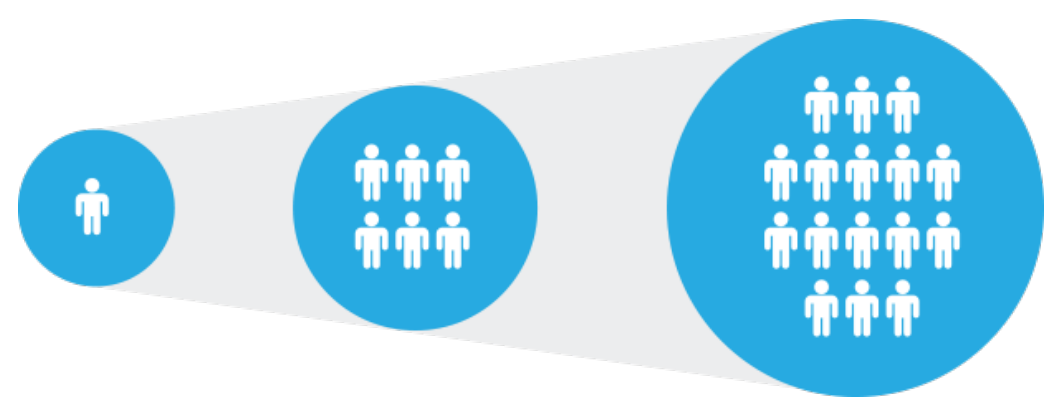
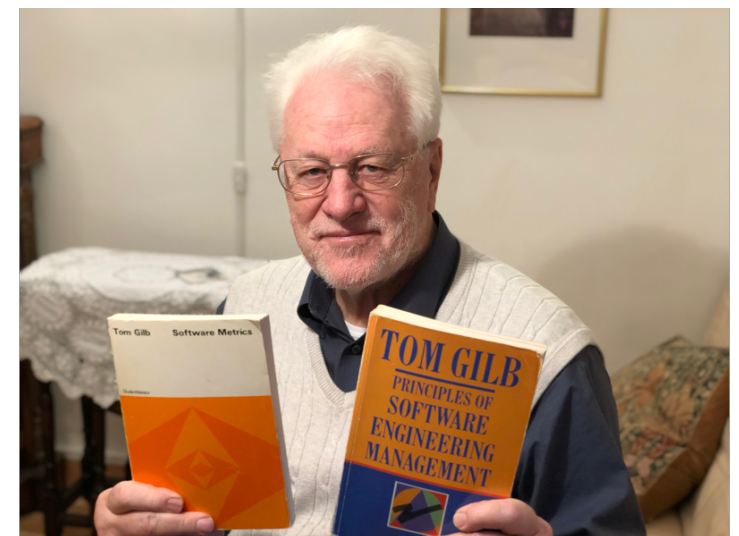




Scalability Metrics: An Engineering Structure, and Principles, for an Agile World

for
Smidig skalerbarhet
Nettverksmøte 5. juni 2018 i Oslo
Et seminar om håndtering av skalerbarhet i smidig utvikling

