Repository of All Slides

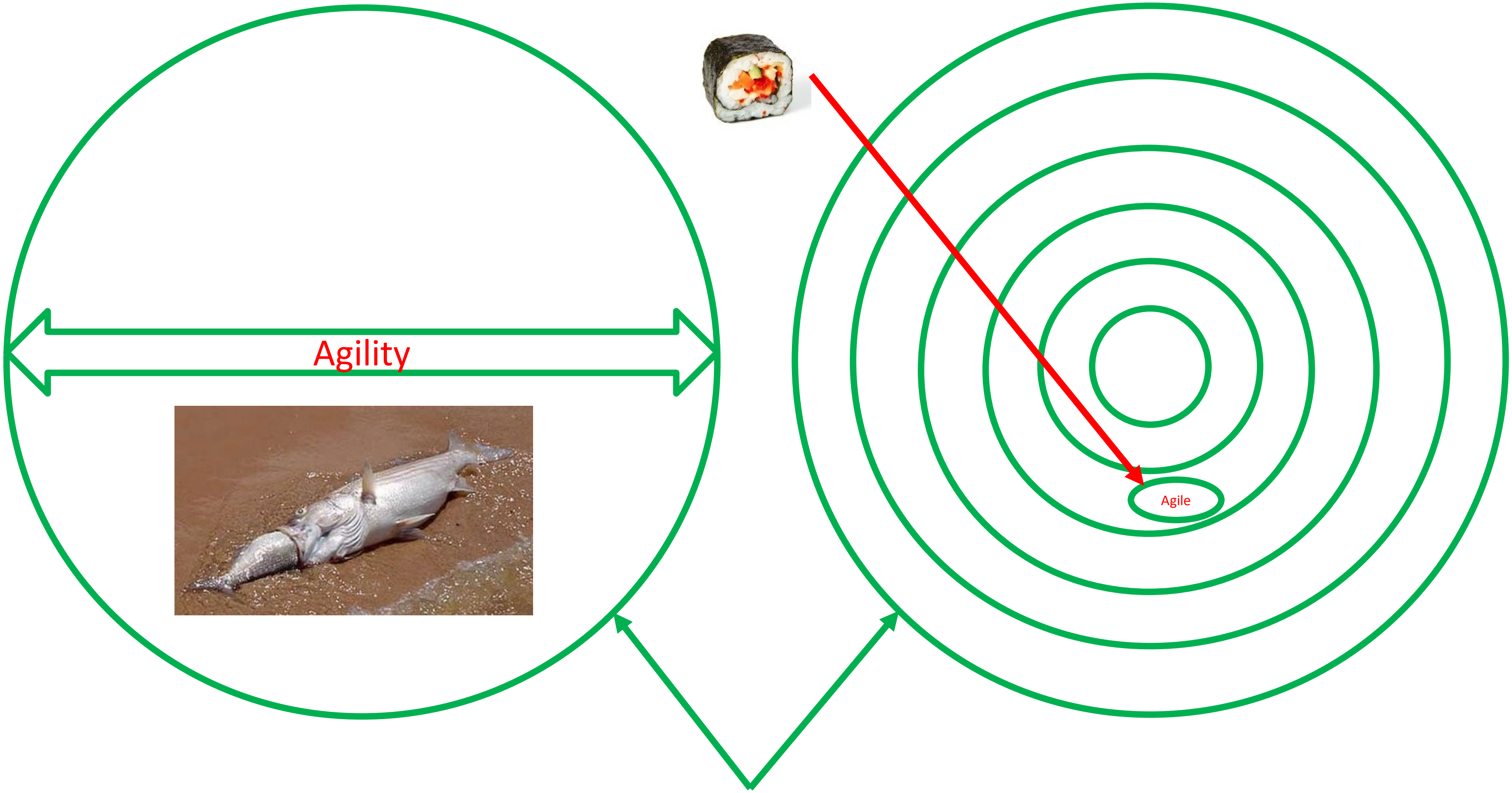
(Training, Presentations, etc)

Organizational Agility

Looking Under the Hood

Where Do We Draw Boundaries?

(of Agile)



Agility

Agile

Outermost Boundaries of Organization

Where to start→

You Get What You Ask (Pay) For

Job Title: Agile Coach/ScrumMaster/Project Manager/Release-Delivery Manager/BA

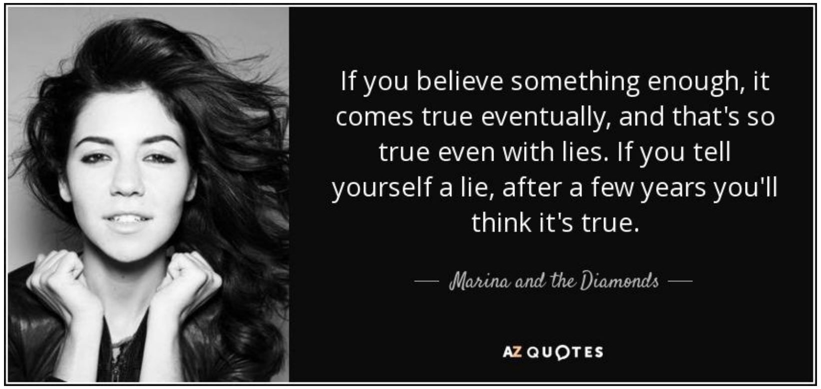
Key Responsibilities:

…EVERYTHING under the Sun…

Certifications/Accreditations:
CSM, CSPO, CSD, CSP, CEC, CTC, CST, PSM, PST, ACP….and of course PMP

Personal Qualifications:
Go-getter, star performer, super-achiever, highly competitive

Compensation:
Salary + Bonus (based on Individual Performance)



If you believe something enough, it comes true eventually, and that's so true even with lies. If you tell yourself a lie, after a few years you'll think it's true.

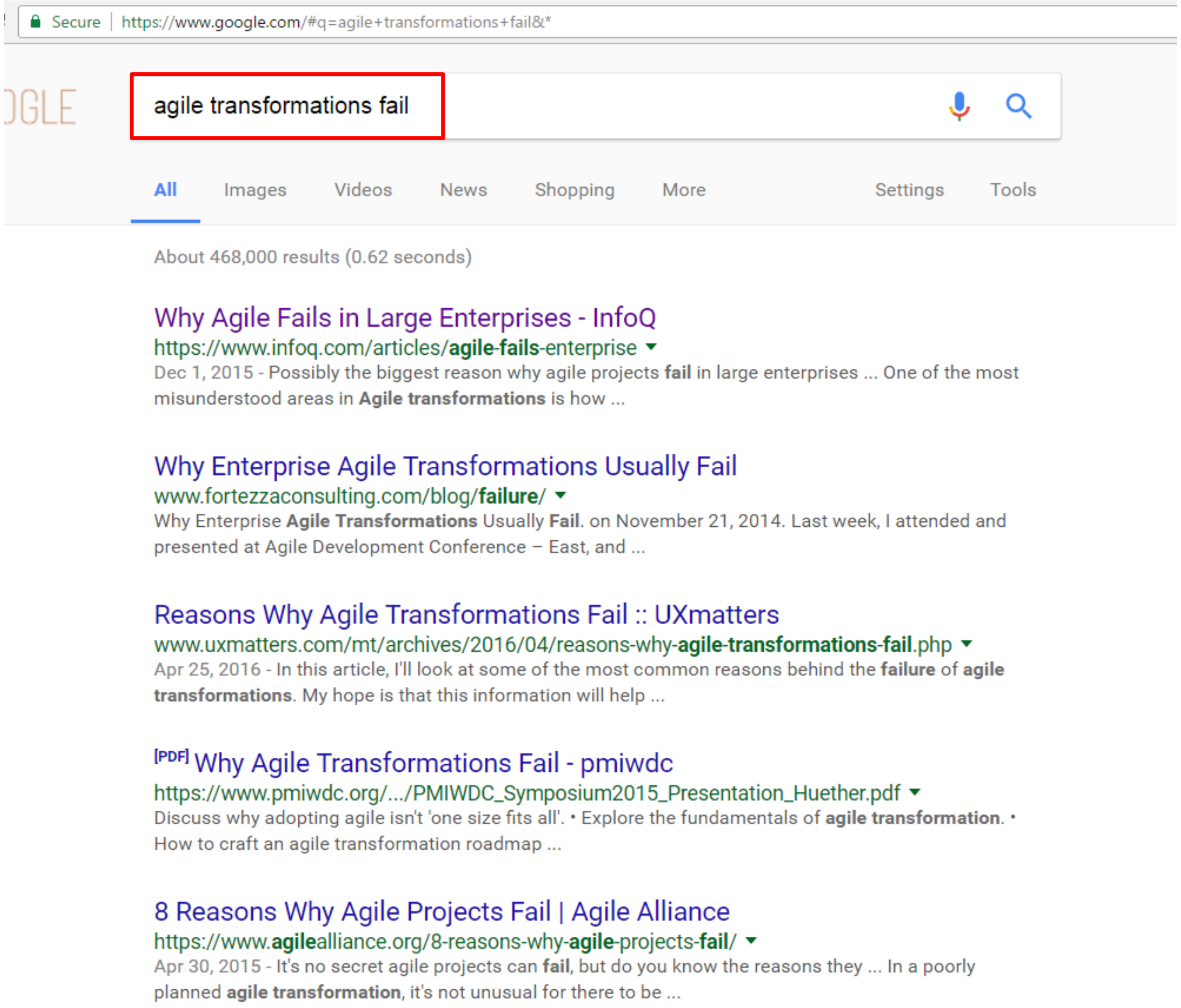
— Marina and the Diamonds —

Ignorance/Arrogance→

Why Many Fake Agile "Adoptions"?

Ignorance

Arrogance



agile transformations fail

About 468,000 results (0.62 seconds)

Why Agile Fails in Large Enterprises - InfoQ
https://www.infoq.com/articles/**agile-fails**-enterprise
Dec 1, 2015 - Possibly the biggest reason why agile projects **fail** in large enterprises ... One of the most misunderstood areas in **Agile transformations** is how ...

Why Enterprise Agile Transformations Usually Fail
www.fortezzaconsulting.com/blog/**failure**/
Why Enterprise **Agile Transformations** Usually **Fail**. on November 21, 2014. Last week, I attended and presented at Agile Development Conference – East, and ...

Reasons Why Agile Transformations Fail :: UXmatters
www.uxmatters.com/mt/archives/2016/04/reasons-why-**agile-transformations-fail**.php
Apr 25, 2016 - In this article, I'll look at some of the most common reasons behind the **failure** of **agile transformations**. My hope is that this information will help ...

[PDF] Why Agile Transformations Fail - pmiwdc
https://www.pmiwdc.org/.../PMIWDC_Symposium2015_Presentation_Huether.pdf
Discuss why adopting agile isn't 'one size fits all'. • Explore the fundamentals of **agile transformation**. • How to craft an agile transformation roadmap ...

8 Reasons Why Agile Projects Fail | Agile Alliance
https://www.**agile**alliance.org/8-reasons-why-**agile**-projects-**fail**/
Apr 30, 2015 - It's no secret agile projects can **fail**, but do you know the reasons they ... In a poorly planned **agile transformation**, it's not unusual for there to be ...

Are You Looking for Big Solutions?

Grow from Bottom → Up

…Instead of Tailoring down….

BUT….

Slice your transformation efforts just like you slice User Stories during PBR

Secret Formula Of “Muda Tolerance”

F(*tolerance to Muda*)

=

1 / Proximity to Real Customer

Tolerance to Muda is Low

Tolerance to Muda is High

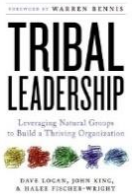
Muda *= “Waste” or failures of* ***people or processes*** *to efficiently deliver product.* ***Mura*** *= “Unevenness”, or failures related to unpredictable or inconsistent* ***outputs****.* ***Muri*** *= “Overburden”, or* ***failures of standardization*** *to create efficient process.*

The Toyota 3M model: Muda, Mura, Muri

Manifesto V →

Organizational Development “Tribal Stages”

By David Logan



“Life is Great”: “Less than **2 percent of workplace** tribal culture is in this stage, wherein members who have made substantial innovations seek to use their potential to make a global impact.”

“We’re great (and they’re not)”: “The transition from “I’m great” to “we’re great” comes in this stage, wherein the tribe members are excited to work together for the benefit of the entire company.”

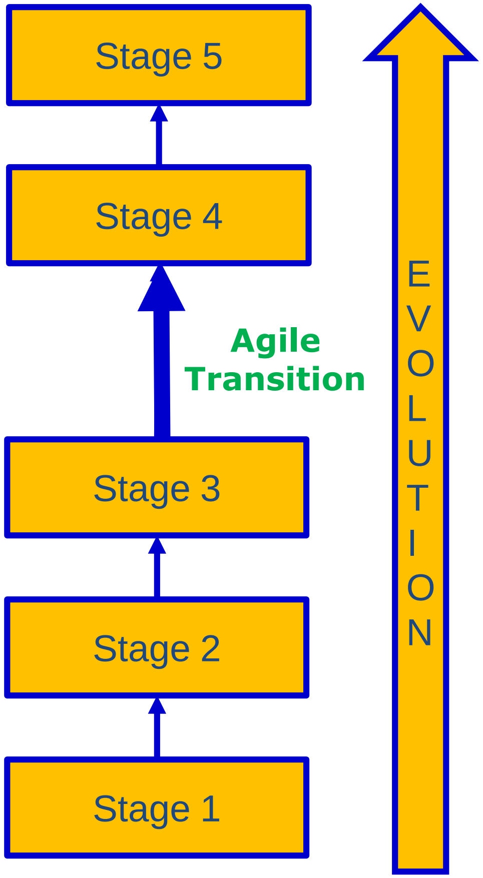
Collective Performers and Team Players, External Competitors

Individual Star Performers, Super Heroes, Internal Competitors

“I’m great (and you’re not)”: “49 percent of workplace tribes are in this stage, marked by knowledge hoarders who want to outwork and outthink their competitors on an individual basis. They are lone warriors who not only want to win, but need to be the best and brightest.”

“My life Sucks”: “The dominant culture for 25 percent of workplace tribes, this stage includes members who are passively antagonistic, sarcastic, and resistant to new management initiatives..”

“Life Sucks”: “The stage most professionals skip, these are tribes whose members are despairingly hostile—they may create scandals, steal from the company, or even threaten violence.”



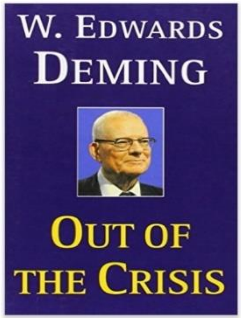
# 5 - Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

Creating “We”
Agile Environments

By bridging the gap between science and business

“The idea of a merit rating is alluring. The sound of the words captivates the imagination: pay for what you get; get what you pay for; motivate people to do their best, for their own good. The effect is exactly the opposite of what the words promise.”

Edward Deming, 1982



Problem We Are Trying to Solve

Internal Contracts (me vs. you)

Finger Pointing/Blame Shifting

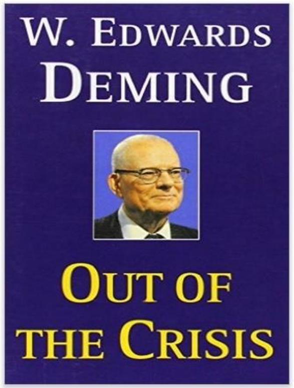
System Gaming, to look good again

This Hurts Companies

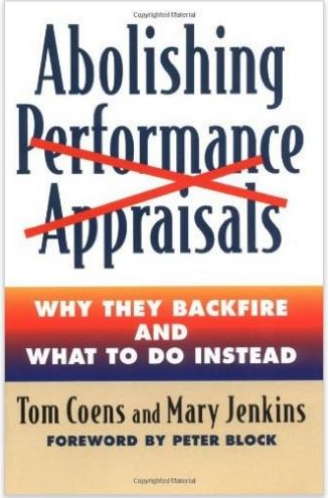
Agile Teams (Scrum)

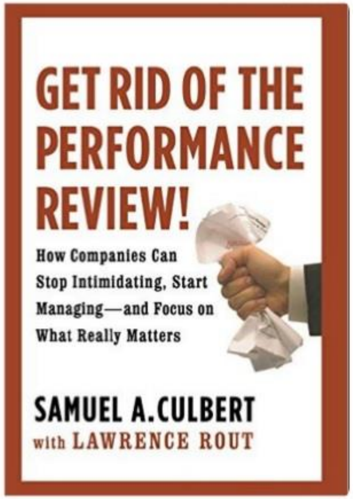
- ❑ Shared Commitment
- ❑ Shared Ownership
- ❑ Shared Responsibilities
- ❑ Shared Deliverables
- ❑ Shared Successes & Failures

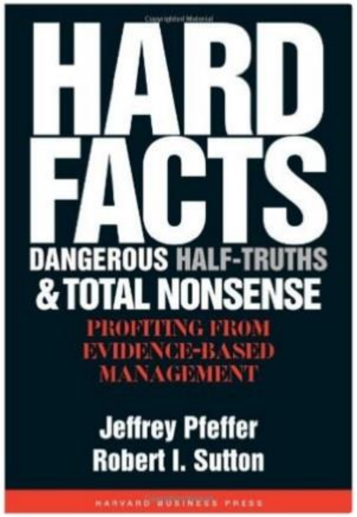
Gap Between Science and Business

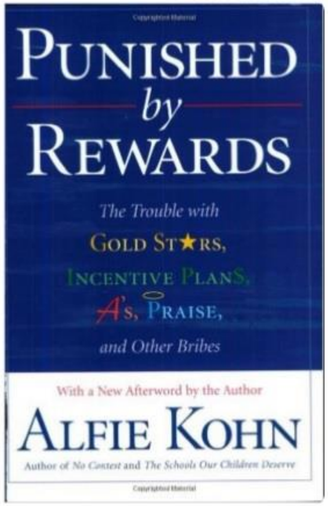


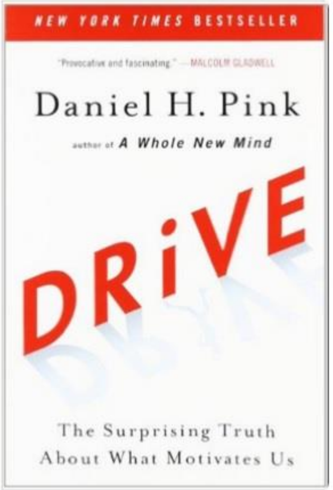
"The idea of a merit rating is alluring. The sound of the words captivates the imagination: pay for what you get; get what you pay for; motivate people to do their best, for their own good. The effect is exactly the opposite of what the words promise."


Abolishing
~~Performance Appraisals~~
Why they backfire and what to do instead
Tom Coens and Mary Jenkins
Foreword by Peter Block


Get rid of the performance review!
How Companies Can Stop Intimidating, Start Managing—and Focus on What Really Matters
Samuel A. Culbert with Lawrence Rout


Hard Facts
Dangerous half-truths & total nonsense
Profiting from evidence-based management
Jeffrey Pfeffer
Robert I. Sutton
Harvard Business Press


Punished by Rewards
The Trouble with Gold Stars, Incentive Plans, A's, Praise, and Other Bribes
With a New Afterword by the Author
Alfie Kohn


New York Times Bestseller
Daniel H. Pink
author of A Whole New Mind
Drive
The Surprising Truth About What Motivates Us

RSA ANIMATE: Drive: The surprising truth about what motivates us (Daniel Pink)

Gap Between Science and Business

Samuel A. Culbert

"…This corporate sham is one of the most insidious, most damaging, and yet most ubiquitous of corporate activities. Everybody does it, and almost everyone who's evaluated hates it. It's a pretentious, bogus practice that produces absolutely nothing that any thinking executive should call a corporate plus…."

Book quotes - summarized

Alfie Kohn

"…Frederick Herzberg observed that managers who emphasize rewards and punishments "offer their own motivational characteristics as the pattern to be instilled in their subordinates. They become the template from which the new recruit to industry learns his motivational pattern…."

Book quotes - summarized

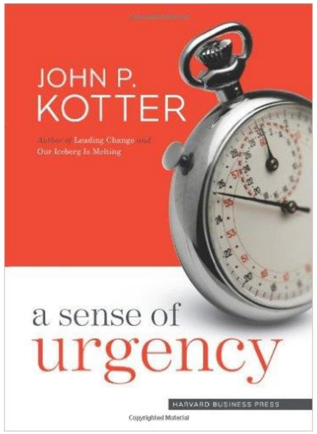
GAP BETWEEN SCIENCE AND BUSINESS

Beware of Resistance

"It is difficult to get a man to understand something, when his salary depends on his not understanding it."

