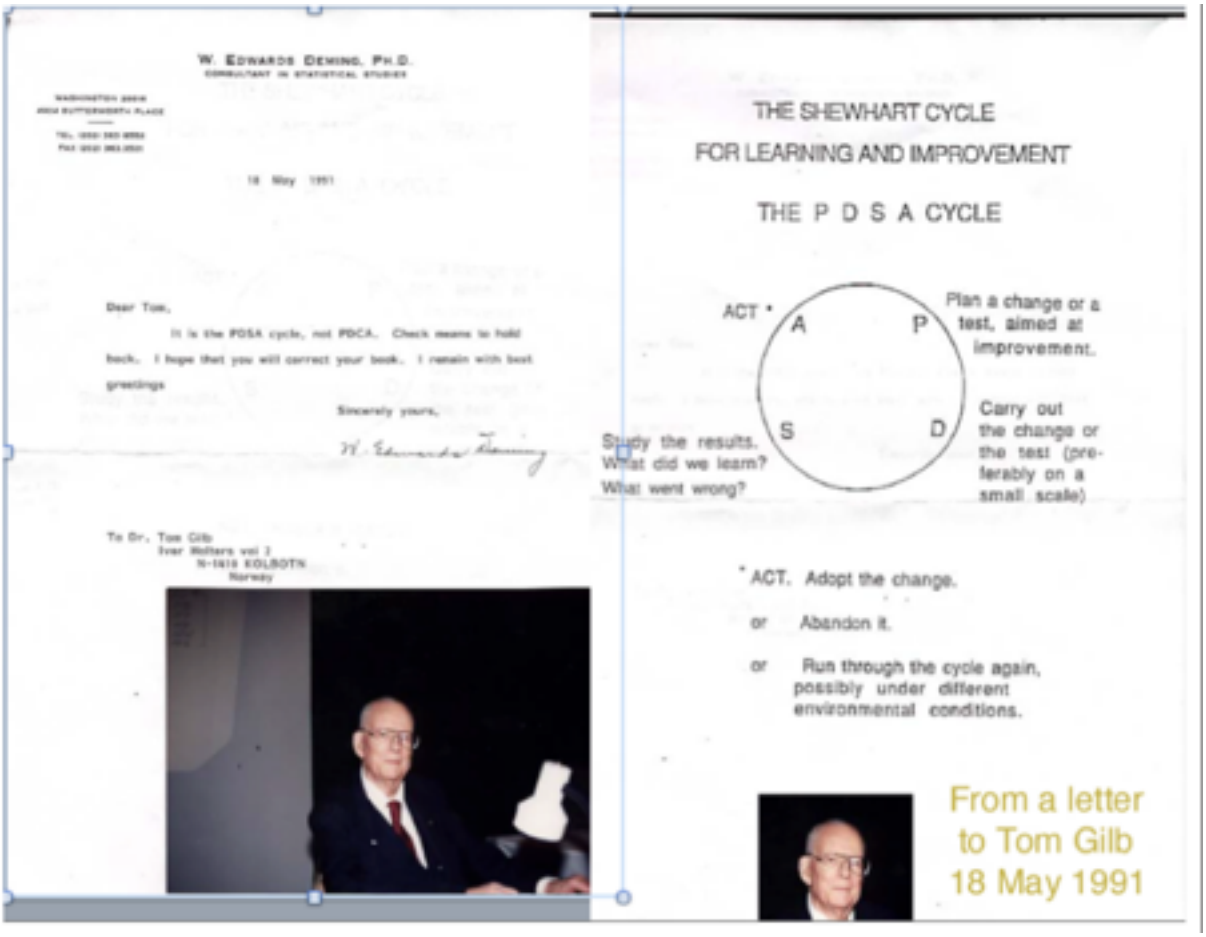
www.complianceautomation.com

Tom, "Here is my other story:
A client of mine had many data points on software projects that could be evaluated like the Gruehl chart. While there was a lot of scatter and some of the projects actually came in under cost and on schedule, an interesting pattern emerged. Overruns for upgrades were much more severe, twice as large or more, than for new projects. The manager who had the data contributed this to the problem of doing an upgrade assuming that it was simple, only to find major problems once the work had begun. This he attributed to lack of documentation of the existing system and lack of access to anyone who had built it, thus no knowledge of what was inside.
Cheers Ivy"