15 minutters innlegg presentert
av
tom@gilb.com



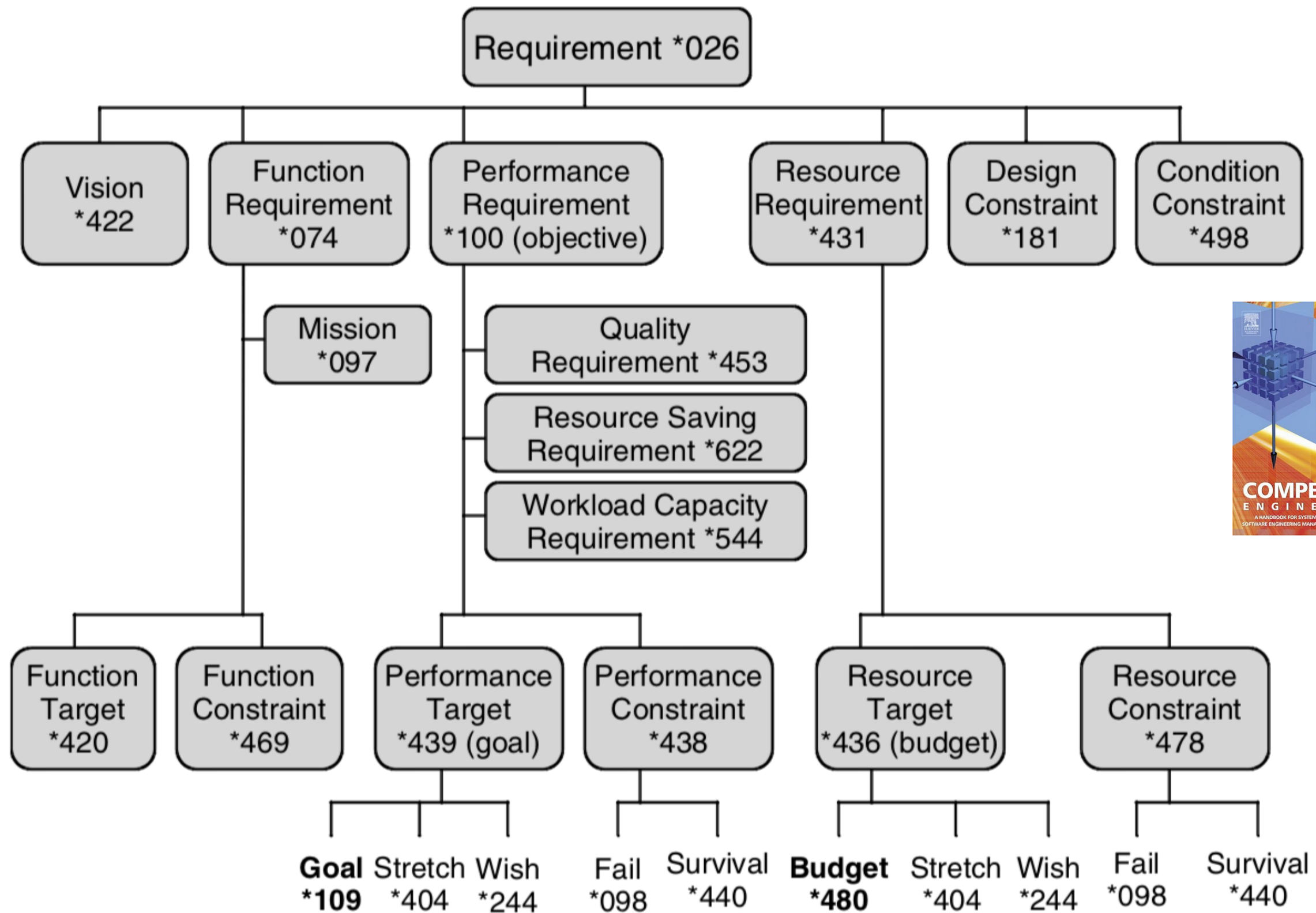
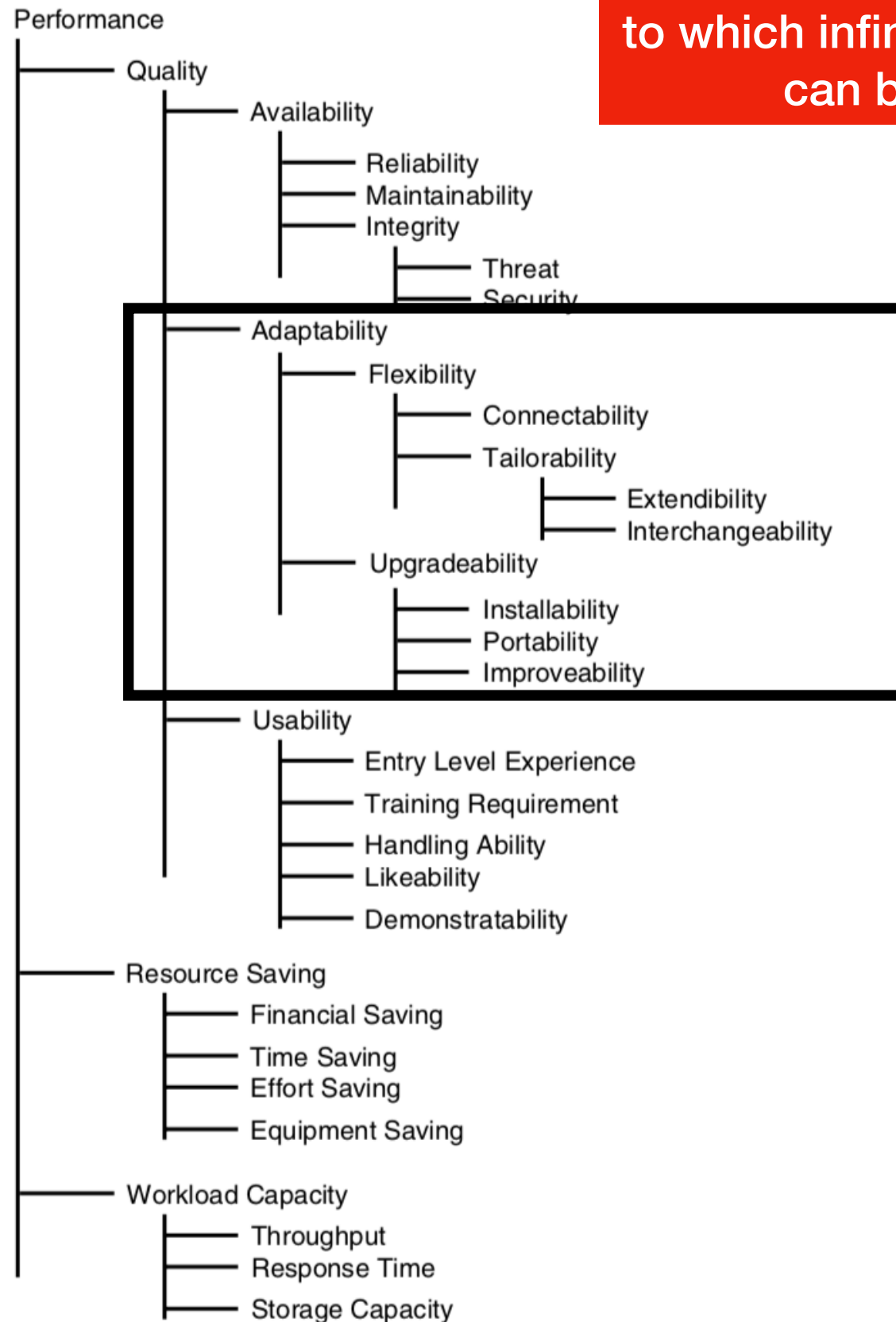


Figure G20

Requirement Concepts.

