Agile Manifesto: 10 Years On, What can we do better? – A rewrite

The intent of Agile has always been to focus on delivering value to our stakeholders.

But,

I think we need to be a lot more specific about what this means, because some people think it means delivering bug free code to a user or customer, even if the stakehoder gets no real value!

By Tom Gilb
Tom@gilb.com
www.gilb.com

ACCU 2011
Conference
Held on 13 April

These slides will be available at ACCU site and Gilb.com downloads, Slides
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.
Value Driven Planning Principles in Detail:

Published in www.agilerecord.com
2010 Part 1 and 2

• *Value-Driven Development: Principles and Values – Agility is the Tool, Not the Master.*


• Part 2

• “Values for Value”

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.
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
Figure 1: Real (NON-CONFIDENTIAL version) example of an initial draft of setting the objectives that engineering processes must meet.

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

Numbers are intentionally changed from real ones.
3. Values are supported by Value Architecture

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

Value levels: 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
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 all’
• 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
• 7. Value can be locked in incrementally

• You can increment the value satisfaction
  – towards longer term Goal levels

• You can spread the value deliveries
  – that are proven in some places,
  – more widely 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*
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
**Strategy Impact Estimation:**
for a $100,000,000 Organizational Improvement Investment

### Objectives

<table>
<thead>
<tr>
<th>Business Objective</th>
<th>Defined in earlier slide</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time to market</td>
<td></td>
</tr>
<tr>
<td>Mid-range</td>
<td></td>
</tr>
<tr>
<td>Platformisation Technology</td>
<td></td>
</tr>
<tr>
<td>Interface</td>
<td></td>
</tr>
<tr>
<td>Operator preference</td>
<td></td>
</tr>
<tr>
<td>Get Torden</td>
<td></td>
</tr>
<tr>
<td>Commoditisation</td>
<td></td>
</tr>
<tr>
<td>Duplication</td>
<td></td>
</tr>
<tr>
<td>Competitiveness</td>
<td></td>
</tr>
<tr>
<td>User experience</td>
<td></td>
</tr>
<tr>
<td>Downstream cost saving</td>
<td></td>
</tr>
<tr>
<td>Platformisation IFace</td>
<td></td>
</tr>
<tr>
<td>Japan</td>
<td></td>
</tr>
</tbody>
</table>

### Benefits on Objectives

<table>
<thead>
<tr>
<th>Resource</th>
<th>Hardware adaptation</th>
<th>Telephony</th>
<th>IFace</th>
<th>Modularity</th>
<th>Defend vs Technology</th>
<th>Tools</th>
<th>User Experience</th>
<th>GUI &amp; Graphics</th>
<th>Security</th>
<th>Defend vs OCD</th>
<th>Enterprise</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time to market</td>
<td>20%</td>
<td>10%</td>
<td>30%</td>
<td>5%</td>
<td>10%</td>
<td>5%</td>
<td>15%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>5%</td>
</tr>
<tr>
<td>Mid-range</td>
<td>15%</td>
<td>5%</td>
<td>5%</td>
<td>10%</td>
<td>5%</td>
<td>10%</td>
<td>5%</td>
<td>5%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>Platformisation Technology</td>
<td>25%</td>
<td>10%</td>
<td>30%</td>
<td>0%</td>
<td>10%</td>
<td>0%</td>
<td>10%</td>
<td>0%</td>
<td>10%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>Interface</td>
<td>5%</td>
<td>15%</td>
<td>15%</td>
<td>0%</td>
<td>5%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>10%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>Operator preference</td>
<td>0%</td>
<td>20%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>20%</td>
<td>0%</td>
<td>0%</td>
<td>10%</td>
<td>10%</td>
<td>20%</td>
</tr>
<tr>
<td>Get Torden</td>
<td>25%</td>
<td>10%</td>
<td>10%</td>
<td>-10%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>10%</td>
<td>-20%</td>
<td>10%</td>
<td>10%</td>
</tr>
<tr>
<td>Commoditisation</td>
<td>20%</td>
<td>10%</td>
<td>10%</td>
<td>-10%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>10%</td>
<td>10%</td>
<td>10%</td>
<td>10%</td>
</tr>
<tr>
<td>Duplication</td>
<td>15%</td>
<td>0%</td>
<td>20%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>5%</td>
<td>0%</td>
<td>20%</td>
</tr>
<tr>
<td>Competitiveness</td>
<td>10%</td>
<td>15%</td>
<td>20%</td>
<td>0%</td>
<td>0%</td>
<td>10%</td>
<td>10%</td>
<td>0%</td>
<td>0%</td>
<td>10%</td>
<td>10%</td>
</tr>
<tr>
<td>User experience</td>
<td>5%</td>
<td>15%</td>
<td>10%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>Downstream cost saving</td>
<td>15%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>Platformisation IFace</td>
<td>10%</td>
<td>10%</td>
<td>20%</td>
<td>40%</td>
<td>0%</td>
<td>20%</td>
<td>5%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>Japan</td>
<td>10%</td>
<td>5%</td>
<td>20%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>10%</td>
<td>5%</td>
<td>0%</td>
<td>0%</td>
</tr>
</tbody>
</table>

