"Agility is the Tool, Not the Master:
Gilb's Ten Key Agile <u>Principles</u>
to deliver stakeholder value,
avoid bureaucracy, and give creative freedom"

Tom Gilb

"Lean and Agile in the Public Sector: Helping to meet financial targets and today's challenges. " (Theme)

London 21 Sept. 2011

tomsgilb@gmail.com

@imtomgilb

www.gilb.com

These slides will be on gilb.com downloads

http://www.gilb.com/tiki-download_file.php?fileId=391

Paper:appeared agilerecord.com Summer 2010.

http://homepage.mac.com/tomgilb/filechute/Agile%20Principles%20and%20Values%20for%20Agile%20Record%202010%20Gilb.doc

1

So, what are Agile methods missing?

- **Stakeholder Focus**
  - **Real projects have dozens of stakeholders**
    - **Not just a customer in the next room**
    - **Not just a user with a use case or story**

- **Results Focus**
  - **It is not about writing code, it is about <u>delivering value</u> to stakeholders**
  - **It is not about programming, it is about making <u>systems</u> work, for <u>real people</u>**

- **Systems Focus**
  - **It is not about coding – (*again* ☺)**
  - **It is about reuse, data, hardware, training, motivation, sub-contracting, Outsourcing, help lines, user documentation, user interfaces, security, etc.**
  - **So, a <u>systems engineering</u> scope is necessary to deliver *results*.**
  - **Systems Engineering needs <u>quantified performance and quality objectives</u>**
    - **To synchronize all necessary disciplines, so that they deliver the results.**

Scrum and Evo

- "Tom Gilb invented Evo, arguably the first Agile process.
- He and his son Kai have been working with me in Norway to align what they are doing with Scrum.
- Kai has some excellent case studies where he has acted as Product Owner. He has done some of the most innovative things I have seen in the Scrum community"
  - Jeff Sutherland, co-inventor of Scrum, 5Feb 2010 in Scrum Alliance Email (recommending us to be invited to Scrum Gathering, Orlando in March 2010, which we did)
  - .http://bit.ly/a5Fd1T #scrum #agile Sutherland credits Gilb in Roots of Scrum slide #accu2010

Gilb credited as Root

THE ROOTS OF SCRUM
How the Japanese lean experience changed global software development

With help from Citrix Online, Google, Yahoo, Microsoft, IBM, Oracle, MySpace, Adobe, GE, Siemens, Disney Animation, BellSouth, Alcatel-Lucent, GSI Commerce, Ulticom, Palm, St. Jude Medical, DigiChart, RosettaStone, Healthwise, Sony/Ericsson, Accenture, Trifork, Systematic Software Engineering, Exigen Services, SirsiDynix, Softhouse, Philips, Barclays Global Investors, Constant Contact, Wellogic, Inova Solutions, Medco, Saxo Bank, Xebia, Insight.com, SolutionsIQ, Crisp, Johns Hopkins Applied Physics Laboratory, Unitarian Universalist Association, Motley Fool, Planon, FinnTech, OpenView Venture Partners, Jyske Bank, BEC, Camp Scrum, DotWay AB, Ultimate Software, Scrum Training Institute, AtTask, Intronis, Version One, OpenView Labs, Central Desktop, Open-E, Zmags, eEye, R... Hamilton, Scrum Alliance, Fortis, DIPS, Program ... Gilb.com, WebGuide Partner, Emergn, NSB (Norweg... egasystems, Wake Forest University, The Economist, iContact, Avaya, Kar...n Marketing, accelare, Tam Tam

Gilb.com

ACCU, Oxford, UK 14 Apr 2010

RUGBY

SCRUM TRAINING INSTITUTE

openview venture partners

ScrumAlliance
transforming the world of work

Tuesday, April 13, 2010   1

First Attempt to Teach a Scrum Front End Using Evo ideas

- A 1-day front-end for 'Product Managers' before a 1-day Scrum Overview course for Product Managers
- Commissioned by and co-authored by Gabriella Benefield (Scrum Alliance) 2009
- Detailed training exercises available at
  - http://www.gilb.com/tiki-download_file.php?fileId=353
  - Value Planning slides for Scrum (Oct 09)
- The dozen slides at end of this slide set are Tom's attempt to describe the relationship of
  - Scrum and the Value Planning front end
  - based on Evo
  - These slides were not part of the training G. B. and I held in 2009)

Value Planning   (+ Scrum)

A better 'front end' to Scrum, and other agile variants

BASED ON IDEAS FROM THE 'EVO' METHOD

Efficient Value Organisation/Options

Evolving Value in Organizations

Evolving Value Optimization ,

Efficient Value Optimization

= EVO

Value-Driven Scrum
(one of your options for smart Product Ownership)

- **Defined As:**
  - **The real world interface to the Scrum Product Owner**
  - **The Businesses 'Organizational Value' Management**
  - **The Business Function Management**
  - **The Technical Architecture Management**

- **All in a pipeline to the Scrum Product Owner (PO)**
  - **Fully designed, from the *organizational* point of view**
  - **Allowing additional design at the level of *programming*, chunking, and data**
    - **By the Scrum Team**
  - **Prioritized from the *Organizational* Point of View**

The 'Scrum Product Owner'

- Needs to get enough information about the product
  - To allow the Scrum team to build, test, make technical detailed decisions

- Here is one set of tools to allow the Product Owner
  - Perhaps, in larger environments, a PO 'team'
  - To collect information, to plan, so that
    - **We really deliver the best value for money, as soon as possible**

What is new?
What is Value-Planning (VP) ?

- **Dominant focus on Value Delivery Management** –
  - Not from a programming point of view
  - But from a business and management non technical point of view
  - Which critical value improvements do we need first, and next

- Stakeholder Values-and-Priorities <u>Integration</u>*
  - Of management, marketing, IT, Systems Engineering,
  - Including Sales, Customer Service and ALL Critical Stakeholders

- **Systems View** – Systems Architecture – Systems Engineering

- * integration: defined as: Alignment and reasonable balance of competing interests, through intelligent dynamic prioritization.

See ppt note for depth papers on priority, including:
http://www.gilb.com/tiki-download_file.php?fileId=60

Value Driven Scrum

System Owner			
Stakeholders Values	Business Values	System Functions	
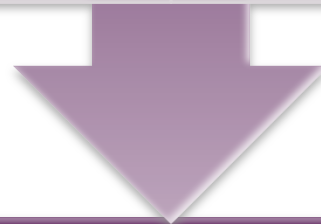
Product Owner	
Build Test Maintain	Detailed Technical Design

Value Decision Tables

Business Goals	Stakeholder Value 1	Stakeholder Value 2
Business Value 1	-10%	40%
Business Value 2	50%	10%
Resources	20%	10%

Stakeholder Val.	Product Value 1	Product Value 2
Stakeholder Value 1	-10%	50 %
Stakeholder Value 2	10 %	10%
Resources	2 %	5 %

Product Values	Solution 1	Solution 2
Product Value 1	-10%	40%
Product Value 2	50%	80 %
Resources	1 %	2 %

Prioritized List
1. Solution 2
2. Solution 9
3. Solution 7

Product Backlog · Sprint Backlog · Sprint · Working increment

24 h · 30 days

Scrum Develops

We measure improvements Learn and Repeat

Copyright: Kai@Gilb.com

Value Decision Tables

Business Goals	Training Costs	User Productivity
Profit	-10%	40%
Market Share	50%	10%
Resources	20%	10%

Stakeholder Val.	Intuitiveness	Performance
Training Costs	-10%	50 %
User Productivity	10 %	10%
Resources	2 %	5 %

Product Values	GUI Style Rex	Code Optimize
Intuitiveness	-10%	40%
Performance	50%	80 %
Resources	1 %	2 %

Jeffsutherland Twitter: Very cool product backlog management by Tom and Kai Gilb http://ad.vu/2h4d *Sat 28 March 2009*

Prioritized List
1. Code Optimize
2. Solution 9
3. Solution 7

Product Backlog

Scrum Develops

24 h

30 days

Sprint Backlog   Sprint   Working increment of the software

We measure improvements Learn and Repeat

Copyright: Kai@Gilb.com
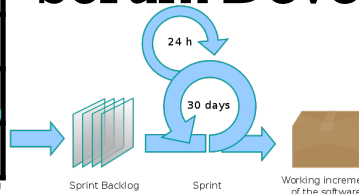
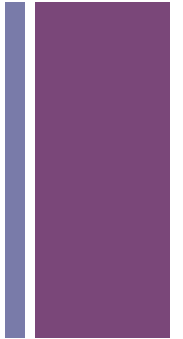## **Value Management (Evo)**

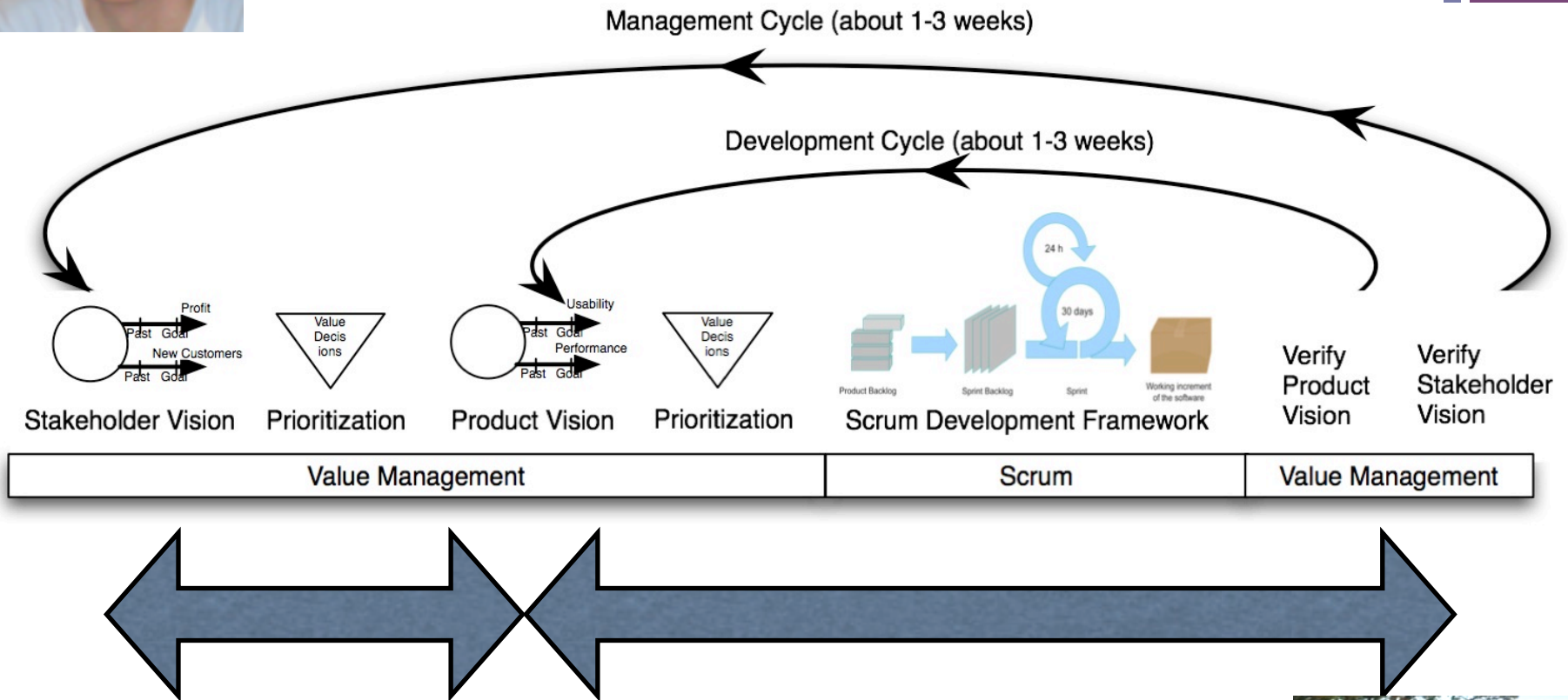Focus towards challenges

Stakeholder requirements quantified

Both Goal and Tolerable levels specified.

