Department of Computer Science and Creative Technologies,
School of Computing,
London Metropolitan University,
166-220 Holloway Road, London N7 8DB, UK.

"Principles of Software Engineering”

tom@Gilb.com

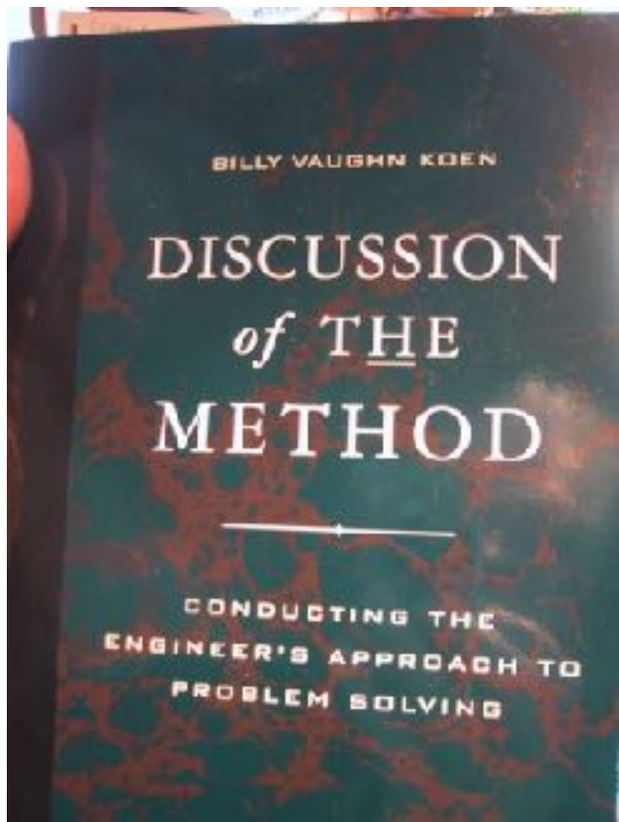
45 minutes, 2 PM

Dec. 9 2016

1. Engineering: Koens definition.
2. The Impact EstimationTable. The core tool
3. Quantified Top Level Critical Requirements
4. Evolutionary agile project management
5. Objective Quality Control of all SE Specs
6. Key Principles of Software Engineering

Koen on Risk Control

- Make small changes in the sota:
  - ‘Sota’ = Engineering State Of The Art Heuristics <-Koen, Discussion, p. 48
- Always give yourself a chance to retreat; and
- Use feedback to stabilize the design process



The engineering method

The engineering method is

the use of engineering heuristics

to cause the best change in a poorly understood situation

within the available resources.

- Source: Toward a definition of the engineering method (Engineering Education, Dec. 1984). Billy V Koen, U of Austin TX

Tom's Rewrite of Koen's

Engineering **Concept *224 June 28, 2003**

Engineering is

- an Evolutionary Process,
- using practical Principles,
- in order to determine,
- and identify the Means to deliver,
- the best achievable Performance and Cost levels balance,
- for optimal Stakeholder satisfaction,
- in a complex risk-filled environment.

Koen's Heuristics

Make small changes in the sota:
'Sota' = Engineering State Of The Art Heuristics <-Koen, Discussion, p. 48

Always give yourself a chance to retreat; and

Use feedback to stabilize the design process

Niels Malotaux

Impact Estimation principle

How much % of what we want to achieve do we achieve by this solution

At what cost ?

Possible solutions to achieve it

Could we get all, within the budgets of time and cost ?

		Design Idea #1	Design Idea #2	Design Idea #3	Total Impact
What to achieve	Objectives	Impact on Objective	Impact on Objective	Impact on Objective	Sum of Impacts on Objectives
Cost to achieve it	Resources Time Money	Impact on Resources	Impact on Resources	Impact on Resources	Sum of Impact on Resources
Return on Investment	Benefits to Cost Ratio	Benefits / Cost	Benefits / Cost	Benefits / Cost	

From Scales to Solutions

	Solution 1	Solution 2	Solution n	Total Impacts
Objective	Impact on Objective	Impact on Objective	Impact on Objective	Total Impact on Objective
Resources	Impact on Budget	Impact on Budget	Impact on Budget	Total Impact on Budget
Benefits-to-Cost Ratio	Ratio	Ratio	Ratio	

Courtesy Rolf Goetz

From Scales to Solutions

	Solution 1	Solution 2	Solution n	Total Impacts
Objective	Impact on Objective	Impact on Objective	Impact on Objective	Total Impact on Objective
Resources	Impact on Budget	Impact on Budget	Impact on Budget	Total Impact on Budget
Benefits-to-Cost Ratio	Ratio	Ratio	Ratio	

Courtesy Rolf Goetz

Impacts on Objectives

	Facebook	Profiler	Umantis BM	Total Impacts on Objectives
Attract Talents 271 -> 700	**70%** *± 10%*	**0%** *±10%*	**50%** *±5%*	**120%** *±25%*
Win Talents 53 -> 100	**30%** *± 20%*	**50%** *± 10%*	**30%** *± 10%*	**110%** *±40%*
Perfect Match 25% -> 75%	**10%** *± 10%*	**30%** *± 10%*	**30%** *± 10%*	**70%** *±30%*
Total Impact of Solutions	**110%** *± 40%*	**80%** *± 30%*	**110%** *± 25%*	

Courtesy Rolf Goetz

Impacts on Resources and Totals

	Facebook	Profiler	Umantis BM	Total Impacts on Objectives
Money 100.000€	**70%** *± 10%*	**0%** *±10%*	**50%** *±5%*	**120%** *±25%*
Time 12 months	**30%** *± 20%*	**50%** *± 10%*	**30%** *± 10%*	**110%** *±40%*
Total Impact of Solutions	**100%** *± 30%*	**50%** *± 20%*	**80%** *± 15%*	
Benefit/Cost	110/100 =1.1 Best 2.1 Worst 0.5	80 / 50 = 1.6 Best 3.7 Worst 0.7	110 / 80 = 1.4 Best 2.1 Worst 0.9	

Courtesy Rolf Goetz

Mars Mission Business School Project Lviv, Ukraine

Safari File Edit View History Bookmarks Window Help 75 % Tom Gibs

Inbox (734 messages, 136 unread)

app.needsandmeans.com/iet/IET-9SVOF9S?subpage=table

Untitled

Settings... | + Add to table | Sort designs | Show Sidebar

Requirements	ship_...	MSTM3_Software_C...	Sum
MSTM3_Supplies Past: **75** → Wish: **95** % Succesful... % Succesful Supply of defined [Goods] for defined [Requestors] within defined [Period] with defined [Delay] during Mars mission	cc... 0	**78** ± 1 % Succ... 0 Δ%: **15** ± 5 %	Σ%: **295** ± 85 %
MSTM3_Energy Past: **90** → Wish: **99.9** % of requir... % of [Generators] online time, "energy storage" [Capacity], self-repair,	r... 0	**90** ± 0 % of r... 0 Δ%: **0** ± 0 %	Σ%: **212** ± 555 %
MSTM3_Arrival Past: **40** → Wish: **95** % Successfu... % Successful arrival of [Things] by [Process] under [Conditions]	ucc... 0	**60** ± 3 % Succ... 0 Δ%: **36** ± 5 %	Σ%: **141** ± 32 %
MSTM3_Landing Past: **50** → Wish: **90** % successfu... % successful landing for defined [Objects] under defined [Conditions] using approved [Landing Process]	cc... 0	**54** ± 1 % succ... 0 Δ%: **10** ± 3 %	Σ%: **181** ± 34 %
Sum Of Performance:		Σ%: **61** ± 13 %	
MSTM3_Time Past: **0** → Wish: **100** %	0	**45** ± 20 % 0 Δ%: **45** ± 20 %	Σ%: **101** ± 40 %
MSTM3_Budget Past: **0** → Wish: **100** % from prog...	om... 0	**15** ± 5 % from... 0 Δ%: **15** ± 5 %	Σ%: **125** ± 33 %
Sum Of Resources:		Σ%: **60** ± 25 %	

Safari File Edit View History Bookmarks Window Help 76 % Tom Gilbs

Inbox (733 messages, 135 unread)