**‘Performance’ attributes, a Generic set ;
to which infinite tailored definitions of attributes
can be added for a given system**



**Adaptability
← --- Metrics**



Suggested scales of measure
for all Performance attributes in this
diagram, are provided in Chapter 5
of 'Competitive Engineering' book

Figure 5.3

One decomposition possibility for performance attributes with emphasis on the detail of the quality attributes.

Scalability Principles

© 2018 Tom@Gilb.com

- **MEASURABLE:** Scalability is a *measurable and observable* quality attribute, or set of scalability attributes, of any system
- **DESIGNABLE:** Scalability *can be designed (engineered, architected)* into a system
- **SCALE-FREE:** If all required system attributes are maintained or improved, at *any* degree or type of scale, then the system is '100% scalable', or - the system is 'scale free'
- **DEGRADATION:** Scalability is related to the 'defined set of *system attributes* which are potentially degraded by scaling', at a defined set of *scale change degrees**:
 - *low attribute degradation = high scalability*
- **SIDE-EFFECTS:** We might specify 'fragile attributes , which we worry will be negatively impacted by scaling', in our scalability evaluations, but this does not limit our professional responsibility, for areas of concern, which *might also* be impacted, aka 'scalability side-effects', by scale changes.
- **UNCERTAINTY:** Uncertainty of future needs and changes, implies that Scaling strategies should be 'open-ended' (low cost modular options) rather than 'built-in up-front costs'

*defined **set of scale changes**: changes can include ~~A~~ volume, speed, backup activity, diagnosis activity, security activity

Scalability Definition

© 2018 tom@Gilb.com



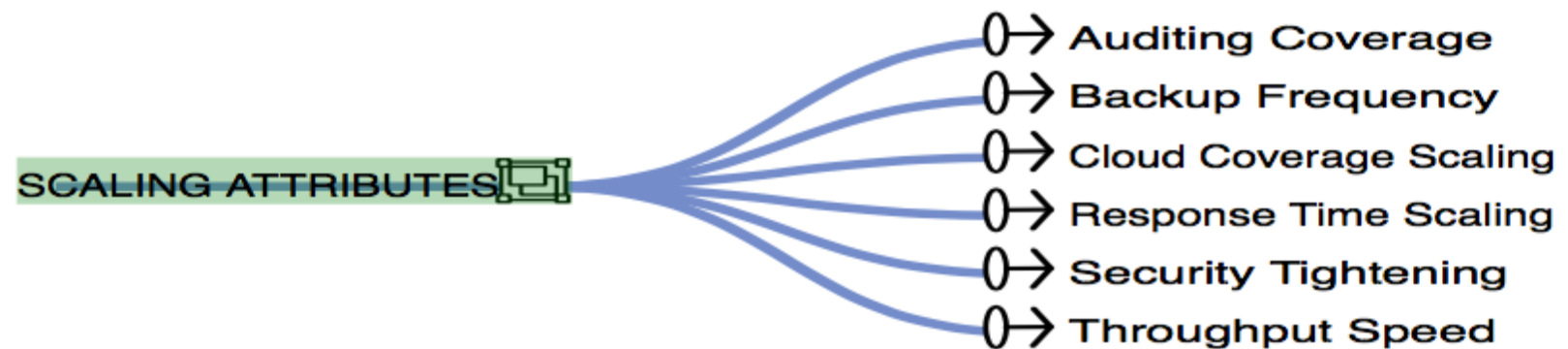
- **Scalability is a ‘system quality’,**
 - **which includes different *types*** of scaling**
 - **and different *degrees* of scaling**
 - **for defined stakeholders**
 - **using a defined set of scaling strategies**

** types of scaling include, but are not limited to:

volume, speed, backup activity, diagnosis activity, security activity. See next slide for bigger list of types.

Types of 'Scaling'

1. ***transaction volume,***
2. ***database size***
3. ***throughput speed,***
4. ***responsiveness***
5. ***backup activity,***
6. ***diagnosis activity,***
7. ***security activity***
8. ***auditing activity***
9. ***self test activity***
10. ***concurrent partner
system activity or
scaling***
11. ***etc.***



Tag:

Auditing Coverage

Level:

Business

Type:

Value

0.0.1

Label?

by tomgilb - 2 minutes ago

Is Part Of: [SCALING ATTRIBUTES](#)

Ambition Level: ability to turn on any degree of any type of automated auditing, without system problems or failure to meet to..

Scale: (by tomgilb - 3 minutes ago) 0

% Increase in Automatic Auditing by [Audit Type] over the Minimum Baseline for [Duration]

Short Description: % Audit Scaling, **Time Units:** Calendar Date

Audit Type: defined as:

Financial, Dates, Identities, Outstanding Balances, Complaints, ...