Table shows relationship requirements and design

Testing during and after deliver cycles

A more-advanced and more-comprehensive way to apply Scrum

Management Cycle (about 1-3 weeks)

Development Cycle (about 1-3 weeks)

Profit							
Past Goal		Usability		24 h			
New Customers	Value Decis ions	Past Goal Performance	Value Decis ions	30 days	Verify Product Vision	Verify Stakeholder Vision
Past Goal		Past Goal				
Stakeholder Vision	Prioritization	Product Vision	Prioritization	Product Backlog → Sprint Backlog → Sprint → Working increment of the software — Scrum Development Framework		

| Value Management | | Scrum | Value Management |

September 21, 2011

Gilb's Ten Key Agile Principles
to avoid bureaucracy and give creative freedom

1. Control projects by quantified critical-few results. 1 Page total !

    (not stories, functions, features, use cases, objects, ..)

2. Make sure those results are <u>business</u> results, not technical

    Align your project with your financial sponsor's interests!

3. Give developers freedom, to find out *how* to deliver those results

4. Estimate the impacts of your designs, on *your* quantified goals

5. Select designs with the best impacts in relation to their costs, do them first.

6. Decompose the workflow, into weekly (or 2% of budget) time boxes

7. Change designs, based on quantified experience of implementation

8. Change requirements, based in quantified experience, new inputs

9. Involve the stakeholders, every week, in setting quantified goals

10. Involve the stakeholders, every week, in *actually using* increments

Gilb's Agile Principles

to avoid bureaucracy and give creative freedom   (1 sentence summary)

Main Idea:

Get   early, and frequent, real, stakeholder net-<u>value</u> - delivered

	VALUE TO CREATE	VALUE TO PRESERVE	VALUE TO SACRIFICE
EMPLOYEES			
CUSTOMERS			
SUPPLIERS AND PROFESSIONAL ADVISERS			
INVESTORS			
TRADES UNIONS			
GOVERNMENT			
MEDIA			
COMMUNITY			
OTHER STAKEHOLDER GROUPS			

Deliver Value !

1. Control projects by quantified critical-few results. 1 Page total !

(not stories, functions, features, use cases, objects, ..)
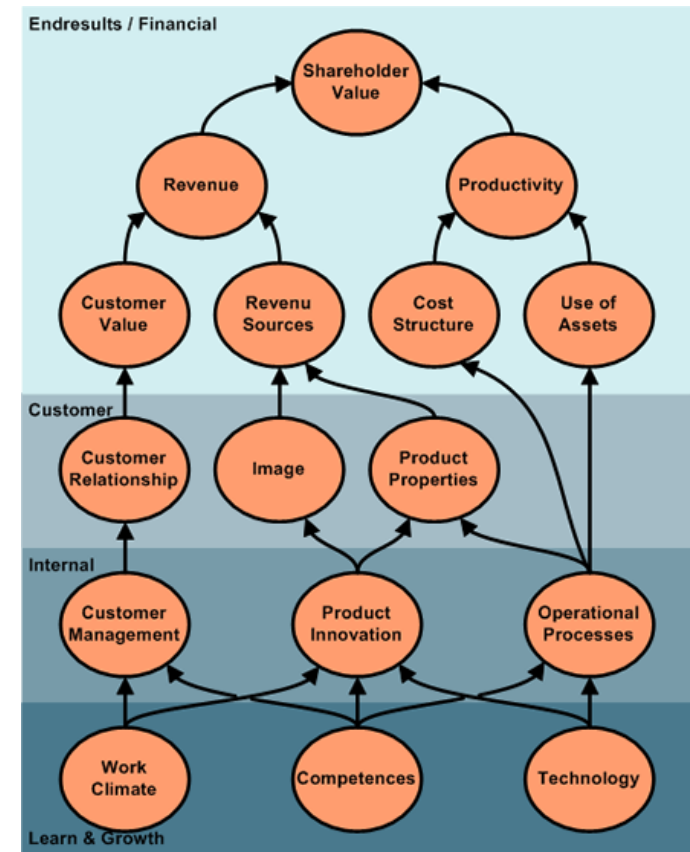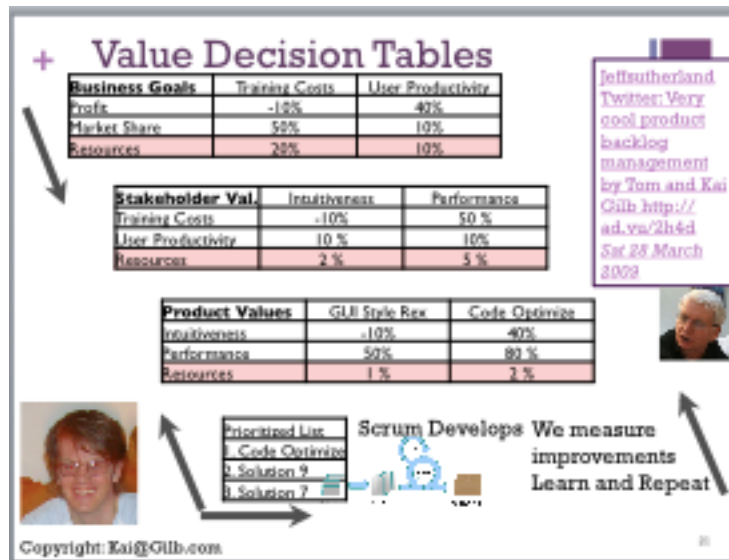
NOT LIKE THIS! Project Objectives
'Unquantified few'

Real Example of *Lack* of Scales

▶ **Defined** Scales of Measure:

◦ Demands *comparative* **thinking.**
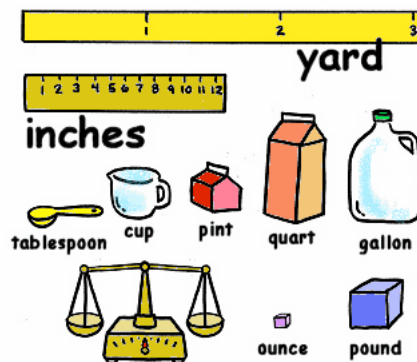
◦ Leads to requirements that are unambiguously **clear**

◦ Helps Team be **Aligned** with the Business

*1. Central to The Corporations business strategy is to be the world's **premier** integrated_<domain> service **provider**.*

*2. Will provide a much more efficient **user** experience*

*3. Dramatically scale back the **time** frequently needed after the last data is acquired to time align, depth correct, splice, merge, recompute and/or do whatever else is needed to **generate** the desired **products***

*4. Make the system much **easier** to **understand** and **use** than has been the case for previous system.*

*5. A primary goal is to provide a much more **productive** system **development** environment than was previously the case.*

*6. Will provide a richer set of functionality for **supporting** next-generation logging **tools** and applications.*

*7. **Robustness** is an essential system requirement (see rewrite in example below)*

*8. Major improvements in **data quality** over current practices*

This lack of clarity cost them $100,000, 000

More like this! (Real case).

Business objective	Measure	Goal (200X)	Stretch goal ('0X)	Volume	Value	Profit	Cash
Time to market	Normal project time from GT to GT5	<9 mo.	<6 mo.	X		X	X
Mid-range	Min BoM for The Corp phone	<$90	<$30	X		X	X
Platformisation Technology	# of Technology 66 Lic. shipping > 3M/yr	4	6				
Interface	Interface units	>11M	>13M	X		X	X
Operator preference	Top-3 operators issue RFQ spec The Corp	1	2	X		X	X
Productivity							
Get Torden	Lyn goes for Technology 66 in Sep-04	Yes		X			X
Fragmentation	Share of components modified	<10%	<5%		X	X	X
Commoditisation	Switching cost for a UI to another System	>1yr	>2yrs				
Duplication	The Corp share of 'in scope' code in best-selling device	>90%	>95%		X	X	X
Competitiveness	Major feature comparison with MX	Same	Better	X		X	X
User experience	Key use cases superior vs. competition	5	10	X	X	X	X
Downstream cost saving	Project ROI for Licensees	>33%	>66%	X	X	X	X
Platformisation IFace	Number of shipping Lic.	33	55	X		X	X
Japan	Share of of XXXX sales	>50%	>60%	X		X	X

Numbers are intentionally changed from real ones

<- Business Objectives Quantified

2. Make sure those results are business results, not technical

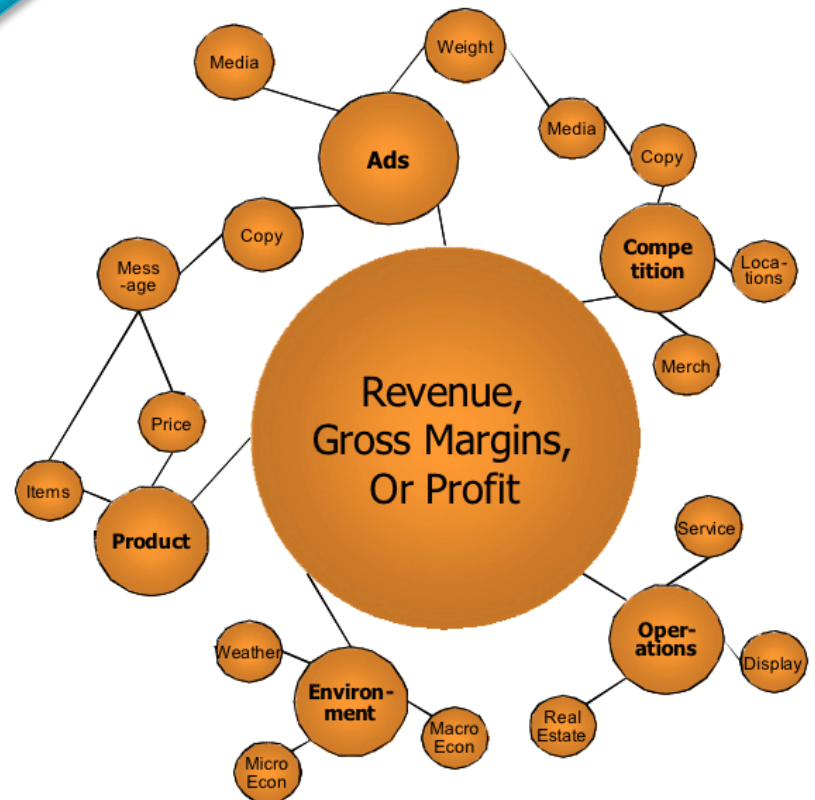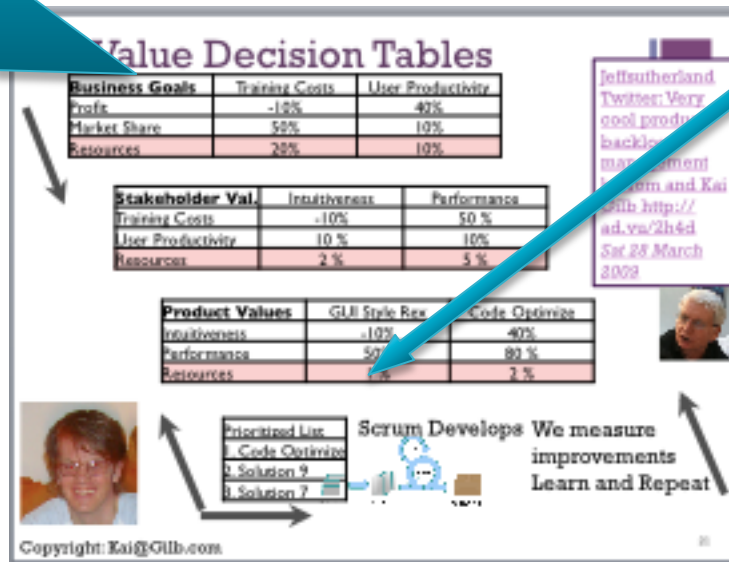**Align your project with your financial sponsor's interests!**

Figure 1. The "Mother of All Models". © 2006 MarketingNPV LLC. All Rights Reserved.

3. Give developers freedom,
to find out *how* to deliver those results

Value Decision Tables

Business Goals	Training Costs	User Productivity
Profit	-10%	40%
Market Share	50%	10%
Resources	20%	10%

Stakeholder Val.	Intuitiveness	Performance
Training Costs	-10%	50 %
User Productivity	10 %	10%
Resources	2 %	5 %

Product Values	GUI Style Rex	Code Optimize
Intuitiveness	-10%	40%
Performance	50%	0 %
Resources	1 %	1 %

Jeffsutherland Twitter: Very cool product backlog management by Tom and Kai Gilb http://ad.vu/2h4d Sat 28 March 2009

Prioritized List
1. Code Optimize
2. Solution 9
3. Solution 7

Scrum ... ...sure improvements Learn and Repeat

Copyright: Kai@Gilb.com

4. Estimate the impacts of your designs, on *your* quantified goals

What values and ways of working are institutionalised in the workforce? What business climate do you operate within – highly competitive, regulated, fast-changing etc?