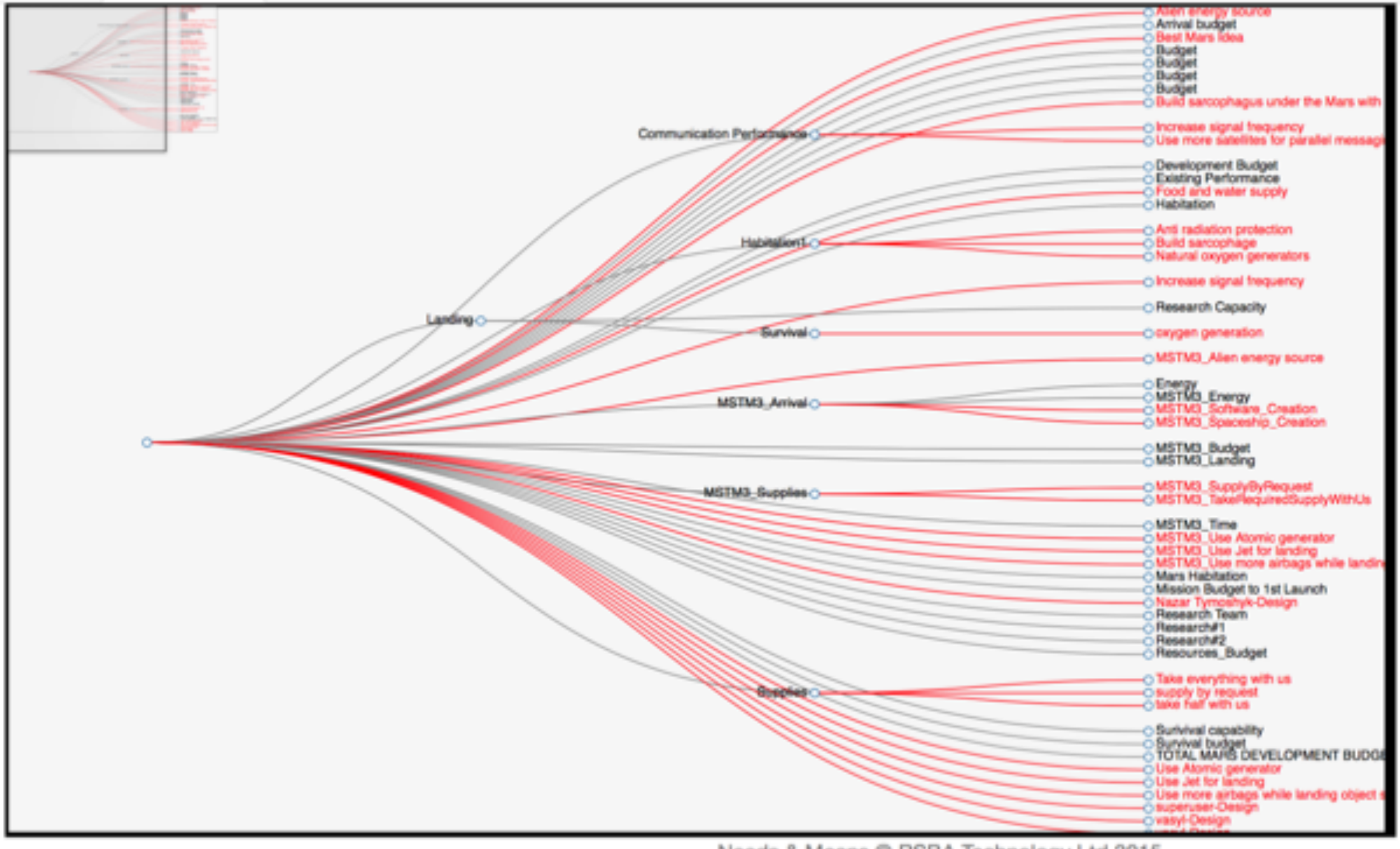
app.needsandmeans.com/jet/IET-9SVOF9S

Untitled

Settings... | + Add to table | Sort designs | Show Sidebar

Requirements	MSTM3_Use more a...	MSTM3_Use Jet fo...	MSTM3_TakeRequir...	MSTM3_Alien ener...	MSTM3_Use Atomic...	MSTM3_SupplyByRe...
MSTM3_Supplies Past: 75 → Wish: 95 % Succesful...	76 ± 1 % Succ... 0 Δ%: 5 ± 5 %	77 ± 2 % Succ... 0 Δ%: 10 ± 10 %	95 ± 5 % Succ... 0 Δ%: 100 ± 25 %	78 ± 2 % Succ... 0 Δ%: 15 ± 10 %	76 ± 1 % Succ... 0 Δ%: 5 ± 5 %	99 ± 2 % Succ... 0 Δ%: 120 ± 10 %
MSTM3_Energy Past: 90 → Wish: 99.9 % of requir...	91 ± 1 % of r... 0 Δ%: 10 ± 10 %	90 ± 0 % of r... 0 Δ%: 0 ± 0 %	90 ± 0 % of r... 0 Δ%: 0 ± 0 %	100 ± 50 % of... 0 Δ%: 101 ± 505 %	98 ± 3 % of r... 0 Δ%: 81 ± 30 %	90 ± 0 % of r... 0 Δ%: 0 ± 0 %
MSTM3_Arrival Past: 40 → Wish: 95 % Succes	43 ± 1 % Succ... 0 Δ%: 5 ± 2 %	50 ± 4 % Succ... 0 Δ%: 18 ± 7 %	40 ± 0 % Succ... 0 Δ%: 0 ± 0 %	40 ± 0 % Succ... 0 Δ%: 0 ± 0 %	40 ± 0 % Succ... 0 Δ%: 0 ± 0 %	40 ± 0 % Succ... 0 Δ%: 0 ± 0 %
MSTM3_Landing Past: 50 → Wish: 90 % successfu...	76 ± 5 % succ... 0 Δ%: 65 ± 13 %	87 ± 5 % succ... 0 Δ%: 93 ± 13 %	50 ± 0 % succ... 0 Δ%: 0 ± 0 %	50 ± 0 % succ... 0 Δ%: 0 ± 0 %	50 ± 0 % succ... 0 Δ%: 0 ± 0 %	50 ± 0 % succ... 0 Δ%: 0 ± 0 %
Sum Of Performance:	Σ%: 85 ± 30 %	Σ%: 121 ± 30 %	Σ%: 100 ± 25 %	Σ%: 116 ± 515 %	Σ%: 86 ± 35 %	Σ%: 120 ± 10 %
MSTM3_Time Past: 0 → Wish: 100 %	1 ± 1 % 0 Δ%: 1 ± 1 %	2 ± 1 % 0 Δ%: 2 ± 1 %	2 ± 2 % 0 Δ%: 2 ± 2 %	15 ± 3 % 0 Δ%: 15 ± 3 %	5 ± 2 % 0 Δ%: 5 ± 2 %	1 ± 1 % 0 Δ%: 1 ± 1 %
MSTM3_Budget Past: 0 → Wish: 100 % from prog...	1 ± 1 % from... 0 Δ%: 1 ± 1 %	4 ± 1 % from... 0 Δ%: 4 ± 1 %	5 ± 3 % from... 0 Δ%: 5 ± 3 %	5 ± 2 % from... 0 Δ%: 5 ± 2 %	10 ± 1 % from... 0 Δ%: 10 ± 1 %	25 ± 10 % from... 0 Δ%: 25 ± 10 %
Sum Of Resources:	Σ%: 2 ± 2 %	Σ%: 6 ± 2 %	Σ%: 7 ± 5 %	Σ%: 20 ± 5 %	Σ%: 15 ± 3 %	Σ%: 26 ± 11 %
Performance To Cost:	42.50	20.17	14.29	5.80	5.73	4.62
Ratio (Worst Case)	13.75	11.00	6.25	-15.96	2.83	2.97

List | Diagram | Documents



Requirements	ProductDesign	Financials	MarketingStrategy	DistributionMethod	Sum
Demographic Past: **0** → Wish: **50** %	20 ± 5 % 0 Δ%: 40 ± 10 % 40	27 ± 5 % 0 Δ%: 54 ± 10 % 94	23 ± 3 % 0 Δ%: 46 ± 6 % 140	10 ± 0 % 0 Δ%: 20 ± 0 % 160	Σ Δ%: 160 ± 26 %
Millionaire Past: 1 → Wish: **1000000** $	450000 ± 15000 $0 Δ%: 45 ± 15 % 45	400000 ± 10000 $0 Δ%: 40 ± 10 % 85	100000 ± 50000 $0 Δ%: 10 ± 5 % 95	200000 ± 10000 $0 Δ%: 20 ± 10 % 115	Σ Δ%: 115 ± 40 %
MarketSegment Past: **4** → Wish: **1** Market Rank	1 ± 1 Market... 0 Δ%: 100 ± 33 % 100	4 ± 1 Market... 0 Δ%: 0 ± 33 % 100	2 ± 1 Market... 0 Δ%: 67 ± 33 % 167	3 ± 1 Market... 0 Δ%: 33 ± 33 % 200	Σ Δ%: 200 ± 132 %
Geography Past: **0** → Wish: **100** %	5 ± 5 % 0 Δ%: 5 ± 5 % 5	10 ± 4 % 0 Δ%: 10 ± 4 % 15	40 ± 5 % 0 Δ%: 40 ± 5 % 55	30 ± 5 % 0 Δ%: 30 ± 5 % 85	Σ Δ%: 85 ± 19 %
Market Past: **0** → Wish: **100** %	40 ± 10 % 0 Δ%: 40 ± 10 % 40	5 ± 3 % 0 Δ%: 5 ± 3 % 45	40 ± 10 % 0 Δ%: 40 ± 10 % 85	20 ± 5 % 1 Δ%: 20 ± 5 % 105	Σ Δ%: 105 ± 28 %
Sum Of Performance:	Σ%: 230 ± 73 % 230	Σ%: 109 ± 60 % 339	Σ%: 203 ± 59 % 542	Σ%: 123 ± 53 % 665	
TimeToMarket Past: 1 → Wish: **8** Weeks	2 ± 0.5 Weeks 0 Δ%: 14 ± 7 % 14	2 ± 0.5 Weeks 0 Δ%: 14 ± 7 % 28	3 ± 0.75 Weeks 0 Δ%: 29 ± 11 % 57	4 ± 1 Weeks 0 Δ%: 43 ± 14 % 100	Σ Δ%: 100 ± 39 %
ShowMeTheMoney Past: **0** → Wish: **5005** £	1200 ± 200 £ 0 Δ%: 24 ± 4 % 24	205 ± 200 £ 0 Δ%: 4 ± 4 % 28	2100 ± 500 £ 0 Δ%: 42 ± 10 % 70	1500 ± 0 £ 0 Δ%: 30 ± 0 % 100	Σ Δ%: 100 ± 18 %
Sum Of Resources:	Σ%: 38 ± 11 % 38	Σ%: 18 ± 11 % 56	Σ%: 71 ± 21 % 127	Σ%: 73 ± 14 % 200	
Performance To Cost:	6.05	6.06	2.86	1.68	
Ratio (Worst Case)	3.20	1.69	1.57	0.80	