Automatic Auditing: defined as:

Use of a computer, central or decentralized, or cloud to perform auditing checks for consistency, validity, suspiciousness and other items of interest to an audit function

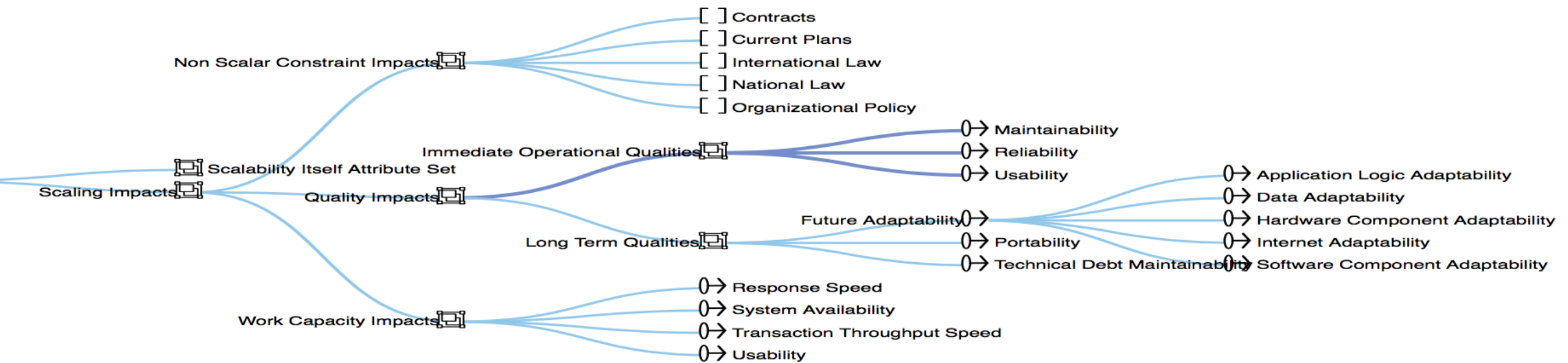
Duration: defined as:

Less than 1 hour, Less than 1 day, Undefined duration, Unknown Duration, Inntil Further Notice, Probably Permanently

Minimum Baseline: defined as:

Zero activity, Minimal allowed activity, Previous status level

Scalability. Auditing Coverage
example of defining a scaling measure



A systematic generic structure of some of the quality variables we might consider when modelling a scalability problem

The '1 liner' specification format

Tag:

Level:

Type:

Label?

Is Part Of:  [Immediate Operational Qualities](#)

 **Ambition Level:** minimise technical debt at all times, and do not incur significantly bad maintenance problems

 **Scale:** Mean time to carry out a defined [Type Of Repair] to a defined [System] using defined [Repair Method]

 **Stakeholders:** Contracted Software Builders, QA Release Testers.

 **Status:** Level: **10** ± 5 ?? Hours to Fix Correctly [Type Of Repair = **All**, System = **Backup Systems**, Repair Method = **All**]

 **Wish:** Level: **1** ± 0.5 ? Hours to Fix Correctly [Type Of Repair = **Code**, System = **Application Code**, Repair Method = **Programmer changes code**]

 **Tolerable:** Level: **5** ± 4 ? Hours to Fix Correctly [Type Of Repair = **Code**, System = **Application Code**, Repair Method = **Programmer changes code**]

Smaller font and more complete text of the above enlargement of the requirement

Tag:

Level:

Type:

Label? (by tomgilb - a few seconds ago)

Is Part Of:  [Immediate Operational Qualities](#)

 **Ambition Level:** minimise technical debt at all times, and do not incur significantly bad maintenance problems due to scaling up or down

 **Scale:** Mean time to carry out a defined [Type Of Repair] to a defined [System] using defined [Repair Method] under defined [Conditions] until Successful Fix.

 **Stakeholders:** Contracted Software Builders, QA Release Testers.

 **Status:** Level: **10** ± 5 ?? Hours to Fix Correctly [Type Of Repair = **All**, System = **Backup Systems**, Repair Method = **All**, Conditions = **10X Scale Up**] When 29th May 2018

 **Wish:** Level: **1** ± 0.5 ? Hours to Fix Correctly [Type Of Repair = **Code**, System = **Application Code**, Repair Method = **Programmer changes code**, Conditions = **10X Scale Up**] When 29th May 2018

 **Tolerable:** Level: **5** ± 4 ? Hours to Fix Correctly [Type Of Repair = **Code**, System = **Application Code**, Repair Method = **Programmer changes code**, Conditions = **10X Scale Up**] When 29th May 2018

Tag: Maintainability

Level: Product

Type: Value

[Label?](#)

(by tomgilb -)

Is Part Of: [Immediate Operational Qualities](#)

Ambition Level: minimise technical debt at all times, and do not incur significantly bad maintenance problems due to scaling up or down

Tag **Scale:** Change...

(by ton)

Scale Description: ?

Mean time to carry out a defined [Type Of Repair] to a defined [System] using defined [Repair Method] under defined [Conditions] until

Conditions: defined as:

10X Scale Up, 10 Scale Down

Definition of the Scale Qualifiers

Repair Method: defined as:

Programmer changes code, Testers Validate Change, Data Base Recovery, Hardware Replacement,

Successful Fix: defined as:

Initially succeeds in all QC and Tests, and pilots; and never is found a bad fix thus far

System: defined as:

Database, Application Code, Software Components, Internet, Backup Systems

Type Of Repair: defined as:

Code, Software Component, Database Recovery, Database Correction, Database Update

Short Description:

Hours to Fix Correctly

Time Units:

Calendar

Tag: Maintainability

Level: Product

Type: Value

[Label?](#)

(by tomgilb - a minute)

Is Part Of: [Immediate Operational Qualities](#)

Ambition Level: minimise technical debt at all times, and do not incur significantly bad maintenance problems due to scaling up or down

Scale: Mean time to carry out a defined [Type Of Repair] to a defined [System] using defined [Repair Method] under defined [Conditions] until S

Stakeholders: Contracted Software Builders, QA Release Testers.