Are the controls and measurements sufficient for managing the project delivery and software quality?

What technical and managerial test processes, procedures and standards are used to ensure quality?

IMPACT™

Environment

Metrics

Processes

Capability

Technology

Organisation

Which skills exist / what gaps exist in the IT (development & test) organisation

Which tools and facilities are used to ensure and/or improve quality and productivity?

How effective is the IT (development & test) structure/organisation?

If you cannot, then your knowledge is of a meagre and unsatisfactory kind (Lord Kelvin)

Strategy Impact Estimation:
for a $100,000,000 Organizational Improvement Investment

Technical Strategies

Objectives

"Benefits"

Strategy Impacts on Objectives

Cost

Benefit/Cost

ratio

358 !

Viking Deliverables

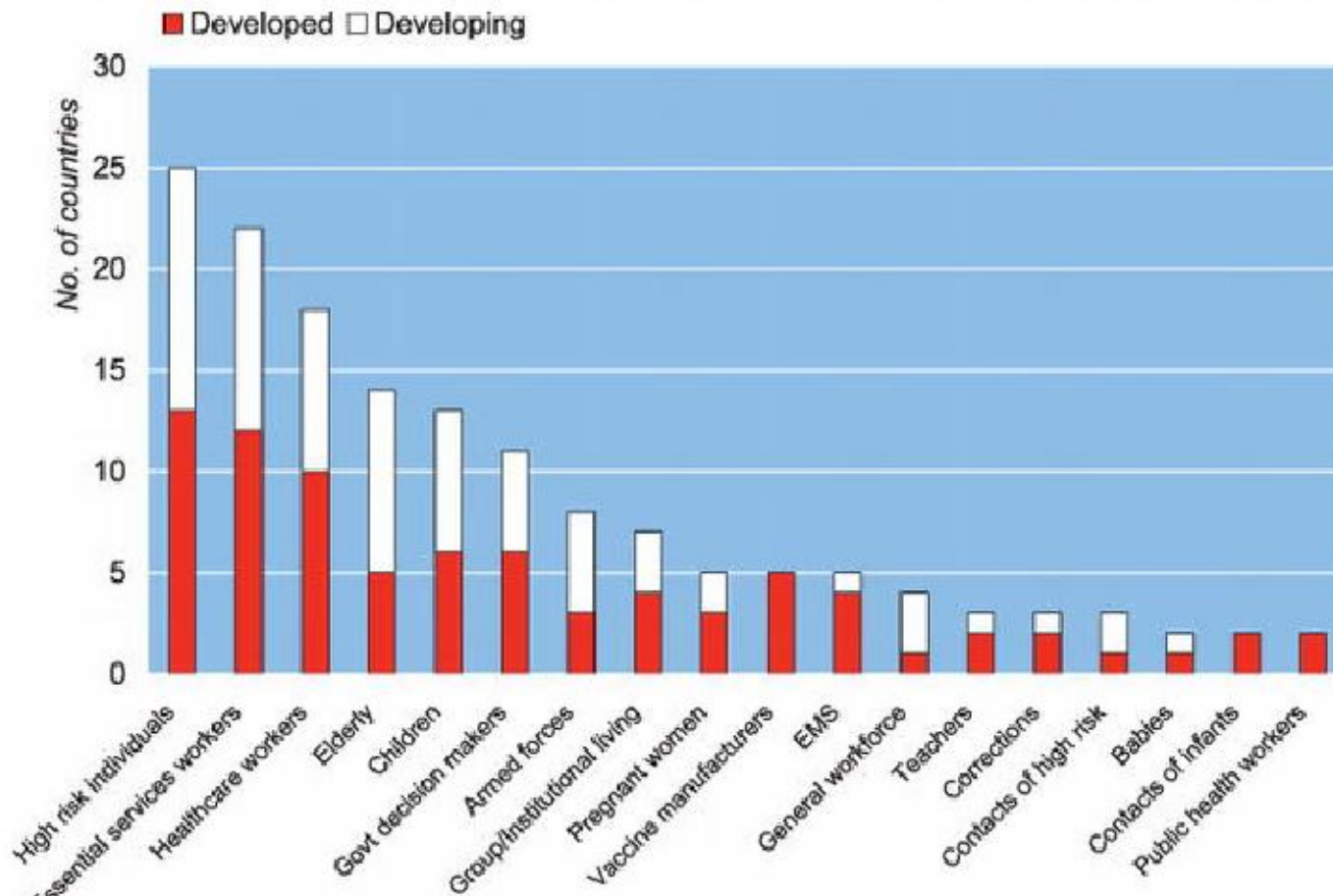| Business Objective		hardware adaptation	Telephony	Reference designs	IFace	Modularity	Defend vs Technology 66	Tools	User Exper'ce	GUI & Graphics	Security	Defend vs OCD	Enterprise
Time to market		20%	10%	30%	5%	10%	5%	15%	0%	0%	0%	5%	5%
Mid-range		15%						5%	10%	5%	5%	0%	0%
Platformisation Technology		25%	10%	30%	0%		10%	0%	5%	0%	10%	0%	5%
Interface		5%	15%	15%	0%	5%	0%	5%	0%	0%	10%	0%	10%
Operator preference		0%						20%	5%	10%	10%	20%	5%	10%
Get Torden		25%	10%	10%	-10%	0%	20%	0%	10%	-20%	10%	10%	5%
Commoditisation		20%	10%	20%	10%	-20%	25%	15%	0%	0%	5%	10%	5%
Duplication		15%		0%	0%	40%		0%	0%	5%	20%	5%
Competitiveness		10%	15%	20%	0%	10%	20%	10%	10%	20%	10%	10%	10%
User experience		5%	0%	0%	20%	0%		0%	30%	10%	0%	0%	0%
Downstream cost saving		15%							10%	0%	0%	0%	10%	5%
Platformisation IFace		10%	10%	20%	40%	0%	20%	5%	0%	0%	0%	0%	5%
Japan		10%	5%	20%								0%	0%
Contribution to overall result		15%	9%	17%	4%								5%
Cost (£M)		£ 2.85	£ 0.49	£ 3.21	£ 2.54	£ 1.92	£ 2.31	£ 0.81	£ 1.21	£ 2.68	£ 0.79	£ 0.62	£ 0.60
ROI Index (100=average)		106	358	109				148	107	10	152	202	174

Target diagrams: 1 2 3 4 5

5. Select designs with the best impacts in relation to their costs, do them first.

Figure 1: Vaccine Priority Groups by Development Status - Listed in at Least Two National Plans


Legend: ■ Developed □ Developing

Y-axis: No. of countries (0, 5, 10, 15, 20, 25, 30)

X-axis categories: High risk individuals, Essential services workers, Healthcare workers, Elderly, Children, Govt decision makers, Armed forces, Group/Institutional living, Pregnant women, Vaccine manufacturers, EMS, General workforce, Teachers, Corrections, Contacts of high risk, Babies, Contacts of infants, Public health workers

Source: Uscher-Pines et al. Priority setting for pandemic influenza: An analysis of national preparedness
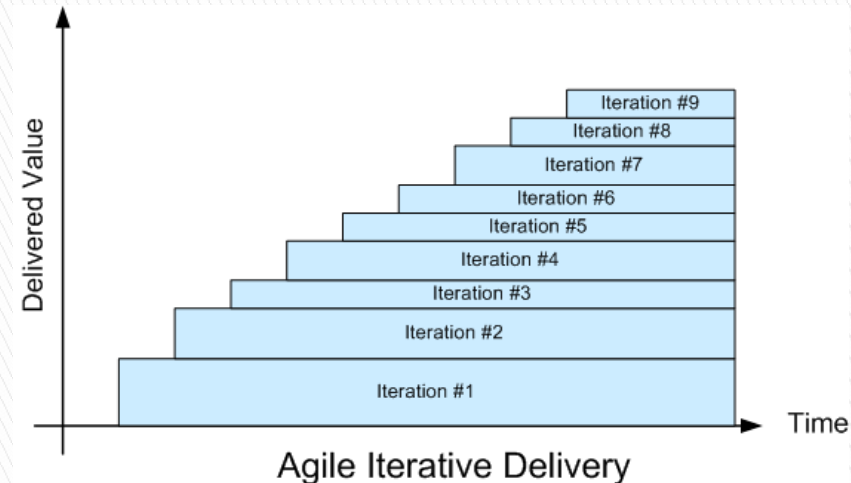
6. Decompose the workflow, into weekly (or 2% of budget) time boxes

▶

Decomposition of Projects: How to Design Small Incremental Steps INCOSE 2008

http://www.gilb.com/tiki-download_file.php?fileId=41

Agile Iterative Delivery

Non-agile Project Delivery
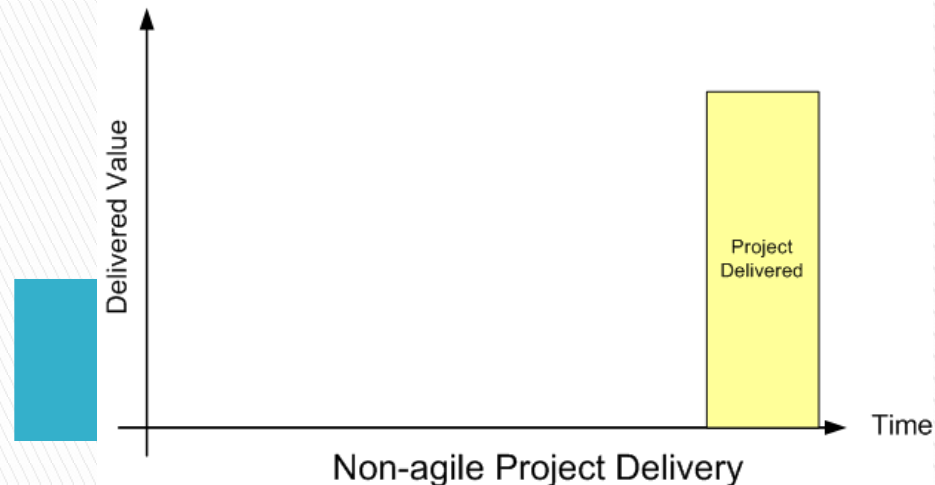
7. Change designs, based on quantified experience of implementation

Design is the servant of the requirement. If it does not work 'fire' it.

Bank Case of 'Evo' (Gilb's Lean Agile method)

- "The proof is in the pudding;
- I have used Evo (albeit in disguise sometimes) on two large, high-risk projects in front-office investment banking businesses, and several smaller tasks.
- On the largest critical project,
  - the original business functions & performance objective requirements document,
  - which *included no design*,
  - essentially remained unchanged over the 14 months the project took to deliver,
- but the detailed designs
  - (of the GUI, business logic, performance characteristics)
  - changed many many times,
  - guided by lessons learnt and feedback gained by delivering a succession of early deliveries to real users.
- In the end,
  - the new system responsible for 10s of USD billions of notional risk,
  - successfully went live over one weekend
  - for 800 users worldwide,
  - and was seen as a big success by the sponsoring stakeholders."