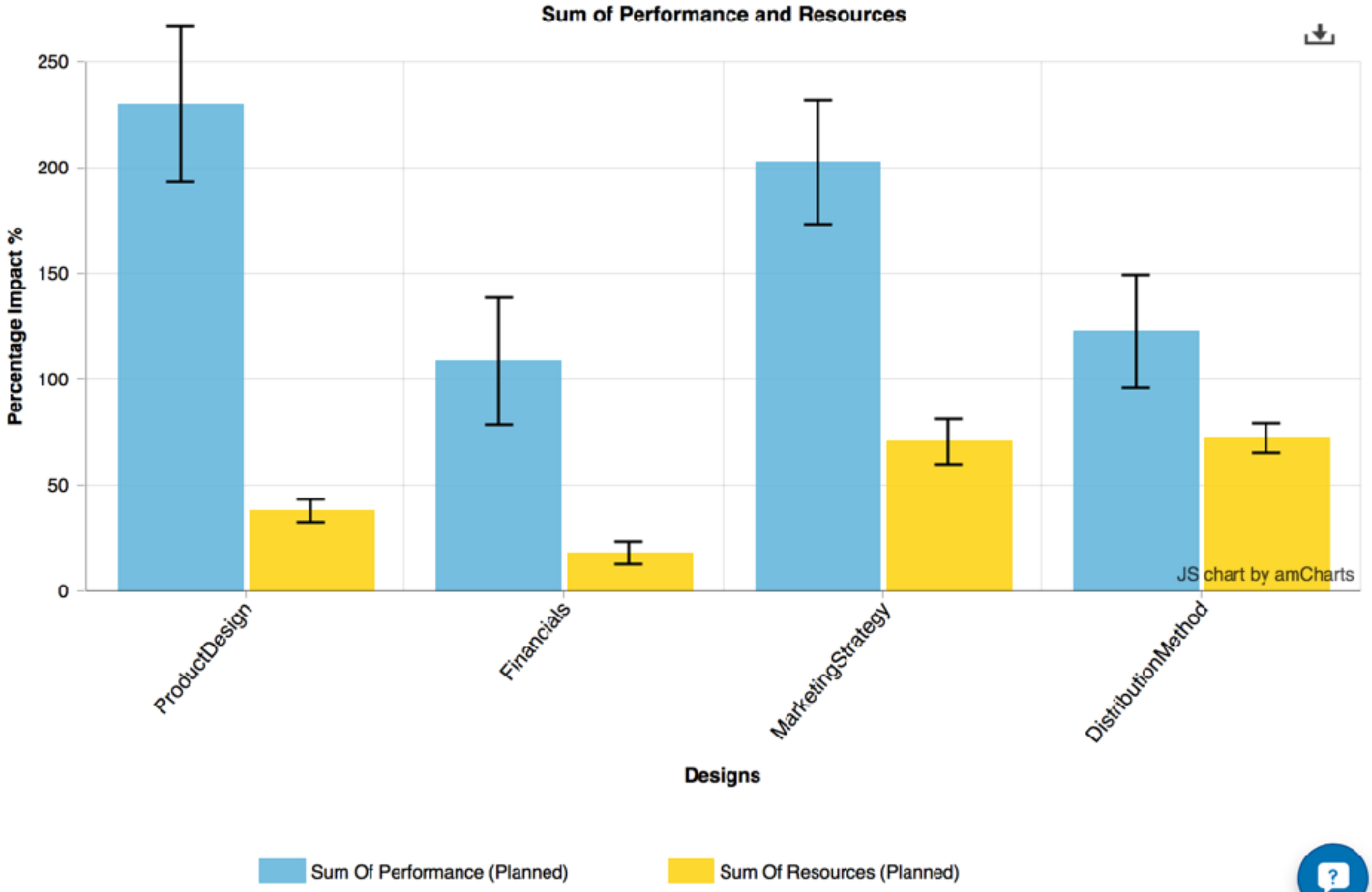
app.needsandmeans.com/iet/IET-AFGGXOT

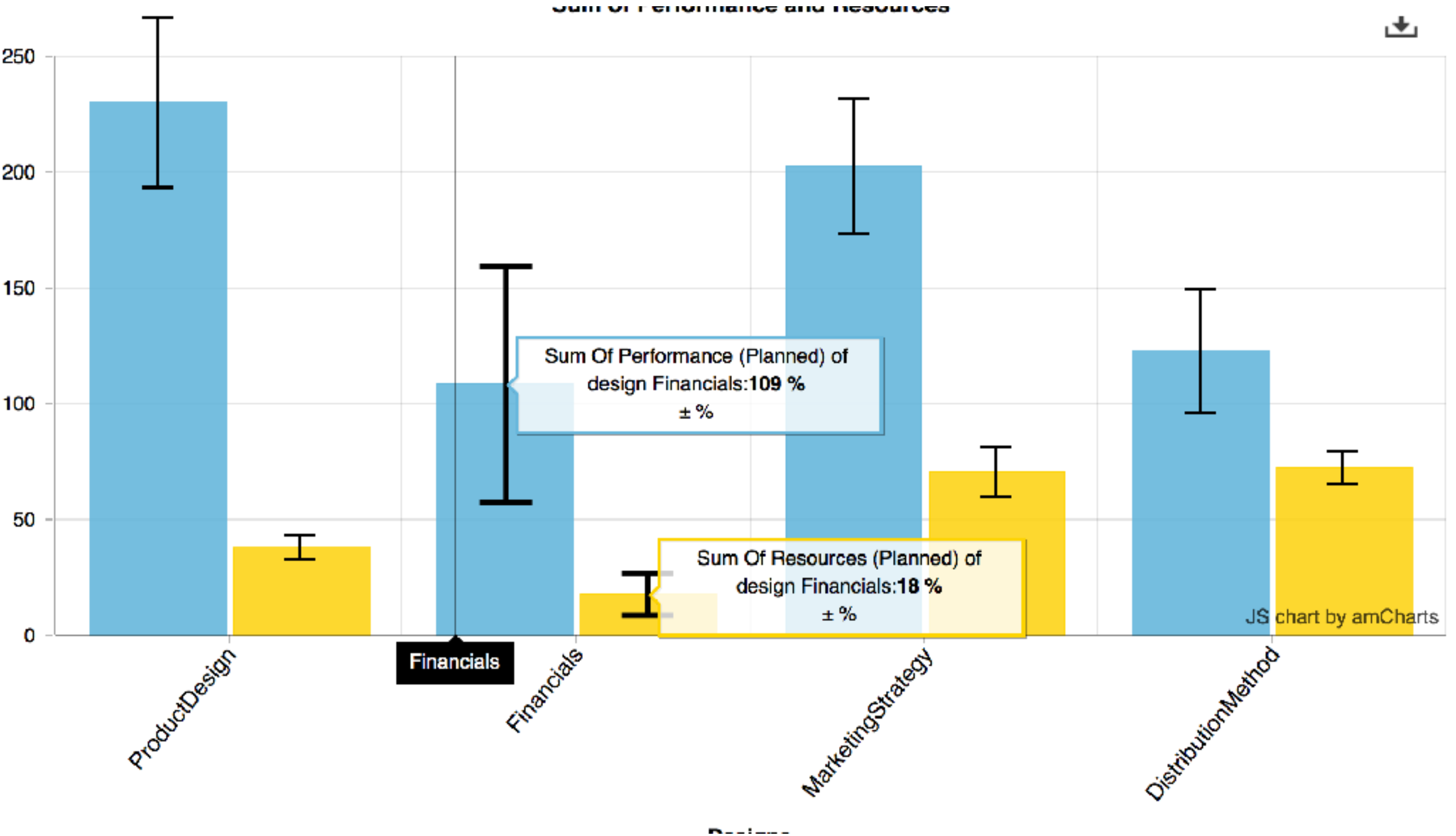
Untitled

Requirements	ProductDesign	Financials	MarketingStrategy	DistributionMethod	Sum
Demographic Past: **0** → Wish: **50** %	**20** ± 5 % 0 Δ%: **40** ± 10 % 40	**27** ± 5 % 0 Δ%: **54** ± 10 % 94	**23** ± 3 % 0 Δ%: **46** ± 6 % 140	**10** ± 0 % 0 Δ%: **20** ± 0 % 160	Σ Δ%: **160** ± 26 %
Millionaire Past: 1 → Wish: **1000000** $	**450000** ± 150000 $0 Δ%: **45** ± 15 % 45	**400000** ± 100000 $0 Δ%: **40** ± 10 % 85	**100000** ± 50000 $0 Δ%: **10** ± 5 % 95	**200000** ± 100000 $0 Δ%: **20** ± 10 % 115	Σ Δ%: **115** ± 40 %
MarketSegment Past: **4** → Wish: **1** Market Rank	**1** ± 1 Market... 0 Δ%: **100** ± 33 % 100	**4** ± 1 Market... 0 Δ%: **0** ± 33 % 100	**2** ± 1 Market... 0 Δ%: **67** ± 33 % 167	**3** ± 1 Market... 0 Δ%: **33** ± 33 % 200	Σ Δ%: **200** ± 132 %
Geography Past: **0** → Wish: **100** %	**5** ± 5 % 0 Δ%: **5** ± 5 % 5	**10** ± 4 % 0 Δ%: **10** ± 4 % 15	**40** ± 5 % 0 Δ%: **40** ± 5 % 55	**30** ± 5 % 0 Δ%: **30** ± 5 % 85	Σ Δ%: **85** ± 19 %
Market Past: **0** → Wish: **100** %	**40** ± 10 % 0 Δ%: **40** ± 10 % 40	**5** ± 3 % 0 Δ%: **5** ± 3 % 45	**40** ± 10 % 0 Δ%: **40** ± 10 % 85	**20** ± 5 % 1 Δ%: **20** ± 5 % 105	Σ Δ%: **105** ± 28 %
Sum Of Performance:	Σ%: **230** ± 73 % 230	Σ%: **109** ± 60 % 339	Σ%: **203** ± 59 % 542	Σ%: **123** ± 53 % 665	
TimeToMarket Past: 1 → Wish: **8** Weeks	**2** ± 0.5 Weeks 0 Δ%: **14** ± 7 % 14	**2** ± 0.5 Weeks 0 Δ%: **14** ± 7 % 28	**3** ± 0.75 Weeks 0 Δ%: **29** ± 11 % 57	**4** ± 1 Weeks 0 Δ%: **43** ± 14 % 100	Σ Δ%: **100** ± 39 %
ShowMeTheMoney Past: **0** → Wish: **5005** £	**1200** ± 200 £ 0 Δ%: **24** ± 4 % 24	**205** ± 200 £ 0 Δ%: **4** ± 4 % 28	**2100** ± 500 £ 0 Δ%: **42** ± 10 % 70	**1500** ± 0 £ 0 Δ%: **30** ± 0 % 100	Σ Δ%: **100** ± 18 %
Sum Of Resources:	Σ%: **38** ± 11 % 38	Σ%: **18** ± 11 % 56	Σ%: **71** ± 21 % 127	Σ%: **73** ± 14 % 200	
Performance To Cost:	6.05	6.06	2.86	1.68	
Ratio (Worst Case)	3.20	1.69	1.57	0.80	

157/49 = 3.2

Settings... | + Add to table | Sort designs




250
200
150
100
50
0
Sum Of Performance (Planned) of design Financials:109 %
± %
Sum Of Resources (Planned) of design Financials:18 %
± %
JS chart by amCharts
ProductDesign
Financials
Financials
MarketingStrategy
DistributionMethod

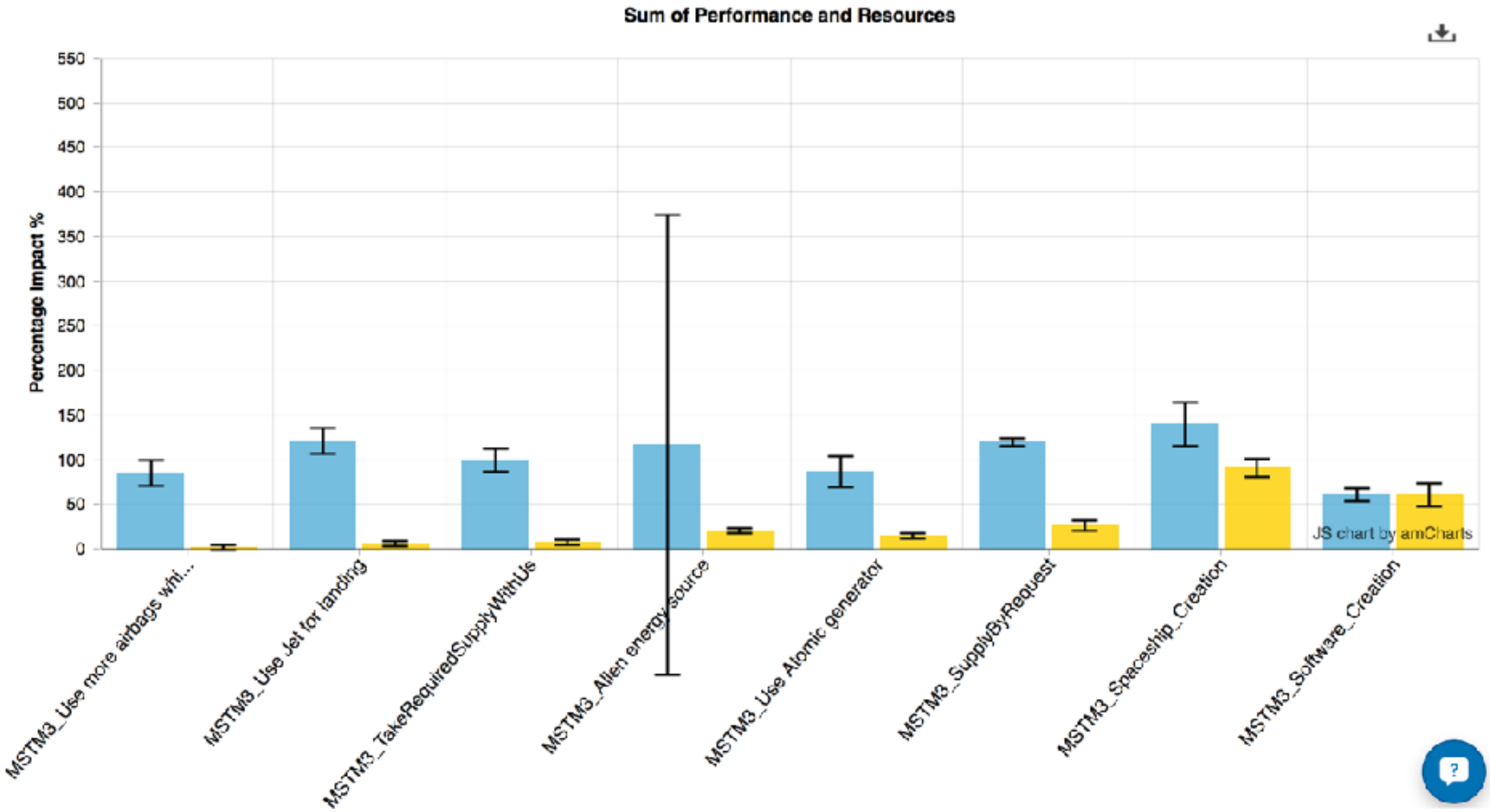
Safari File Edit View History Bookmarks Window Help 76 % Tom Gilbs

Inbox (733 messages, 135 unread)

app.needsandmeans.com/let/IET-9SVOF9S?subpage=graph&graph=sums

Untitled

Settings... Add to table Sort designs



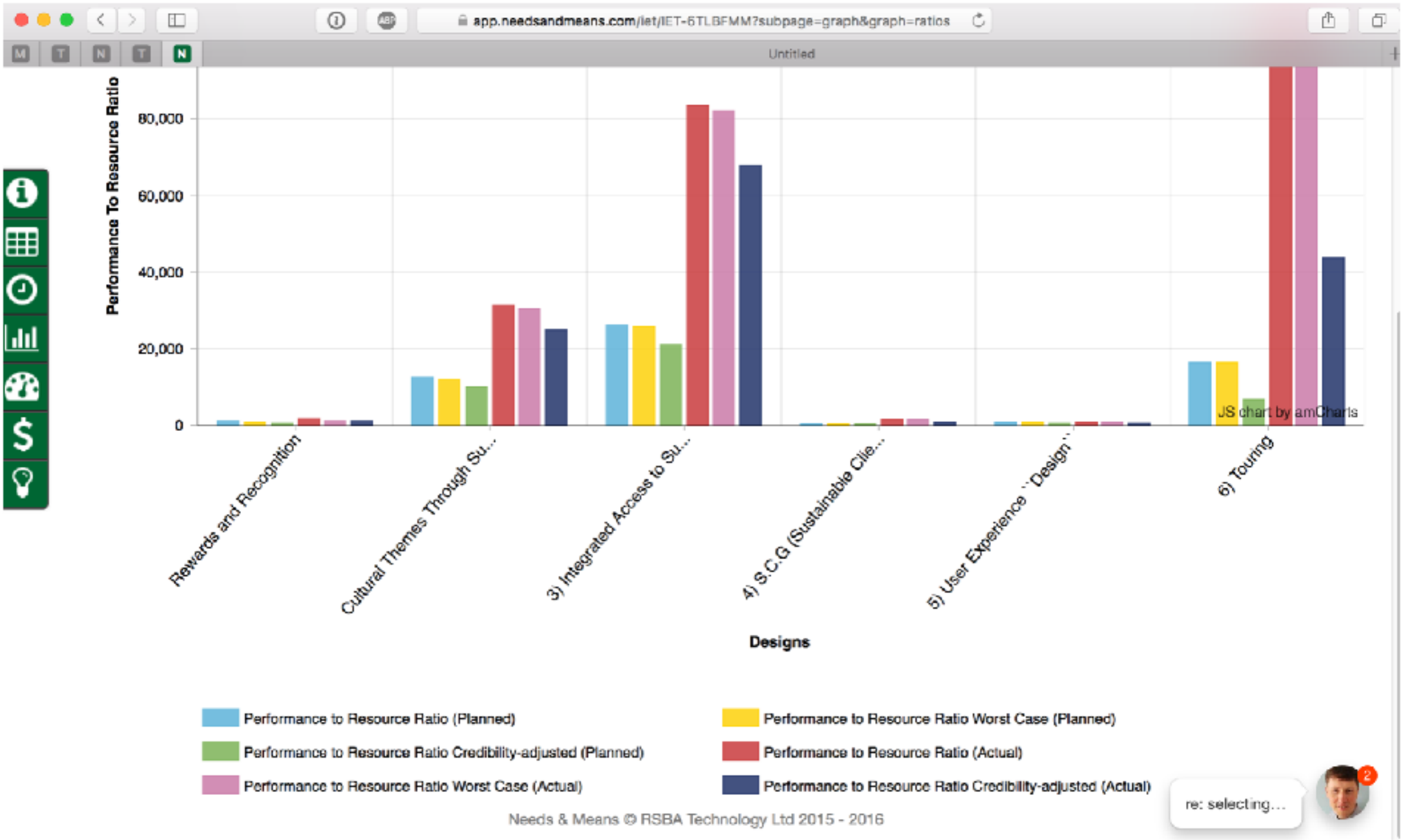
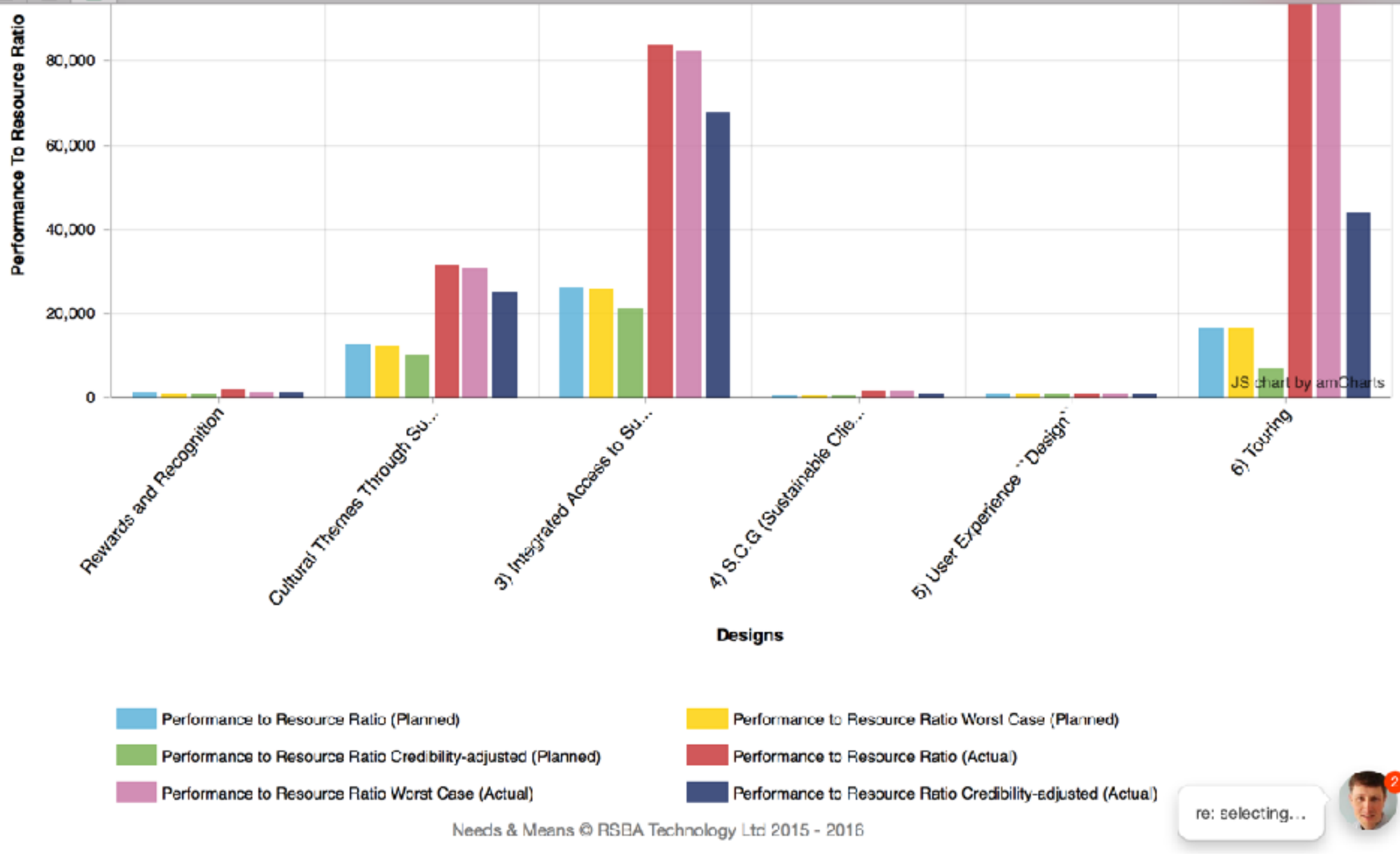