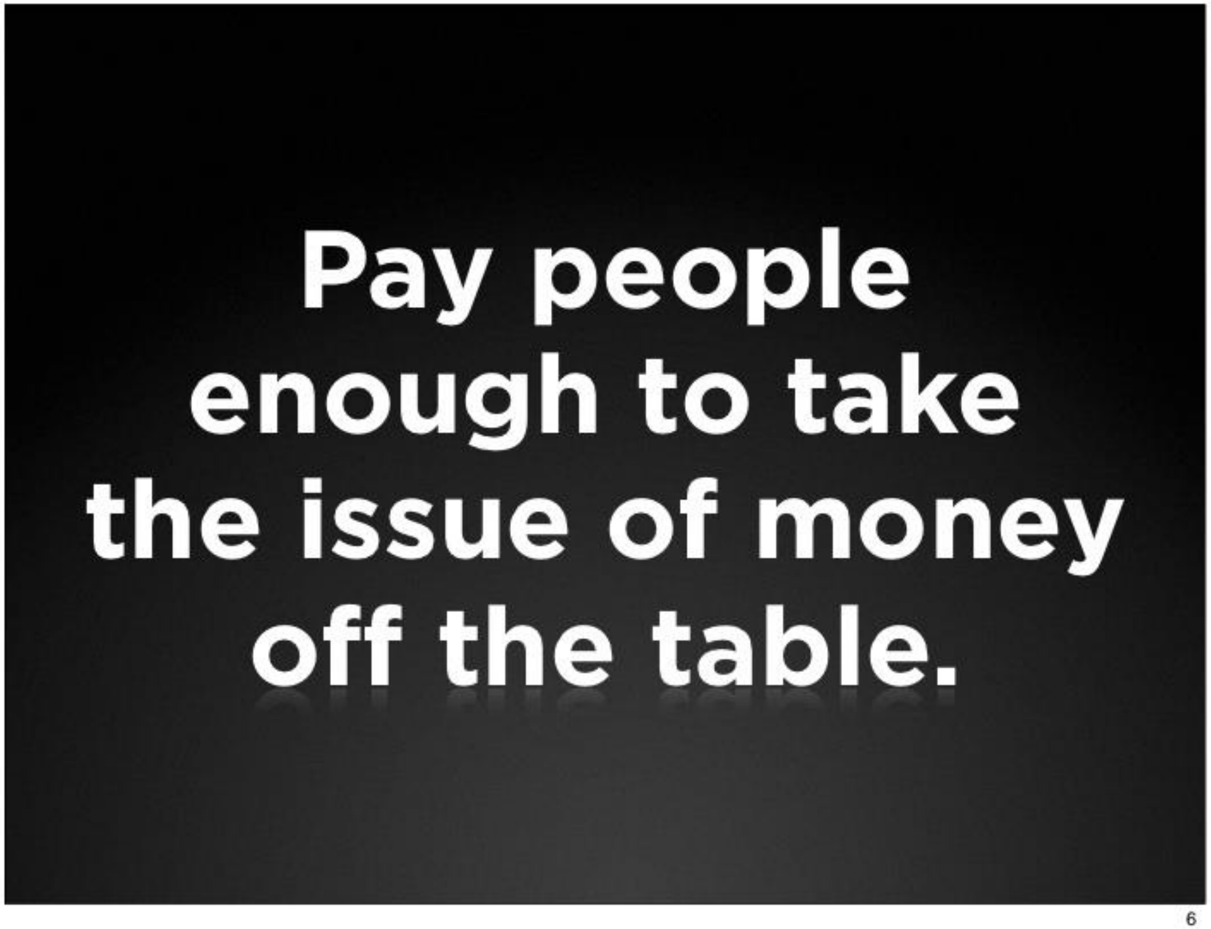
Upton Sinclair

Resistance to Change
By John Kotter



=Full Comp?
(base + bonus)

How to Fix It? → Plan A



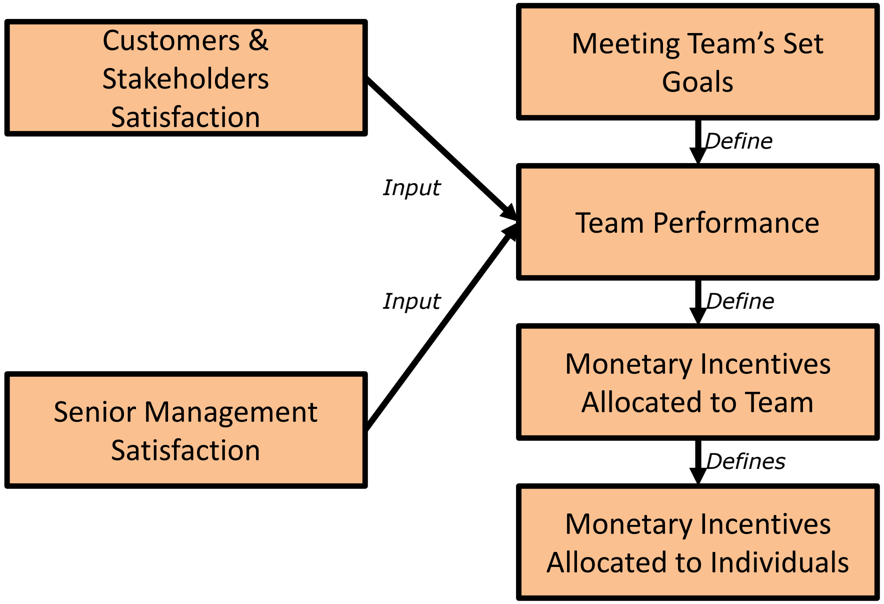
Pay people enough to take the issue of money off the table.

6

…Exceptions could be jobs in sales/marketing, etc….

…But still, look out for and avoid internal competition….

How to Fix It? → Plan B


Customers & Stakeholders Satisfaction
Senior Management Satisfaction
Input
Input
Meeting Team's Set Goals
Define
Team Performance
Define
Monetary Incentives Allocated to Team
Defines
Monetary Incentives Allocated to Individuals

This Monetary Incentives Allocation Schema would be appropriate for Organizations:

- That have decided to mature from Tribal Level 3 to Tribal Level 4, as per David Logan's Tribal Evolution Scale
- Where work performed and deliverables produced by a single individual carry significantly less intrinsic business value than what is produced by jointly working team
- That see more value in promoting and incentivizing Team Level Performance than Individual Level Performance, as a way to prevent internal competition and rivalry
- That recognize that concentrating all decision-making power in hands of a single person (e.g. team-level Line Manager) is prone to bias, favoritism and subjectivity
- That are serious about supporting and developing Kaizen culture that penetrates all organizational layers

Option 1: Monetary Incentives Are Equally Allocated

Use Case:
-5 team members: Members A, B, C, D, E
-Salaries of all team members are very comparable
-Team gets $1000 incentive
-Each team member gets $200 ($1000 / 5)

Note: This option is ideal for equally (or close to equal) compensated, highly skilled, cross-functional team members that feel on par with each other (organizational hierarchy, seniority), are not in competition with each other, passed far beyond forming/storming and deeply understand the benefits of collective efforts

Option 2: Monetary Incentives Are Allocated in Proportion to Base Salary

Use Case:
-5 team members: Members A, B, C, D, E
-Salaries of all team members are NOT comparable
-Team's Total base compensation is equal to base A + base B + base C + base D + base E
-Member's Allocation Coefficient is identified as Member Base/Total Base (e.g. Member A Base/Total Base)
-Team gets $1000 incentive
-Each member gets $1000 x Member's Allocation Coefficient (e.g. Member A gets $1000 x Member A Allocation Coefficient)

Note: This option makes more sense when team members are not equally compensated (junior vs. senior) and are not fully cross-functional. This option is similar to Option 1 in a sense that individuals still do not compete among each other and value of collective work.
With this option, individuals cannot earn bonuses that are disproportionally high, with respect to their base salaries that were defined at the time when people were hired.

Option 3: Monetary Incentives Are Allocated based on Team's Internal Voting

Use Case:
-5 team members: Members A, B, C, D, E
-Salaries of all team members are NOT comparable
-During 360 Team Review, each team member is assigned 100 arbitrary units that he/she can confidentially allocate to other 4 team members (excluding himself).
-This is a periodic event and its cadency decided by a team.
-Points accumulated by each member are not disclosed. "Tale of the Tape" can be monitored by a third party (e.g. Line Manager, ONLY to prevent "one hand washes another" situations, where, for example, members A and B privately agree to give each other all 100 points, reciprocally, to maximize each other's count).
At the end of a year, each team member reveals accumulated total individual points. Points are then added up to produce Team's Total Points. Then, each member's accumulated total points are divided by Team's Total Points to derive Member's Allocation Coefficient – this is the approach similar to Option 2.
From here, the steps from Option 2 are followed.

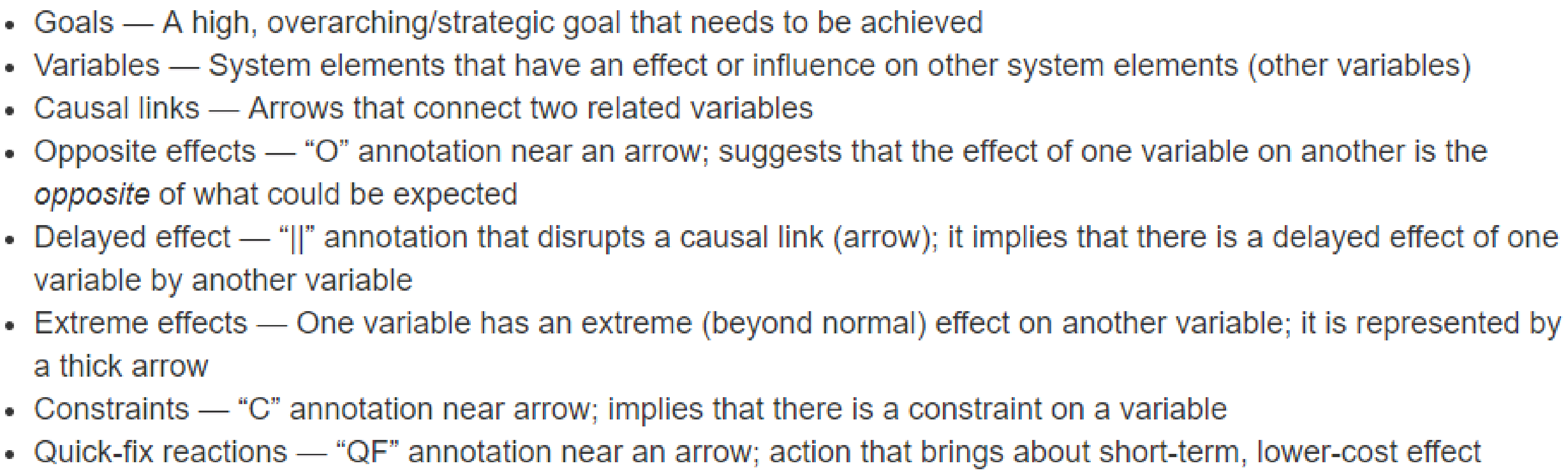
Note: This option introduces the element of self-governance and self-management to a team. It makes more sense to use it on teams, where individuals are not cross-functional and there is a higher potential of individual "slacking" and "free riding", as this approach encourages team members to continuously learn and advance.

Adaptive (Agile) Planning & Reporting

Supported by

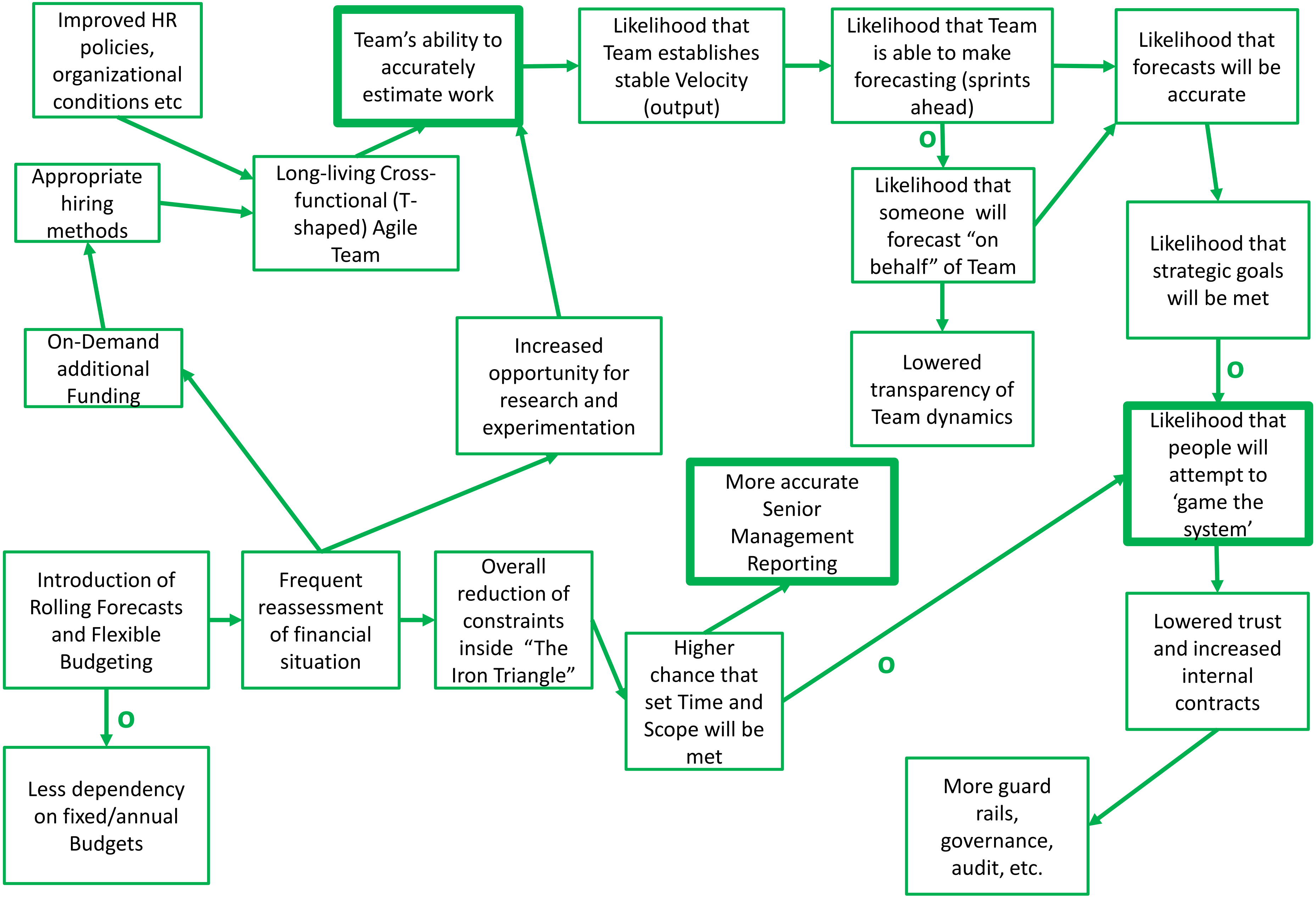
Flexible Budgeting & Rolling Forecasts

CLD Legend



- Goals — A high, overarching/strategic goal that needs to be achieved
- Variables — System elements that have an effect or influence on other system elements (other variables)
- Causal links — Arrows that connect two related variables
- Opposite effects — “O” annotation near an arrow; suggests that the effect of one variable on another is the *opposite* of what could be expected
- Delayed effect — “||” annotation that disrupts a causal link (arrow); it implies that there is a delayed effect of one variable by another variable
- Extreme effects — One variable has an extreme (beyond normal) effect on another variable; it is represented by a thick arrow
- Constraints — “C” annotation near arrow; implies that there is a constraint on a variable
- Quick-fix reactions — “QF” annotation near an arrow; action that brings about short-term, lower-cost effect

CLD



Improved HR policies, organizational conditions etc

Team's ability to accurately estimate work

Likelihood that Team establishes stable Velocity (output)

Likelihood that Team is able to make forecasting (sprints ahead)

Likelihood that forecasts will be accurate

Appropriate hiring methods

Long-living Cross-functional (T-shaped) Agile Team

Likelihood that someone will forecast "on behalf" of Team

Likelihood that strategic goals will be met

On-Demand additional Funding

Increased opportunity for research and experimentation

Lowered transparency of Team dynamics

Likelihood that people will attempt to 'game the system'

More accurate Senior Management Reporting

Introduction of Rolling Forecasts and Flexible Budgeting

Frequent reassessment of financial situation

Overall reduction of constraints inside "The Iron Triangle"

Higher chance that set Time and Scope will be met

Lowered trust and increased internal contracts

Less dependency on fixed/annual Budgets

More guard rails, governance, audit, etc.

Estimating Work

Option 1

Many agile teams, each doing their own thing (e.g. many unrelated Scrum teams). Usually, a result of “let’s do Agile” mindset, due to a lack of system thinking

- Velocities of different teams are not comparable and not ‘addable’ because teams don’t have a shared understanding of work size and complexity.
- BOW/forecasting/planning can be done, based on whole features, not on Story Points (units of work)
- **Note**: in order to ‘add’ Story Points of teams that don’t work together, complex *normalization* techniques should be used, to convert arbitrary units of measure (story points, features) into **TIME → MONEY**
- Much higher chance that Product Portfolio is **FAKE**

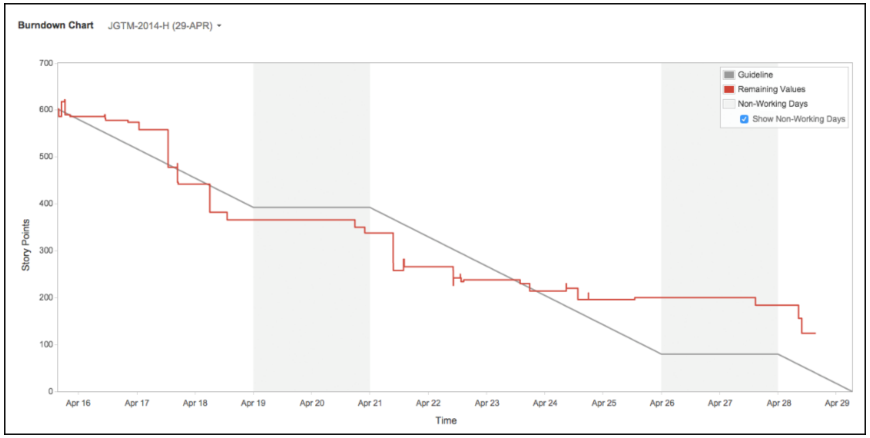
Any unit of measure/scale, used in iterative/incremental/agile product development, when plotted against time, can produce a “burn” rate, a.k.a. *rate of output*. Based on burn rate, and assuming that BOW (a.k.a. Product Backlog) is estimated, forecasting can be done (next sprint only or a longer period of time)

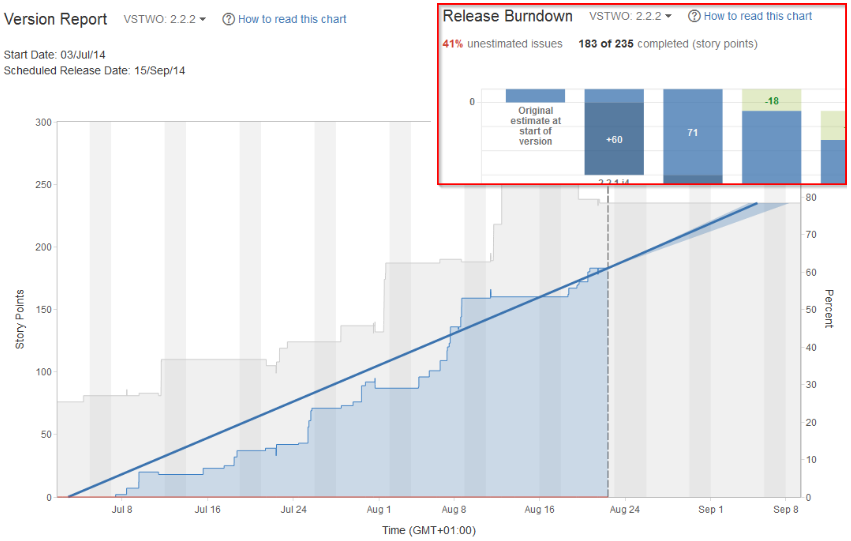
Option 2

Multiple teams working closely together (e.g. on the same product, facing the same customer). E.g., properly designed Large Scale Scrum organization

- Velocities of different teams are not comparable but ‘addable’ because teams have shared understanding of work size and complexity.
- To the extent teams estimate work together, BOW, forecasting and planning can be done for multiple teams, by using Story Points (units of work) estimation.
- Then, beyond the point of teams working together, BOW/forecasting/planning can be done, based on whole features
- Much higher chance that Product Portfolio is **REAL**

Reporting Examples

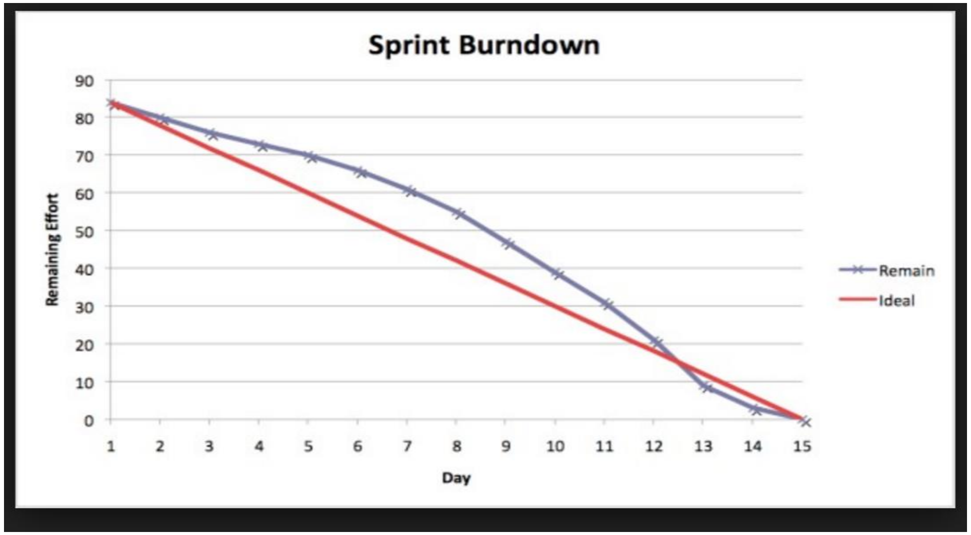




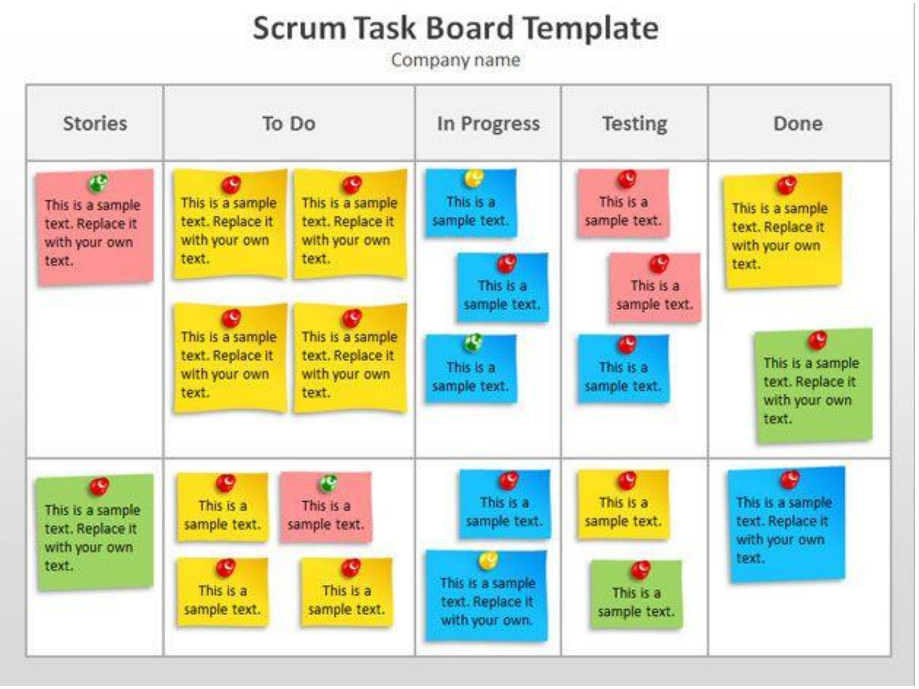
Start Date: 03/Jul/14
Scheduled Release Date: 15/Sep/14

41% unestimated issues 183 of 235 completed (story points)

RAGs - how reliable are they?