- Richard Smith, London,
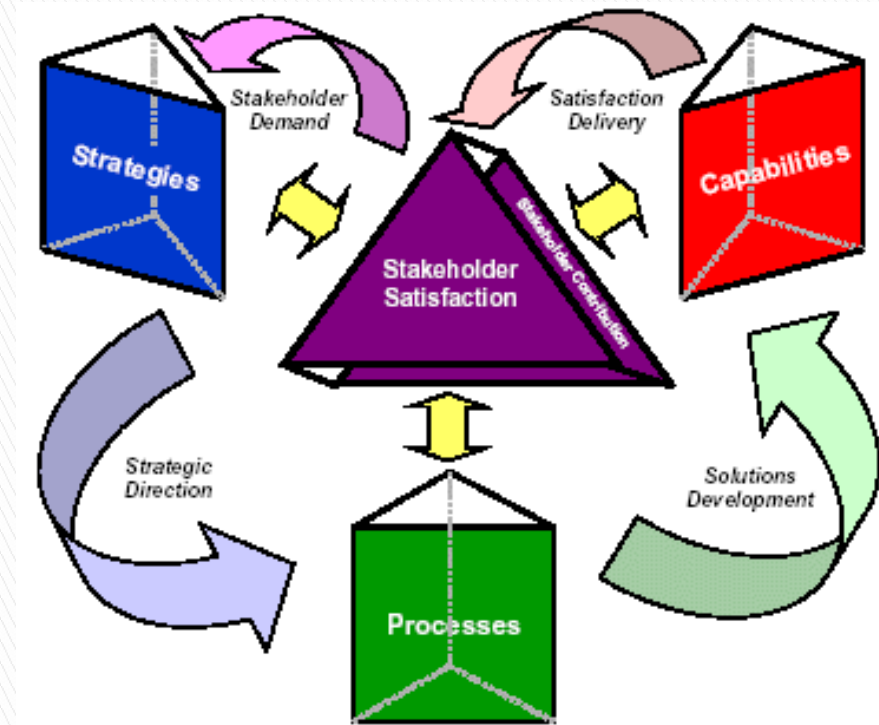 http://rsbatechnology.co.uk/blog:8
  Sept 10 2011

8. Change requirements, based on quantified experience, new inputs: intelligent tradeoff.

▶ Reduce the level or delivery time, of lower-priority requirements, in order to deliver high priority requirements on time, within budget, or at Goal levels.
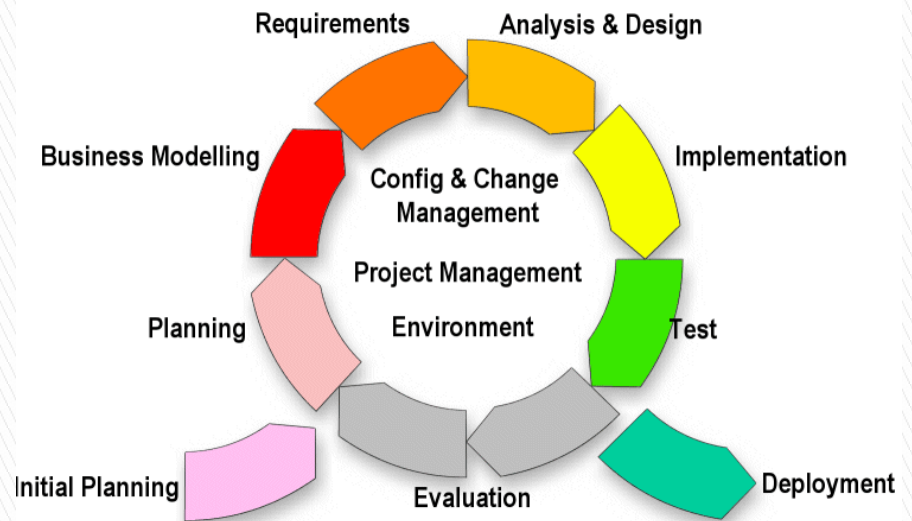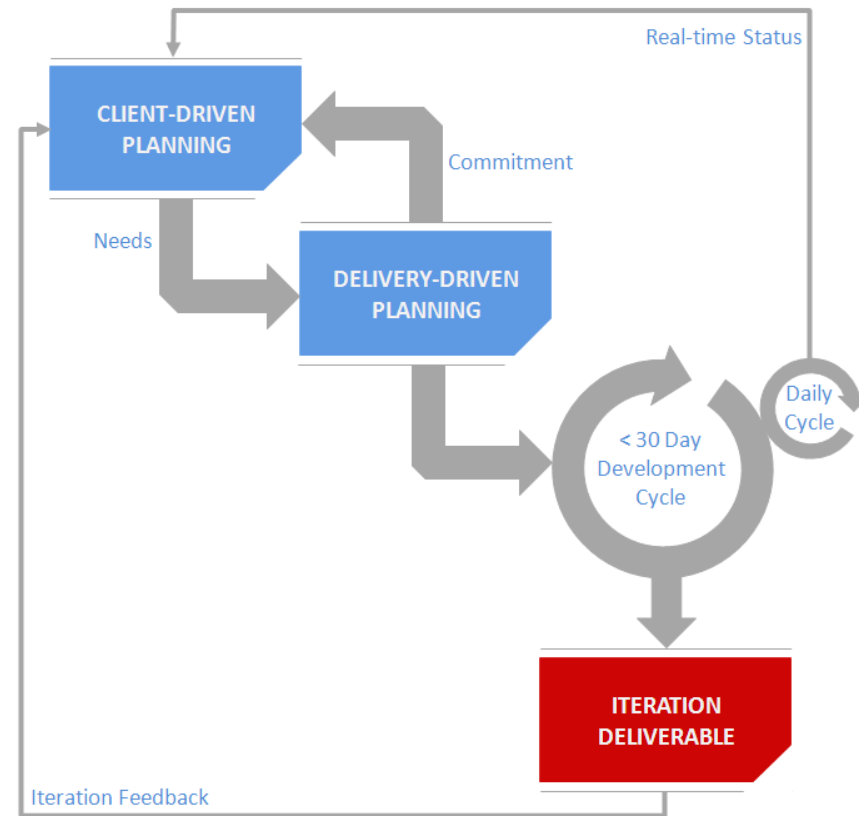
9. Involve the stakeholders, every week, in setting quantified goals

It is much easier to determine requirements with a little hindsight!

The eternal cycle of stakeholder priorities

10. Involve the stakeholders, every week, in actually using increments

My 10 Agile Values?

- **Simplicity**
  - 1. Focus on real stakeholder values
- **Communication**
  - 2. Communicate stakeholder values quantitatively
  - 3. Estimate expected results and costs for weekly steps
- **Feedback**
  - 4. Generate results, weekly, for stakeholders, in their environment
  - 5. Measure all critical aspects of the improved results cycle.
  - 6. Analyze deviation from your initial estimates
- **Courage**
  - 7. Change plans to reflect weekly learning
  - 8. Immediately implement valued stakeholder needs, next week
    - *Don't wait, don't study (analysis paralysis), don't make excuses.*
    - *Just Do It!*
  - 9. Tell stakeholders exactly what you will deliver next week
  - 10. Use any design, strategy, method, process that works quantitatively well - to get your results
    - Be a <u>systems</u> <u>engineer</u>, not a just programmer (a 'Softcrafter').
    - Do not be limited by your craft background, in serving your paymasters
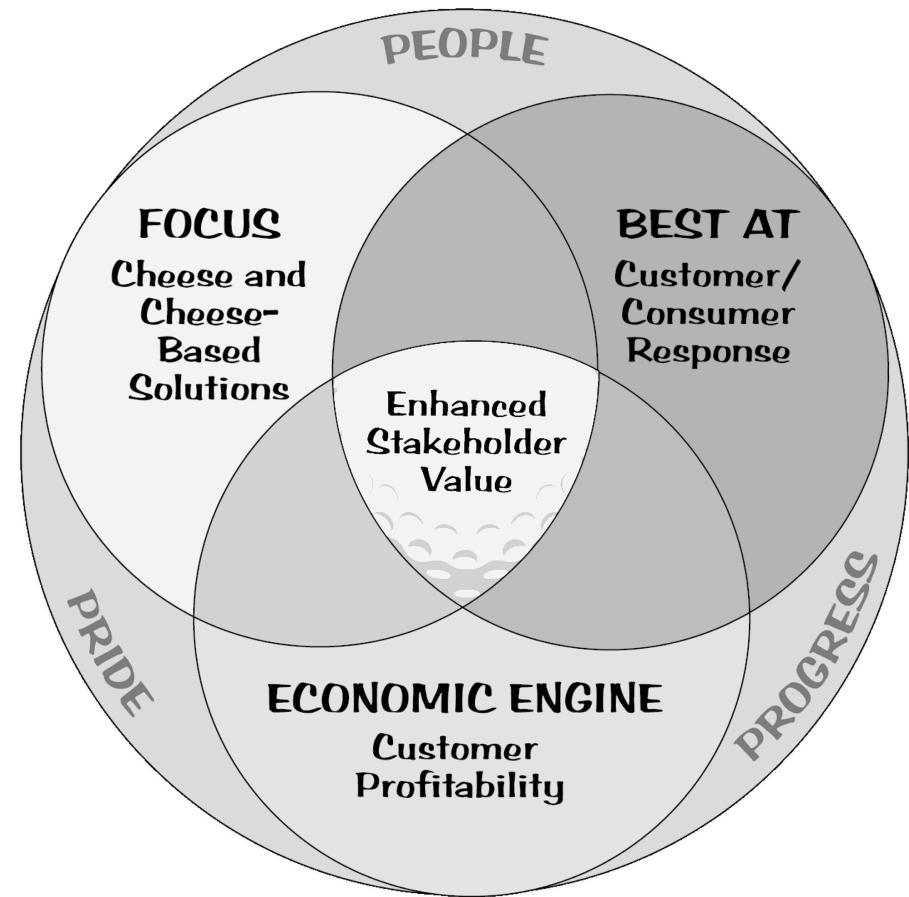
                September 21, 2011

My 10 Agile *Values*? (Detail)

- **Simplicity**
- **Communication**
- **Feedback**
- **Courage**

September 21, 2011

Simplicity

- ◦ 1. Focus on real stakeholder values

- ◦

Communication

- **2. Communicate stakeholder values quantitatively.**

-

Kura - Kano Rice Value Chain

© Gilb.com   Agility is the Tool
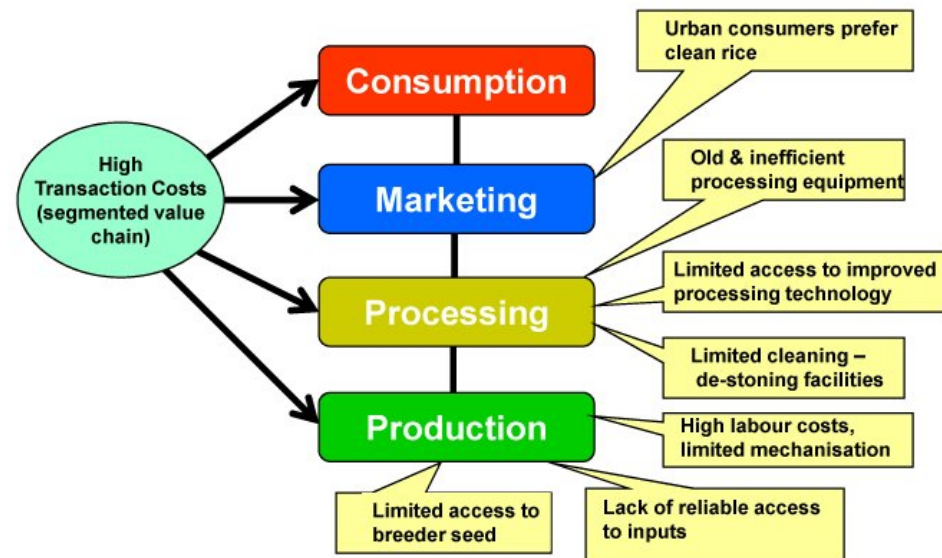
Estimate Often