Figure W3: Real planning example. A 'bottom line' summary of the estimated impacts of a set of strategies, where the impact on all top level critical quantified performance objectives is considered. Sometimes with respect to estimated set of budgeted costs. Sometimes with respect to risks with the strategies. Courtesy Incognito Startup Project, Oslo (Gottfried Osei) January 8 2016.

app.needsandmeans.com/iet/IET-6TLBFMM?subpage=graph&graph=ratios

Untitled



re: selecting...

Quantification of Qualities and Values

Quantifying Music

Lean QA Audience at ACCU 2012 “Surely you cannot quantify ‘Music’ ?”

- I claimed
  - –we can quantify any variable quality of any system
- I replied:
  - –*I’ll do it in a lightening talk here at ACCU*

- Can you quantify this music?

Black-Eyed Peas song ”I gotta Feeling” gets 8.9 of 10 from Hit Song Science software

Frank Micelotta/Getty Image

The Black Eyed Peas' single "I Gotta Feeling" received a hit score of 8.9 out 10 with Music Intelligenc Solutions' new software Hit Song Science.

“There's no magic in that; it's math”

- "[It's] a series of **algorithms that we use**
  - to look at what's **the potential of a song**
  - to **be sticky with a listener ...**
- To have **those patterns in the music** that would
  - ***correspond*** with what **human brain waves would find pleasing**”

CEO David Meredith

- A study conducted by the Harvard Business School found that the software was accurate 8 out of 10 times.http://www.npr.org/templates/story/story.php?storyId=113673324

Measurable Attributes of Hits

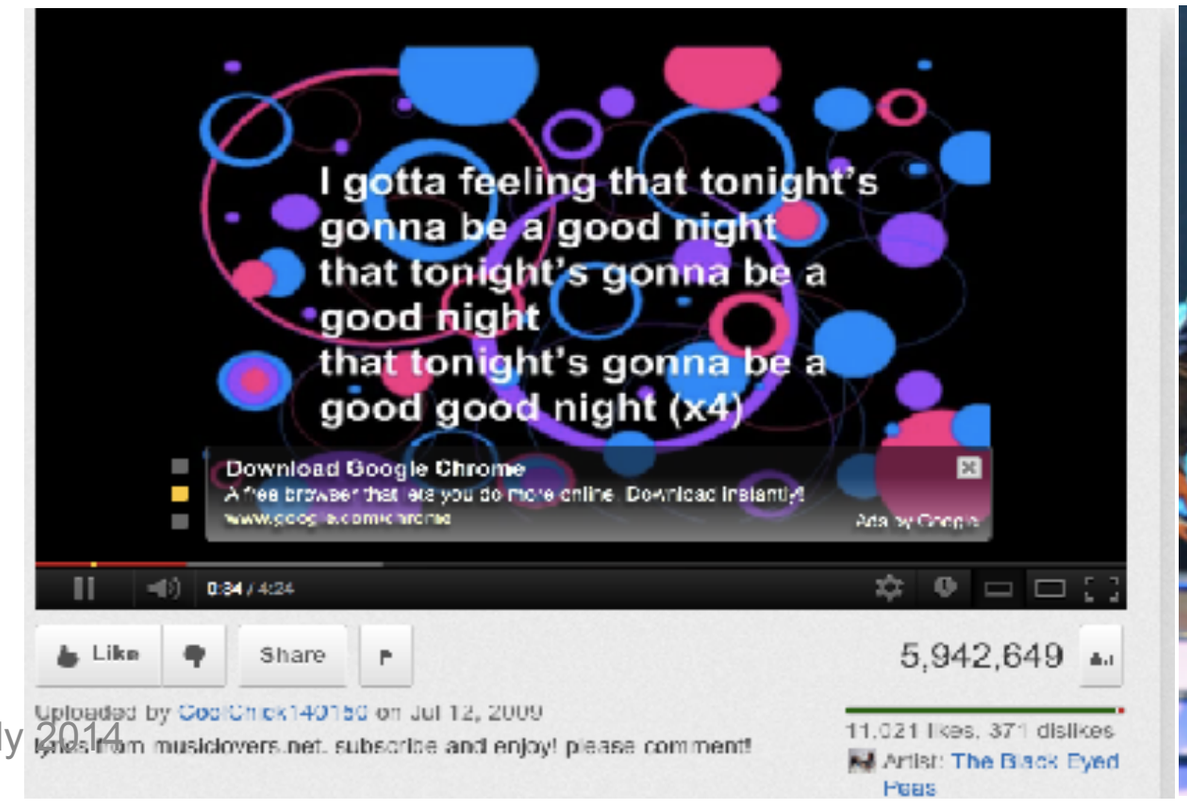
Meredith says his software evaluates songs over sixty elements including

Melody
Harmony
Tempo
Pitch
Octave
Beat
Rhythm
Fullness of sound
Noise
Brilliance
Chord progression

http://edition.cnn.com/2008/WORLD/europe/03/07/spiritof.music/

YouTube Measures

- Number of Likes and Dislikes
  11,021 Likes, 371 Dislikes (April 26, 2012)
- Number of times video has been viewed
  5,942.649 Views (April 26, 2012)



By **Survey**: Most Wanted Attributes

- Yudkin reports on a web-based survey into American musical tastes conducted by Komar and Melamid in 1996
- If you want to please the greatest number of Americans (72% ± 12%) consider
  - Male and female solo voices
  - R&B with a love theme
  - Small ensemble of musicians
  - Length of about 5 minutes
  - Moderate pitch, tempo and volume

http://www.bu.edu/cfa/music/faculty/yudkin/

Most Unwanted Attributes

To appeal to only about 200 Americans

- Extreme length
- Wide range of dynamics, tempo and pitch in abrupt succession
- An operatic soprano singing atonally
- A cowboy song with political slogans
- A children's choir singing holiday songs
- Large orchestra featuring harp, accordion and bagpipes

http://www.bu.edu/cfa/music/faculty/yudkin/

There are samples of two songs written by David Soldier with lyrics by Nina Mankin to these wanted and unwanted guidelines about 19 minutes into Yudkin's lecture

Some potentially quantifiable Quality dimensions of Music

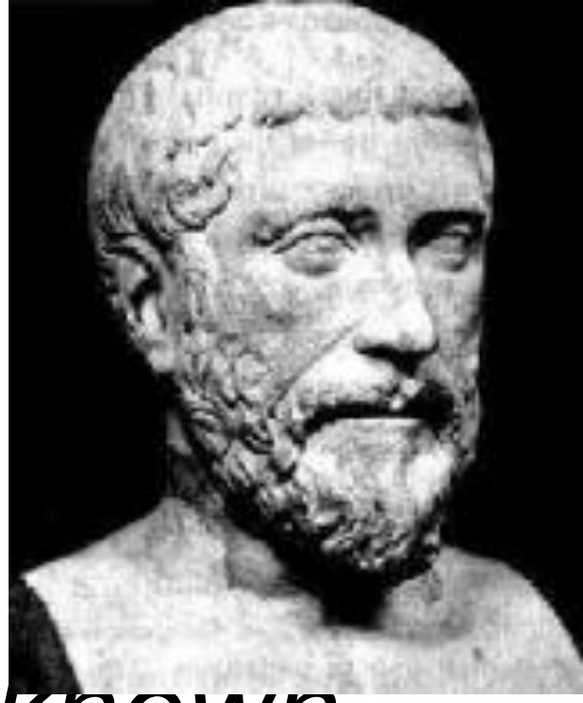
Brainstormed by Steve F. and Rachel D. At lunch

- In tune
- Applause
- Moving
- Encores
- Repeat Gigs
- Busking Hat Collection
- MRI Brain Scan
- Downloads
- Utube Reviews
- Royalties
- … (many more!!)

Examples in Planguage

- **Music.Moving:**
- **Type: primary music quality attribute**
- **Ambition Level: the majority of listeners feel moved to tears or strong physical emotional reactions.**
- **Scale: the % of defined [Listeners] hearing defined [Music] under defined [Environments] who reports a defined [Emotion] at a defined [Strength]**
- **Goal [1st UK Release, Music = Hip Hop, Environment = Itunes, Emotion = {Tears,**

Philolaus on Numbers



- Over four hundred years BC,
- a Greek by the name of
- **Philolaus** of Tarentum said :
- *”Actually, everything that can be known has a Number;*
  - *for it is impossible to grasp anything with the mind or to recognize it without this (number).”*

Best regards (Aug 2005),
N.V.Krishnawww.microsensesoftware.com

How to Quantify any Qualitative Requirement

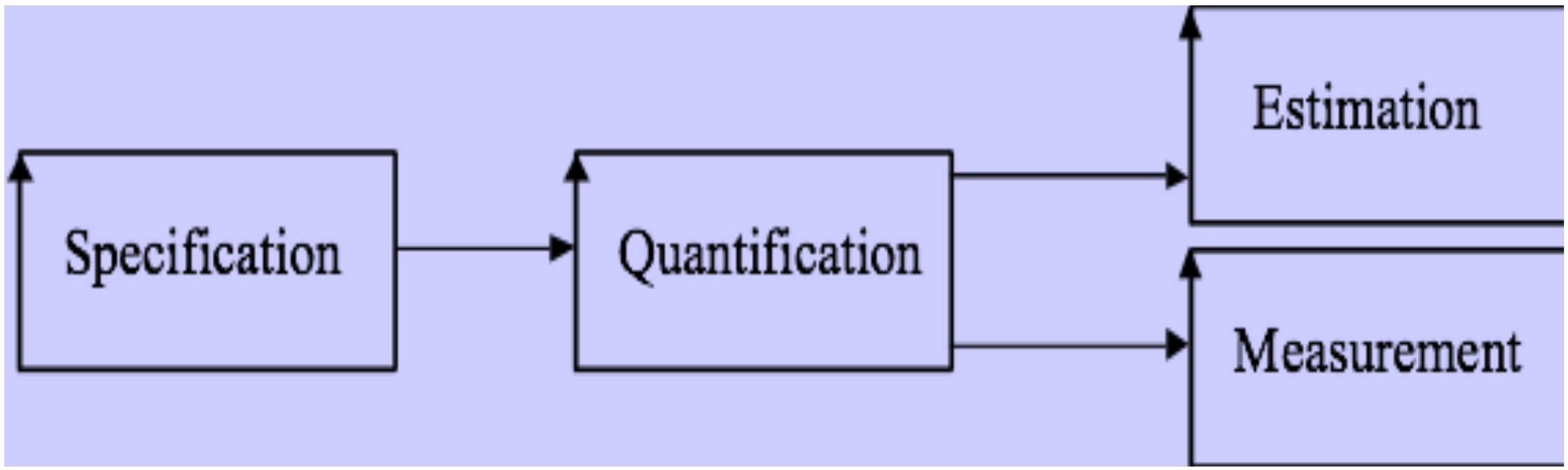


Diagram from 'Competitive Engineering.' book.