2nd Degree of accuracy



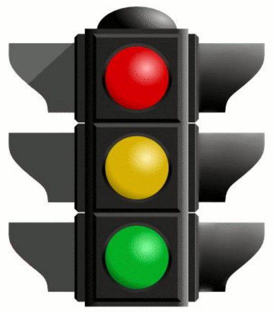
Scrum Task Board Template

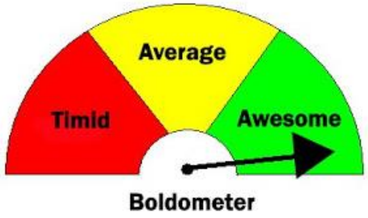
Company name

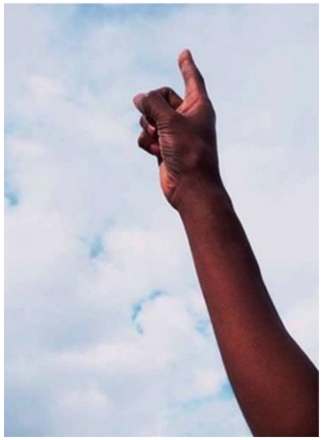
Stories	To Do	In Progress	Testing	Done
This is a sample text. Replace it with your own text.	This is a sample text. Replace it with your own text. This is a sample text. Replace it with your own text. This is a sample text. Replace it with your own text. This is a sample text. Replace it with your own text.	This is a sample text. This is a sample text. This is a sample text.	This is a sample text. This is a sample text. This is a sample text.	This is a sample text. Replace it with your own text. This is a sample text. Replace it with your own text.
This is a sample text. Replace it with your own text.	This is a sample text. This is a sample text. This is a sample text. This is a sample text.	This is a sample text. This is a sample text. Replace it with your own.	This is a sample text. This is a sample text.	This is a sample text. Replace it with your own text.

1st Degree of accuracy

RAGs - how reliable are they?







RAGs are 3^{rd} Degree of Accuracy

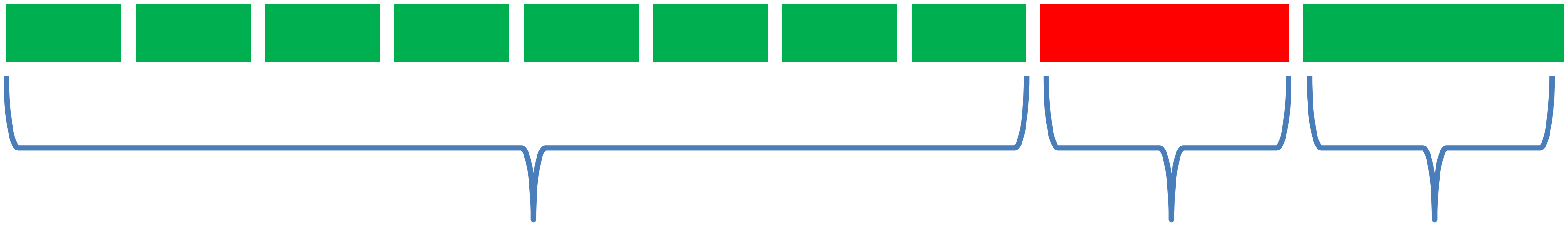
- Red, Yellow, Green or RYG/RAG Reports: How They Hide the Truth
- The Fallacy of Red, Amber, Green Reporting
- Agile Metrics

RAGs - how reliable are they?

We are on-track!!!

We are dead under water!!!

We are back on track again!!!

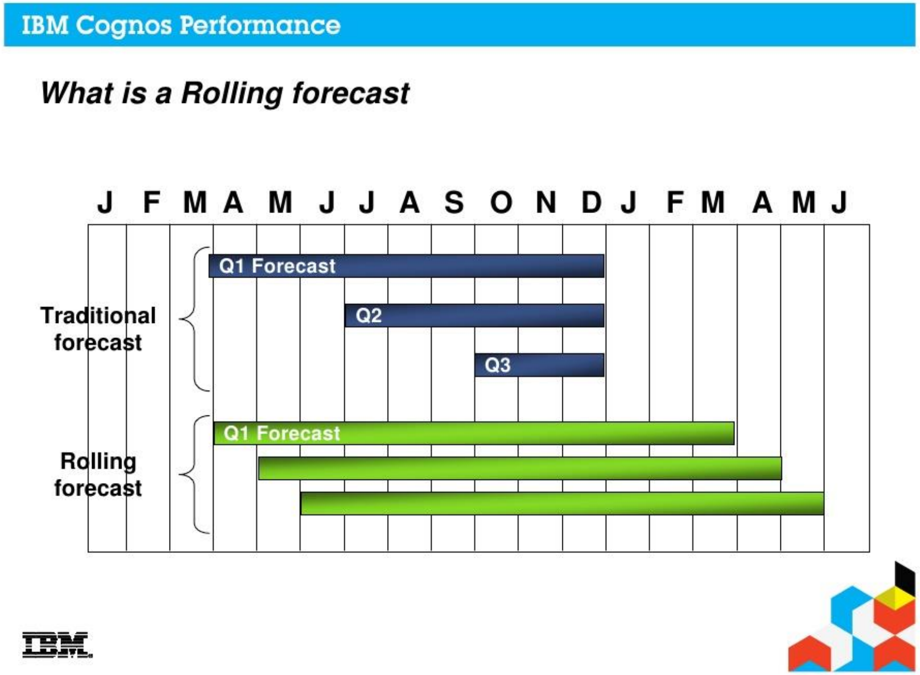


-Long status meetings (time waste for many)
-Weekly RAG reporting (mostly green)
-”Effective” Risk, Resource, Cost management
-Multi-tiered communication b/w IT and Biz

-“Oh My God”
-”You promised”
-”No we did not”

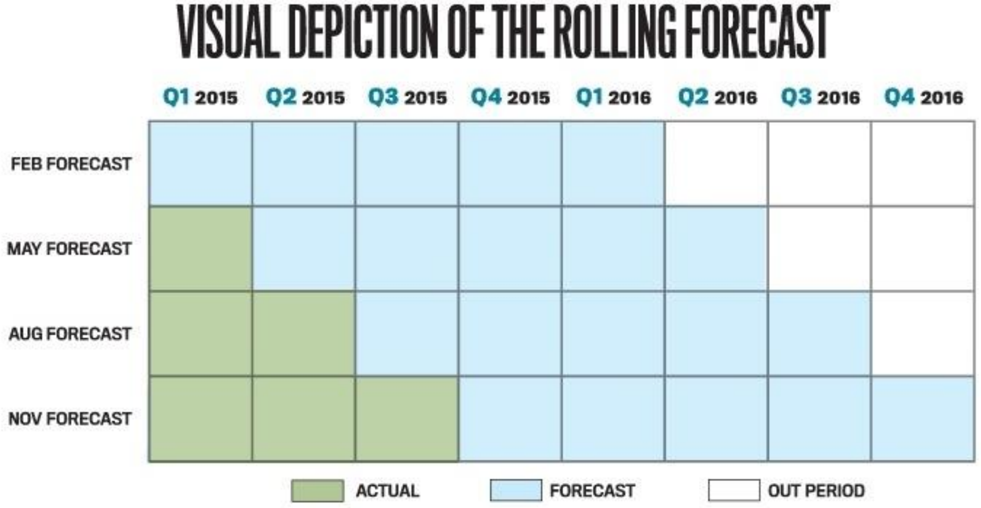
-Surprise!!!
-Miracle!!!
-Scope-cutting
-Secret “Toolbox”
-Compromising Quality

Rolling Forecast -Examples



IBM Cognos Performance

What is a Rolling forecast



VISUAL DEPICTION OF THE ROLLING FORECAST

	Q1 2015	Q2 2015	Q3 2015	Q4 2015	Q1 2016	Q2 2016	Q3 2016	Q4 2016
FEB FORECAST	🟦	🟦	🟦	🟦	🟦	□	□	□
MAY FORECAST	🟩	🟦	🟦	🟦	🟦	🟦	□	□
AUG FORECAST	🟩	🟩	🟦	🟦	🟦	🟦	🟦	□
NOV FORECAST	🟩	🟩	🟩	🟦	🟦	🟦	🟦	🟦

🟩 ACTUAL 🟦 FORECAST □ OUT PERIOD

About Coaches and Coaching

About Coaches and Coaching

At Team(s) Level, Coaches Are Expected To Be
Supporting single or multiple teams in improving their dynamics and maturity
Coaching individual team members, Scrum Masters, Product Owners
Assisting to establish agile roles, ceremonies, day-to-day interactions
Focusing on engineering practices, coding standards, test quality
Advising teams on agile requirements, living documentation, metrics, communication
Advising teams with adoption of basic agile frameworks (e.g. Kanban, Scrum, XP)
Challenging inappropriate locally manifested (in isolation) behavioral patterns

At Enterprise/Organizational Level, Coaches Are Expected To:
Seed agility across an entire organization, trying to influence/educate senior leadership and executives
Assess team(s) and organization(s) for effectiveness of applying agile principles and practices
Develop leadership and organizational agility through guided self-discovery and growth
Challenge organizational and leadership status quo and enabling Kaizen culture
Analyze systemic patterns, including norms, standards and behaviors
Help Organization link deep systemic (“back-end”) impediments to peripheral (“front-end”) symptoms
Educate senior leadership on inter-connection of various organizational elements within one Organizational Ecosystem

Gene Gendel, Certified Enterprise & Team Coach (CEC & CTC) | LeSS Friendly Scrum Trainer (LSFT) | Certified in Agile Leadership (CAL)

Source: https://www.infoq.com/articles/agile-coaching-lessons

Coaching “Bad Smells” and Why They Should Be Avoided

Bad Smell	Why Does it Smell Badly?
Continuously resolving clients’/coachees' problems for them. Engaging as "doers" for too long and continuously giving complete solutions. Exhibiting Command & Control behavior.	Initial 'leading by example' is OK. Part of teaching comes through initial training. However, any prolonged engagement as a 'doer' puts a client/coachee into a comfort zone and prevents learning, independence and autonomy. Commanding and excessive directive telling also implies that a coach has an authoritative position with a client. This makes a client less independent in its decisions.
Assuming authoritative, long-lasting position with client/coachee that does not translate timely into tangible results. Establishing dependence.	A coach ends up generating financial benefits, whereas a client gains very little value
Publicly criticize/scrutinize individuals of any level, especially in presence of their superiors	This behavior creates hostility and mistrust between a coach and individual clients/coachees. If a client/coachee feels comfortable to be openly coached in front of others, without becoming defensive, they should explicitly invite a coach to do so, before it is done in public. This is a key sign of coaching immaturity.
Using team-based metrics to judge individual team members	Excessive use of metrics is a simplistic false dichotomy. While certain health checks/indicators can be used as a way of reflecting to individual teams, using the same metrics to judge individuals is counter-productive and misleading.
Using team-based metrics to compare teams to each other, establishing competition between teams	A prime example of local optimization, establishing cross-team competition will not optimize the performance of the entire organization and will instead encourage information hoarding and closed communication patterns.
Getting involved in activities and feedback that influence individual performance appraisals, incentives, compensation, bonuses, promotions	Being able to explicitly influence compensation/financial well-being of an individual is a violation of an individual's safety space. A coachee's desire to become autonomous and independent in making his/her own decisions will be significantly diminished. Individuals will feel "obligated" to follow recommendations of a coach. Recommendations will be perceived as instructions/mandates

Source: https://www.infoq.com/articles/agile-coaching-lessons

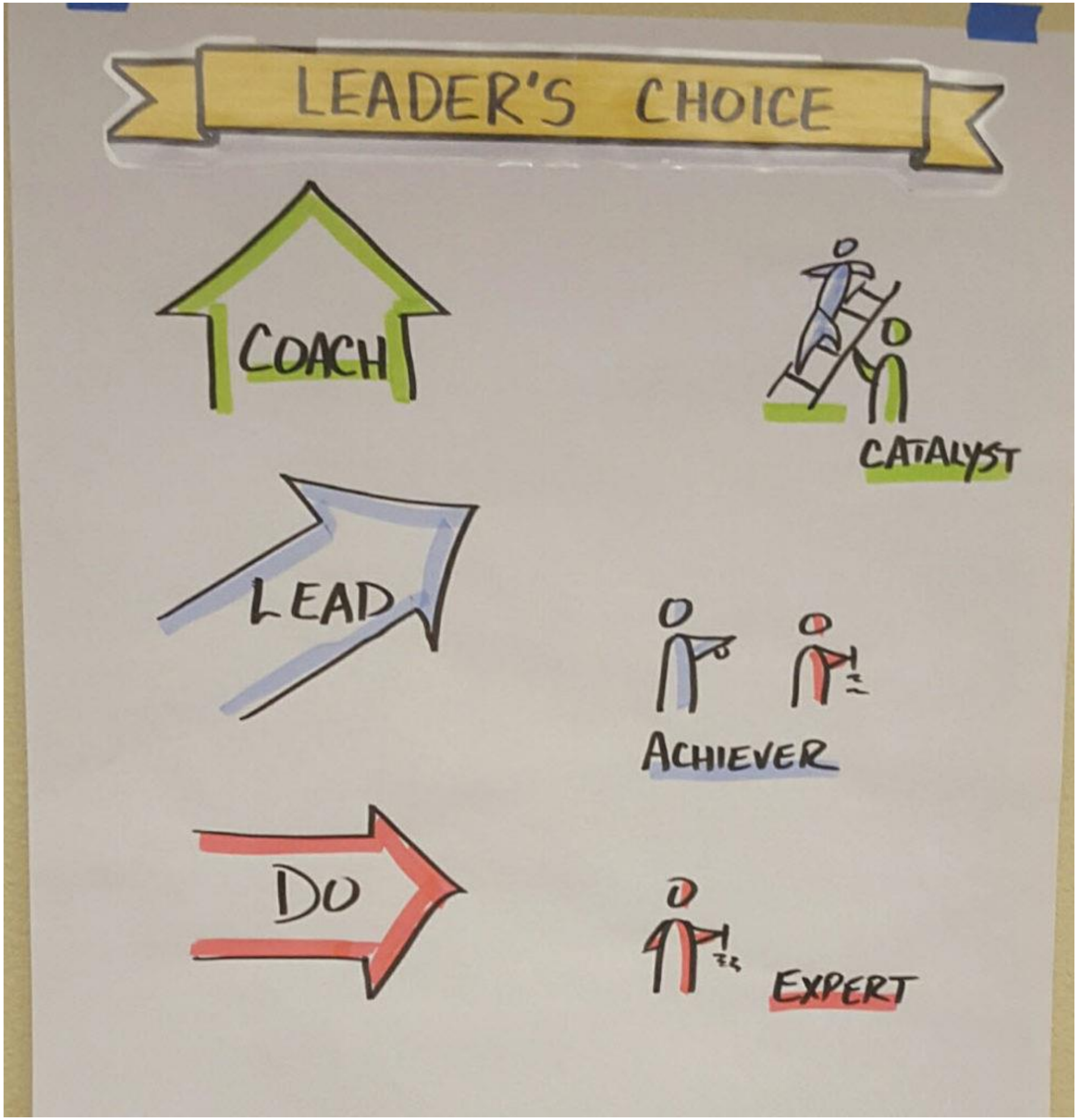
Coaching “Bad Smells” and Why They Should Be Avoided

Bad Smell	Why Does it Smell Badly?
Quantify/numerically estimate ('status'/'checkbox') things that cannot be quantified. Applying a numerical scale (metrics) to human/cultural element of agile transformation	Similar to the above: there is a huge human factor that is responsible for successful agile implementation. A human factor is not quantifiable and cannot be easily plotted on a scale. Checks and balances produce a false sense of completion (or lack of such). This diminishes opportunity for continuous improvement. Examples may include “Using Scrum” or “Converted to Agile” checkboxes for teams.
Tracking individual recommendations given to clients/coachees and assessing if recommendations are followed, reporting to senior management when they are not.	This is an indication of command & control, micro-management and lack of trust. Policing individuals and enforcing things to be done, contradicts agile principles and prevents Kaizen adoption. This also erodes relationship between a coach and client/coachee. Also, if a ratio of coaches to coachees is low (many coachees for a single coach), scaling coaching efforts becomes challenging. Ideally, in cases like these, a "pull" system must be used, instead of "push": coachees should pursue with Coach, proactively asking to provide feedback to their improvements, instead of Coach chasing them
Withhold views and observations about pivotal organizational dysfunctions from organizational leaders to avoid personal risks and repercussions	By virtue of being a change agent and organizational transformer, a coach is expected to speak openly about organizational dysfunctions and impediments that most of employees are not comfortable discussing. This is paramount for a coaching job. The job of Coach is sometimes risky. This is why, in majority of cases, a coaching role is consultative in nature (external to an organization), as Coach has to bring to light organizational dysfunctions that may put a coach in the position of scrutiny and/or criticism - something that consultants care less about than full time employees.

Source: https://www.infoq.com/articles/agile-coaching-lessons

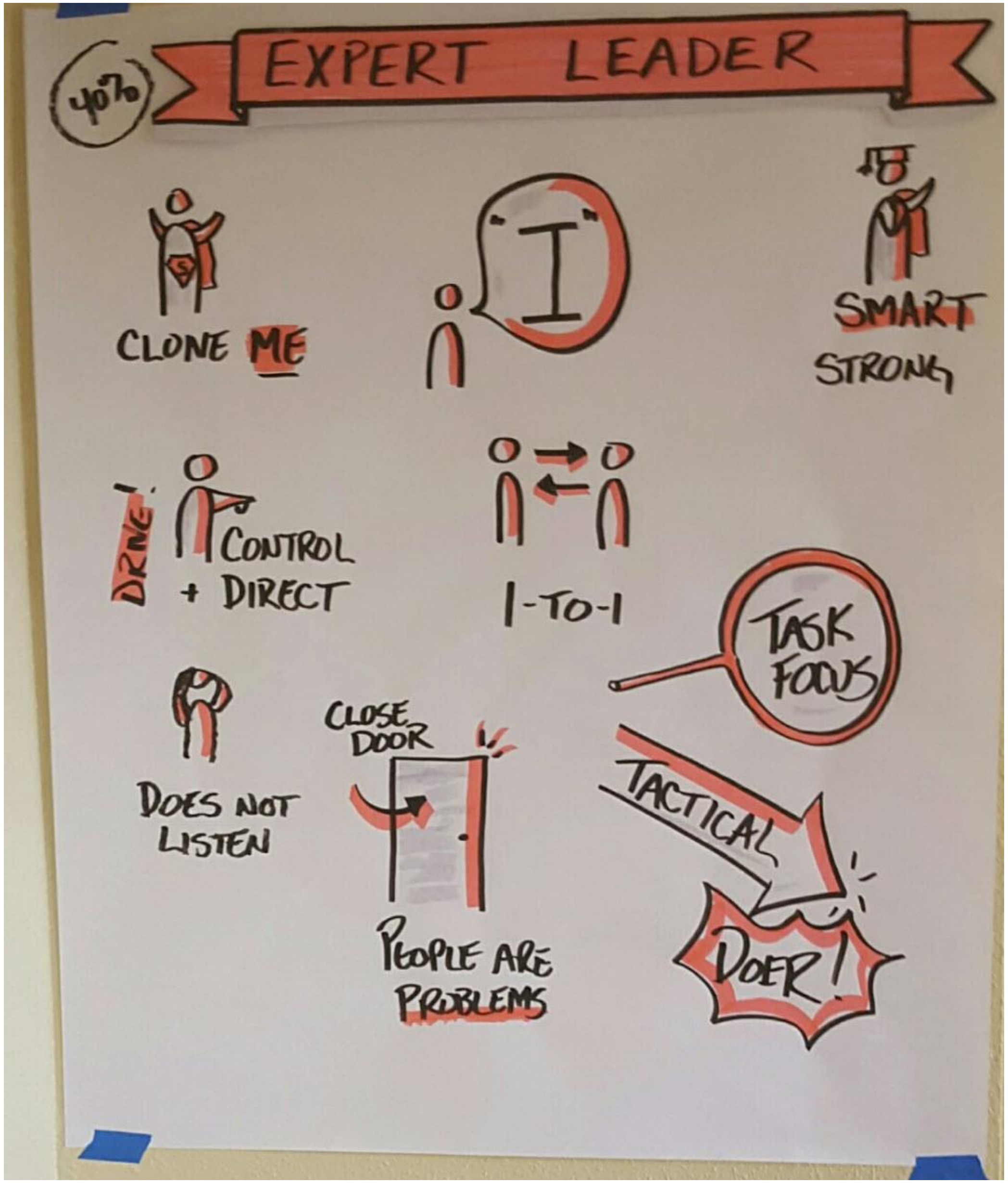
Agile Leadership

Are you ready to lead your Organization towards better Agility?