- **3. Estimate expected results and costs for weekly steps**
-
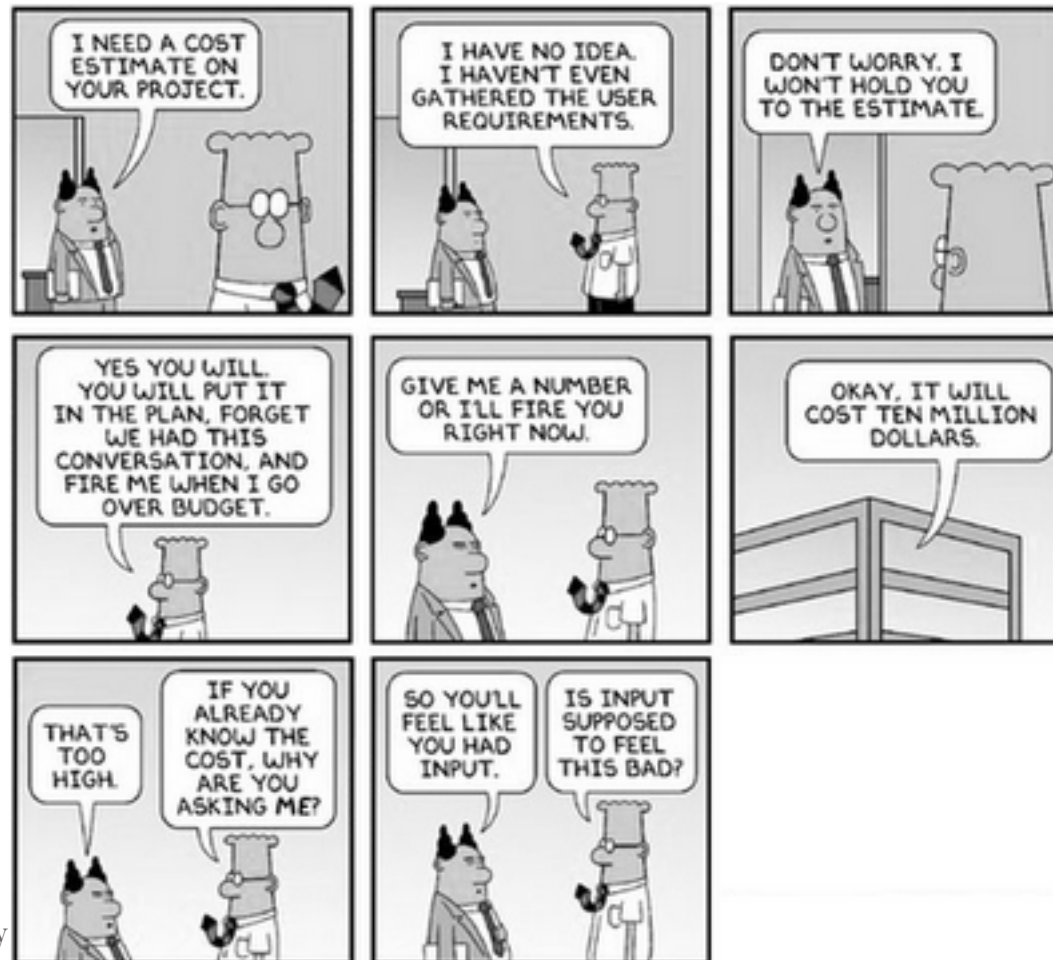
ember 21, 2011

Feedback

- **4. Generate results, weekly, for stakeholders, in *their* environment**

© Gilb.com   Agility is the Tool   September 21, 2011

Measure Critical Stuff

- **5. Measure all critical aspects of the improved results cycle.**
-

How may a man measure his own happiness?

He must first go to his cupboard and take out all his neckties.

Then he must lay them out on the ground, end to end.

Then he must measure the length of this line of neckties.

And that measurement; that distance is exactly the same

..... as his distance from true happiness.

© Gilb.com   Agility is the Tool
September 21, 2011
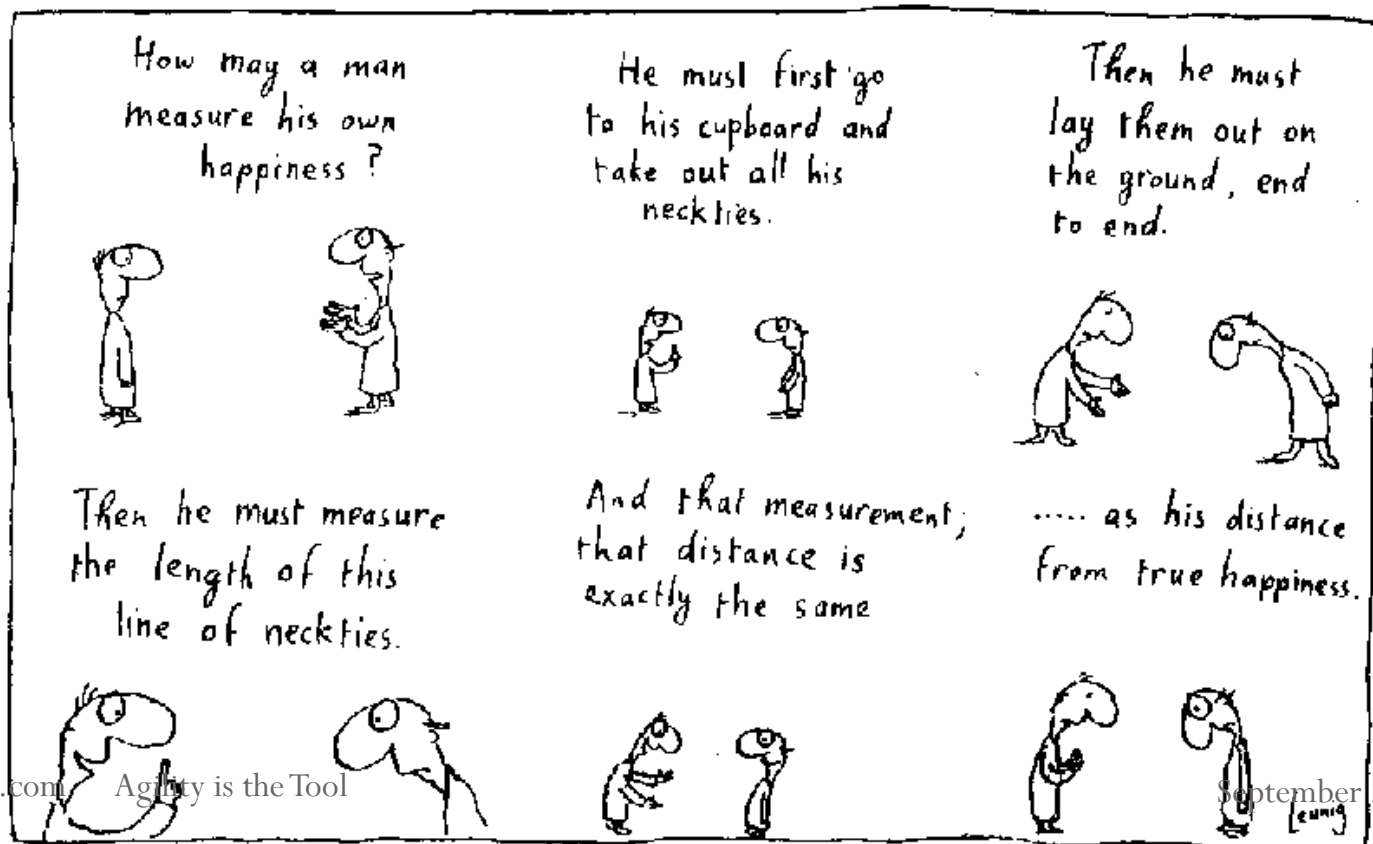
Learn from Deviations

- **6. Analyze deviation from your initial estimates.**
-

© Gilb.com   Agility is the Tool   September 21, 2011

- **7. Change plans to reflect weekly learning.**
-

© Gilb.com   Agility is the Tool

September 21, 2011

- **8. Immediately implement valued stakeholder needs, next week**
  - *Don't wait, don't study (analysis paralysis), don't make excuses.*
  - *Just Do It!*
-

© Gilb.com   Agility is the Tool

September 21, 2011

Tell Stakeholders What's next

- **9. Tell stakeholders exactly what you will deliver next week**

-

© Gilb.com   Agility is the Tool

September 21, 2011
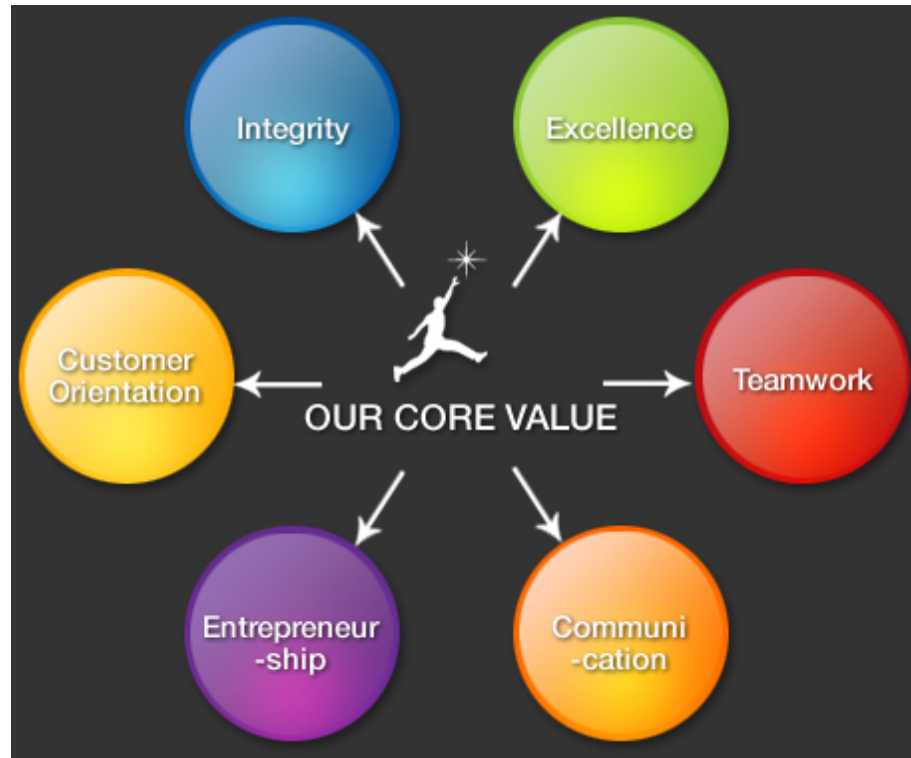
If it works, do it!

- **10. Use <u>any</u> design, strategy, method, process that works quantitatively well - to get your <u>results</u>**
  - **Be a <u>systems</u> <u>engineer,</u> not a just programmer (a 'Softcrafter').**
  - **Do not be limited by your craft background, in serving your paymasters.**
-

© Gilb.com   Agility is the Tool

September 21, 2011

So, what are Agile methods missing?

- ## Stakeholder Focus
  - **Real projects have dozens of stakeholders**
    - **Not just a customer in the next room**
    - **Not just a user with a use case or story**
- ## Results Focus
  - **It is not about writing code, it is about <u>delivering value </u>to stakeholders**
  - **It is not about programming, it is about making <u>systems</u> work, for <u>real people</u>**
- ## Systems Focus
  - **It is not about coding – (*again* ☺)**
  - **It is about reuse, data, hardware, training, motivation, sub-contracting, Outsourcing, help lines, user documentation, user interfaces, security, etc.**
  - **So, a <u>systems engineering </u>scope is necessary to deliver <u>*results*</u>.**
  - **Systems Engineering needs <u>quantified performance and quality objectives</u>**
    - **To synchronize all necessary disciplines, so that they deliver the results.**

September 21, 2011

- Ecstatic Stakeholder!

-

That's All Folks!

- Questions?
- Remarks?
- For free digital copy of this book, and 4 of my Agile papers
- Email me subject "Book"
- Tom@Gilb.com

COMPETITIVE ENGINEERING

TOM GILB

COMPETITIVE ENGINEERING

A HANDBOOK FOR SYSTEMS ENGINEERING, REQUIREMENTS ENGINEERING, AND SOFTWARE ENGINEERING USING PLANGUAGE

GILB

17 Feb 2010

End of 1 Hour Lecture

- **Discussion Remarks Questions ?**
  - **Now, and throughout the conference**

- **And by email**
  - **TomsGilb@Gmail.com**
  - **+47 92066 705, +44 (0) 77 1267 0707**
  - **@ ImTomGilb**

- **For another Norwegian case study of doing it right, see Confirmit**
  - **http://www.gilb.com/tiki-download_file.php?fileId=278**
  - **http://www.gilb.com/tiki-download_file.php?fileId=50**

- **See Value slides, following these, as an extra reserve, another angle.**
  - **From London BCS SPA Lecture 2009**

"AGILE HAS DOOMED ITSELF - TO BECOME YET ANOTHER FAD ".