- *Customer-Centered Products, Creating Successful Products through Smart Requirements Management,* Ivy Hooks and Kristin A. Farry, Amacom, 2001

F5: Deming

- About 1984 in London, Tom had the pleasure of a week long class (with 600 other people) with Dr. Deming. I even managed to take him to the Leningrad Ballet, visiting London, after class was over! At the end of the week, I tried, for fun to see if I could summarize all his profound wisdom that week in a purely pictorial diagram. My diagram looked very much like the diagrams 7.8 A&B (Control charts, showing improved performance and reduced variance). "Is this the message? Is this the essence?" I asked him. I got a clear 'Yes' answer to that. Dr. Deming was quite minimalistic. For his one week class he had only one slide: his 14 points!



- The photographs were taken in our classroom, on that course. He was 84,and still going strong! His last lecture was done a week before he died peacefully ten year later. I liked his style: it was about sharing ideas, and making the world a better place. [URL42, re PDSA]

F6: Tesla

- **The author has been happily driving our Tesla Model S from December 2013**. What a delight! Musk keeps on giving extra guarantees (drive train, 8 years, Summer 2014) after the fact, free Superchargers, after the fact, monthly updates of the car driving, safety, entertainment etc. characteristics, constant improvement; after we paid for the car! And the car is initially 'perfect' anyway! He really practices this relentless improvement, not just for future products, but for 'old' customers.

F7. Wilmot and Books, and the half life of ideas and principles

- Wilmot remarked to me one *"I never read books, they are bound to be out of date, in these fast moving times" [F7] in 8.2*

- In 1960 my sister, Wendy Bartlett, arrived in Norway bearing a book by Jevons, The Principles of Science (Dover edition). She had it as a textbook from University of California, Berkeley. I thought it was strange to see the Steam Engines in the book, until I saw it was a reprint from 1874, earlier.

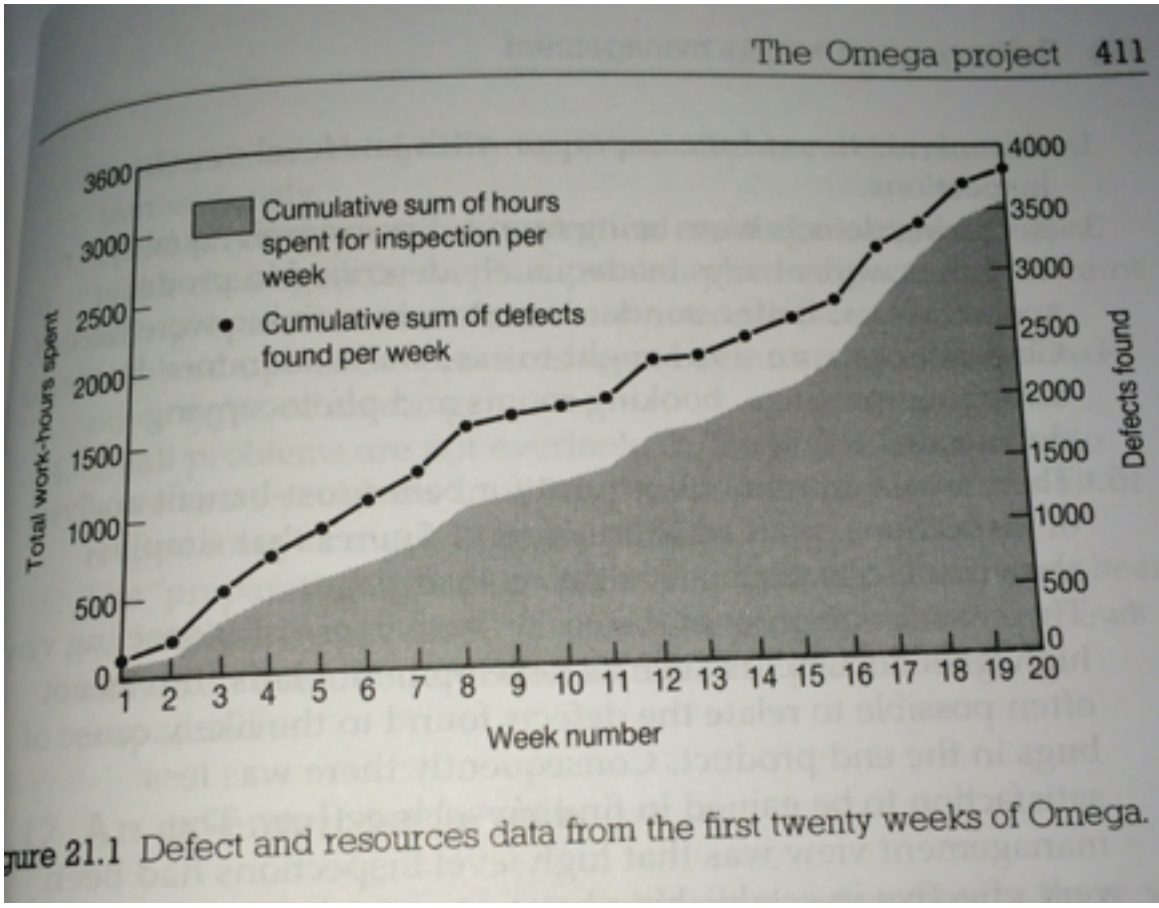
http://en.wikipedia.org/wiki/William_Stanley_Jevons
and