Quality Quantification Methods #1

- Common Sense, Domain Knowledge
  - Decompose “until quantification becomes obvious”.
  - Then use Planguage specification:
    - **Scale**: define a measurement scale
    - **Meter**: define a test or process for measuring on the scale
    - **Past**: define benchmarks, old system, competitors on the scale
    - **Goal**: define a committed level of future stakeholder quality, on your scale.

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: This above is an extension of some basic ideas from Ireson, Editor, Reliability Handbook, *McGraw Hill, 1966* (Ireson 1966).

Quality Quantification Methods #**2**, **Look it up in a book**

Chapter
5

SCALES OF MEASURE

How to Quantify

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}.

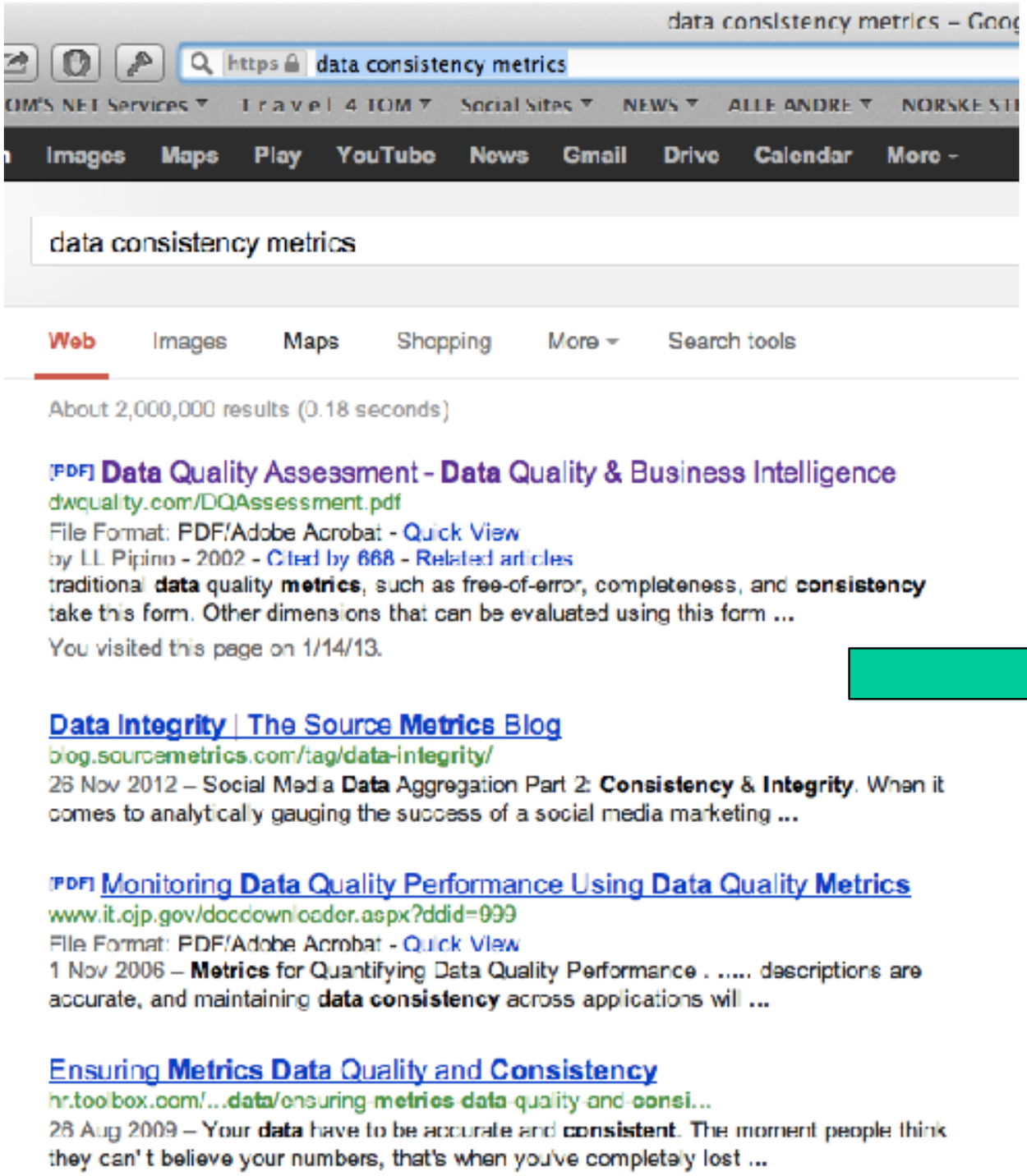
Problem Recognition:
Scale: C
system]
electroni
Adminis
Scale: C
assigned
Tool Co
Scale: C
acquire
and qua
Problem
Scale: C
toms an
Change
Scale: C
the nece
Note: Th
Quality
Scale: C
Modifica
Scale: C
correctio
Modifica
Unit T
Scale:
Integr
Scale:
Beta T
Scale:
releas
Syste
Scale:
Recovery:
Scale: Clock 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.

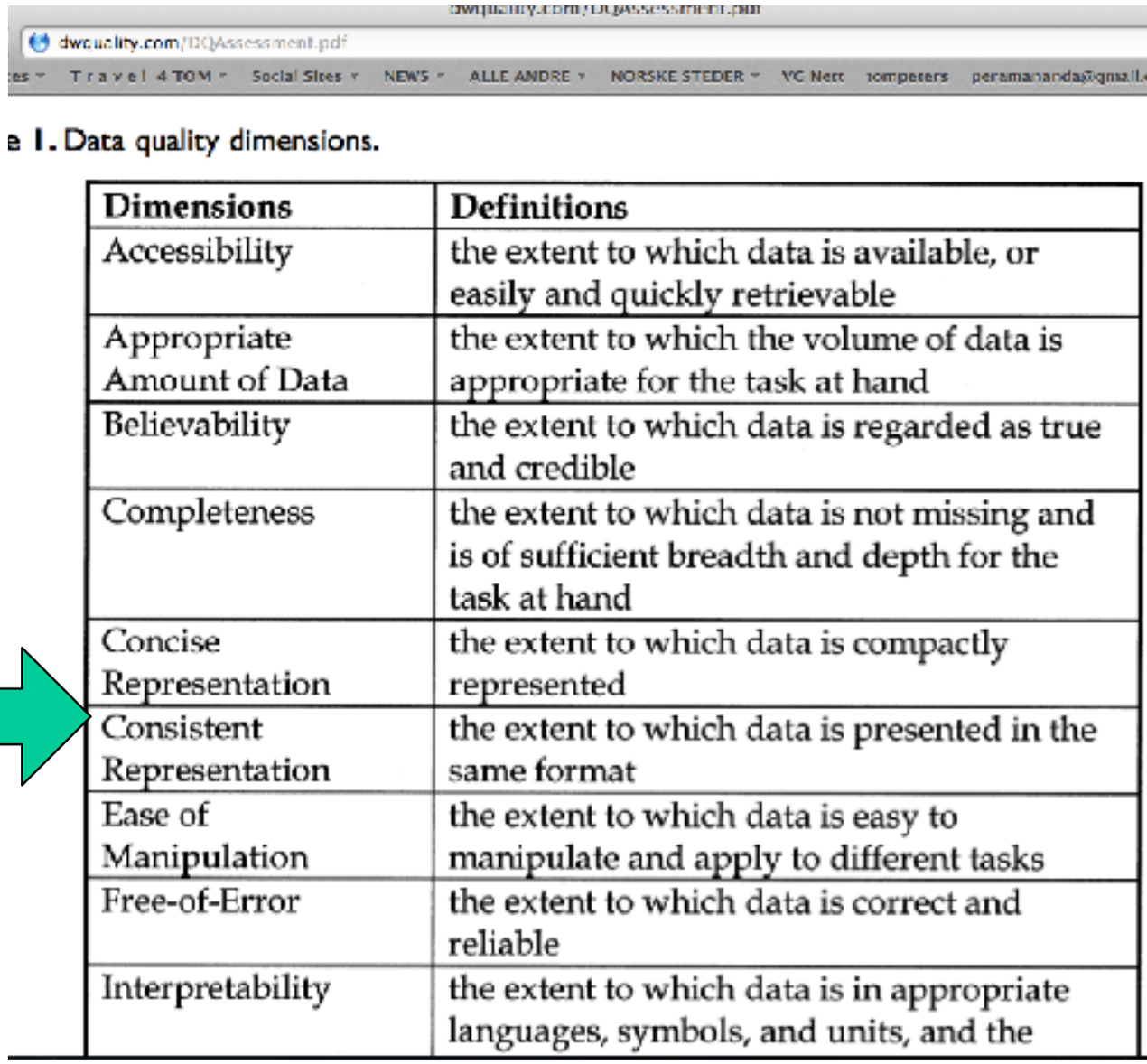
Source: This above is an extension of some basic ideas from Ireson, Editor, Reliability Handbook, *McGraw Hill, 1966* (Ireson 1966).

Quality Quantification Methods #**2**,
Look it up in a book

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].

Quality Quantification Methods #**3**, Google It





e 1. Data quality dimensions.

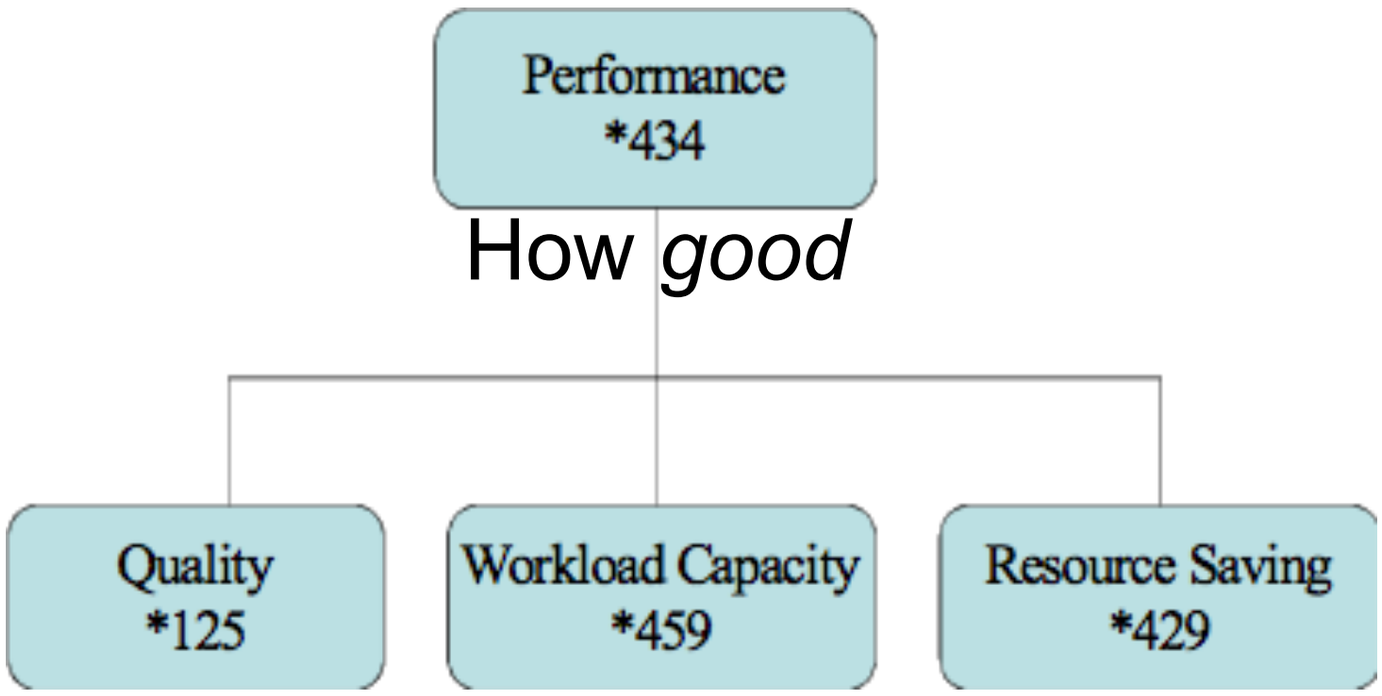
Dimensions	Definitions
Accessibility	the extent to which data is available, or easily and quickly retrievable
Appropriate Amount of Data	the extent to which the volume of data is appropriate for the task at hand
Believability	the extent to which data is regarded as true and credible
Completeness	the extent to which data is not missing and is of sufficient breadth and depth for the task at hand
Concise Representation	the extent to which data is compactly represented
Consistent Representation	the extent to which data is presented in the same format
Ease of Manipulation	the extent to which data is easy to manipulate and apply to different tasks
Free-of-Error	the extent to which data is correct and reliable
Interpretability	the extent to which data is in appropriate languages, symbols, and units, and the