What is Seriously Wrong with Agile practices and interpretations - why AGILE, AS CURRENTLY PRACTICED, is PROJECT-failure-prone as a culture

"What is Tom's advice, his own more value-oriented 'agile' principles and values (see below) and metrics-oriented agile practices in Evo?
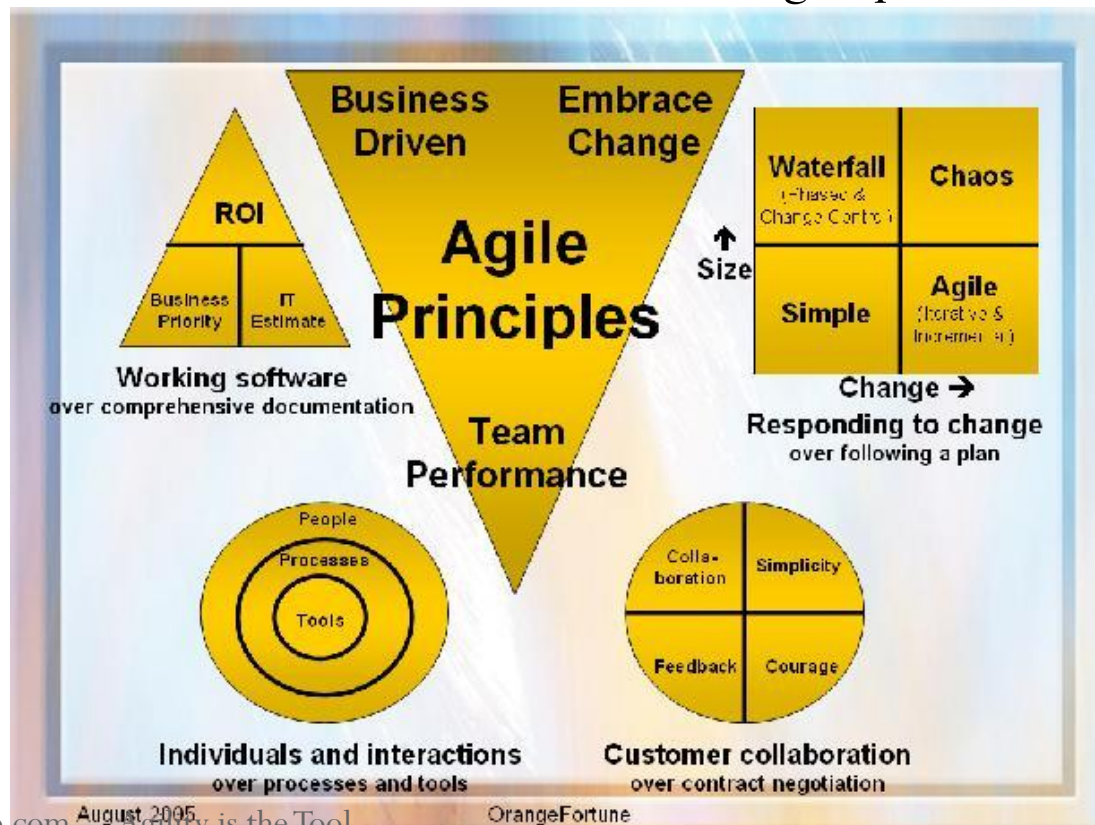
Gilb's 'Value Driven Planning' Principles:

1. Critical Stakeholders determine the values

2. Values can and must be quantified

3. Values are supported by Value Architecture

4. Value levels are determined by timing, architecture effect, and resources

5. Value levels can differ for different scopes (where, who)

6. Value can be delivered early

7. Value can be locked in incrementally

8. New Values can be discovered (external news, experience)

9. Values can be evaluated as a function of architecture (Impact Estimation)

10. Value delivery will attract resources.

September 21, 2011

Value Driven Planning Principles in Detail:

September 21, 2011

1. Critical Stakeholders determine the values

Critical: "having a decisive or crucial importance in the success or failure of something" <Dictionary

- The primary and prioritized values we need to deliver are determined by
  - analysis of the needs and values of stakeholders
    - stakeholders who can determine whether we *succeed* or *fail*.
- We cannot afford to satisfy *other (less critical)* levels, at other times and places, yet.
  - Because that might undermine our ability to satisfy the more critical stakeholders –
  - and consequently threaten our overall project success.

September 21, 2011
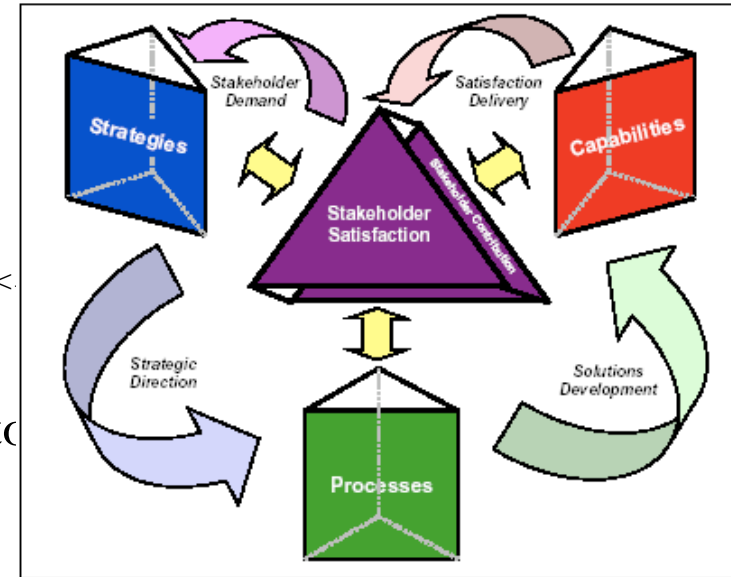
2. 'Values' can and must be *quantified*

- Values can, if you want, be expressed numerically.
  - With a defined scale of measure
  - with a deliverable level of performance
  - and with qualifier info [Where, When, If]
- Quantification is useful:
  - to clarify your own thoughts
  - to get real agreement to one clear idea
  - to allow for varied targets and constraints
  - to allow direct comparison with benchmarks
  - to put in Request for bids, bids and contracts
  - to manage project evolutionarily : track progress
  - as a basis for measurement and testing
  - to enable research on methods

CSR - score per module

Purchasing
100%
80%
60%
40%
20%
0%

Communication
Production
Sales
HRM
Finance
Strategy

© Gilb.com   Agility is the Tool
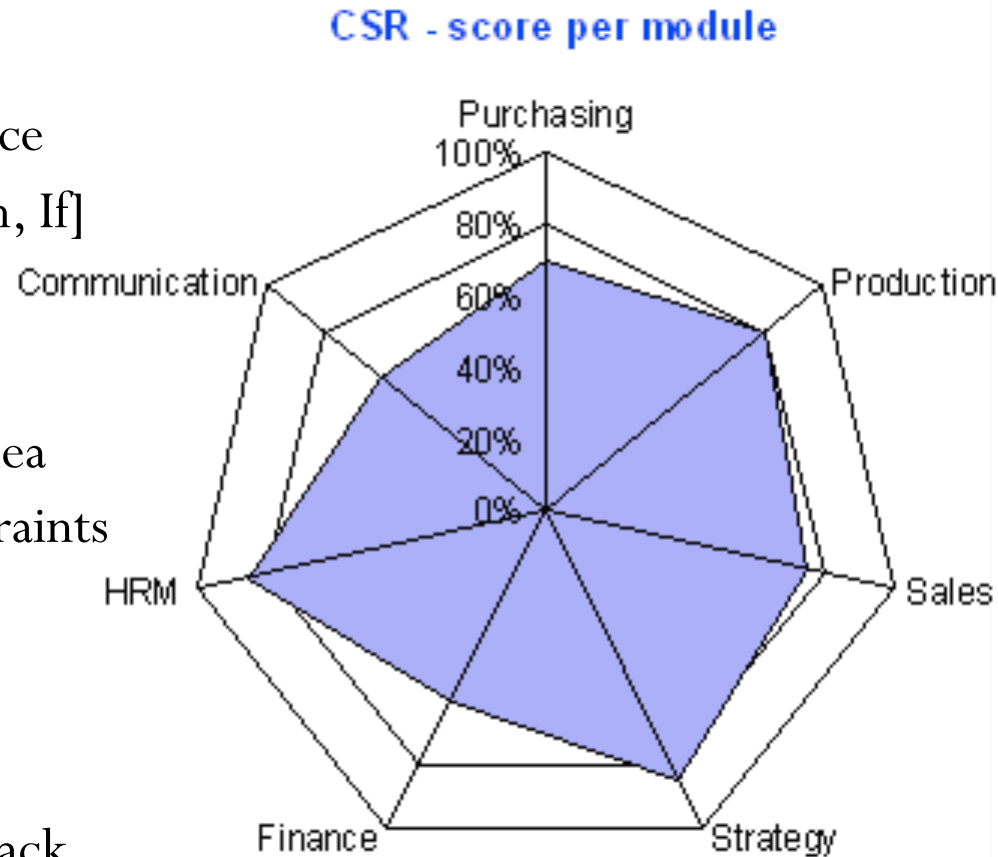
•Figure 1: Real (NON-CONFIDENTIAL version) example of an initial draft of setting the objectives that engineering processes must meet.

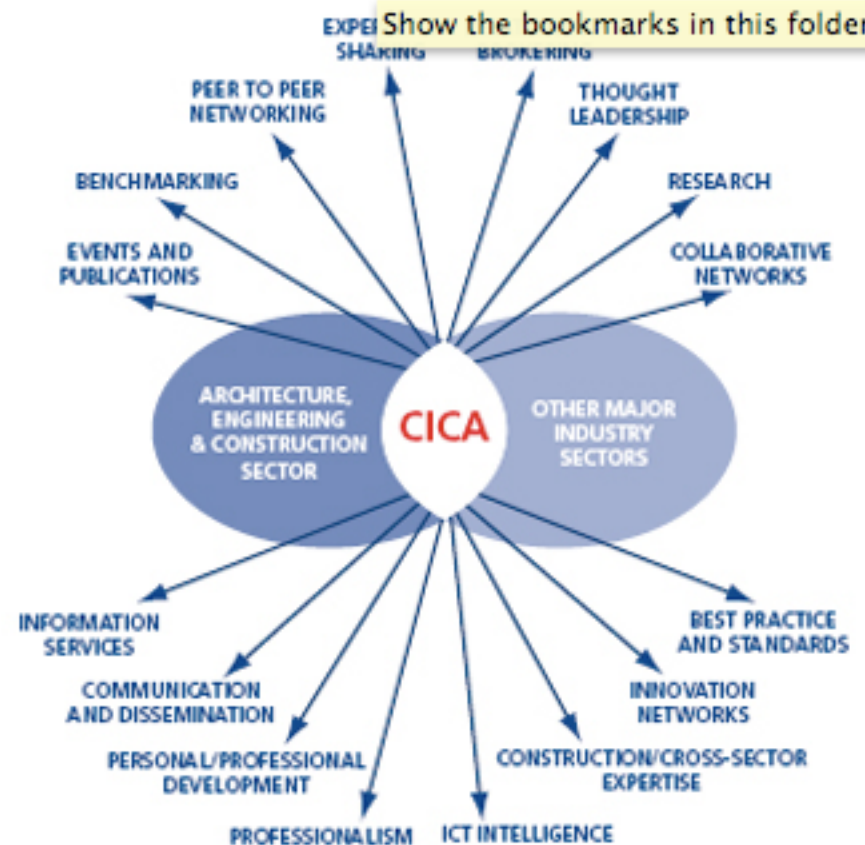| Business objective	Measure	Goal (200X)	Stretch goal ('0X)	Volume	Value	Profit	Cash
Time to market	Normal project time from GT to GT5	<9 mo.	<6 mo.	X		X	X
Mid-range	Min BoM for The Corp phone	<$90	<$30		X	X	X
Platformisation Technology	# of Technology 66 Lic. shipping > 3M/yr	4	6	X		X	X
Interface	Interface units	>11M	>13M	X		X	X
Operator preference	Top-3 operators issue RFQ spec The Corp	1	2			X	X
Productivity							X
Get Torden	Lyn goes for Technology 66 in Sep-04	Yes		X		X	X
Fragmentation	Share of components modified	<10%	<5%		X	X	X
Commoditisation	Switching cost for a UI to another System	>1yr	>2yrs			X	X
Duplication	The Corp share of 'in scope' code in best-selling device	>90%	>95%		X	X	X
Competitiveness	Major feature comparison with MX	Same	Better	X		X	X
User experience	Key use cases superior vs. competition	5	10	X	X	X	X
Downstream cost saving	Project ROI for Licensees	>33%	>66%	X	X	X	X
Platformisation IFace	Number of shipping Lic.	33	55	X		X	X
Japan	Share of of XXXX sales	>50%	>60%	X		X	X
Numbers are intentionally changed from real ones							

September 21, 2011

Business Values Quantified

3. Values are supported by Value Architecture

- Value <u>Architecture</u>: defined as:
  - anything you *implement* with a view to satisfying stakeholder values.