http://history-computer.com/ModernComputer/thinkers/Jevons.html

This inspired my future work and books. I keep on searching for principles, and other long term knowledge, that never went out of style. I hope the reader appreciates this; and realizing that the tools in this book, not least the Principles, are intended to wax eternal, and help the reader in the rest of their career and life.

F8: **ICL Inspection. Alan Brown**

- Chapter 21 in PoSEM book [5 D]



This is one example of the Grass Roots work I was asked to do by the top management of ICL. It is using the Specification Quality Control method (aka then as Inspection). They are removing about 400 defects, at a cost of 1 hour each. The downstream cost of removal, in testing or the field with customers is about 10x that.

F9: On the 'quotation', 'It is difficult to estimate correctly, especially about the future'. In 9.4

There are many variations and attributions of this.
http://chaosbook.blogspot.no/2010/06/lundskovdk-citater.html?m=1
Niels Bohr, and Yogi Berra among them.
Sources point back to Denmark and Norway in 1918
But just to be on the safe side, I wrote my own unique variation, using *estimate* and *correctly*, so that variant can be attributed to Tom Gilb, 2014. With a nod to Bohr and his Danish sources.

F10. Eleanor Roosevelt (1884-1962)

For more very pithy wisdom from ER.
http://www.goodreads.com/author/quotes/44566.Eleanor_Roosevelt

PHOTO SOURCE

http://en.wikipedia.org/wiki/Anna_Roosevelt_Halsted#mediaviewer/File:Eleanor_Roosevelt,_Anna_Roosevelt,_and_John_Boettiger,_Jr_-_NARA_-_195584.jpg

- I could not resist this photo, after choosing the wonderful E. R. quote. John Boettinger, Jr. is a friend of the author, as we met, as teenagers crossing the Atlantic in a ship in Summer 1956. I got off at Southampton, UK, but he and his cousin went on to Holland, and the next day in the *New York Herald Tribune*, front page, there were the boys, cruising the Amsterdam Canals with the Grandmother! It has been a pleasure to renew that friendship in Norway and California.

F11: **Alan Bartlett Shepard**

- Bartlett is my mother's father's family name. Ruth Bartlett, and we proudly are related to Josiah Bartlett who signed the Declaration of Independence. We were amongst the earliest Settlers (invaders?) I have a son who chose Per Bartlett, and a sister Wendy Bartlett. So I guess, without researching it, that Alan Bartlett Shepard and I are somehow distantly related. Perhaps this explains my independence of thought?

F12: Roald Amundsen Statue & Airplane.

A. This statue of Roald Amundsen is 4 minutes walk from my home in Norway. I pass it daily, on the way to a shop of the train or bus to Oslo. Amundsen was a local hero, and lived nearby, say 15 minutes drive from me. He gained the admiration of local boys by skiing that distance (maybe 10 km?) from his home in Svartskog to Kolbotn train station, behind the statue, for his daily trips to Oslo. My grandchildren go to the local Roald Amundsen school.