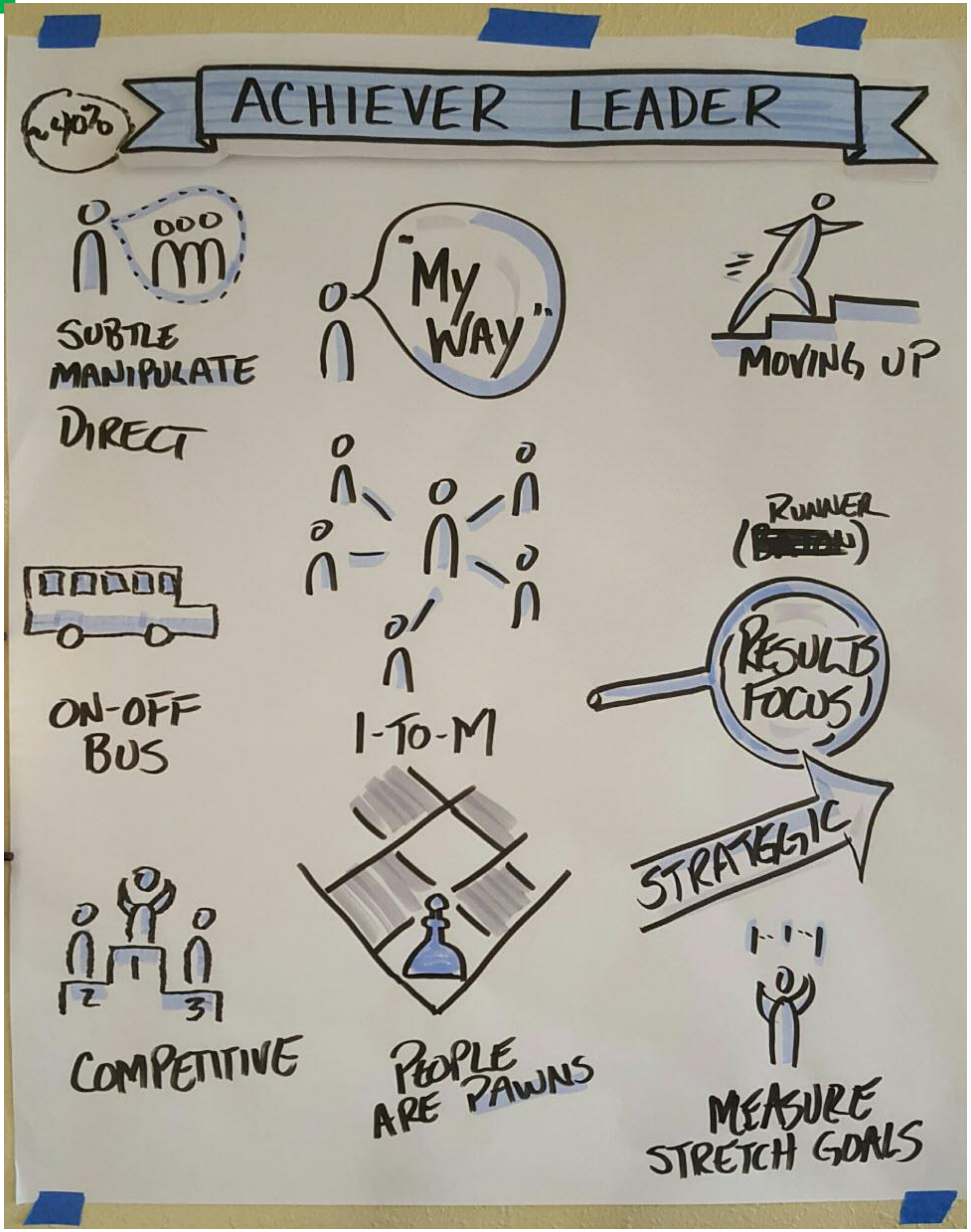


How do you want to lead???

Some of these may not be applicable at certain organizational levels

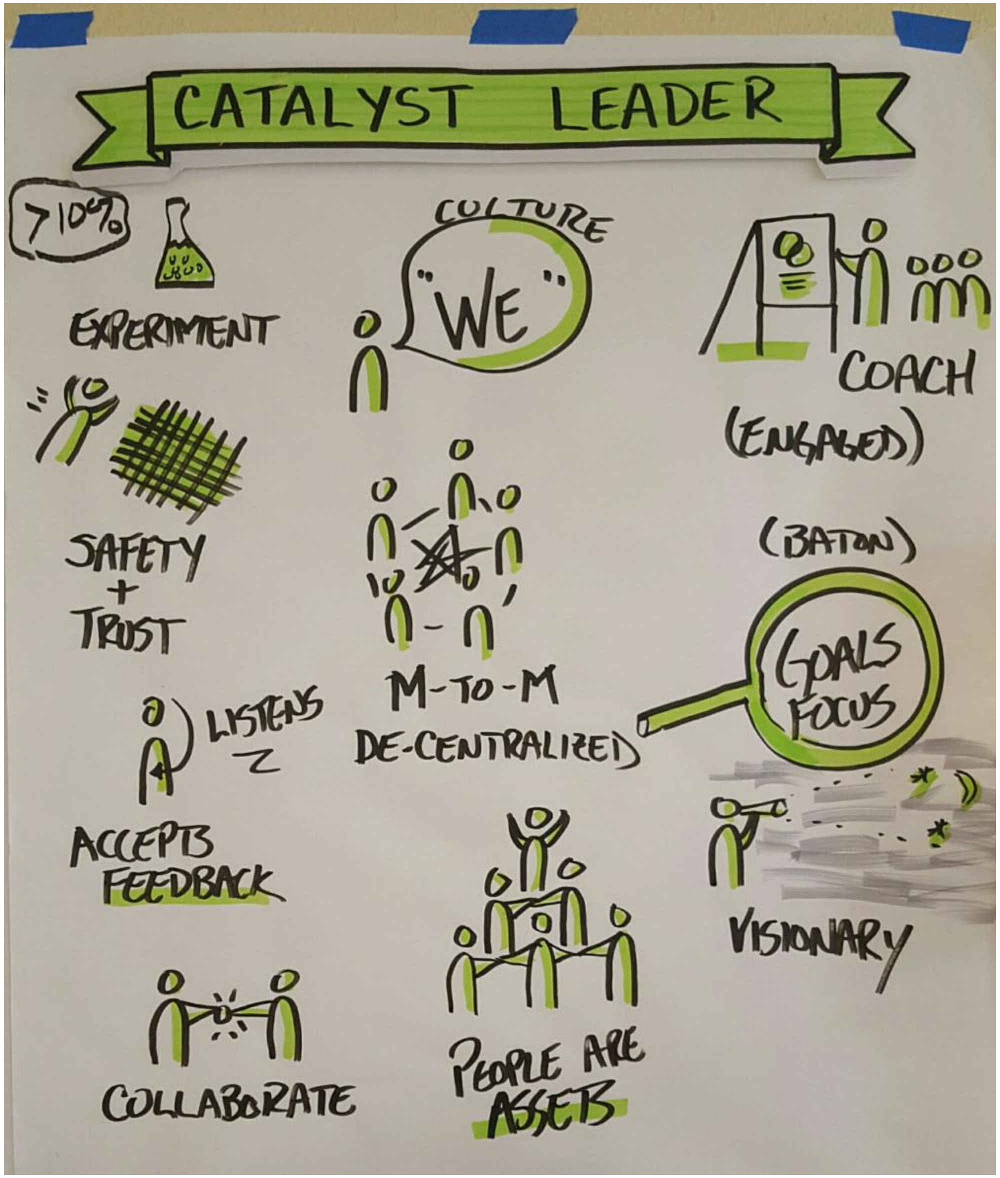


Stage 3 ("primitive") of Tribal Development (David Logan's model) : "**I am the great and you are not**"



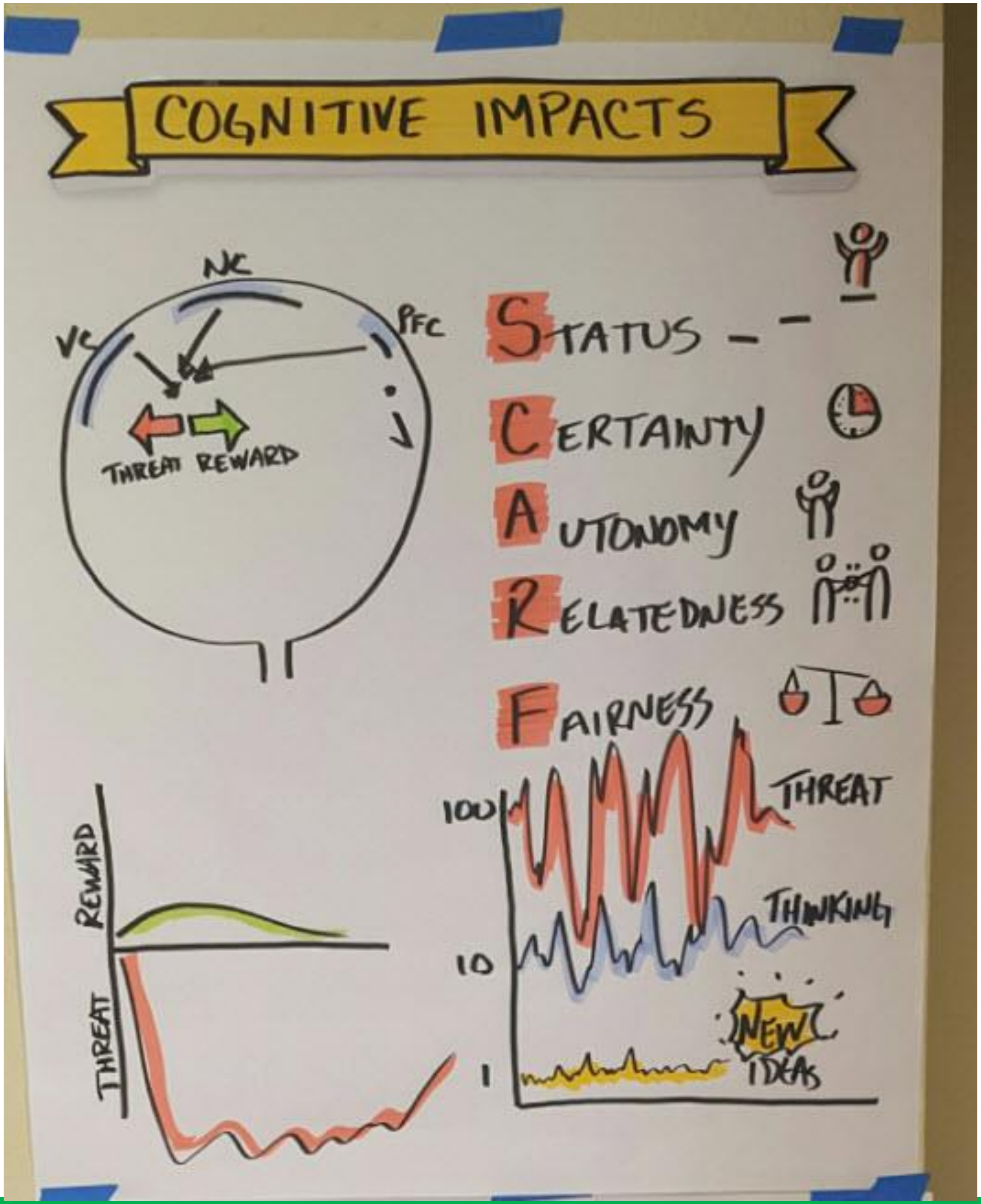
Stage 3 ("advanced") of Tribal Development (David Logan's model) : "**I am the great and you are not**" (but at your expense)

Mostly seen in **Orange Companies** (Frederic Laloux)

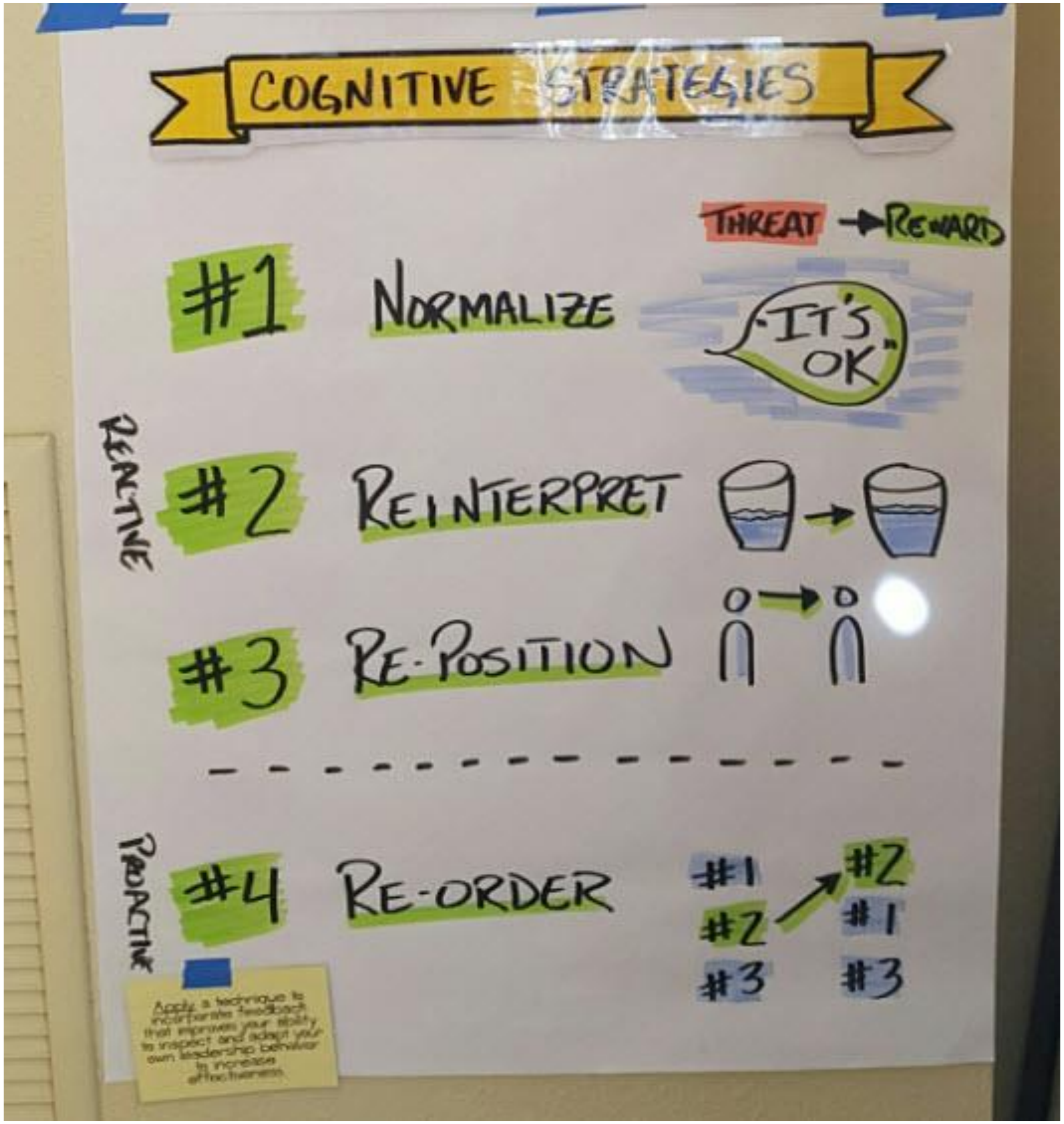


Stage 4 of Tribal Development (David Logan's model) : "**We are great and they are not**"

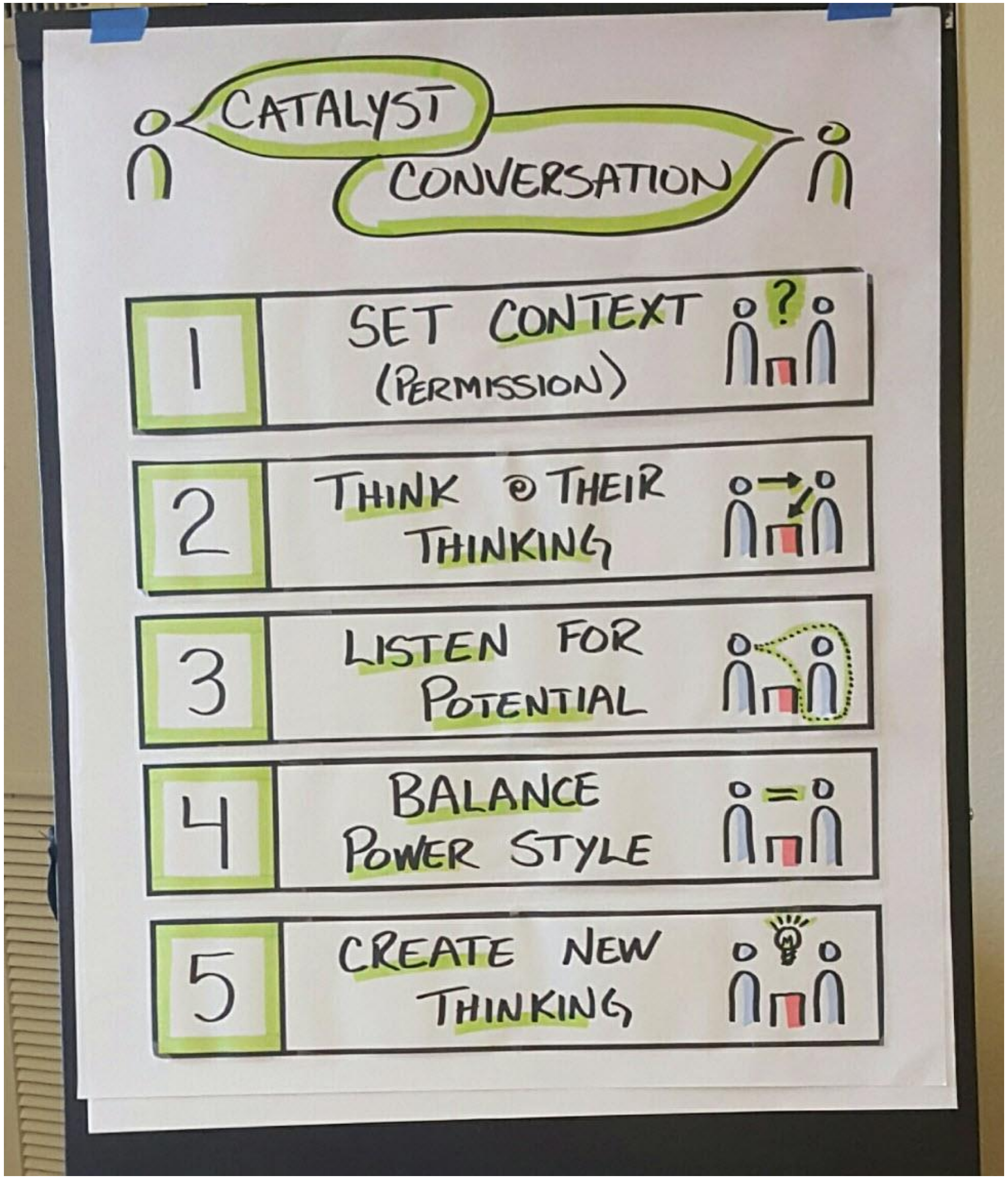
Mostly seen in **Green Companies** (Frederic Laloux)


COGNITIVE IMPACTS
NC
VC
PFC
THREAT
REWARD
STATUS
CERTAINTY
AUTONOMY
RELATEDNESS
FAIRNESS
REWARD
THREAT
100
10
1
THREAT
THINKING
NEW IDEAS

...skip page...