- Value Architecture:
  - includes product/system objectives
    - Which are a 'design' for satisfying stakeholder values
  - Has a multitude of performance and cost impacts
  - can impact a given system differently, depending on what is in the system, or what gets put in later
  - Needs to try to maximize value delivered for resources used.

Show the bookmarks in this folder.

EXPE... BROKERING
SHARING

PEER TO PEER NETWORKING
THOUGHT LEADERSHIP

BENCHMARKING
RESEARCH

EVENTS AND PUBLICATIONS
COLLABORATIVE NETWORKS

ARCHITECTURE, ENGINEERING & CONSTRUCTION SECTOR
CICA
OTHER MAJOR INDUSTRY SECTORS

INFORMATION SERVICES
BEST PRACTICE AND STANDARDS

COMMUNICATION AND DISSEMINATION
INNOVATION NETWORKS

PERSONAL/PROFESSIONAL DEVELOPMENT
CONSTRUCTION/CROSS-SECTOR EXPERTISE

PROFESSIONALISM   ICT INTELLIGENCE
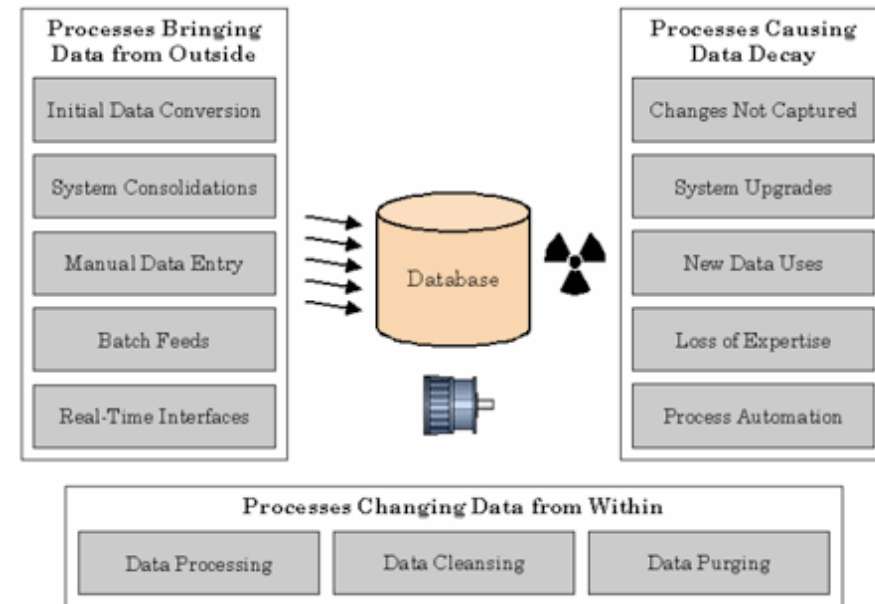
September 21, 2011

4. Value *levels* are determined by *timing*, *architecture effect*, and *resources*

Value <u>levels</u>: defined as:

the degree of satisfaction of value needs.

Value level:

- depends on *when* you observe the level
  - The environment, the people, other system performance characteristics (security, speed, usability)
- depends on the *current incremental power* of *particular value architecture* components
- depends on *resources available* both in development and operation

Processes Bringing Data from Outside	Processes Causing Data Decay
Initial Data Conversion	Changes Not Captured
System Consolidations	System Upgrades
Manual Data Entry	New Data Uses
Batch Feeds	Loss of Expertise
Real-Time Interfaces	Process Automation

Database

Processes Changing Data from Within		
Data Processing	Data Cleansing	Data Purging

September 21, 2011

5. Required Value *levels* can differ for different scopes (where, who)

The level of value needed, and the level of value delivered - for a single attribute dimension (like Ease of Use) can vary for

- different stakeholders
- at different times
  - (peak, holiday, slack, emergency, early implementation)
- for different 'locations'
  - countries, companies, industries

There is nothing simple like 'one level for a

September 21, 2011
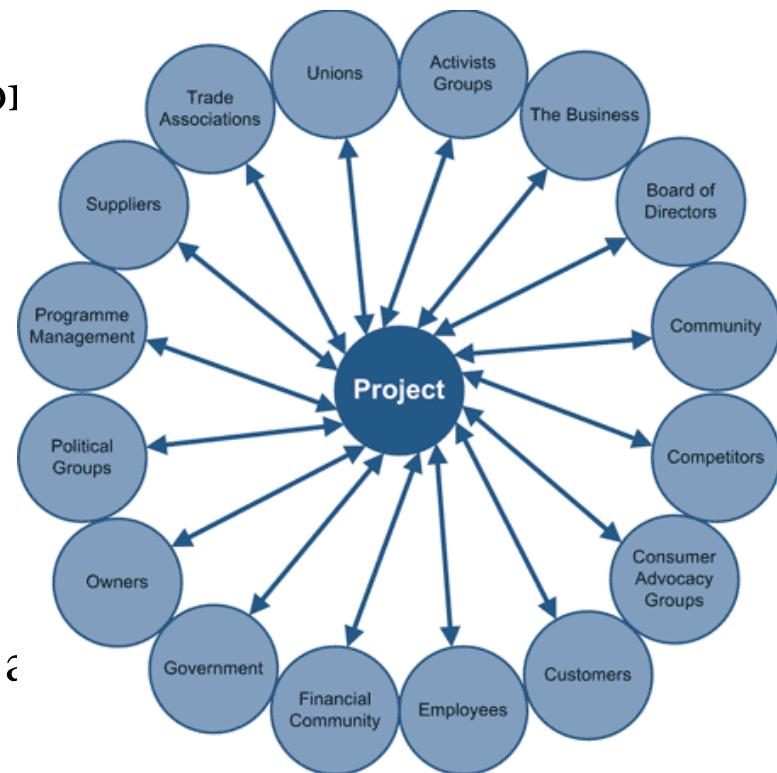
6. Value can be delivered _early_

You do not have to wait until 'the project is done' to deliver useful stakeholder value satisfaction.

You can intentionally target the highest priority stakeholders, and their highest priority value area, and levels.

You can deliver them early and continuously

You can learn what is possible

And what stakeholders really value.

Discover new value ideas

Discover new stakeholders

Discover new levels of satisfaction

Agile Iterative Delivery

Non-agile Project Delivery
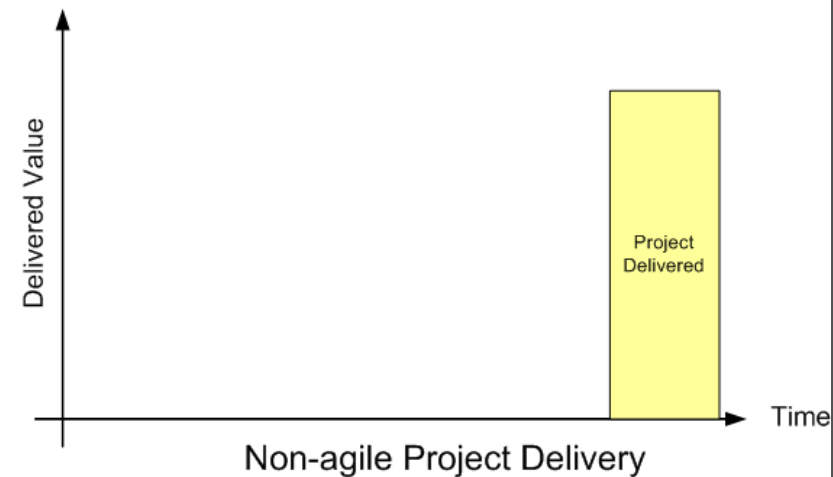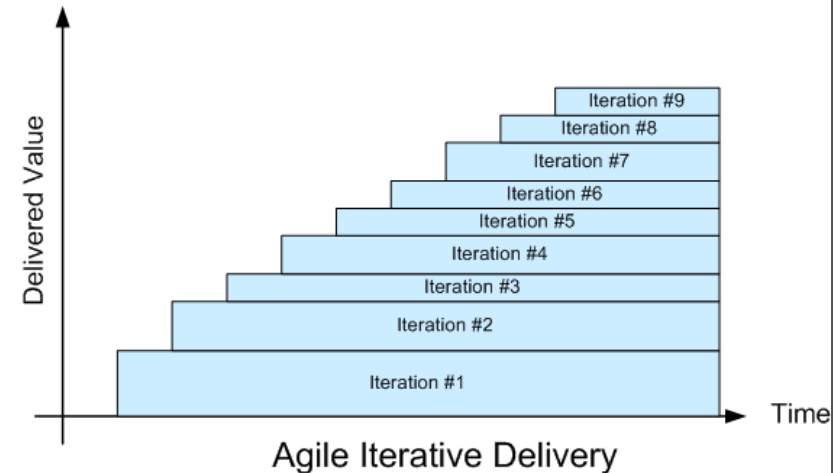
7. Value can be locked in incrementally

- You can increment the value satisfaction
  - *towards* <u>longer term Goal levels</u>
- You can spread the value deliveries
  - that are *proven* in *some* places,
  - more <u>widely</u> in the next increments
- This probably assumes that you have really handed over real results to real people.
  - Not just developed systems without delivery

8. New Values can be discovered
(external news, experience)

- *Expect*, and try to discover,
  - entirely new stakeholder values.
- These will of course emerge *after you start delivering* some satisfaction, because:
  - Stakeholders believe you can help
  - Things *change*

Discovery: Multiple Dimensions — Explore, Affinity, Refer, Serendipity

September 21, 2011
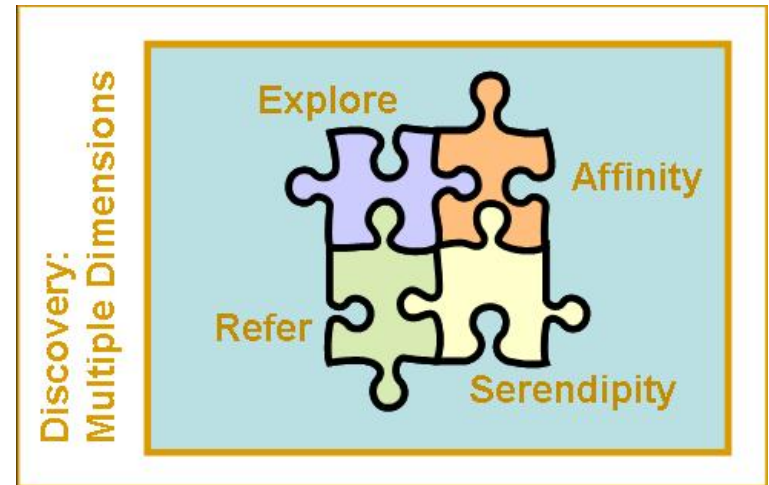
9. Values can be *evaluated* as a function of *architecture* (using 'Impact Estimation')

- It is possible to get an **overview** of
  - the totality of impacts
  - that your **architecture**
  - (all designs and strategies)
  - **might** have
  - on all your defined stakeholder **needs**.