Quality: the concept, the noun

Planguage Concept *125, Version: March 20, 2003

A '**quality**' is

- a scalar attribute -|-|-|-|- (Scale symbol)
- reflecting 'how *well*' ------Past Level<---------->
- a system functions. (Fn)------Past Level<-------->



Quality *is characterized by these traits (from CE book)*

1. **Quality describes 'how well' a function is done.**
2. **Quality describes the *partial effectiveness* of a function (as do all other performance attributes).**
3. **Quality is *valued* to *some* degree by *some* stakeholders of the system**
4. ***More* quality is generally *valued* by stakeholders; especially if the increase is free, or lower cost, than the value of the increase.**
5. **Quality attributes can be *articulated* independently of the particular means (designs) used for reaching a specific quality level –**
6. **even though all quality levels *depend* on the particular designs used to achieve them.**
7. **A particular quality can be a described in terms of a *complex* concept, consisting of multiple elementary quality concepts.**
8. **Quality is *variable* (along a definable scale of measure: as are all scalar attributes).**
9. **Quality levels are capable of being specified *quantitatively* (as are all scalar attributes).**
10. **Quality levels can be *measured* in practice.**
11. **Quality levels can be traded off to some degree; with other system attributes valued more by stakeholders.**
12. **Quality can never be perfect (100%), in the real world.**
13. **There are some levels of a particular quality that may be outside the state of the art; at a defined time and circumstance.**
14. **When quality levels increase towards perfection, the resources needed to support those levels tend towards infinity.**

Quality *is characterized by these traits*

1. Quality describes 'how well' a function is done.
2. Quality describes the *partial effectiveness* of a function (as do all other performance attributes).
3. Quality is *valued* to *some* degree by *some* stakeholders of the system
4. *More* quality is generally *valued* by stakeholders; especially if the increase is free, or lower

9. Quality levels are capable of being specified *quantitatively* (as are all scalar attributes).

8. Quality is *variable* (along a definable scale of measure: as are all scalar attributes).
9. Quality levels are capable of being specified *quantitatively* (as are all scalar attributes).
10. Quality levels can be *measured* in practice.
11. Quality levels can be traded off to some degree; with other system attributes valued more by stakeholders.
12. Quality can never be perfect (100%), in the real world.
13. There are some levels of a particular quality that may be outside the state of the art; at a defined time and circumstance.
14. When quality levels increase towards perfection, the resources needed to support those levels tend towards infinity.

Love Quantification

a 4.5 minute lightening Talk at ACCU Conference, Oxford April 15 2010

Class Exercise: Aspects of Love, or Love is a many splendored thing!

- METHOD
  - Make a list of love's many aspects
  - Quantify *one* random requirement, for love
    - To show that all of the aspects can be similarly quantified

Love Attributes:
Brainstormed By Dutch Engineers

- **Kissed-ness**
- **Care**
- **Sharing**
- **Respect**
- **Comfort**
- **Friendship**
- **Sex**
- **Understanding**
- **Trust**

- Support
- Attention
- Passion
- Satisfaction
- ...
- ...
- ...

Trust Defined

- **Love.Trust.Truthfulness**

  Ambition: No lies.

  Scale:

  Average **Black** lies/month from [defined sources].

  Meter:

  independent confidential log from sample of the defined sources.

  Past Lie Level:

  Past [My Old Mate, 2004] 42 <-Bart

  Goal

  [My Current Mate, Year = 2005] Past Lie Level/2

  Black: Defined: Non White Lies

- Other aspects of Trust:
- 1. 'Truthfulness'

  2. Broken Agreements

  3. Late Appointments

  4. Late delivery

  5. Gossiping to Others

Camaraderie (Real Case UK)

Ambition: *to maintain an exceptionally high sense of good personal feelings and co-operation amongst all staff: family atmosphere, corporate patriotism. In spite of business change and pressures.*

Scale: **probability that individuals enjoy the working atmosphere so much that they would not move to another company for less than 50% pay rise**.

Meter: Apparently real offer via CD-S

Past [September 2001] 60+ % <- R & CD

Goal [Mid 2002] 10%, [End 2002] **<1%** <- R & CD

Rationale:

maintain staff number, and morale as core of business and business predictability for customers.

My ‘Christian’ Friend

- Lawrence Day. Seattle Washington
- “Love is not quantifiable”
  - Not in Bible
  - Little guidance from God and Jesus

Love: Biblical Dimensions
<- Lawrence Day, Boeing

The biblical citation (Book of First Corinthians, Chapter 13) I included gives the quantification of the term "love" (agape in Greek). The 'quantification' for love would be as follows:

---------->

A person who loves acts the following way toward the person being loved:

1. **suffereth long**
2. **is kind**
3. **envieth not**
4. **vaunteth not itself, vaunteth...:**
   or, is not rash (Vaunt = extravagant self praise)
5. **is not puffed up**
6. **Doth not behave itself unseemly**
7. **seeketh not her own**
8. **is not easily provoked**
9. **thinketh no evil**
10. **Rejoiceth not in iniquity (=an unjust act)**
11. **rejoiceth in the truth**
12. **Beareth all things**
13. **believeth all things**
14. **hopeth all things**
15. **endureth all things**
16. **never faileth**

A Paper on ‘Love Quantified’

http://www.gilb.com/tiki-download_file.php?fileId=335

Love Quantified

By:

Lawrence E. Day

for

Dr. Larry Beebe

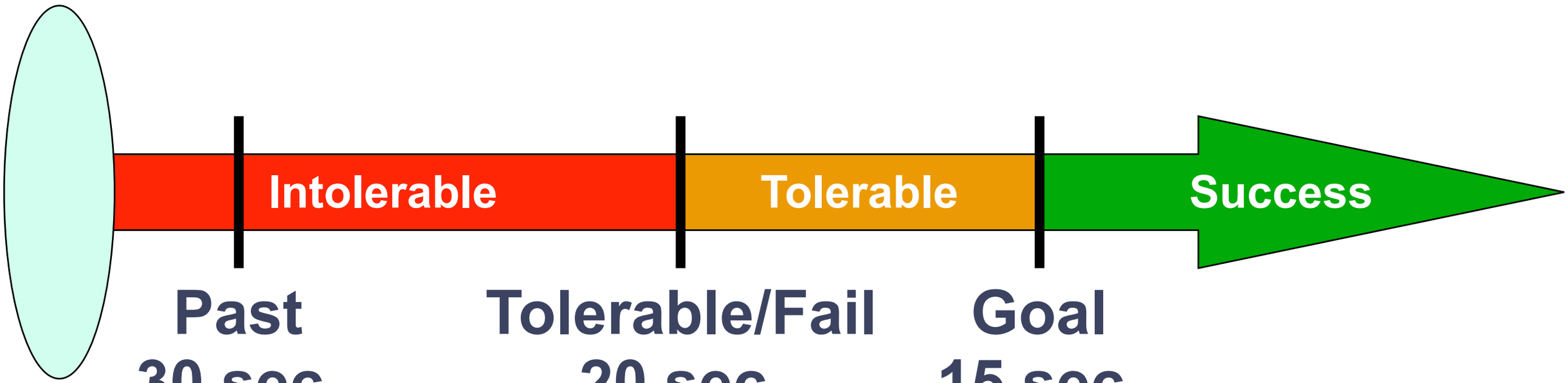
And

Dr. Raghu Korrapati

Table of Con

The ‘Evo’ Agile method: Value Focus

Evolutionary Delivery is driven by meeting Stakeholder Value & Product Quality Requirements

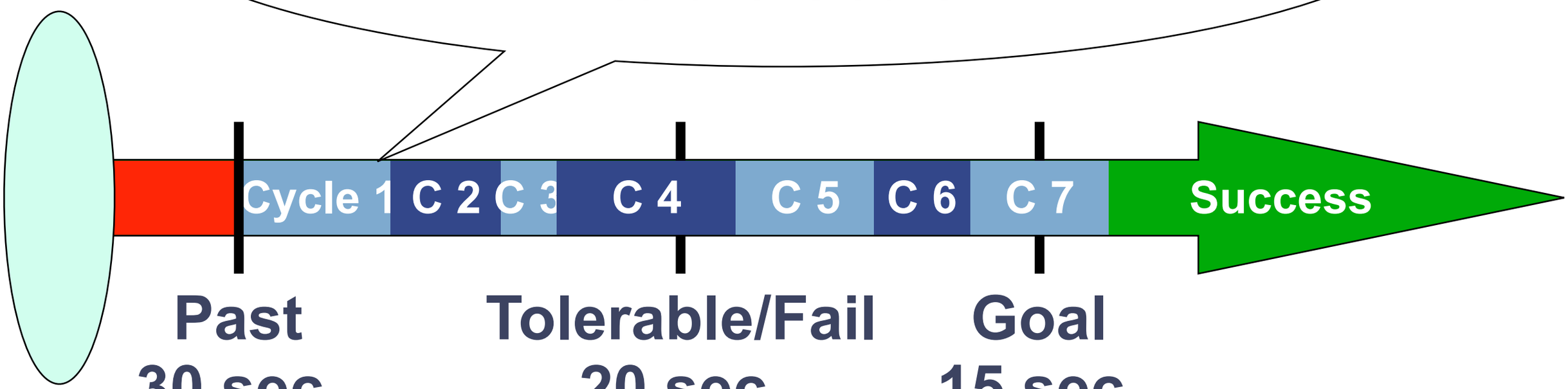


Speed

Scale: seconds to do task

Evolutionary Delivery is driven by meeting Stakeholder Value & Product Quality Requirements