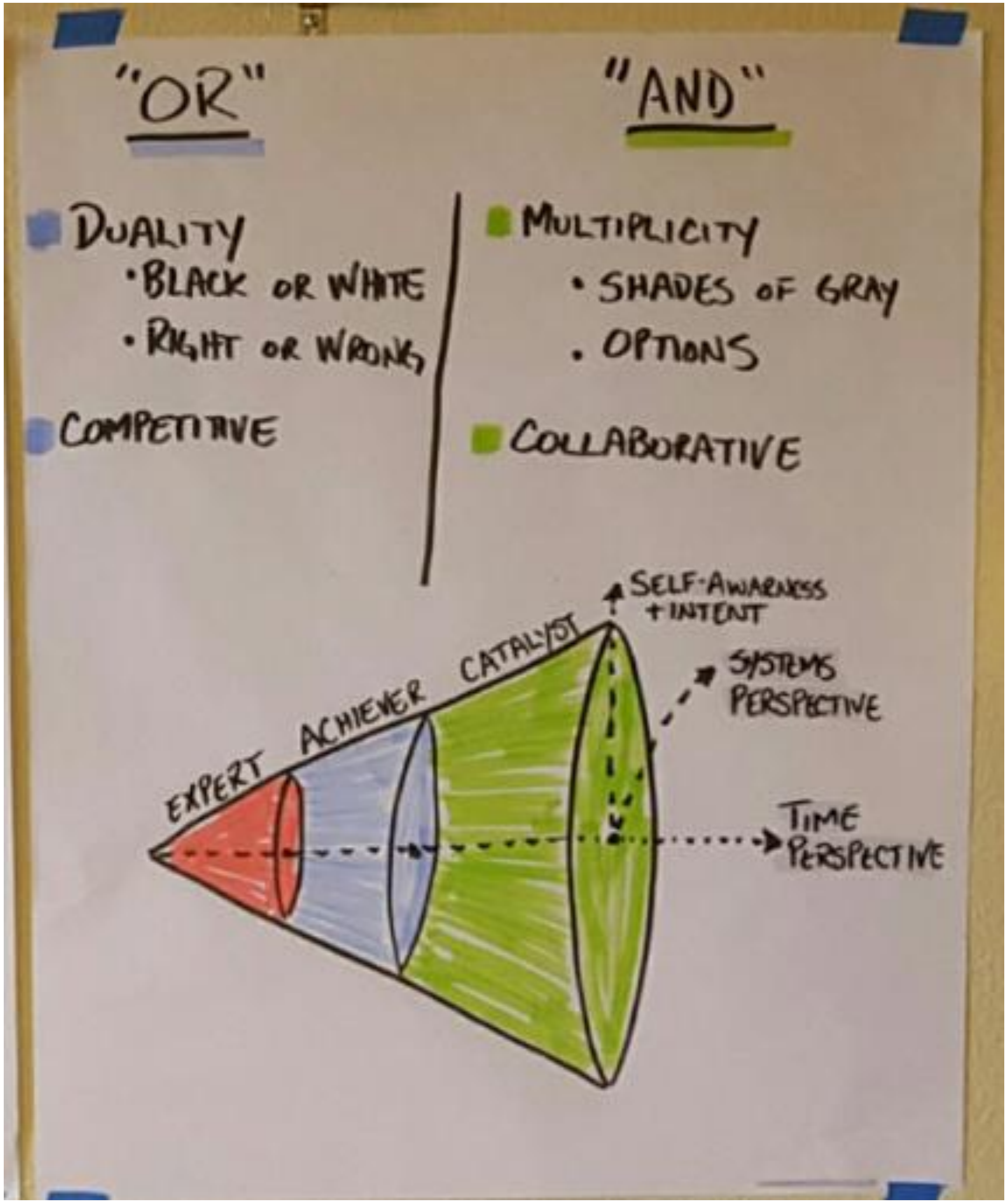

COGNITIVE STRATEGIES
THREAT → REWARD
#1 NORMALIZE
"IT'S OK"
REACTIVE
#2 REINTERPRET
#3 RE-POSITION
PROACTIVE
#4 RE-ORDER
#1 #2 #3
#2 #1 #3
Apply a technique to incorporate feedback that improves your ability to inspect and adapt your own leadership behavior to increase effectiveness

...skip page...



How a great **Catalyst Leader** should talk

Standard Coaching Approach



"OR"

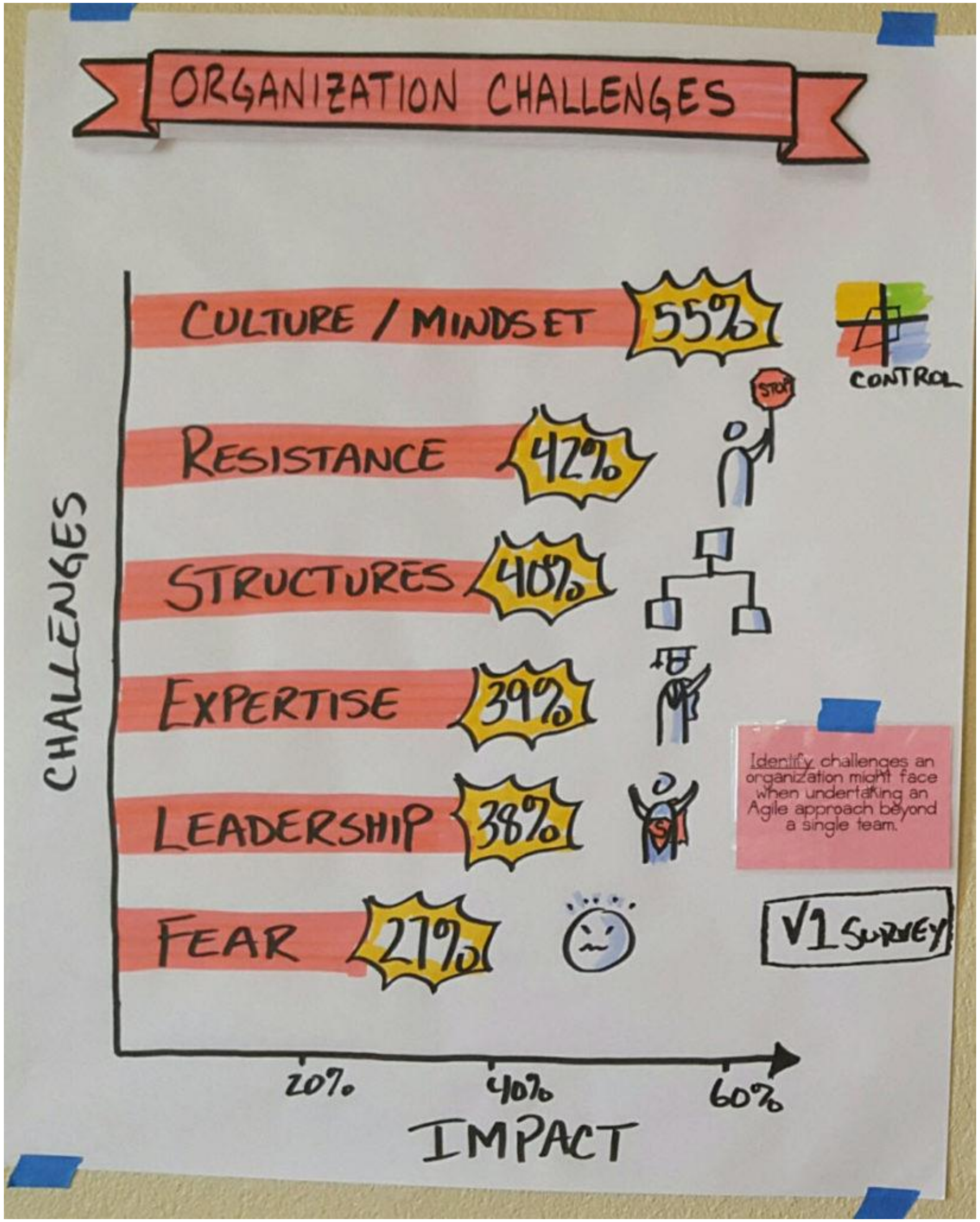
- 🟦 DUALITY
  - BLACK OR WHITE
  - RIGHT OR WRONG
- 🟦 COMPETITIVE

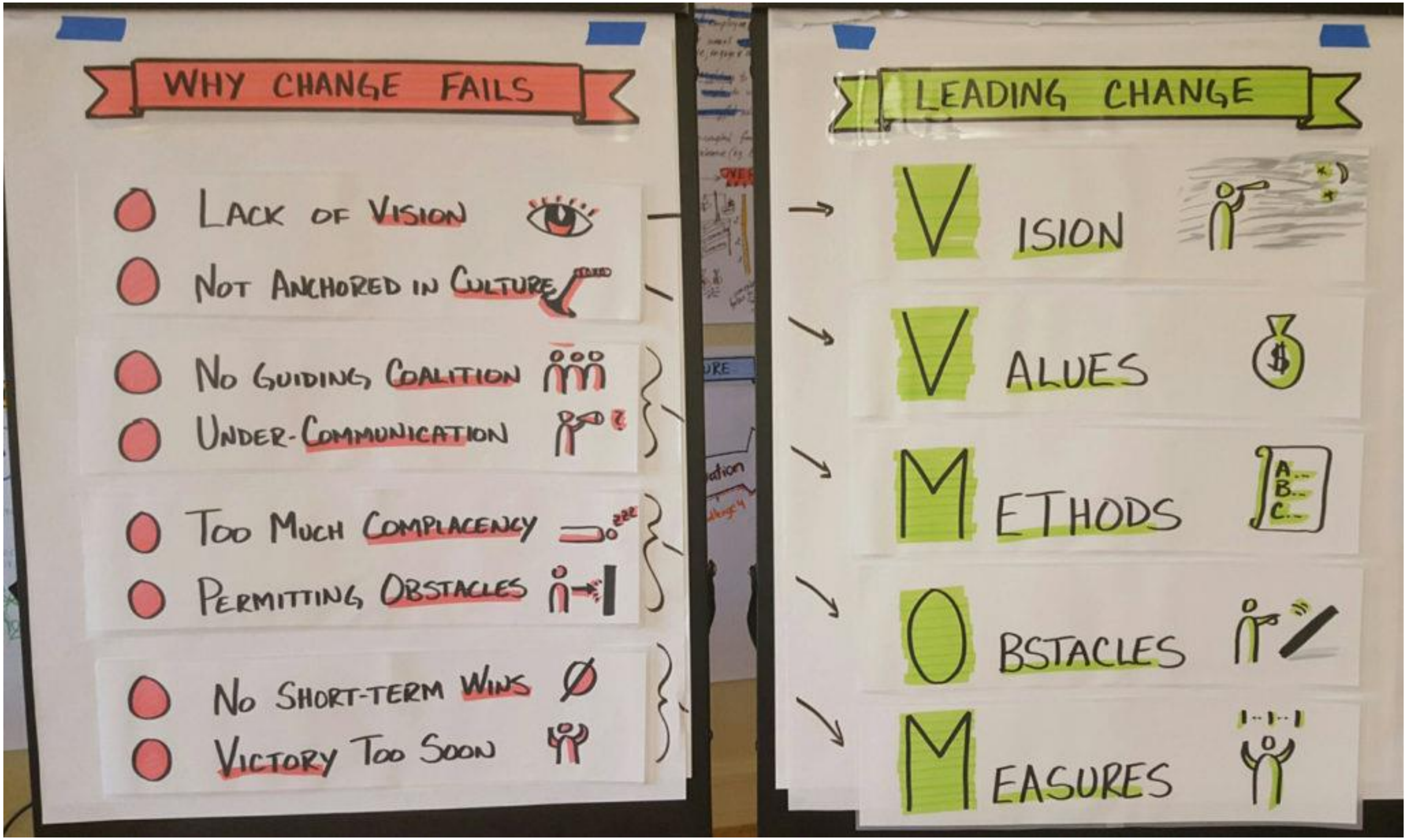
"AND"

- 🟩 MULTIPLICITY
  - SHADES OF GRAY
  - OPTIONS
- 🟩 COLLABORATIVE

Achievers
&
Catalysts
prefer
different
“SQL clause”

Industry Statistics





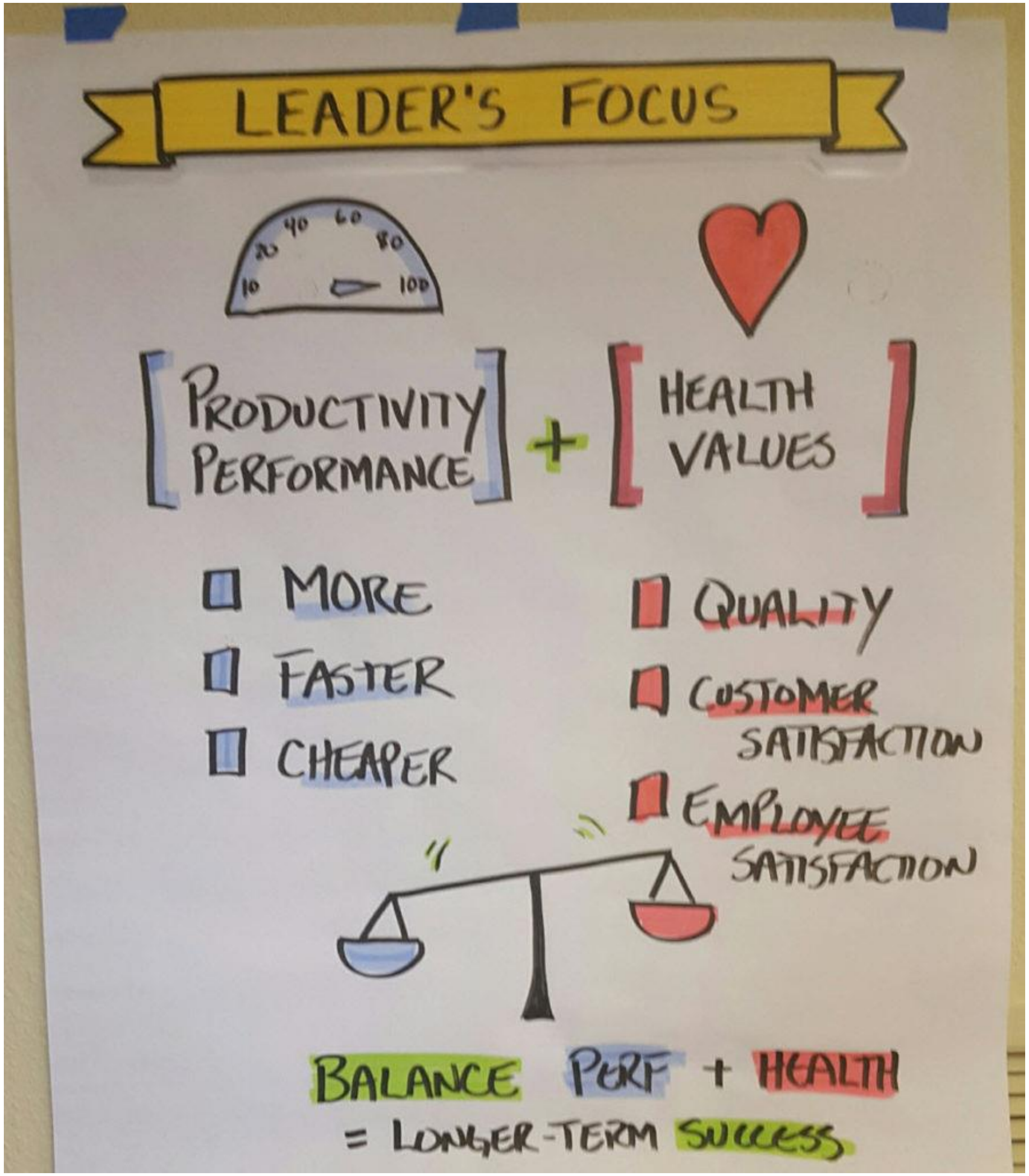
WHY CHANGE FAILS

- Lack of Vision
- Not Anchored in Culture
- No Guiding Coalition
- Under-Communication
- Too Much Complacency
- Permitting Obstacles
- No Short-term Wins
- Victory Too Soon

LEADING CHANGE

- V ISION
- V ALUES
- M ETHODS
- O BSTACLES
- M EASURES

Striking a Happy Balance



LEADER'S FOCUS

[PRODUCTIVITY PERFORMANCE] + [HEALTH VALUES]

□ MORE
□ FASTER
□ CHEAPER

□ QUALITY
□ CUSTOMER SATISFACTION
□ EMPLOYEE SATISFACTION

BALANCE PERF + HEALTH
= LONGER-TERM SUCCESS

Getting Language Straight

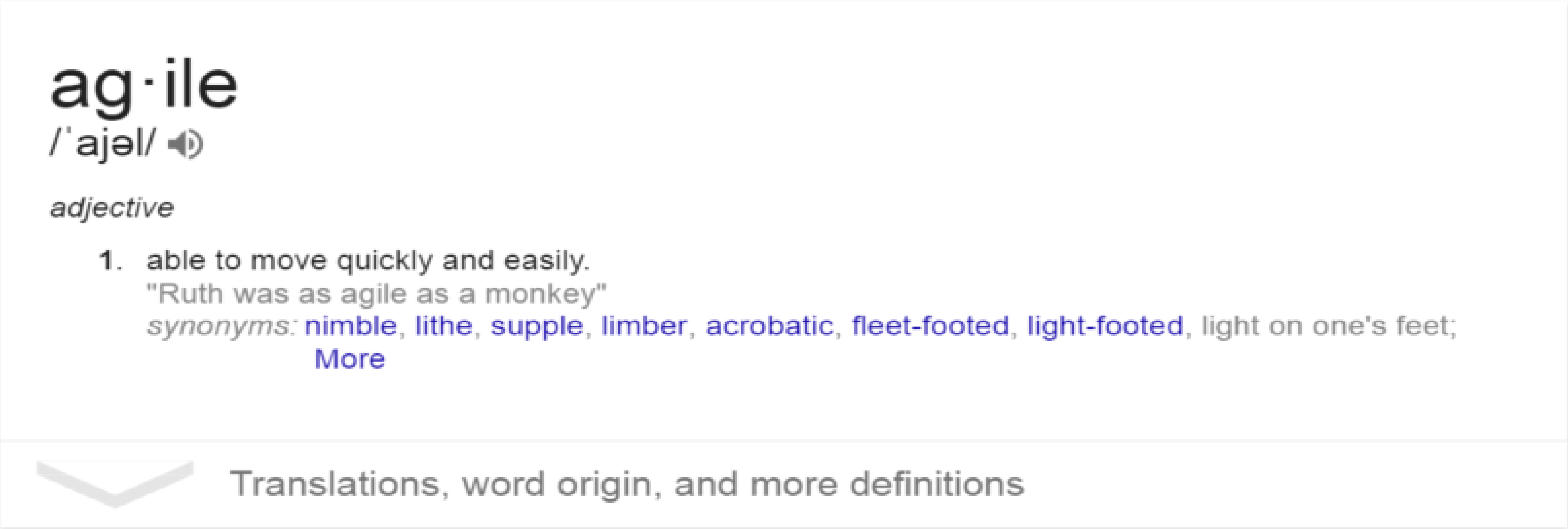
Agile

VERB

You cannot "DO" Agile?

ADJECTIVE

You can "Be" Agile?



ag·ile

/ˈajəl/

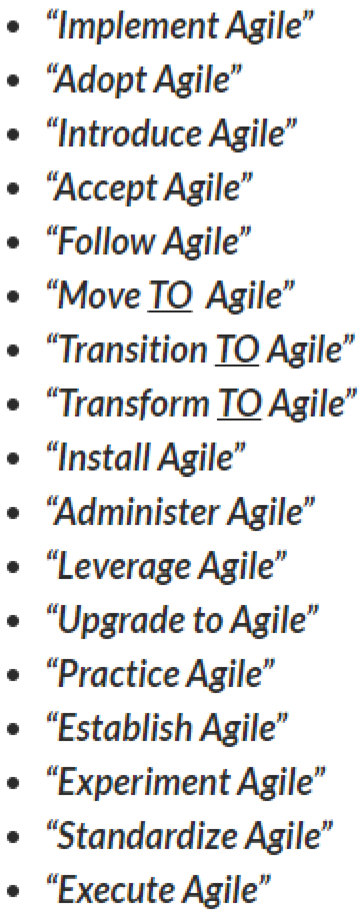
adjective

1. able to move quickly and easily.
   "Ruth was as agile as a monkey"
   synonyms: nimble, lithe, supple, limber, acrobatic, fleet-footed, light-footed, light on one's feet;
   More

Translations, word origin, and more definitions

Getting Language Straight

Don’t try to be fancy with verbs:



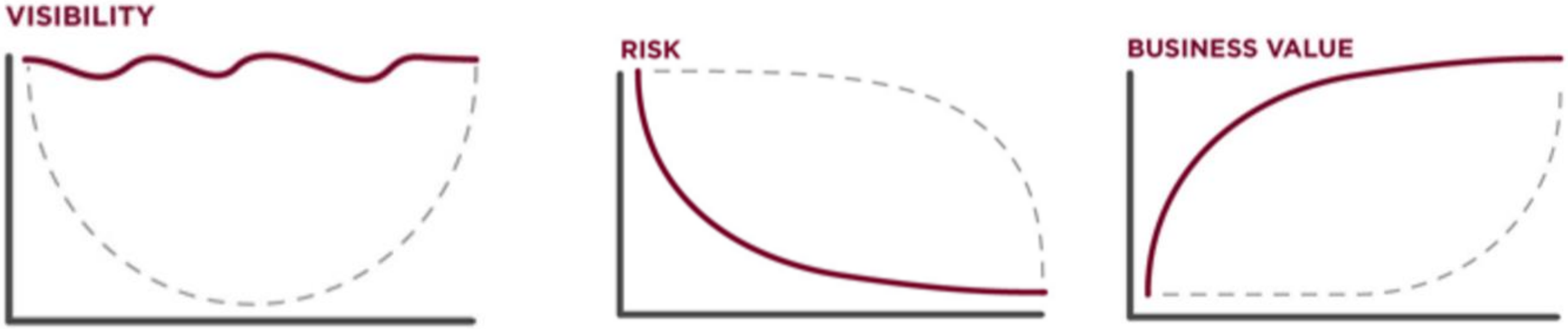
- “Implement Agile”
- “Adopt Agile”
- “Introduce Agile”
- “Accept Agile”
- “Follow Agile”
- “Move TO Agile”
- “Transition TO Agile”
- “Transform TO Agile”
- “Install Agile”
- “Administer Agile”
- “Leverage Agile”
- “Upgrade to Agile”
- “Practice Agile”
- “Establish Agile”
- “Experiment Agile”
- “Standardize Agile”
- “Execute Agile”