Status: Level: 10 ±5 ?? Hours to Fix Correctly [Type Of Repair = All, System = Backup Systems, Repair Method = All, Conditions = 10X Sca

Tag [.Wish:](#) Change... (by tomgilb - a

Scale Level: Hours to Fix Correctly

1

±0.5 ?

By When:

29/05/2020

Date format: dd/mm/yyyy (e.g. "5/2/2017" for 5th Feb 2017)

Qualifiers: [Copy from...](#)

[Type Of Repair] =

* Code

[Repair Method] =

* Programmer changes code

[System] =

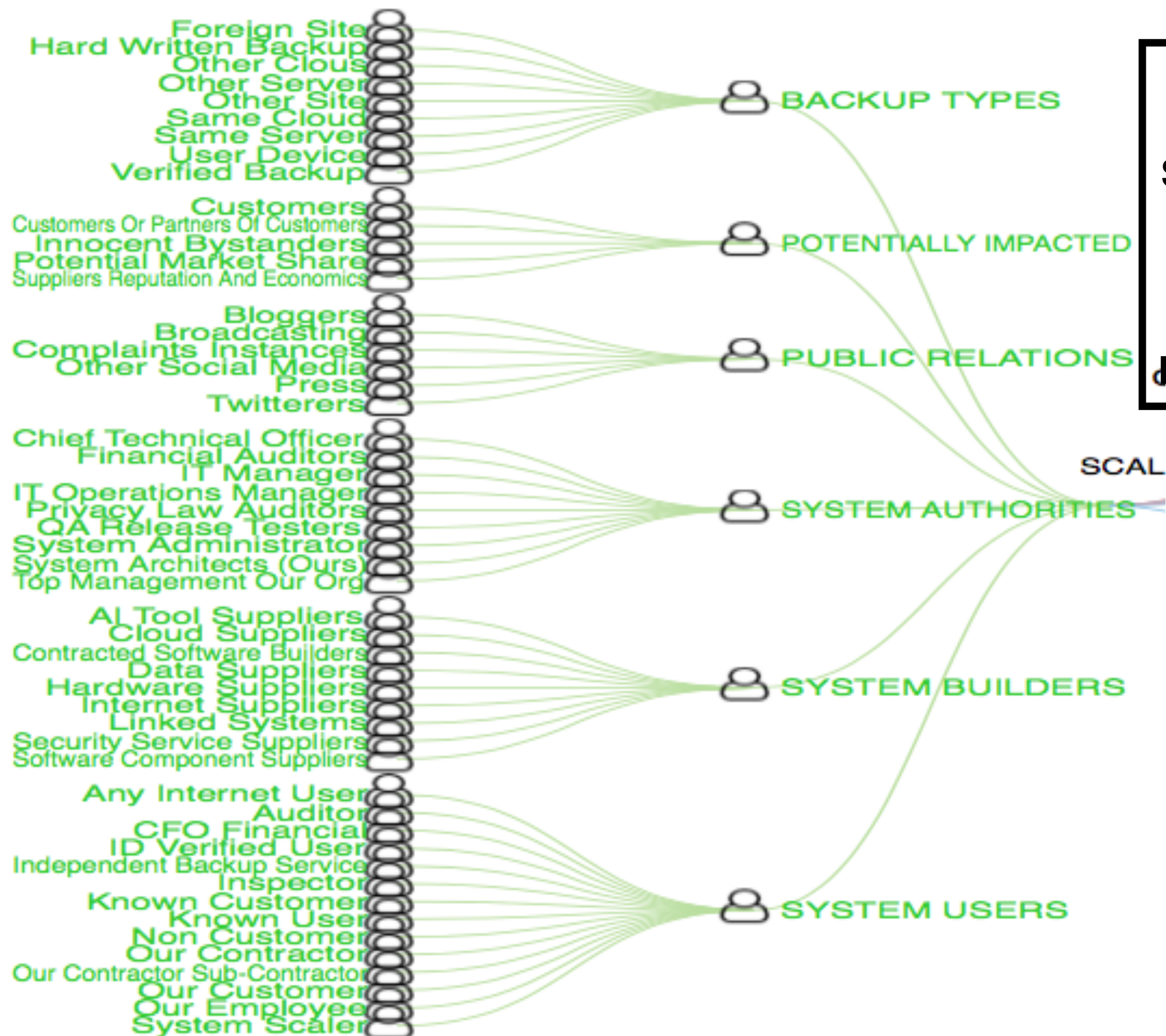
* Application Code

[Conditions] =

* 10X Scale Up

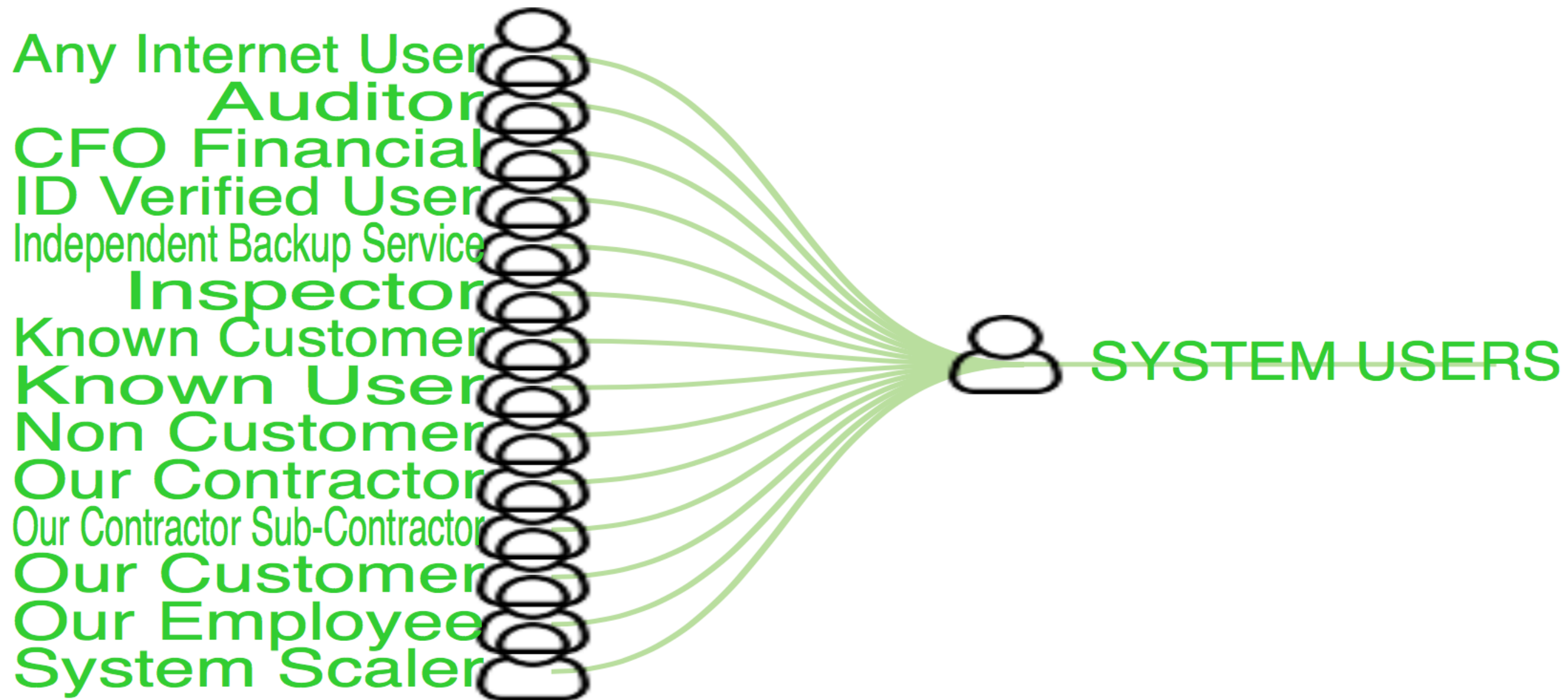
Detail of the Wish level
Notice the selected Scale Qualifiers
Notice the Delivery Date '2020'

Notice the stakeholder-required level is 10X better than today 2018