### Contribution to overall result

<table>
<thead>
<tr>
<th>Resource</th>
<th>Contribution to overall result</th>
<th>Cost (£M)</th>
<th>ROI Index (100=average)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time to market</td>
<td>15%</td>
<td>£ 2.85</td>
<td>106</td>
</tr>
<tr>
<td>Mid-range</td>
<td>9%</td>
<td>£ 0.49</td>
<td>358</td>
</tr>
<tr>
<td>Platformisation Technology</td>
<td>17%</td>
<td>£ 3.21</td>
<td>109</td>
</tr>
<tr>
<td>Interface</td>
<td>4%</td>
<td>£ 2.54</td>
<td>33</td>
</tr>
<tr>
<td>Operator preference</td>
<td>7%</td>
<td>£ 1.92</td>
<td>76</td>
</tr>
<tr>
<td>Get Torden</td>
<td>15%</td>
<td>£ 2.31</td>
<td>137</td>
</tr>
<tr>
<td>Commoditisation</td>
<td>6%</td>
<td>£ 0.81</td>
<td>146</td>
</tr>
<tr>
<td>Duplication</td>
<td>6%</td>
<td>£ 1.21</td>
<td>107</td>
</tr>
<tr>
<td>Competitiveness</td>
<td>1%</td>
<td>£ 2.68</td>
<td>10</td>
</tr>
<tr>
<td>User experience</td>
<td>6%</td>
<td>£ 0.79</td>
<td>152</td>
</tr>
<tr>
<td>Downstream cost saving</td>
<td>6%</td>
<td>£ 0.62</td>
<td>202</td>
</tr>
<tr>
<td>Platformisation IFace</td>
<td>5%</td>
<td>£ 0.60</td>
<td>174</td>
</tr>
</tbody>
</table>

April 13, 2011

SPA BCS 2 Sept 2009 London

14
10. Value delivery will attract

• 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
End of ACCU Lecture

• In practice after 40 minutes.
• The rest of the slides are for additional documentation
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.

Source “Value Delivery in Systems Engineering” available at www.gilb.com
The Value Delivery Problem

• Sponsors who order and pay for systems engineering projects, must justify their money spent based on the expected consequential effects (hereafter called ‘value’) of the systems.

• The value 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”
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.
THESE ARE SAME PRINCIPLES AND VALUES

• BUT WITH NO DETAILED TEXT FOR EACH AND NO GILB EXAMPLES

• FOR USE WHEN LITTLE TIME AND NOT TOO DEMANDING AUDIENCE
Gilb’s Ten Key Agile Principles

to avoid bureaucracy and give creative freedom (Summary)

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

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

Make sure those results are **business** results, not technical

Align your project with your financial sponsor’s interests!

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

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

Select designs with the best impacts for their costs, do them first.

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

Change designs, based on quantified experience of implementation

Change requirements, based on quantified experience, new inputs

Involve the stakeholders, every week, in setting quantified goals

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

Copyright 2004-8 Gilb, may be used citing source

SPA BCS 2 Sept 2009 London
Gilb’s Ten Key Agile Principles (Sum)
to avoid bureaucracy and give creative freedom

Main Idea:
Get early and frequent real stakeholder net value delivered

<table>
<thead>
<tr>
<th></th>
<th>VALUE TO CREATE</th>
<th>VALUE TO PRESERVE</th>
<th>VALUE TO SACRIFICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPLOYEES</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CUSTOMERS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SUPPLIERS AND PROFESSIONAL ADVISERS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>INVESTORS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRADES UNIONS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GOVERNMENT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MEDIA</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COMMUNITY</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OTHER STAKEHOLDER GROUPS</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Control projects by quantified critical-few results. 1 Page total!

(not stories, functions, features, use cases, objects, ..)
Make sure those results are business results, not technical

Align your project with your financial sponsor’s interests!
Give developers freedom, to find out *how* to deliver those results.
Estimate the impacts of your designs, on your quantified goals.
Select designs with the best impacts for their costs, do them first.

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

Decompose the workflow, into weekly (or 2% of budget) time boxes.
Change designs, based on quantified experience of implementation
Change requirements, based on quantified experience,
Involve the stakeholders, every week, in setting quantified goals.
Involve the stakeholders, every week, in *actually using* increments.
My 10 Agile Values? (Summary)

• 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 systems engineer, not a just programmer (a ‘Softcrafter’).
    • Do not be limited by your craft background, in serving your paymasters
My 10 Agile Values? (Detail)

- Simplicity
- Communication
- Feedback
- Courage
Simplicity

– 1. Focus on real stakeholder values
– 2. Communicate stakeholder values quantitatively

Kura - Kano Rice Value Chain
Estimate Often

3. Estimate expected results and costs for weekly steps
Feedback

4. Generate results, weekly, for stakeholders, in their environment
Measure Critical Stuff

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

![Comic Strip](chart.png)
Learn from Deviations

- 6. Analyze deviation from your initial estimates
Courage

– 7. Change plans to reflect weekly learning
Deliver Value Now

8. Immediately implement valued stakeholder needs, next week
   • *Don’t wait, don’t study (analysis paralysis), don’t make excuses.*
   • *Just Do It!*
Tell Stakeholders What’s next

9. Tell stakeholders exactly what you will deliver next week
If it works, do it!

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

- Ecstatic Stakeholder!