… PLEASE…

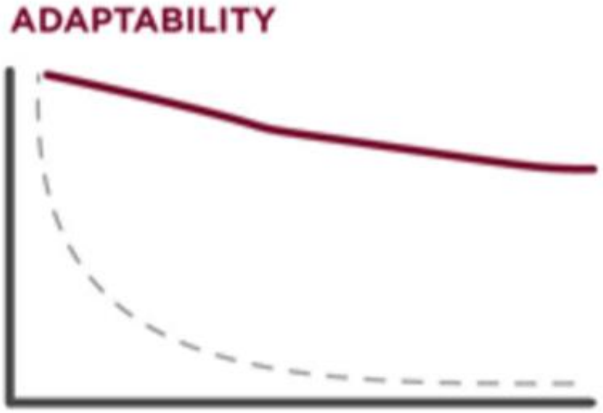
…just stick to BE/BEING agile…

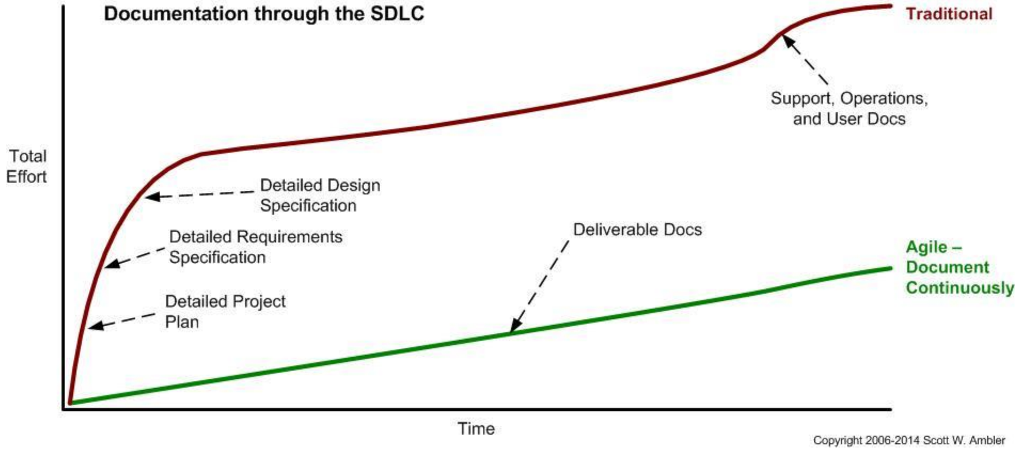
Source: Bad Choice of Verbs Associated with “Agile”, by EFL People

Waterfall vs. Agile



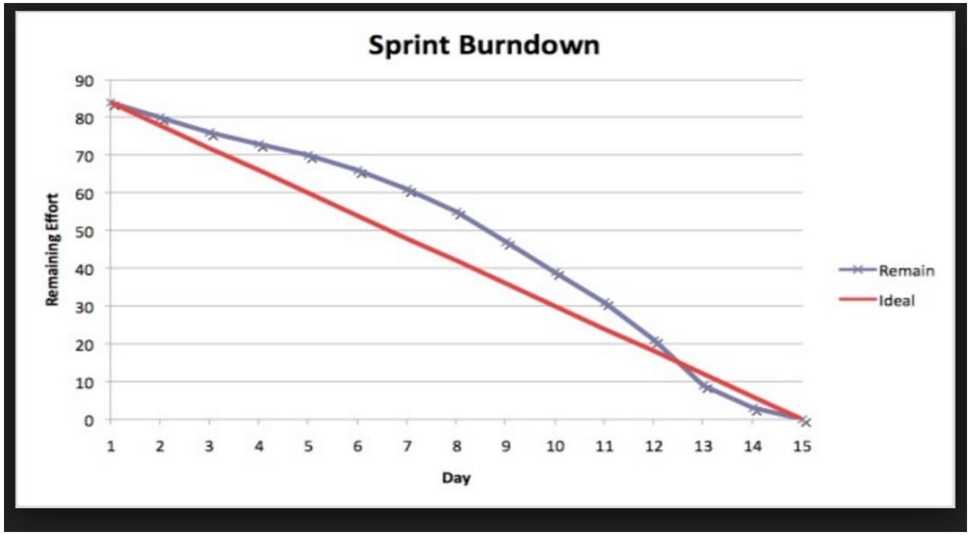
Source: http://image.slidesharecdn.com/ralphjocham-120612034405-phpapp02/95/agile-techniques-as-business-acceleration-tools-20-728.jpg?cb=1339472718



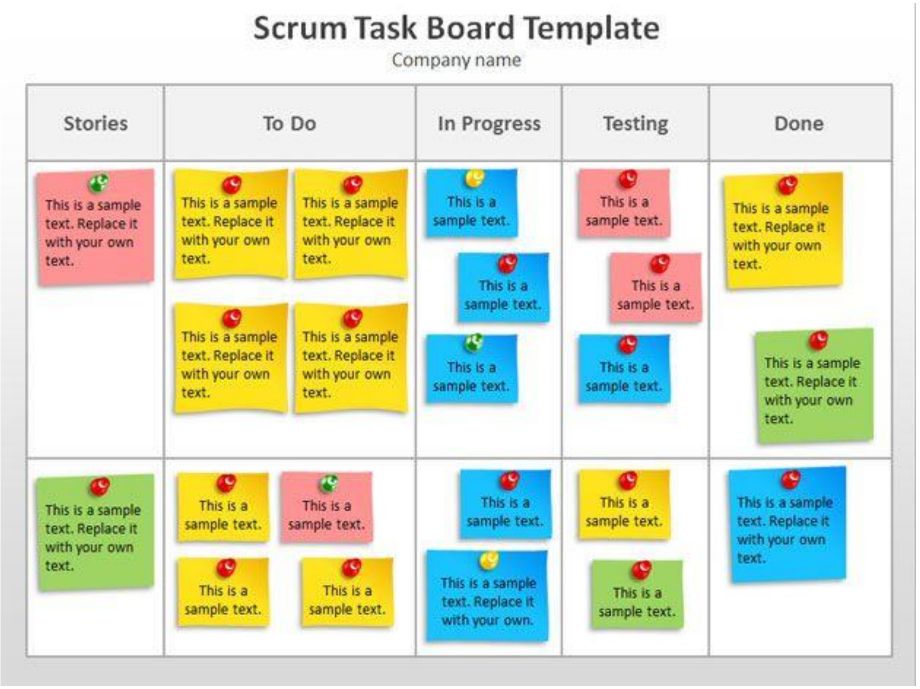


Source: http://www.agilemodeling.com/essays/agileDocumentation.htm

RAGs - how reliable are they?



2nd Degree of accuracy



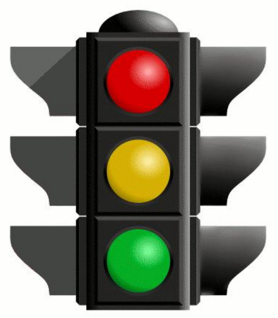
Scrum Task Board Template

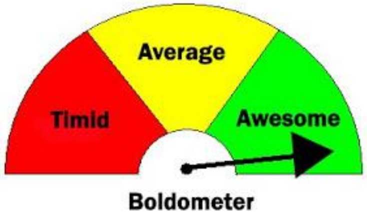
Company name

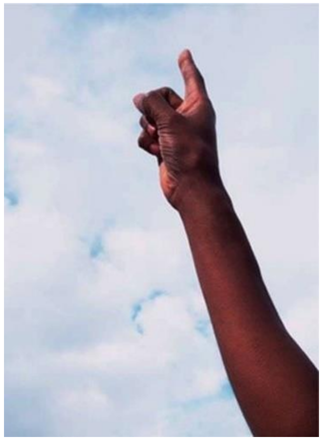
Stories	To Do	In Progress	Testing	Done
This is a sample text. Replace it with your own text.	This is a sample text. Replace it with your own text. This is a sample text. Replace it with your own text. This is a sample text. Replace it with your own text. This is a sample text. Replace it with your own text.	This is a sample text. This is a sample text. This is a sample text.	This is a sample text. This is a sample text. This is a sample text.	This is a sample text. Replace it with your own text. This is a sample text. Replace it with your own text.
This is a sample text. Replace it with your own text.	This is a sample text. This is a sample text. This is a sample text. This is a sample text.	This is a sample text. This is a sample text. Replace it with your own.	This is a sample text. This is a sample text.	This is a sample text. Replace it with your own text.

1st Degree of accuracy

RAGs - how reliable are they?







RAGs are 3^{rd} Degree of Accuracy

- Red, Yellow, Green or RYG/RAG Reports: How They Hide the Truth
- The Fallacy of Red, Amber, Green Reporting
- Agile Metrics

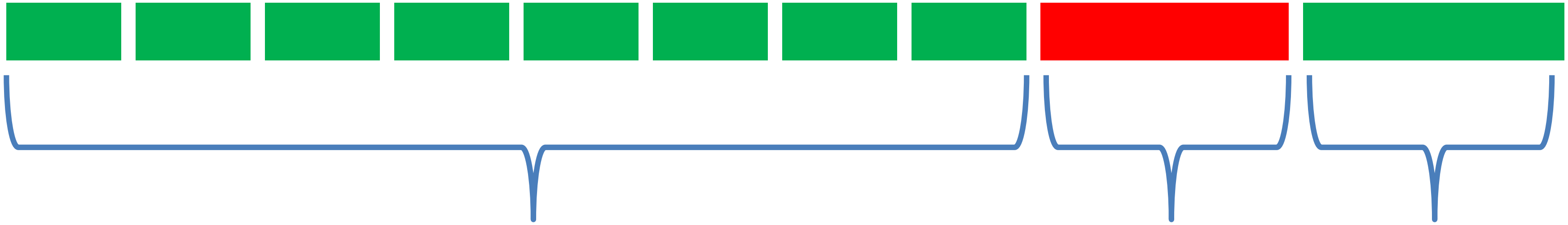
RAGs - how reliable are they?

…Separate Topic…

We are on-track!!!

We are dead under water!!!

We are back on track again!!!

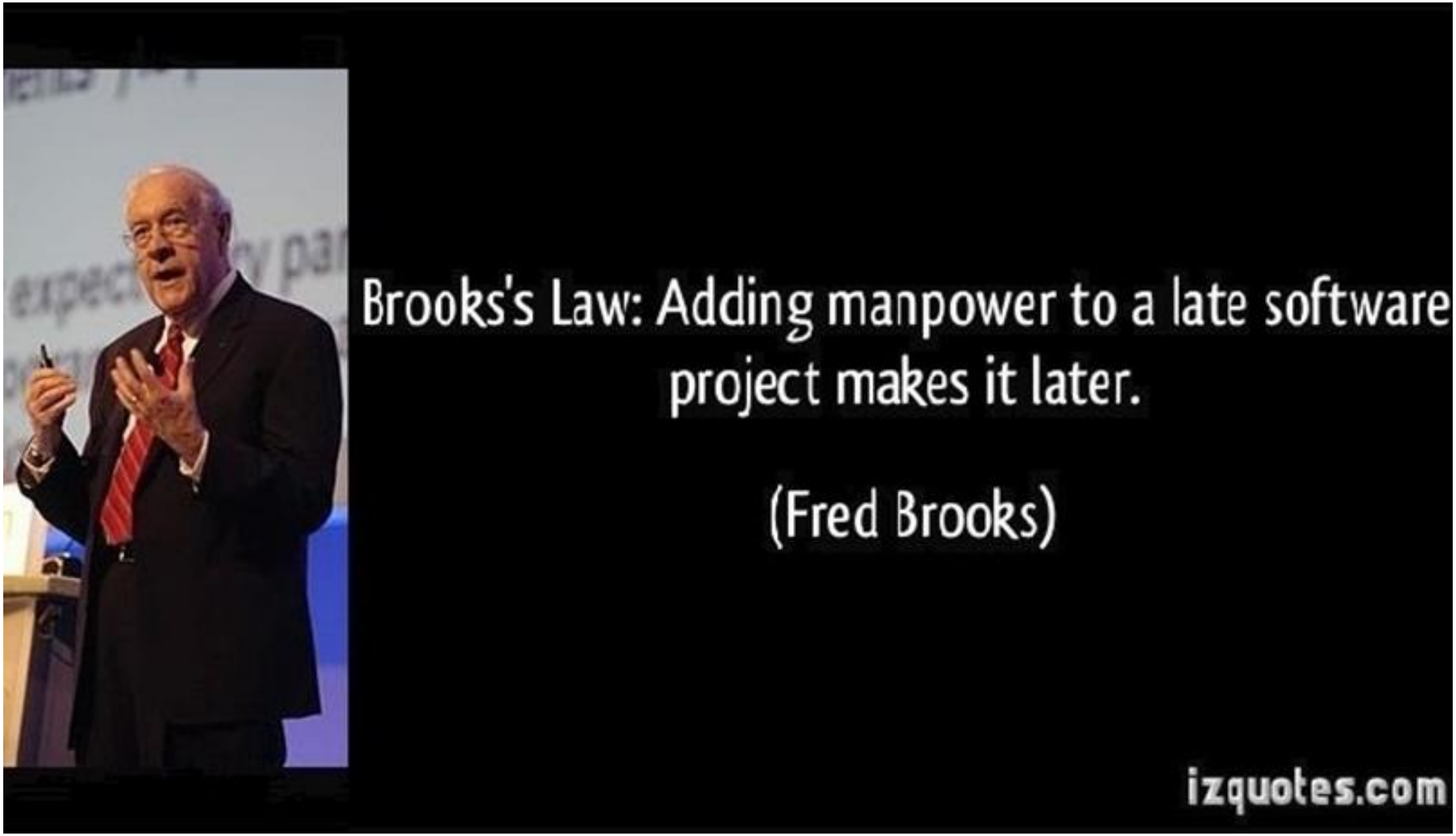


-Long status meetings (time waste for many)
-Weekly RAG reporting (mostly green)
-"Effective" Risk, Resource, Cost management
-Multi-tiered communication b/w IT and Biz

-"Oh My God"
-"You promised"
-"No we did not"

-Surprise!!!
-Miracle!!!
-Scope-cutting
-Secret "Toolbox"
-Compromising Quality

Agile Overview



Brooks's Law: Adding manpower to a late software project makes it later.

(Fred Brooks)

izquotes.com

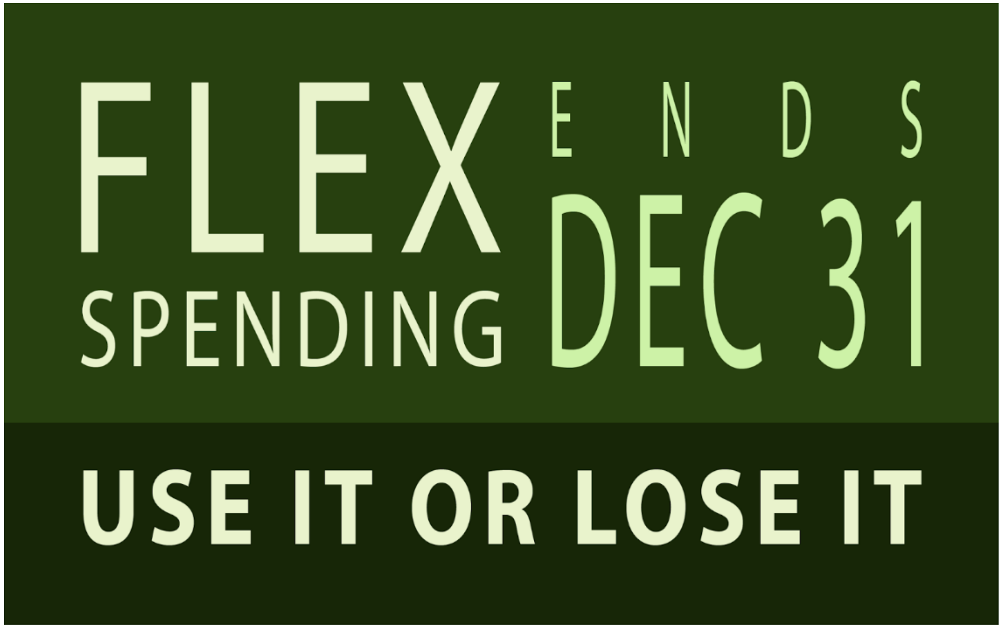
This Is All Tied To…

Budgeting/Finance - Separate Topic…

Budget Ceiling is also Floor (Ironically)

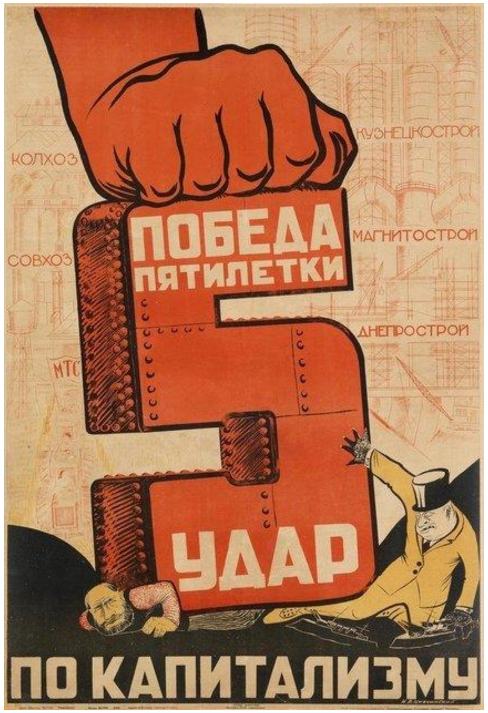
- *“Cost budgets are spent even when initial assumptions are lowered”*
- *“People do not cheat intentionally: system makes them to do so”*

Quotes from “Implementing Beyond Budgeting ” by Bjarte Bogsnes

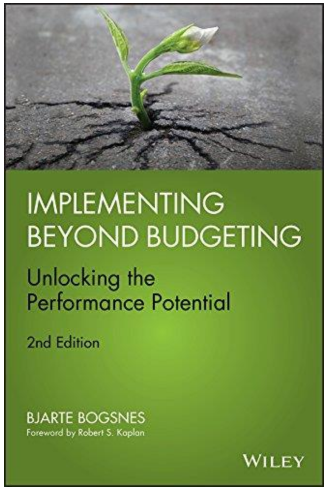

FLEX SPENDING ENDS DEC 31

USE IT OR LOSE IT

This Is All Tied To…



Budgeting/Finance - Separate Topic…



IMPLEMENTING BEYOND BUDGETING

Unlocking the Performance Potential

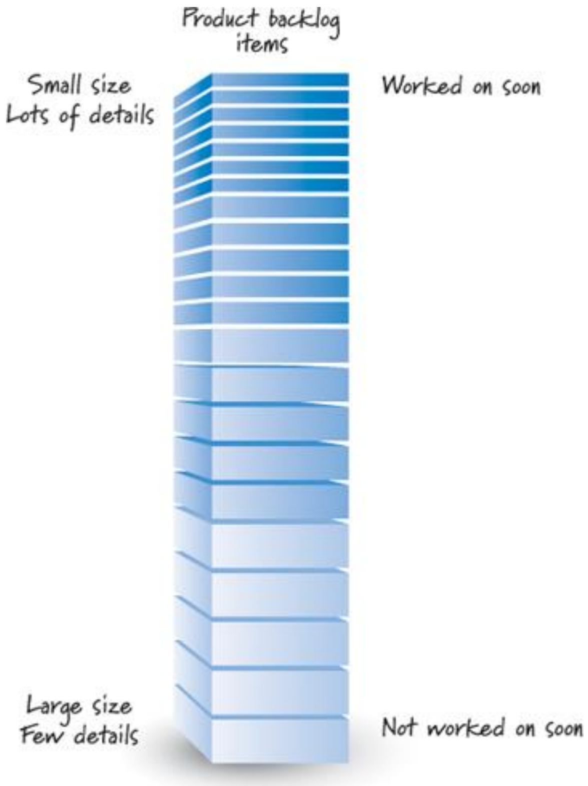
2nd Edition

BJARTE BOGSNES

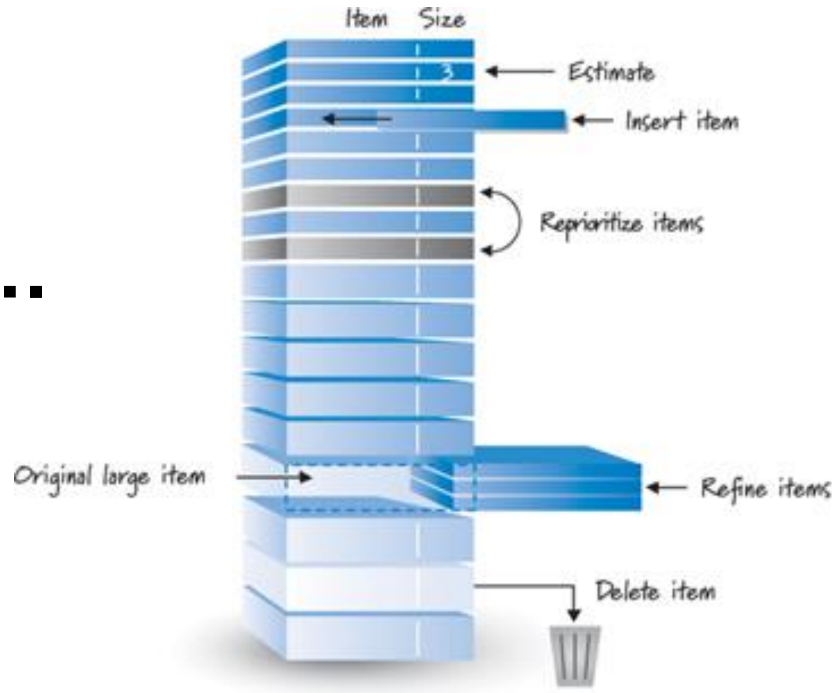
Foreword by Robert S. Kaplan

WILEY

Scrum



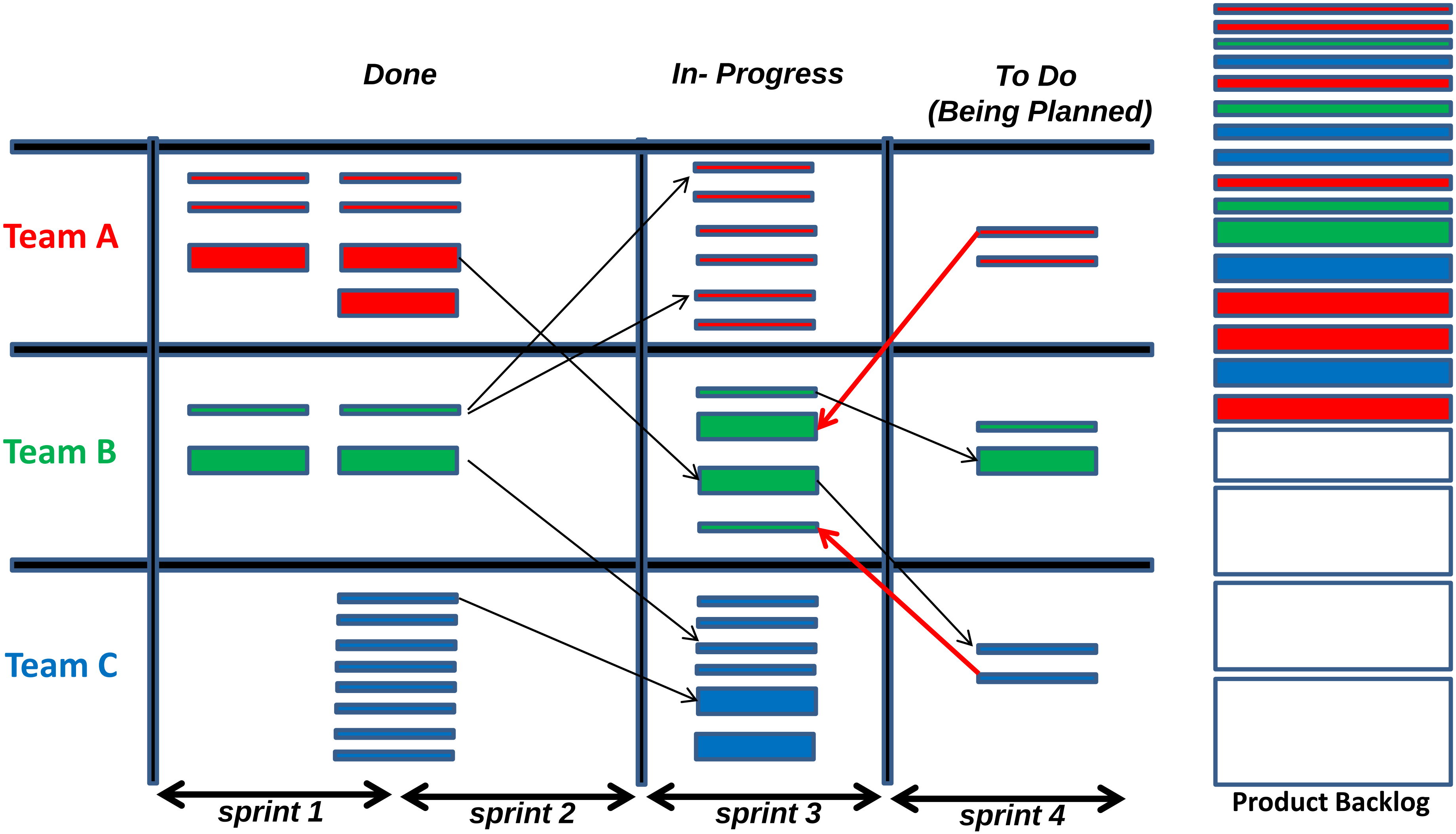
http://www.informit.com/content/images/chap6_9780137043293/elementLinks/06fig03.jpg



http://www.informit.com/content/images/chap6_9780137043293/elementLinks/06fig06.jpg