7 classes of stakeholders from which to derive requirements

This is just a 2nd draft of some stakeholders



Safari File Edit View History Bookmarks Window Help 80 % 29 May 18:04 Tom Gilbs

app.needsandmeans.com/iet/IET-ITTNVY3

EE Find classica...event reviews IET-4U9WCP4...graph=sums CustomerInfo# requirement...gram&page=1 Tom Gilb & Ka...nts-Material Google Docs

Untitled

SCALABILITY... Canvas Diagram Tables List More... Create tomgilb

SCALABILITY TALK EXERCISE June 2018 Tom Gilb / Value Decision Tables / Maintainability Versus Costs

Maintainability Versus Costs

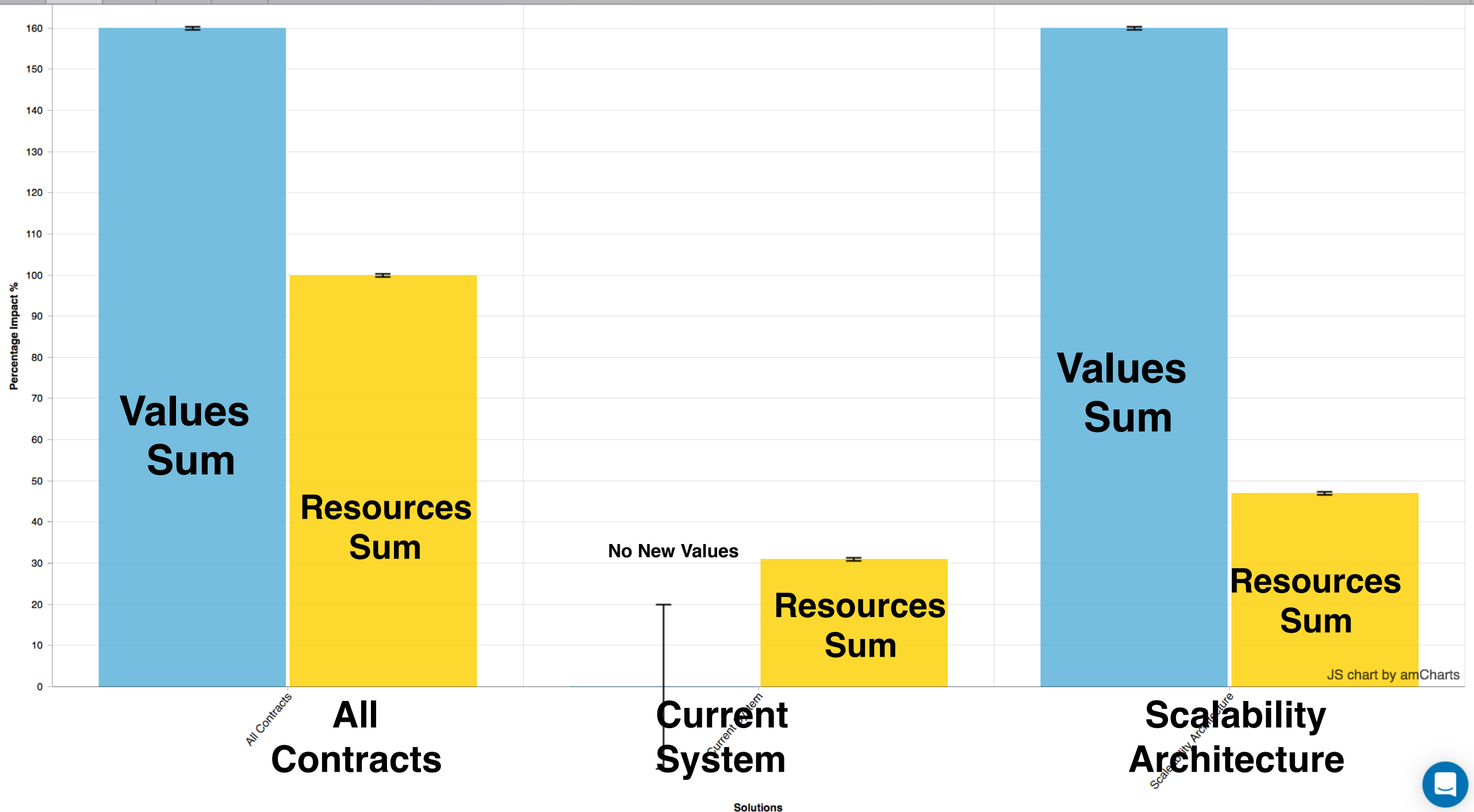
From Level: Product To Level: Level? Show Sidebar