B. The airplane photo.

Roald Amundsen and N-25 near the North Pole, in 1925.
Photo is public property.
Photographer: Anders Beer Wilse (1865–1949) - Galleri NOR
Collection Number: NF.WA 03026

C. Plane lent from French, 'Latham 47', Roald Amundsen 1928, This is shortly before takeoff from Tromsø, Norway on his ill fated mission.
Photo is public property Conditions.
Photographer: Anders Beer Wilse - Galleri Nor
Collection Number: NF.WB 48692

D. "Right away."
Amundsen's answer when asked if he would participate in the attempt to find Nobile.

E. other Amundsen quotes in English
http://en.wikiquote.org/wiki/Roald_Amundsen
including

- I may say that this is the greatest factor — the way in which the expedition is equipped — the way in which every difficulty is foreseen, and precautions taken for meeting or avoiding it. Victory awaits him who has everything in order — luck, people call it. Defeat is certain for him who has neglected to take the necessary precautions in time; this is called bad luck.
  — from The South Pole, by Roald Amundsen

F13: Case Study, Thorn EMI, Trevor Reeve

- Based on Chapter 14, Software Inspection [14] book, 1993
- Company Name History: MEL (Mullard Electric Limited) was part of Phillips UK until taken over by Thorn EMI Radar in the early 90's. Comment about the work Reeve started in MEL continued in Thorn EMI Radar until 1994 when he left the company.
- Trevor can be contacted at: trevorreeve@mail.com
-

F14. Deming
. He gave me a copy of his 1950 book, 'Some Theory of Sampling', in 1994. Refer back to Quote 10.4.

F15. ***Christopher Strachey***. *Referenced in 10.9, 10.10*

- ***Here is my initial inspiration for these ideas:***
- *Christopher Strachey (Oxford) :* "Computing science has been under some attack on the grounds that it isn't software engineering. I propose to attack it on different grounds. I think we should seriously ask ourselves the question: is computing science?
- Recently I did a small survey as to whether computing is suitable as an undergraduate subject in an English university. I did this by grading all the topics I could think of under the headings of relevance and state of development. **The premise is that it is clearly wrong to teach undergraduates the state of the art; one should teach them things which will still be valid in 20 years time: the fundamental concepts and underlying principles. Anything**
- **else is dishonest.**
- The gradings for relevance ran from "clearly relevant and essential" to "part of another subject" (like numerical analysis) and those for state of development from "well developed with theorems, laws and text books" to "a fruitful field for research". Note, incidentally, the importance of text books. They are designed to be taught from; they are quite different from treatises and even further from research papers. Now, it turned out that all those subjects which score highly for relevance score very low on state of development and vice versa.
- Until we have a sufficient body of topics which are important, relevant and well developed we cannot call the subject a "science". I am quite convinced that in fact computing will become a very important science. But at the moment we are in a very primitive state of development; **we don't know the basic principles yet and we**
- **must learn them first. If universities spend their time teaching the state of the art, they will not discover these principles and that, surely, is what academics should be doing.**

I do not for a moment underestimate the importance of the state of the art in engineering. Clearly it is essential and furthermore from engineering practice we must get our experience and material from which we develop theory**. But, before teaching students we must get our basic principles right. [Strachey 1969]**
Strachey 1969:
http://homepages.cs.ncl.ac.uk/brian.randell/NATO/nato1969.PDF

- Software Engineering, Proceedings, page 65
- Bio & Photo

http://www.cs.man.ac.uk/CCS/res/res43.htm#e

F16. **William A. Foster. Quoted in 2.3**

Quality is never an accident; it is always the result of high intention, sincere effort, intelligent direction and skillful execution; it represents the wise choice of many alternatives.
Quoted in 2.3
William A. Foster
http://www.mastercontrol.com/newsletter/quality-inspiration.html

"When an initial Google search didn't turn up anything significant except for a wiki on a military-related William A. Foster, I got really dedicated and clicked over to page two of the search results. I found a Google research thread looking for the same information, with conclusions drawn that indicated the first noted observation came from an advertisement in the June 10th, 1939 edition of the New York Times. Since no other source can be found, the conclusion is that the advertising agency came up with it, which makes sense to me considering the quality of the content itself. As profound as the statement is, a solid assumption would be that someone in QA came up with it! As for the context, I can't argue with this choice as a favorite. Whoever William A. Foster was, if it even was William A. Foster, the idea that quality requires high intention is inarguable. As a general misconception, I've noted only those within the quality profession know of the true level of 'sincere effort, intelligent direction and skillful execution' that those committed to the industry put into their work each day."

F17. Source: Peter Drucker , "Management" (Heinemann, 1975) , Cited in Gilb PoSEM book 1988 [5]. Quote 2.9 B.