Each Evolutionary Cycle aiming to get closer to the Goals

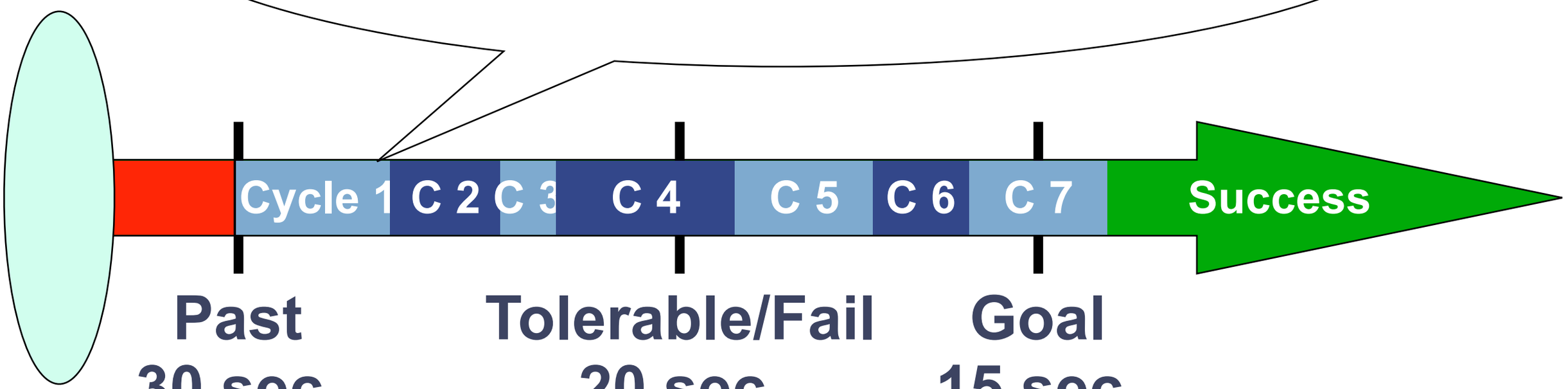


Speed

Scale: seconds to do task

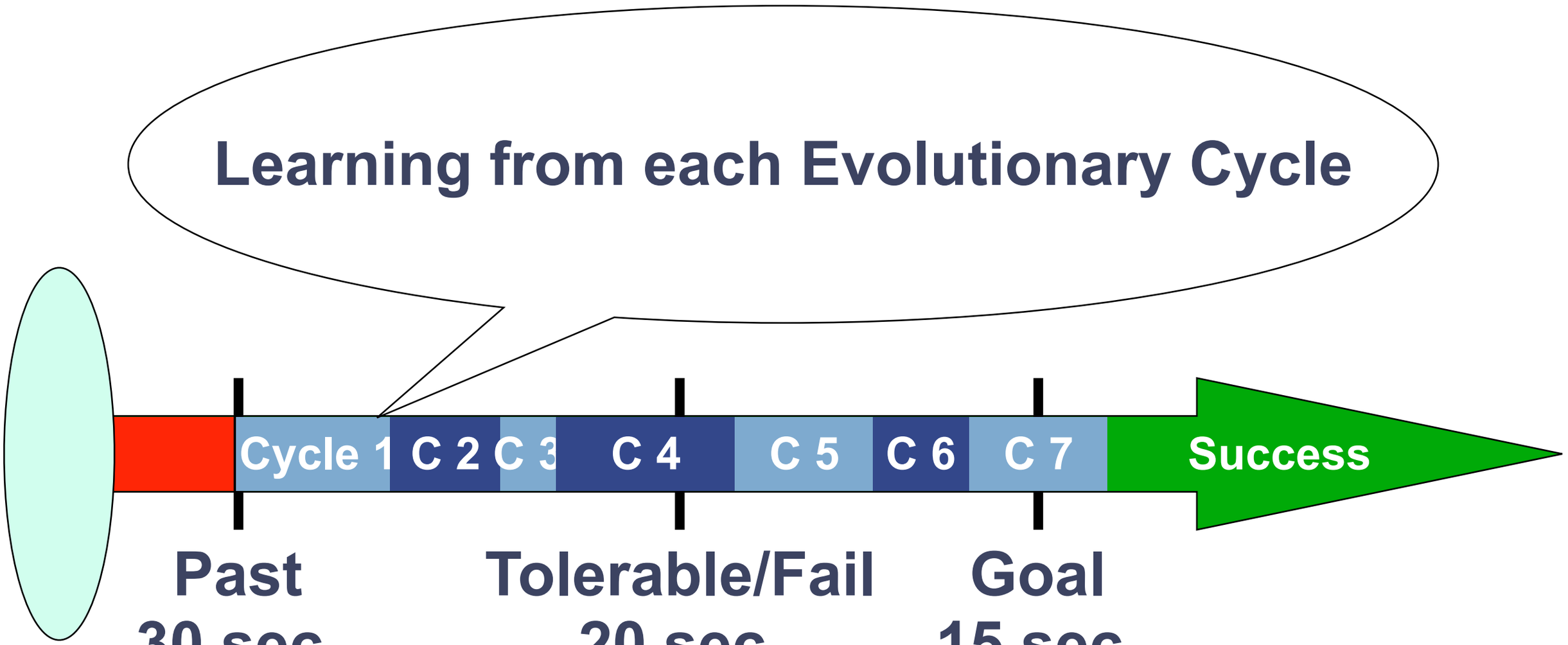
Evolutionary Delivery is driven by meeting Stakeholder Value & Product Quality Requirements

Each Evolutionary Cycle integrated into a 'working' system



<u>Speed</u>
Scale: seconds to do task

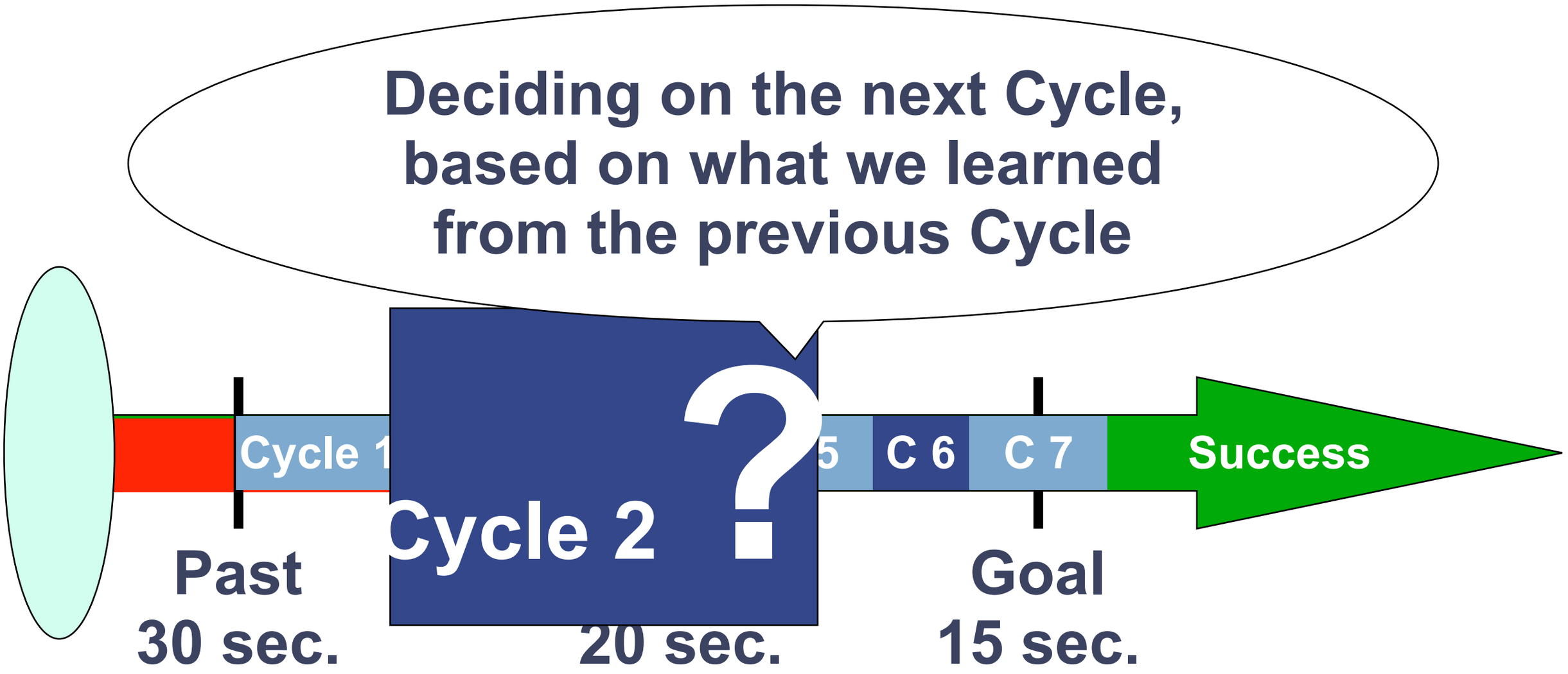
Evolutionary Delivery is driven by meeting Stakeholder Value & Product Quality Requirements



Speed

Scale: seconds to do task

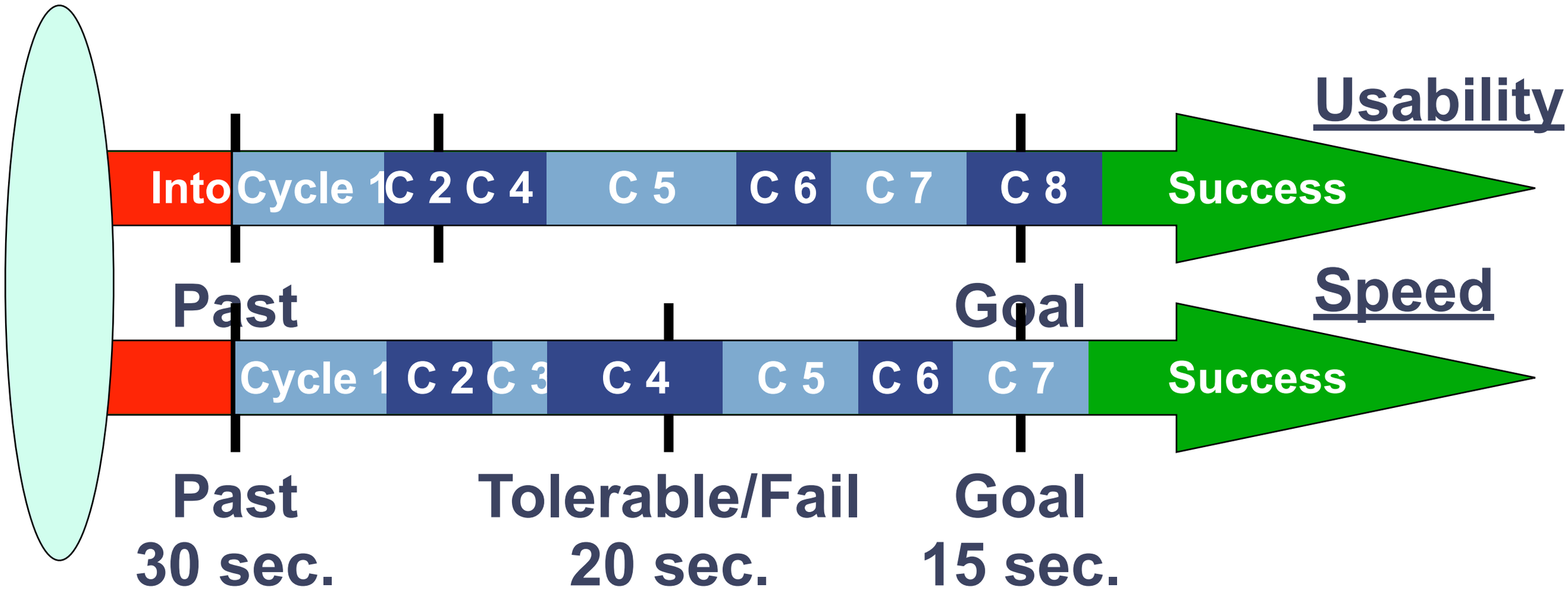
Evolutionary Delivery is driven by meeting Stakeholder Value & Product Quality Requirements



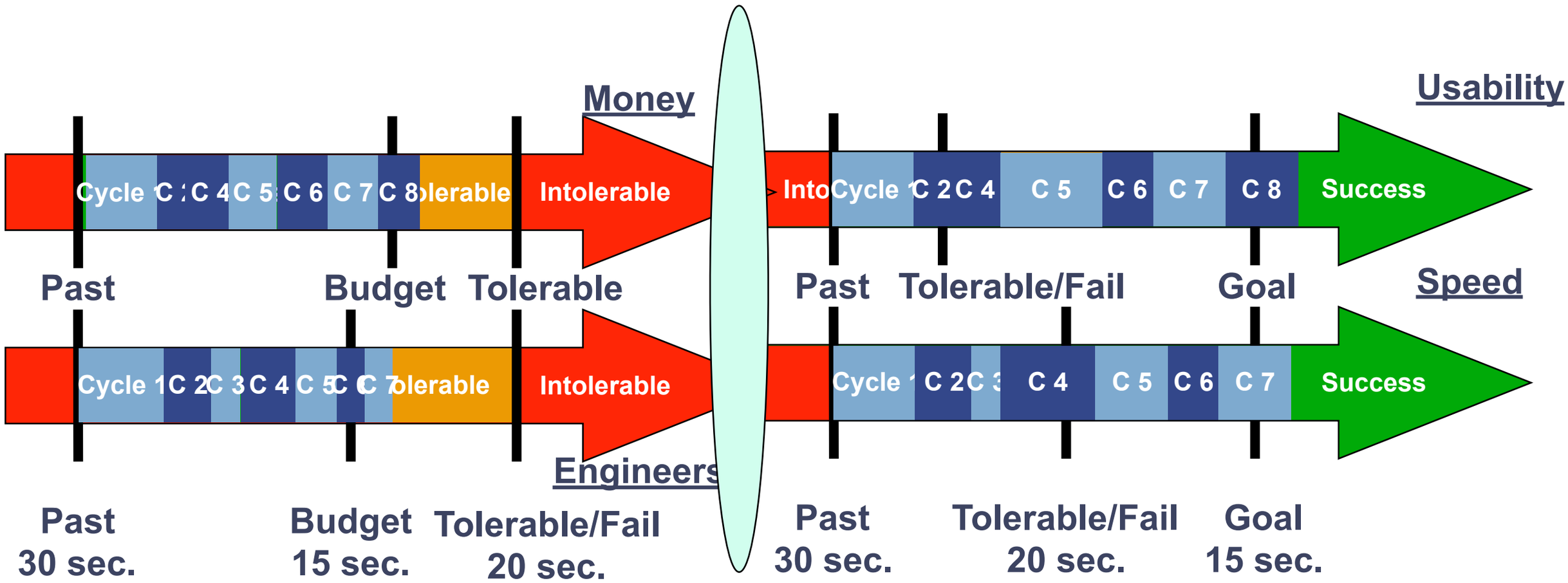
Speed

Scale: seconds to do task

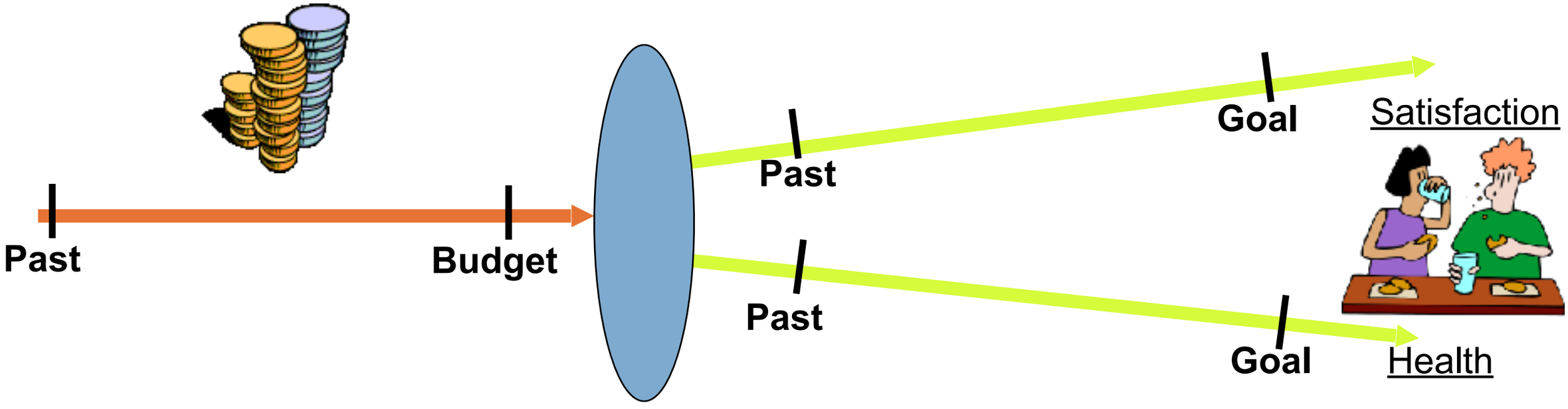
Evolutionary Delivery is driven by meeting Stakeholder Value & Product Quality Requirements Simultaneously