- Use an Impact Estimation table
  - and you will be able to spot *opportunities* for
    - high value and
    - low cost   early deliveries
      - by analyzing the numbers on the table

See next slide
For enlargement

Business Objective	Weight	hardware adaptation	Telephony	Reference designs	IFace	Modularity	Defend vs Technology 66	Tools	User Exper'ce	GUI & Graphics	Security	Defend vs OCD	Enterprise
Time to market	20%	20%	10%	30%	5%	10%	5%	15%	0%	0%	0%	5%	5%
Mid-range	10%	15%	0%	15%	0%	30%	15%	5%	10%	5%	5%	0%	0%
Platformisation Technology	5%	25%	10%	30%	0%	0%	10%	0%	5%	0%	10%	0%	0%
Interface	5%	5%	15%	15%	0%	5%	0%	5%	0%	0%	10%	0%	10%
Operator preference	10%	0%	10%	0%	15%	5%	20%	5%	10%	10%	20%	5%	10%
Get Torden	10%	25%	10%	10%	-10%	0%	20%	0%	10%	-20%	10%	10%	5%
Commoditisation	5%	20%	10%	20%	10%	-20%	25%	15%	0%	0%	5%	10%	5%
Duplication	10%	15%	10%	10%	0%	0%	40%	0%	0%	0%	5%	20%	5%
Competitiveness	5%	10%	15%	10%	0%	10%	20%	10%	10%	20%	10%	0%	10%
User experience	5%	5%	0%	0%	0%	20%	0%	0%	30%	10%	0%	0%	0%
Downstream cost saving	5%	15%	5%	20%	0%	10%	20%	0%	10%	0%	0%	10%	0%
Platformisation IFace	5%	10%	10%	20%	40%	0%	20%	5%	0%	0%	0%	0%	5%
Japan	5%	10%	5%	20%	0%	10%	0%	0%	10%	5%	0%	0%	0%
Contribution to overall result		15%	9%	17%	4%	7%	15%	6%	6%	1%	6%	6%	5%
Cost (£M)		£ 2.85	£ 0.49	£ 3.21	£ 2.54	£ 1.92	£ 2.31	£ 0.81	£ 1.21	£ 2.68	£ 0.79	£ 0.62	£ 0.60
ROI Index (100=average)		106	358	109	33	78	137	148	107	10	152	202	174

Viking Deliverables

© Gilb.com   Agility is the Tool

September 21, 2011

Objectives

Technical Strategies

Defined In earlier slide

"Benefits"

Strategy Impacts on Objectives

Cost

Viking Deliverables

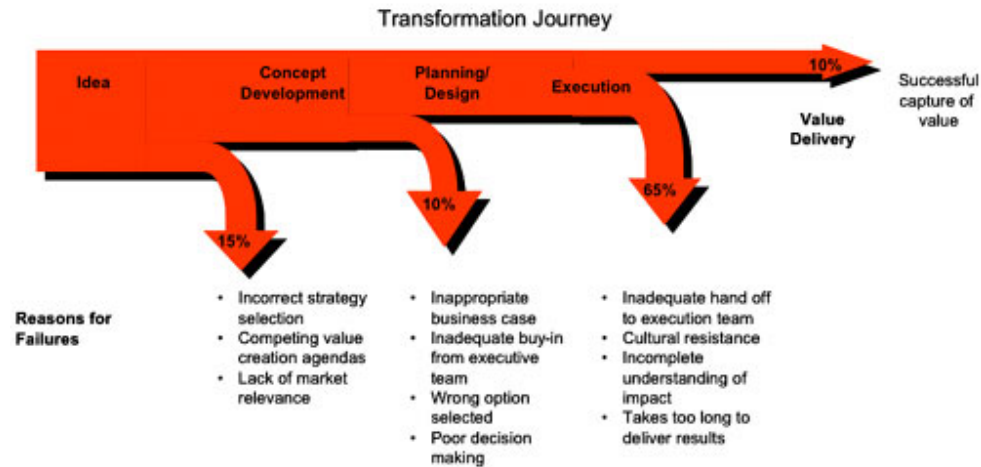| Business Objective	hardware adaptation	Telephony	Reference designs	IFace	Modularity	Defend vs Technology 66	Tools	User Exper'ce	GUI & Graphics	Security	Defend vs OCD	Enterprise
Time to market	20%	10%	30%	5%	10%	5%	15%	0%	0%	0%	5%	5%
Mid-range	15%						5%	10%	5%	5%	0%	0%
Platformisation Technology	25%	10%	30%	0%		10%	0%	5%	0%	10%	0%	5%
Interface	5%	15%	15%	0%	5%	0%	5%	0%	0%	10%	0%	10%
Operator preference	0%					20%	5%	10%	10%	20%	5%	10%
Get Torden	25%	10%	10%	-10%	0%	20%	0%	10%	-20%	10%	10%	5%
Commoditisation	20%	10%	20%	10%	-20%	25%	15%	0%	0%	5%	10%	5%
Duplication	15%		0%	0%	0%	40%	0%	0%	0%	5%	20%	5%
Competitiveness	10%	15%	20%	0%	10%	20%	10%	10%	20%	10%	10%	10%
User experience	5%	0%	0%		20%		0%	30%	10%	0%	0%	0%
Downstream cost saving	15%							10%	0%	0%	10%	5%
Platformisation IFace	10%	10%	20%	40%	0%	20%	5%	0%	0%	0%	0%	5%
Japan	10%	5%	20%	0%	10%	0%	0%	10%	5%	0%	0%	0%
Contribution to overall result	15%	9%	17%	4%	7%	15%	6%	6%	1%	6%	6%	5%
Cost (£M)	£ 2.85	£ 0.49	£ 3.21	£ 2.54	£ 1.92	£ 2.31	£ 0.81	£ 1.21	£ 2.68	£ 0.79	£ 0.62	£ 0.60
ROI Index (100=average)	106	358	109	33	78	137	148	107	10	152	202	174

358 !

10. Value delivery will attract resources.

- If you are really good at delivering value
  - You can expect to attract
    - even more funding
  - Managers like
    - to be credited with success
  - Money seeks
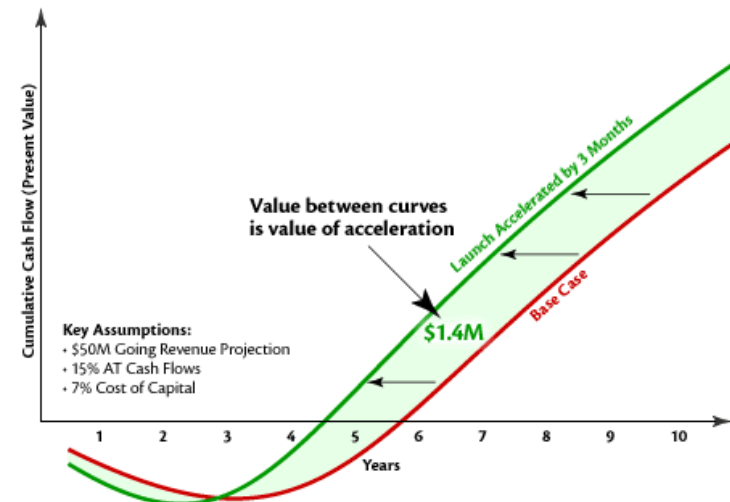   - best interest rates

September 21, 2011

Gilb's Value Manifesto: A Management Policy?

1. **Really useful value, for real stakeholders will be defined measurably.**

   No nice-sounding emotive words please.

2. **Value will be seen in light of total long term costs**

   as a decent return on investment.

3. **Powerful management devices, like motivation and follow-up, will make sure that the value for money is really delivered –**

   or that the failure is punished, and the success is rewarded.

4. **The value will be delivered evolutionarily –**

   not all at the end.

5. **That is, we will create a stream of prioritized value delivery to stakeholders, at the *beginning* of our value delivery projects;**

   and continue as long as the real return on investment is suitably large.

6. **The CEO is primarily responsible for making all this happen effectively.**

   1. **The CFO will be charged with tracking all value to cost progress.**
   2. **The CTO and CIO will be charged with formulating all their efforts in terms of measurable value for resources.**

Transformation Journey

Idea → Concept Development (15%) → Planning/ Design (10%) → Execution (65%) → Value Delivery (10%) → Successful capture of value

Reasons for Failures

- Incorrect strategy selection
- Competing value creation agendas
- Lack of market relevance

- Inappropriate business case
- Inadequate buy-in from executive team
- Wrong option selected
- Poor decision making

- Inadequate hand off to execution team
- Cultural resistance
- Incomplete understanding of impact
- Takes too long to deliver results

Source: Survey 100 Global Companies 2001-2002

Cumulative Present Value of Accelerating Cash Flows

Value between curves is value of acceleration

Launch Accelerated by 3 Months

Base Case

$1.4M

Key Assumptions:
- $50M Going Revenue Projection
- 15% AT Cash Flows
- 7% Cost of Capital

The Value Delivery Problem

- **Sponsors who order and pay for systems engineering projects,**
  - must <u>justify</u> their money spent
  - based on the expected consequential effects (hereafter called 'value') of the systems.

- **The <u>*value*</u> of the technical system is often expressed**
  - in presentation slides and requirements documents
  - as a set of nice-sounding words,
  - under various titles such as "System Objectives", and "Business Problem Definition"

September 21, 2011

Some Assertions

Assertion 1. When top management allows large projects to proceed, with such badly formulated primary objectives, then
- they are responsible as managers for the outcome (failure).
- They cannot plead ignorance.

Assertion 2. The failure of technical staff (project management) to react to the lack of primary objective formulation by top management is also a total failure to do reasonable systems engineering.
- Management might have a poor requirements culture, but we should routinely save them from themselves.

Assertion 3. Both top managers and project personnel can be trained and motivated to clarify and quantify critical objectives routinely.
- But until the poor external culture of education and practice changes, it may take strong CEO action to make this happen in your corporation.
- My experience is that no one else will fight for this.

Assertion 4. All top level system performance improvements, are by definition, variables.
- So, we can expect to define them quantitatively.
- We can also expect to be able to measure or test the current level of performance.
- Words like 'enhanced', 'reduced', 'improved' are not serious systems engineering requirements terms.

September 21, 2011

Slides moved from front to end

© Gilb.com   Agility is the Tool

Value Planning
The _Organizational_ Components

Product Management
(deciding what the product should be)

Scrum Teams
(building the product)

Product Owner

System Architect

Scrum Master

Team Members

September 21, 2011

Value Planning

The _Inputs_

Product Management

(deciding what the product should be)

Scrum Teams

(building the product)

Stakeholders and their Needs

(like: Potential New Users, Usability)

Long Term Quality Needs

(like Portability, Security, Adaptability)

Requirements

(what to build, how well to build)

High Level and Super-ordinate

Designs and Architecture

(how to build, solutions given from others)

September 21, 2011

Value Planning

The Work Products - *outputs*

Product Management
(deciding what the product should be)

Product Owner:

__Requirements__, particularly top critical few improvement requirements

Strategies, __Designs__, Solutions

(How we propose to deliver the improvements)

System Architect:

__Technical Architecture__ to support long term

(like suppliers, interfaces, platforms, languages)

Scrum Teams
(building the product)

Scrum Master:

ensure team __empowerment__

Team Members: (IT)

Code, Tests, System Improvements, Reports on progress, Work Process Improvements

September 21, 2011

The *Product Management* <u>Process</u>
Deciding the exact product content

Gather relevant inputs: Analyze The Market & Related Environment

Stakeholders   Stakeholder needs

Clarify Needs & Organize the Information:

= Clear and Complete Requirements

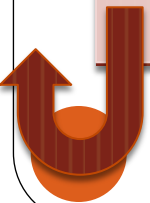Quantify Improvements and Constraints   Add info about risks, sources, priorities

Decide how to deliver the requirements – Product Design

Strategies, Design, Architecture   Estimate expected Impacts on Product Improvements and costs

Iteration assumed!

Analysis: by PM
What You 'have to' know

Market Needs

> **Product Characteristics**
>
> **How good? Qualities**
>
> **Top 10 Critical Improvements**

> **Service Characteristics**
>
> **(help, training, fault support, sales channels, …)**

Other Needs

> **Organizational Needs**
>
> **IT Environment, Sales and marketing Environment, Distribution and Partners, International considerations, …**

> **External Environment Needs**
>
> **(legal, co-operation, image, ..)**

September 21, 2011

Requirements:
Determining What You <u>Want</u>

What you need to determine

- Top Level Critical Objectives
- All other critical requirements

How <u>well</u> you need to determine requirements

- Quantified, Unambiguous, Clear, Testable, Agreed and Approved, Quality Controlled
- With supporting detail to allow analysis, risk understanding, prioritization

September 21, 2011

Design: to meet Requirements:

What you have to design

Choose specific designs of product and service

(detailed enough to hand over to development team**)**

Choose specific architectures to deal with long term needs

(platforms, interfaces, processes, organizational structures, rewards)

How well you have to design it

So that **you reasonably understand all critical** attributes and costs

($\pm20\%$?)

So that the overall long term implications of the product are understood

(recruitment, partnering, international deals)

September 21, 2011

Building:
The _development_ team

What do you have to do?

Build Product

(Software, Dataware, Docuware)

Validate Product

(Does it work well enough?)

How _well_ do you have to do it

To meet all targets, and constraints – for quality and performance.

For new increments, and total system

To reflect on both product attributes and process problems.

To improve their own work environment.

To improve the design, estimates and requirements.

September 21, 2011

Implementation: Integration: Delivery to Market

What the team has to do

Integrate next increment into existing product and field/Beta trial it

Deliver to market as finished product change

How well it has to be done

So that it normally is clean (no bugs!) and impressive. So that we learn, and can tune it, before final market delivery

Rock solid. No problems. Clear improvement to all customers

September 21, 2011