Settings... Add Sort Duplicate... Undo... Δ: INCREASE Help me!

Requirements	All Contracts	Current System	Scaleability Arc...	Sum
→ Maintainability Status: 10 → Tolerable: 5 Hou... Mean time to carry out a defined [Ty... [Type Of Repair = Code 29th May 2019	Δ: -8 ± 0 Δ%: 160 ± 0 % 160%	-0.001 ± 2 0 ± 40 % 0%	Δ: -8 ± 0 Δ%: 160 ± 0 % 160%	✓ ΣΔ%: 320 ± 40 %
Sum Of Values: Worst Case:	Σ%: 160 ± 0 % Σ±%: 160 %	0 ± 40 % -40 %	160 ± 0 % 160 %	
→ Annual Costs Status: 10 → Budget: 100 % A... % of Development Costs cumulated lev... No qualifiers ?	Δ: 45 ± 0 Δ%: 50 ± 0 % 50%	10 ± 0 11 ± 0 % 11%	20 ± 0 22 ± 0 % 22%	✓ ΣΔ%: 83 ± 0 %
→ Development Cost Status: 0 → Budget: 2 € m... € million cumulative investment paym... No qualifiers 4th December 2028	Δ: 1 ± 0 Δ%: 50 ± 0 % 50%	0.4 ± 0 20 ± 0 % 20%	0.5 ± 0 25 ± 0 % 25%	✓ ΣΔ%: 95 ± 0 %
Sum Of Development Resources: Worst Case:	Σ%: 100 ± 0 % Σ±%: 100 %	31 ± 0 % 31 %	47 ± 0 % 47 %	
Value To Cost:	1.60	0.00	3.40	
Ratio (Worst Case)	1.60	-1.30	3.40	

Comments: 0

Add Comment...



A Scalability Architecture Table represented as a Bar Chart presentation:
what is the values-to-costs ratio for a Scalability strategy?

Smidig Utvikling: 'Scalering'

**Agile Systems Development
Scaling**

SCALE-FREE: Practical Scaling Methods for Industrial Systems Engineering

These slides are at <https://tinyurl.com/GilbUnicomScale>

Tom Gilb

LECTURE 30 MINUTES FOR UNICOM DEVOPS

31 OCT 2017 15:20 TO 15:50, LONDON HEATHROW, RADISSON BLU

Based on a paper:

“Some Advanced Tools and Principles for Scaling Agile Projects - Agile Engineering.

40 practical Engineering ideas for scaling agile development successfully all the time.”

A very short pdf paper, supported by references to necessary detail. Not least the The new LeanPub.com/
ValuePlanning book

<http://www.gilb.com/dl865>

The paper

And based on detailed methods in Value Planning book

<http://gilb.com/dl853>

(free digital copy, for YOU)

‘Value Planning’ book

[Commercially available
Leanpub.com/ValuePlanning](http://Leanpub.com/ValuePlanning)

<http://concepts.gilb.com/dl892>
These slides

og Get a Deal at Gilb.com for this and related products like Videos and Courses

My current book manuscript [Value Planning](http://ValuePlanning)

For my friends, use coupon code **VP5** for a €5 discount.

Beyond Scaling: Scale-free Principles for Agile Value Delivery - Agile Engineering.

© tom@gilb.com 2016, Posted at [gilb.com resources/downloads/papers](http://gilb.com/resources/downloads/papers)
<http://www.gilb.com//dl865>

Version March 14 2016, Modified April 11 2016 (XP)

Summary

There is widespread interest in how to make Agile (including Scrum) methods, work better, on a large scale.