Scrum

Cross-Team Work Dependency

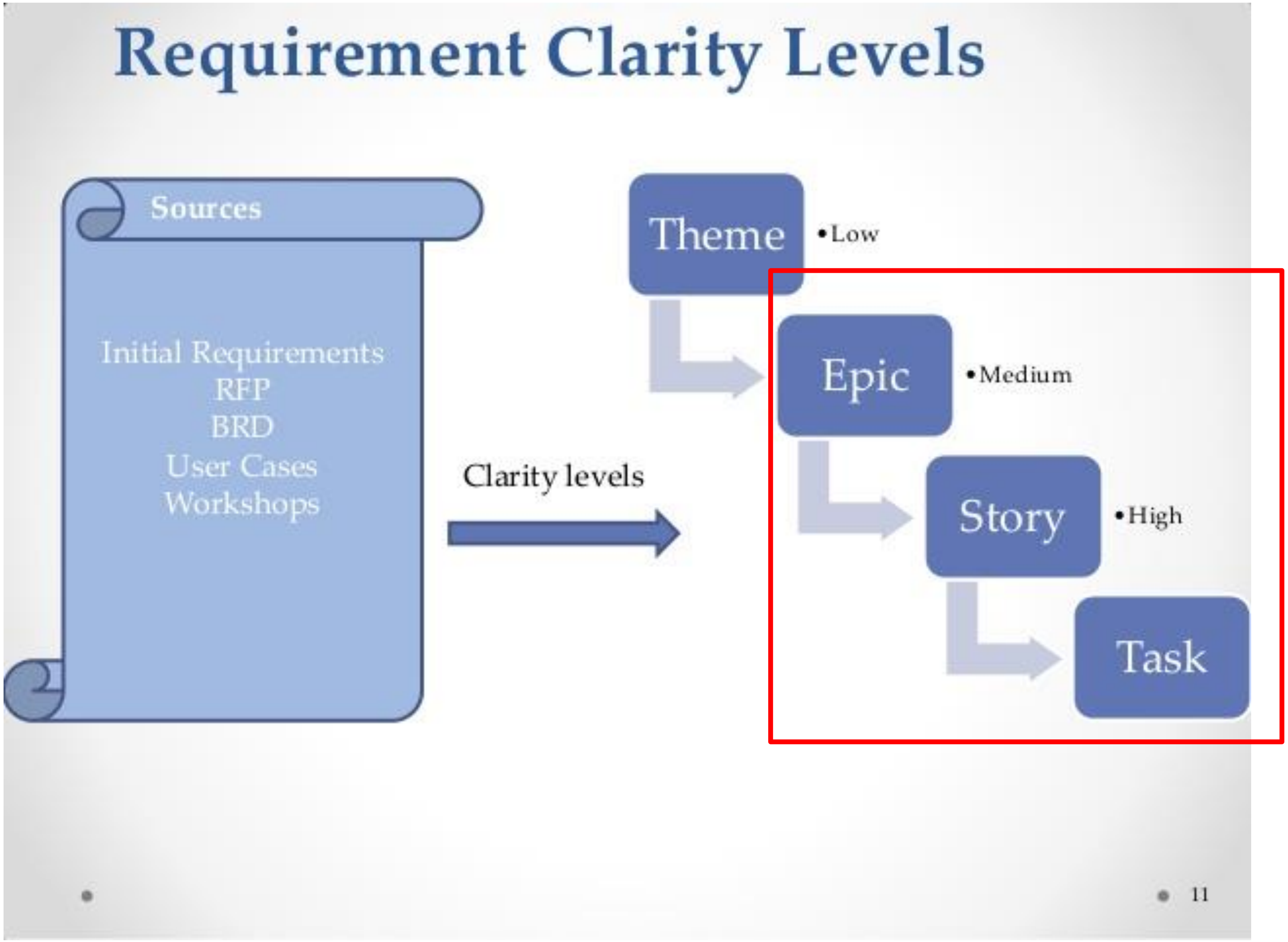


***Note**: Teams do not possess the same skillset and don't estimate work together*

Product Ownership Team

Source: https://www.army.mil/article/145515/usanato_1gnc_soldiers_allies_train_during_exercise_austere_challenge_2015

From Waterfall → To Agile Requirements



Requirement Clarity Levels

Sources

Initial Requirements
RFP
BRD
User Cases
Workshops

Clarity levels

Theme
- Low

Epic
- Medium

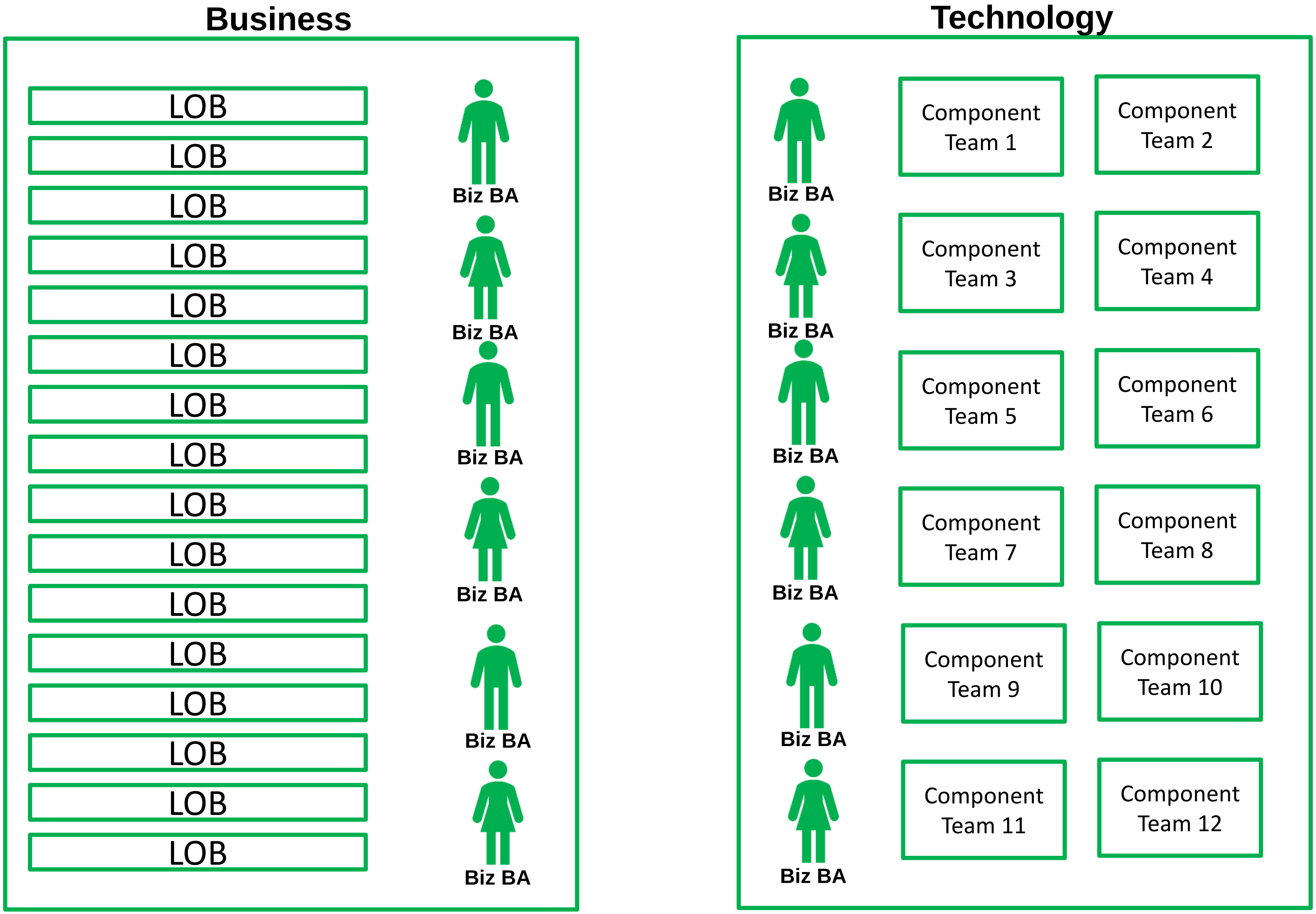
Story
- High

Task

11

Source: https://www.slideshare.net/jayanth72/agile-estimation-for-fixed-price-model

Conventional Organizational Design



Business

LOB

LOB

LOB

LOB

LOB

LOB

LOB

LOB

LOB

LOB

LOB

LOB

LOB

LOB

LOB

LOB

Biz BA

Biz BA

Biz BA

Biz BA

Biz BA

Biz BA

Technology

Biz BA

Biz BA

Biz BA

Biz BA

Biz BA

Biz BA

Component Team 1

Component Team 2

Component Team 3

Component Team 4

Component Team 5

Component Team 6

Component Team 7

Component Team 8

Component Team 9

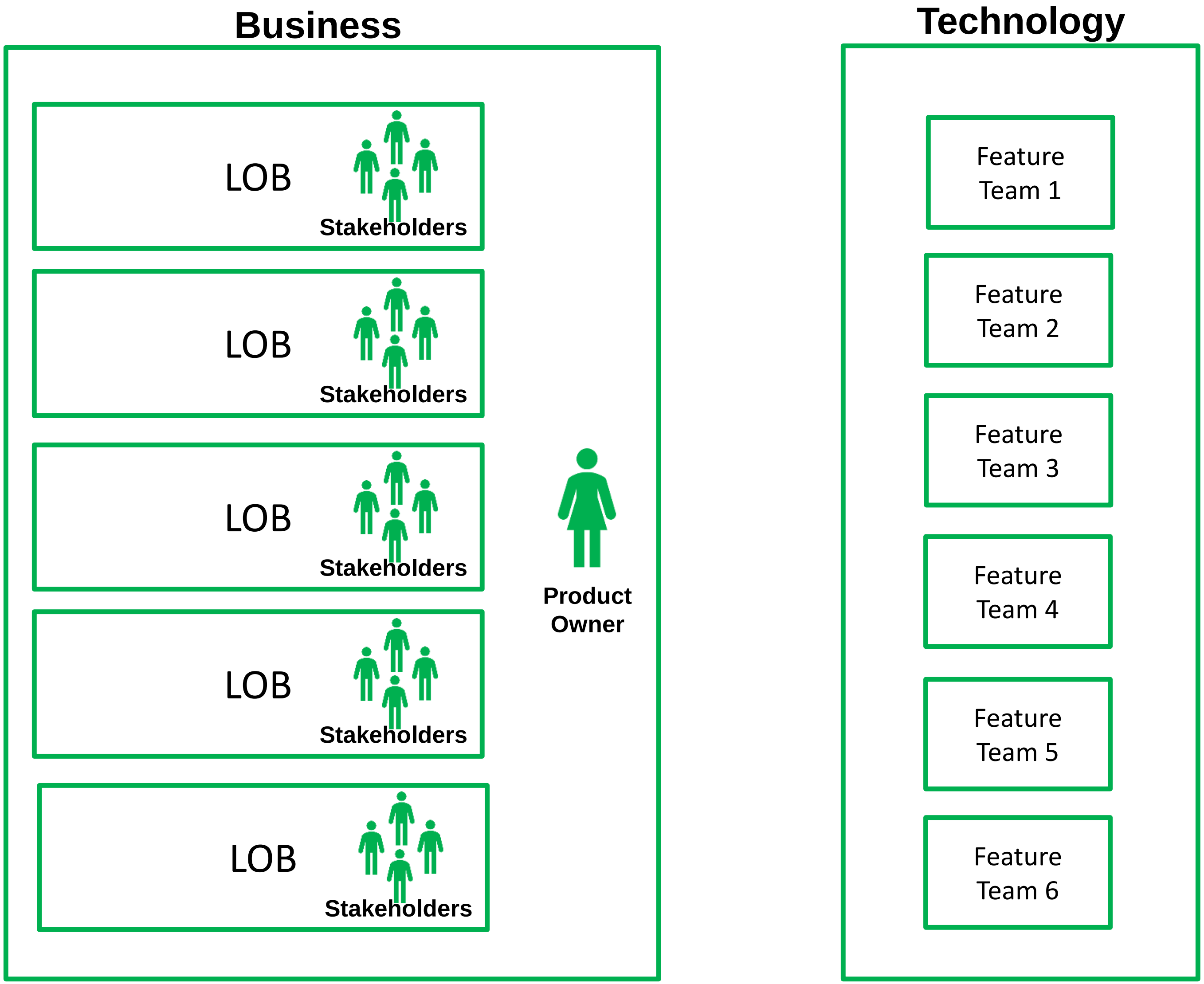
Component Team 10

Component Team 11

Component Team 12

Agile Friendly Organizational Design

(e.g. Large Scale Scrum)



Business

LOB Stakeholders

LOB Stakeholders

LOB Stakeholders

LOB Stakeholders

LOB Stakeholders

Product Owner

Technology

Feature Team 1

Feature Team 2

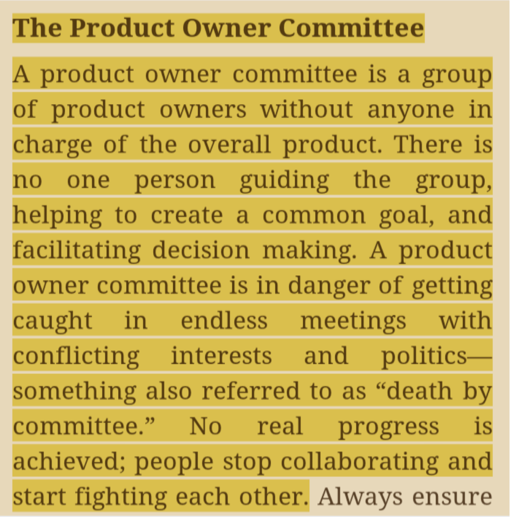
Feature Team 3

Feature Team 4

Feature Team 5

Feature Team 6

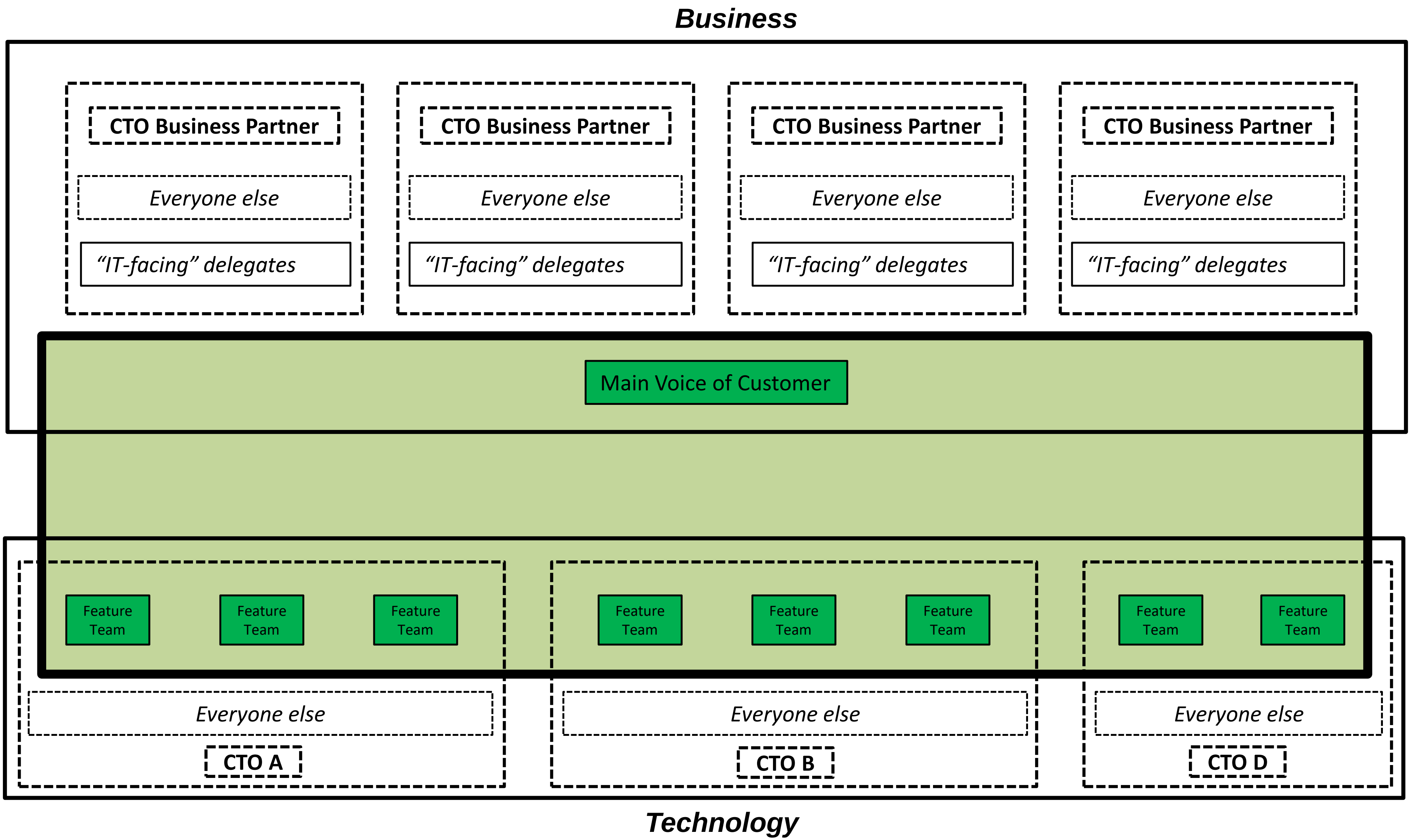
Product Ownership Team



The Product Owner Committee

A product owner committee is a group of product owners without anyone in charge of the overall product. There is no one person guiding the group, helping to create a common goal, and facilitating decision making. A product owner committee is in danger of getting caught in endless meetings with conflicting interests and politics—something also referred to as "death by committee." No real progress is achieved; people stop collaborating and start fighting each other. Always ensure

…and…in Large Scale Scrum environments…

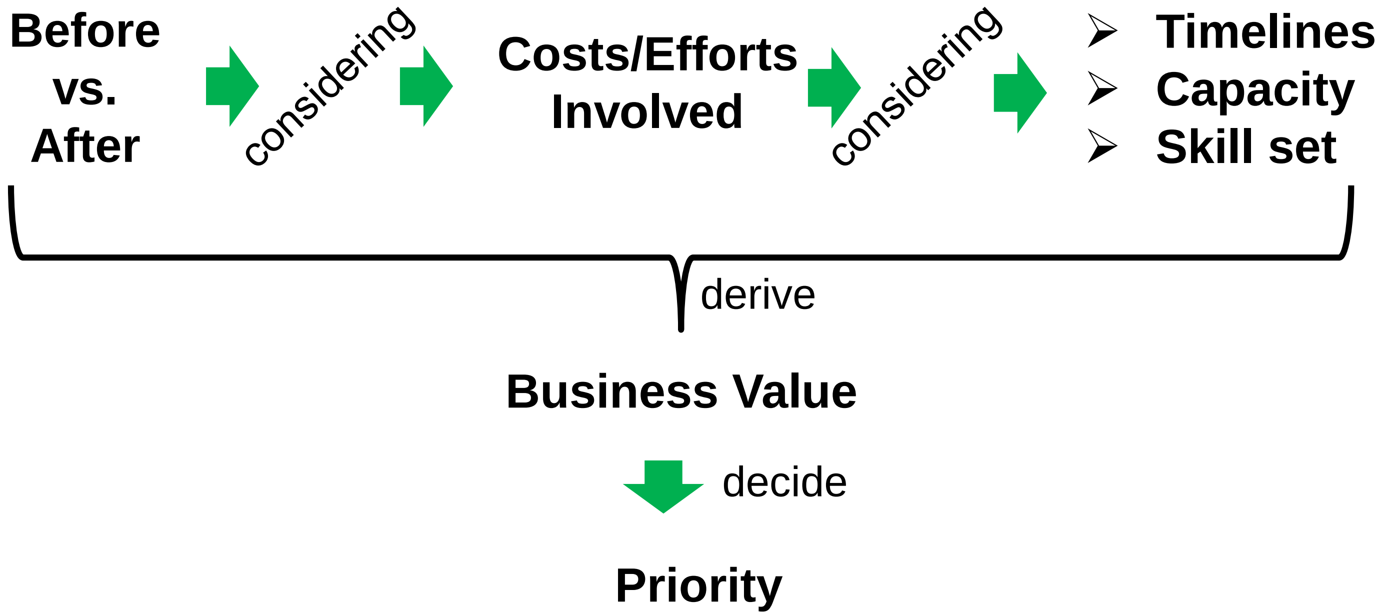


How to Monetize Business Value?