Each Evolutionary Cycle uses a constrained budget of Development Resources

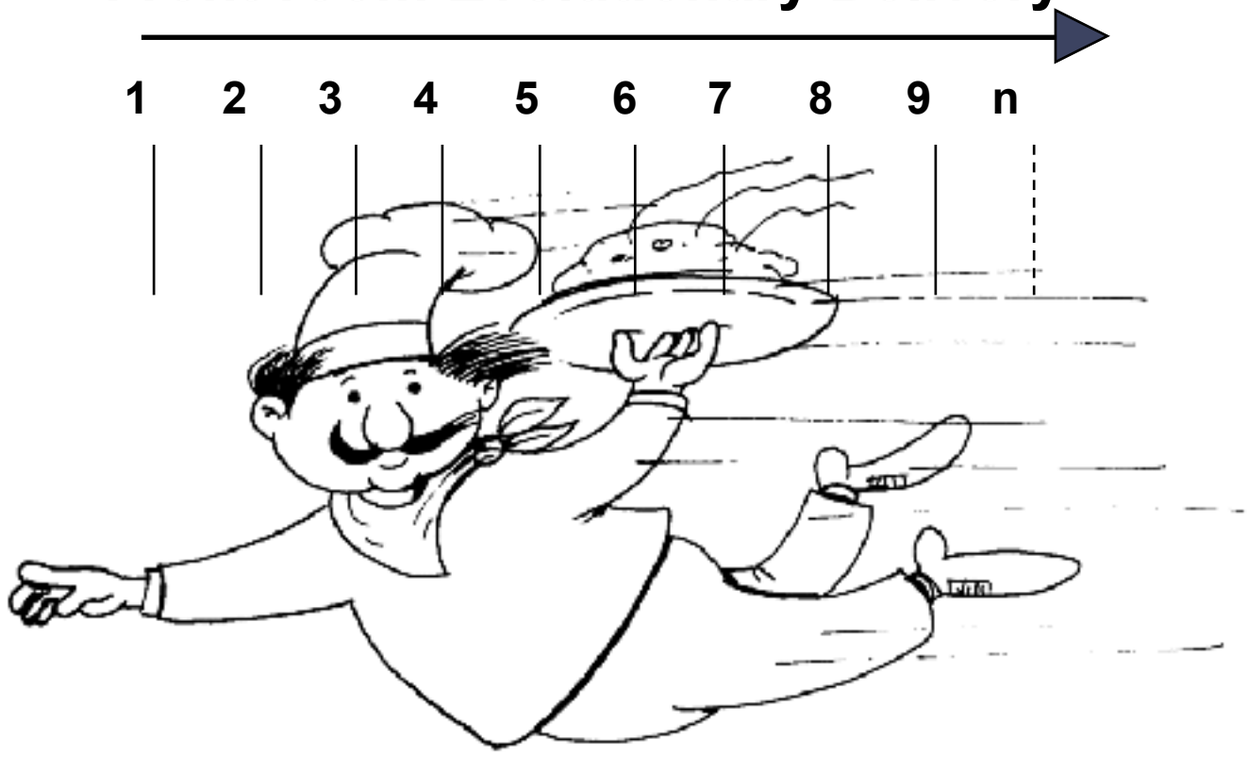


Costs / Effects

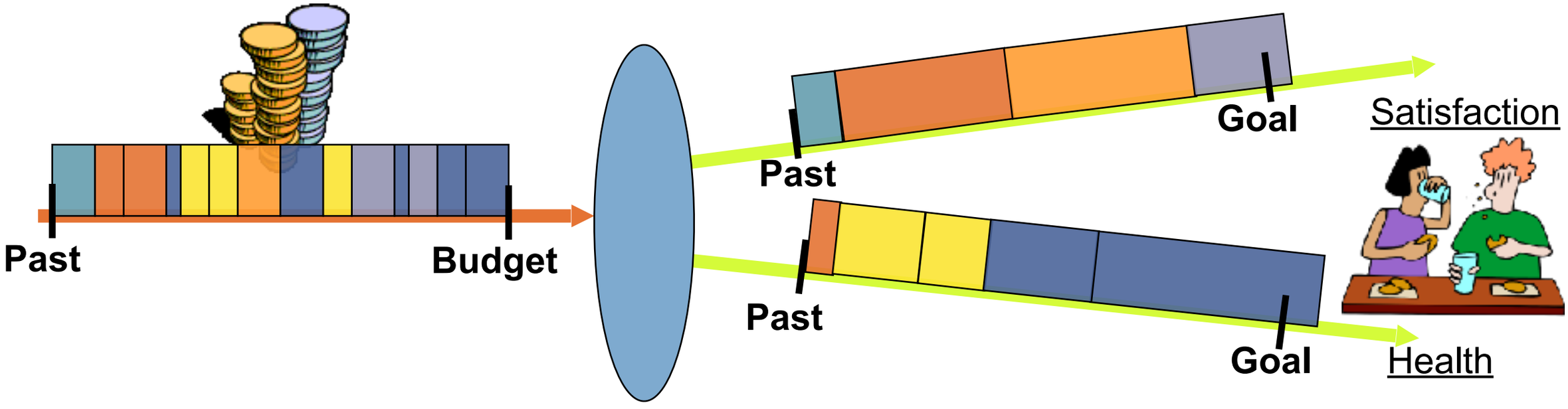


Back-room Design Development

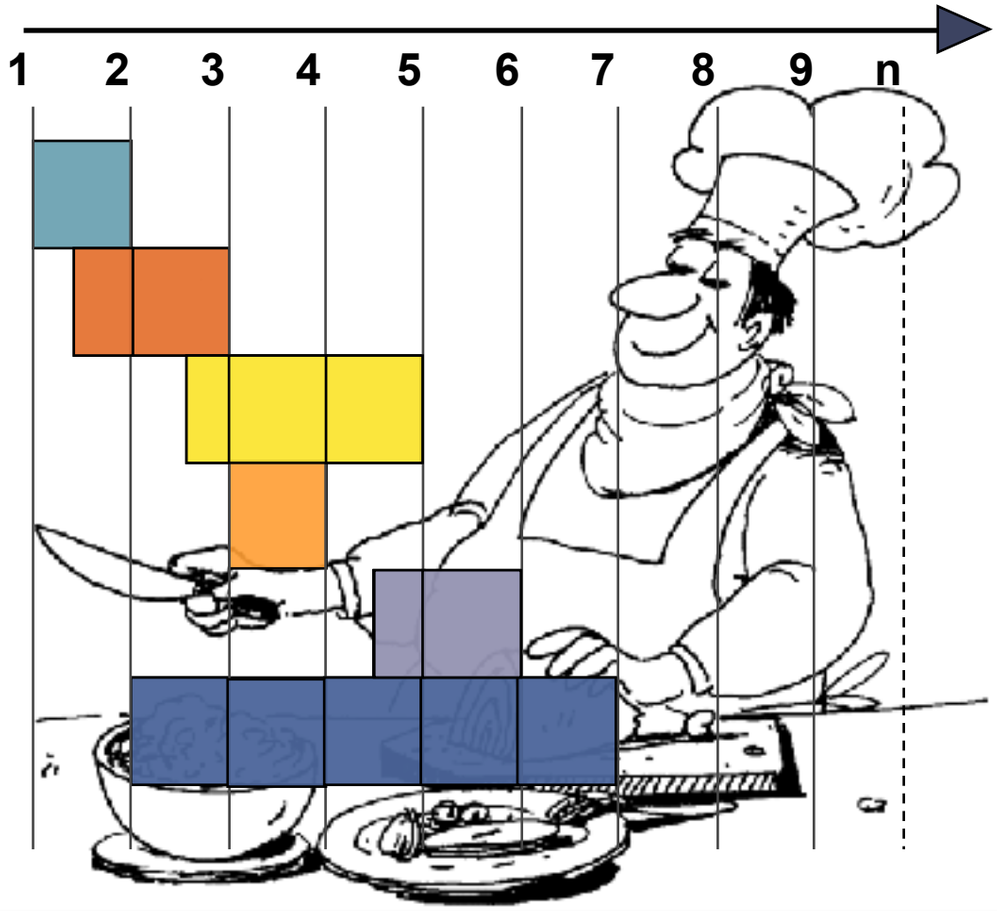
Front-room Evolutionary Delivery



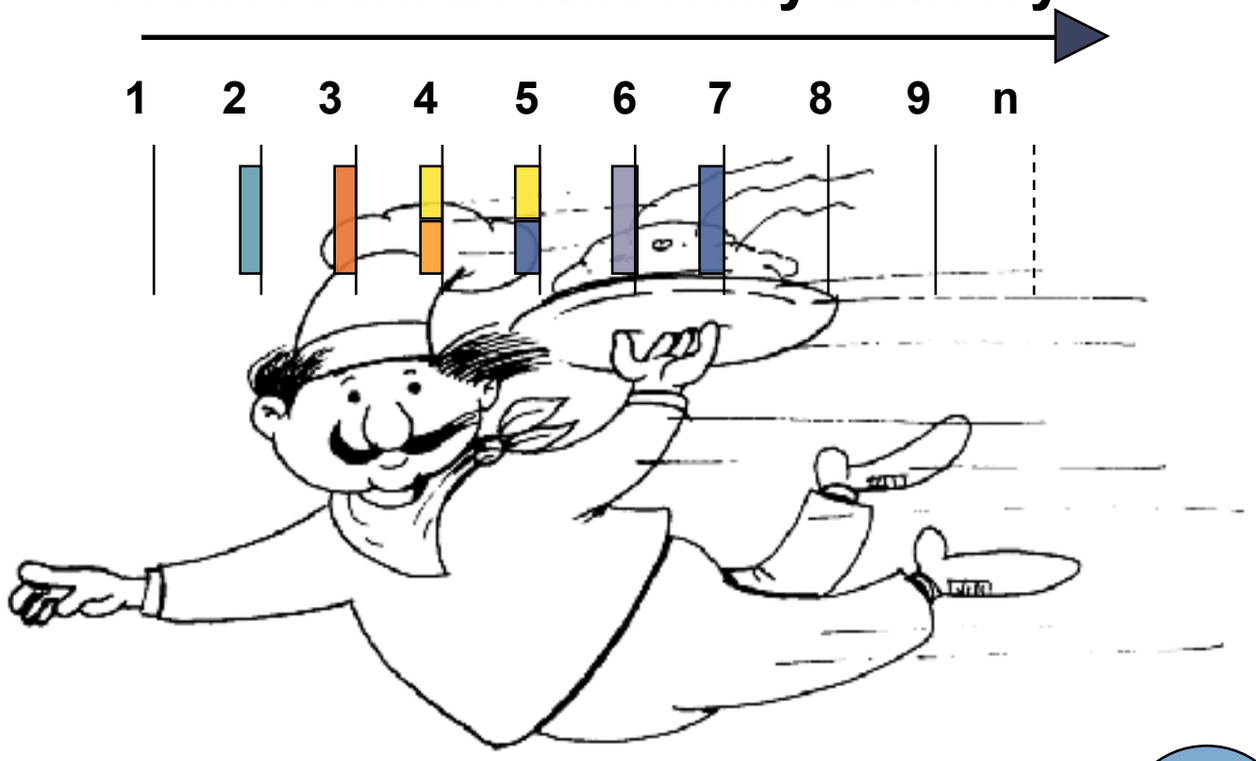
Costs / Effects



Back-room Design Development



Front-room Evolutionary Delivery



Spec QC

Source Eric Simmons, hoofdwerk@gmail.com 25 Oc
Personal Public Communication

A Recent Example

Application of Specification Quality Control by a SW team resulted in the following defect density reduction in requirements over several months:

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%

Downstream benefits:
•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

Software Engineering Principles

Software Engineering Principles

- **Real engineering is about rigorous quantified models of the problem and solutions**

- **All critical objectives and resource constraints must be quantified**

- **Agile (Evo) value delivery is the main point, not ‘programming**

- **all projects have several simultaneous critical quantifiable objectives**

- **all design ideas have many quantifiable, estimable, measurable attributes, that can be managed by an engineering**

- Copyright: 2016 tom@Gilb.com