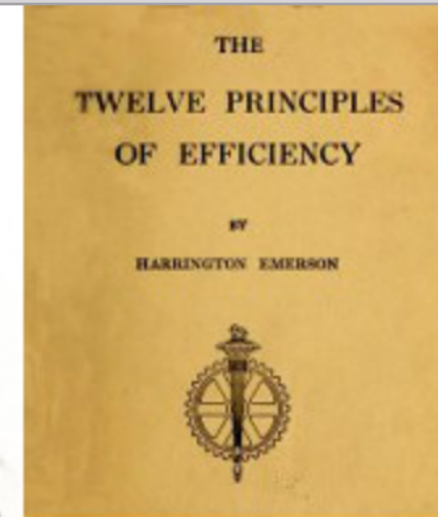
Mike Beedle's paper [1] gives a good *overview (references to much of the agile scaling literature) of many different proposed methods*. I am not going to argue whether these methods are good or bad. No doubt most of the techniques have some value in some circumstances. My concern is *not* this set of 'conventional agile scaling' ideas'. My concern is the large collection of

The Principle that Principles beat methods

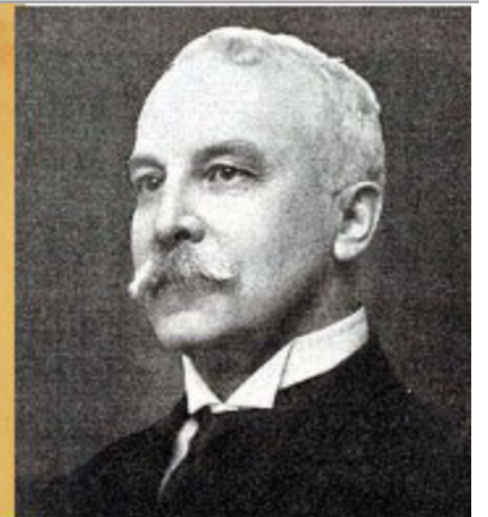
- “As to methods, there may be a million and then some, but principles are few.
- The man who grasps principles can successfully select his own methods”.



R. W. Emerson



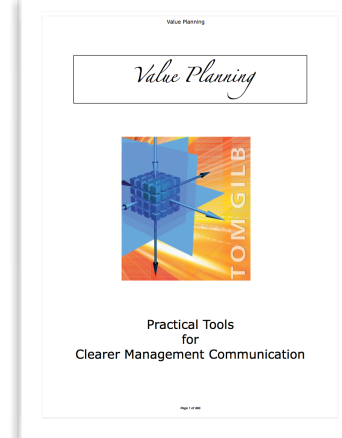
Book Cover



Harrington Emerson

- - Emerson, Harrington
- (Not as thought, R W E)

Scale-free Agile Principles



<http://gilb.com/dl853>
(free digital copy, for YOU)
'Value Planning' book
the VP ref. below

- 1.Keep focus on measurable delivery of critical values and their costs. [3, 4, 5, 6, 9, 10, 12, VP (20) Part 1, VP 10.6]**
- 2.Deliver value early, quickly and regularly: in roughly 2% increments. [14, 11, VP Ch.4, 2, 5]**
- 3.Do NOT focus on code delivery; focus on overall system value and costs. [VP Ch.4, 10D, 10F, 13, VP 3.4, VP 2.10, VP 9.8, 4, 12]**
- 4.Focus on quantified *critical stakeholder* values. [19, VP 3.4, VP 3.7, VP 3.9, VP 3.10 VP 4.2, 10]**
- 5.Synchronize all teams in terms of measurable value delivery. [VP 3.3, VP 3.4, VP Part 1, VP 3.6, VP 3.8, VP 8.4 , 11, 12, 13]**
- 6.Solve big problems through ingenious architecture; not through coding faster. [VP 4.5, VP 5.1, VP 5.3, VP 7.2, 15]**
- 7.Decompose the large problems by incremental value deliveries: not code deliveries. [7, VP Ch. 5, VP 5.1, VP 5.6 , 10, 11, 13, 15]**
- 8.The software component needs to be integrated into the total system of hardware, data, people, culture. [VP 5.2, 10]**
- 9.If your team cannot deliver small increments of real value early, frequently, and predictably; they are incompetent and need to be abandoned for those who can deliver. [7, VP 2.8, 10]**
- 10.Never commit to contracts for *work done* or *code delivered* alone: there must always be a sufficiently large contractual protection, of paying for measurable value delivered. [12, 15].**

Talk Summary

- ‘Scaling’ can be modelled in realistic and ‘tailored’ detail
- Scaling can be *engineered* in
- Scaling’s aspects and attributes can be *quantified*
- *Scale-Free* design is a possibility
- Scaling is a very *multidimensional* problem area

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

5 words,
scaled down

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.

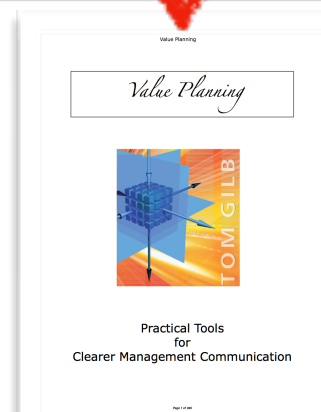
1. **STAKEHOLDERS:** Identify your most critical stakeholders.
2. **OBJECTIVES:** Identify the smart levels of their most critical value improvements.
3. **STRATEGIES:** Identify potential strategies for delivering planned value levels to stakeholders, at lowest cost and risk.
4. **SMALL STEPS:** Decompose 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 Quantities Can Be Quantified”. 18 minutes.



890 pages

‘scaled up’