Making

Saving

Avoiding



Before vs. After

considering

Costs/Efforts Involved

considering

- **Timelines**
- **Capacity**
- **Skill set**

derive

Business Value

decide

Priority

What are Corporate Misfits Good For?

Janet Bumpas
"Container Ships & Speedboats",
Krakow, Poland, May 2017

"If you are scouting for people to put on an Innovation Project, go to HR and ask for a person with the fattest file and take that person" 😁

Mystery of the Century

WHAT

IS

AGILE?

Getting Language Straight

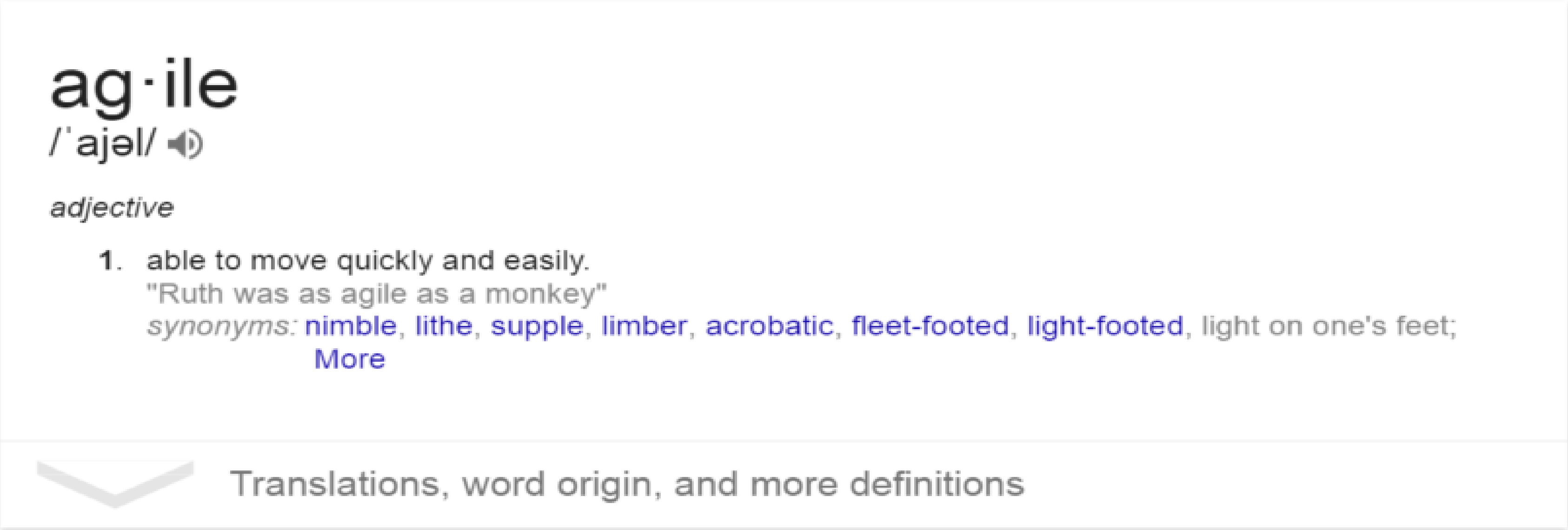
Agile

VERB

You cannot "DO" Agile?

ADJECTIVE

You can "Be" Agile?



ag·ile

/ˈajəl/ 🔊

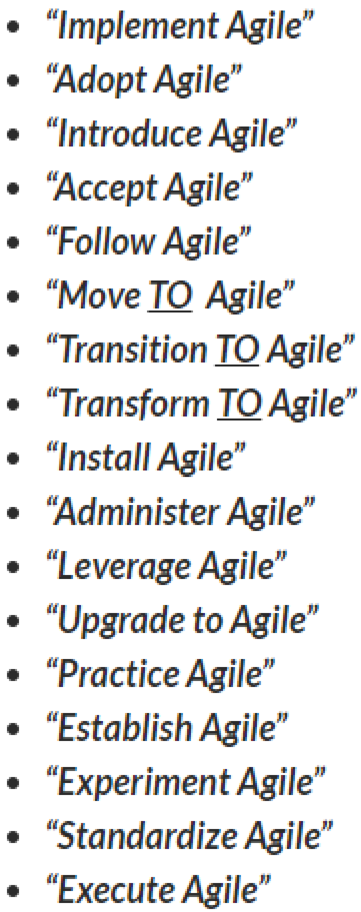
adjective

1. able to move quickly and easily.
   "Ruth was as agile as a monkey"
   synonyms: nimble, lithe, supple, limber, acrobatic, fleet-footed, light-footed, light on one's feet;
   More

Translations, word origin, and more definitions

Getting Language Straight

Don't try to be fancy with verbs:



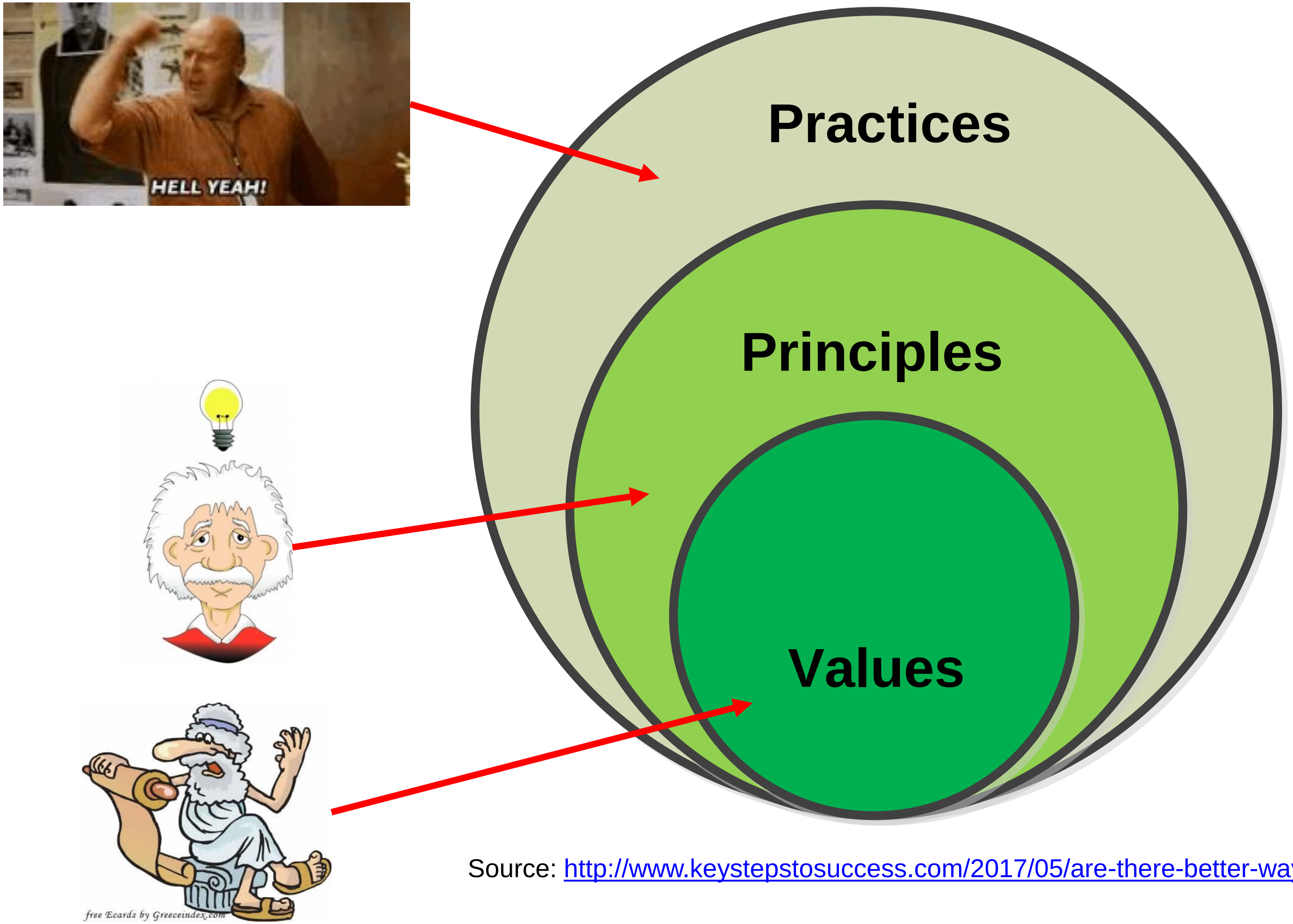
- *"Implement Agile"*
- *"Adopt Agile"*
- *"Introduce Agile"*
- *"Accept Agile"*
- *"Follow Agile"*
- *"Move TO Agile"*
- *"Transition TO Agile"*
- *"Transform TO Agile"*
- *"Install Agile"*
- *"Administer Agile"*
- *"Leverage Agile"*
- *"Upgrade to Agile"*
- *"Practice Agile"*
- *"Establish Agile"*
- *"Experiment Agile"*
- *"Standardize Agile"*
- *"Execute Agile"*

... PLEASE…

…just stick to BE/BEING agile…

Source: Bad Choice of Verbs Associated with "Agile", by EFL People

Where Do You Start?



Practices

Principles

Values

Source: http://www.keystepstosuccess.com/2017/05/are-there-better-ways-to-teach/

Short Cuts

BEST
PRACTICES

Source: https://www.youtube.com/watch?v=FllvssCuLuM

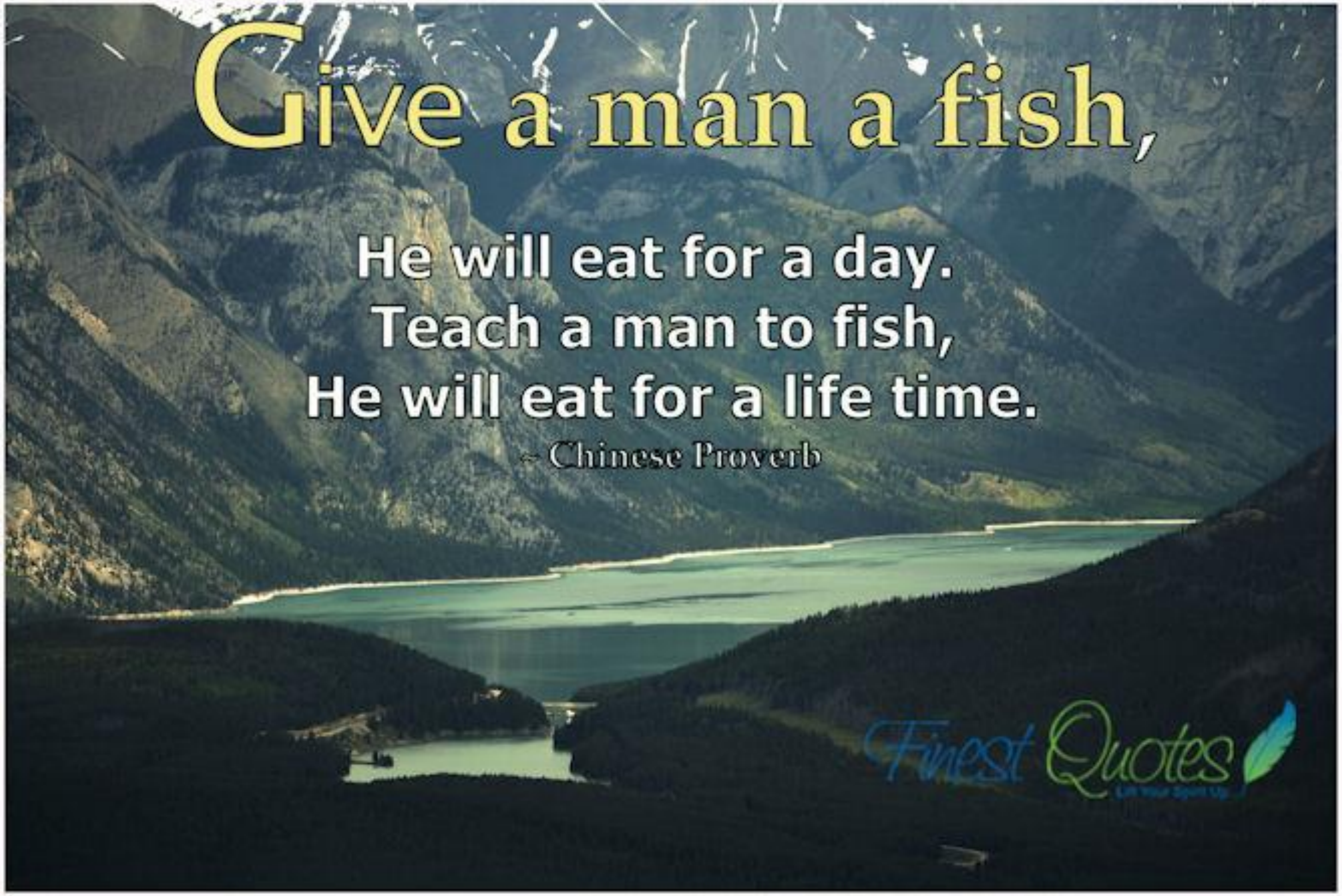
...There are THOUSANDS of great practices...
but NONE of them are "best"

RALLY SOFTWARE VERSIONONE Agile Made Easier JIRA PIVOTAL TRACKER

Owning vs. Renting

Source: https://www.manufacturedhomelivingnews.com/own-vs-rent-modern-housing-options/

Solving vs. Teaching How To Solve



Give a man a fish,

He will eat for a day.
Teach a man to fish,
He will eat for a life time.

~ Chinese Proverb

Source: http://www.lawteched.com/teaching-a-man-to-fish/

So…Why So much Renting?

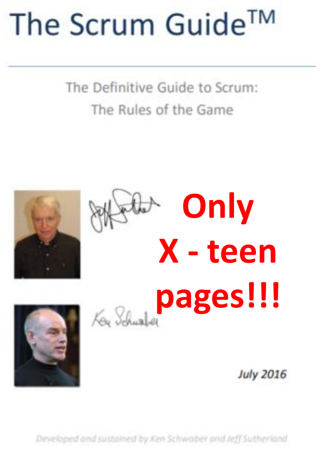
57.3454354%
Agile
by
EOY

☑

Source: http://www.employeerightslawgroup.com/harassment/peer-pressure-just-go-along-with-it-its-easier-for-everyone/

Due to Pressure.
and Peer Pressure.

“Scripted” Play at Agile “Theater”



The Scrum Guide™

The Definitive Guide to Scrum:
The Rules of the Game

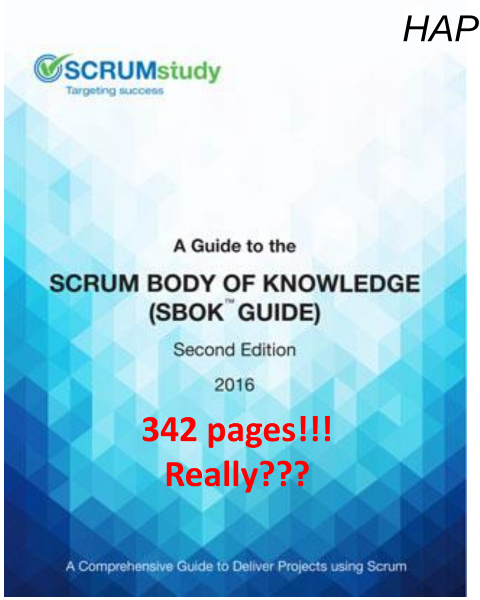
Ken Schwaber

July 2016

Developed and sustained by Ken Schwaber and Jeff Sutherland

Only
X - teen
pages!!!

*HOW DID
THIS
HAPPEN???*



SCRUMstudy
Targeting success

A Guide to the

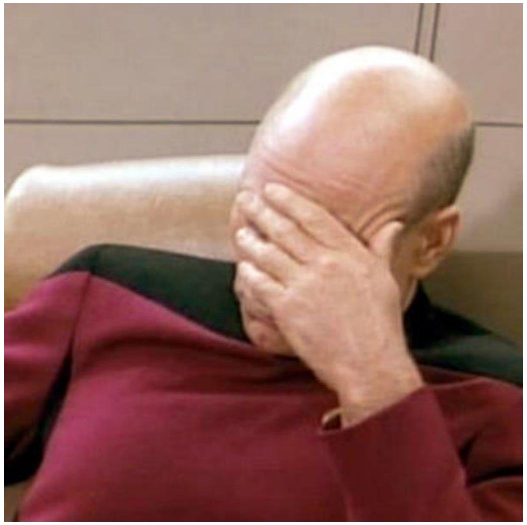
SCRUM BODY OF KNOWLEDGE
(SBOK™ GUIDE)

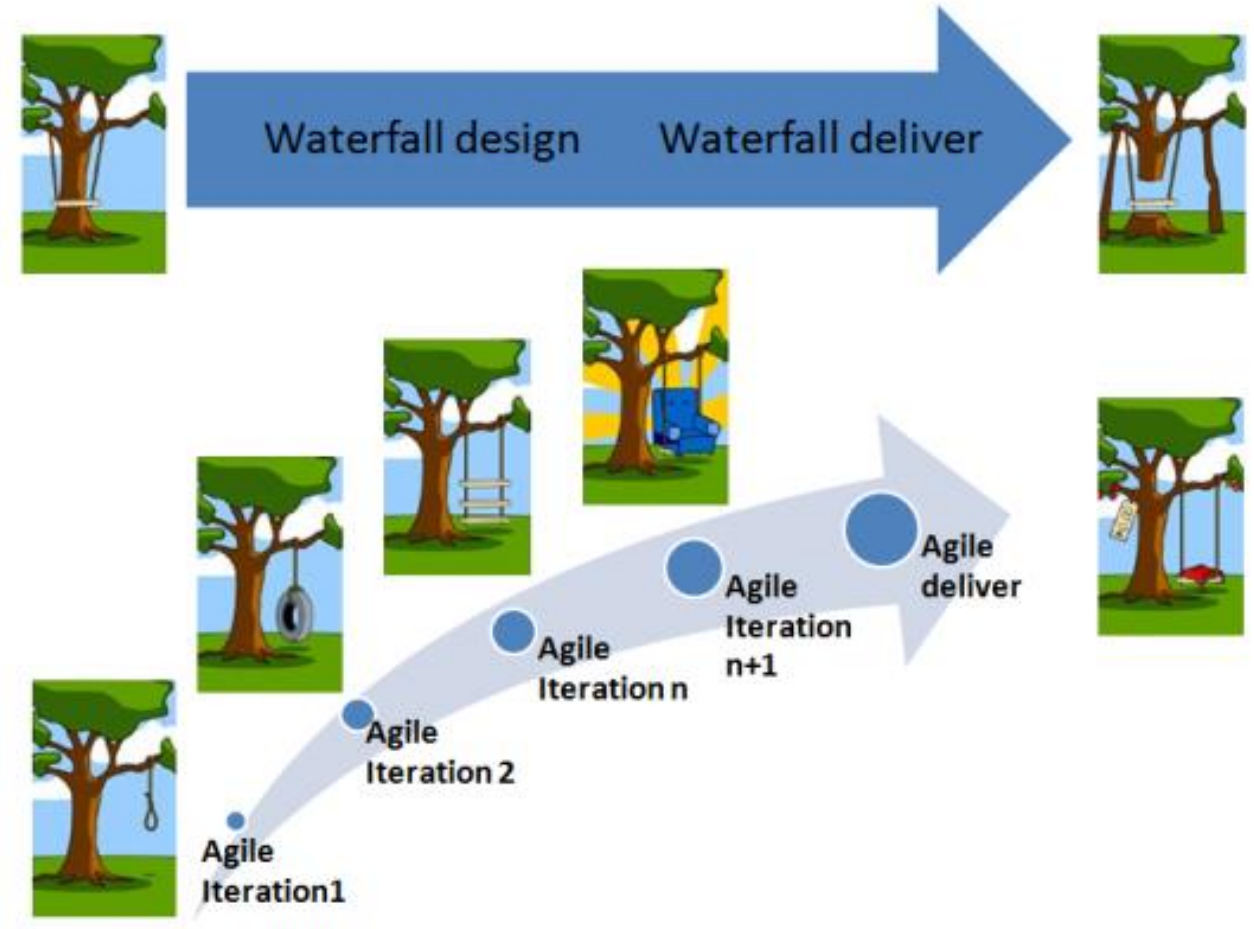
Second Edition

2016

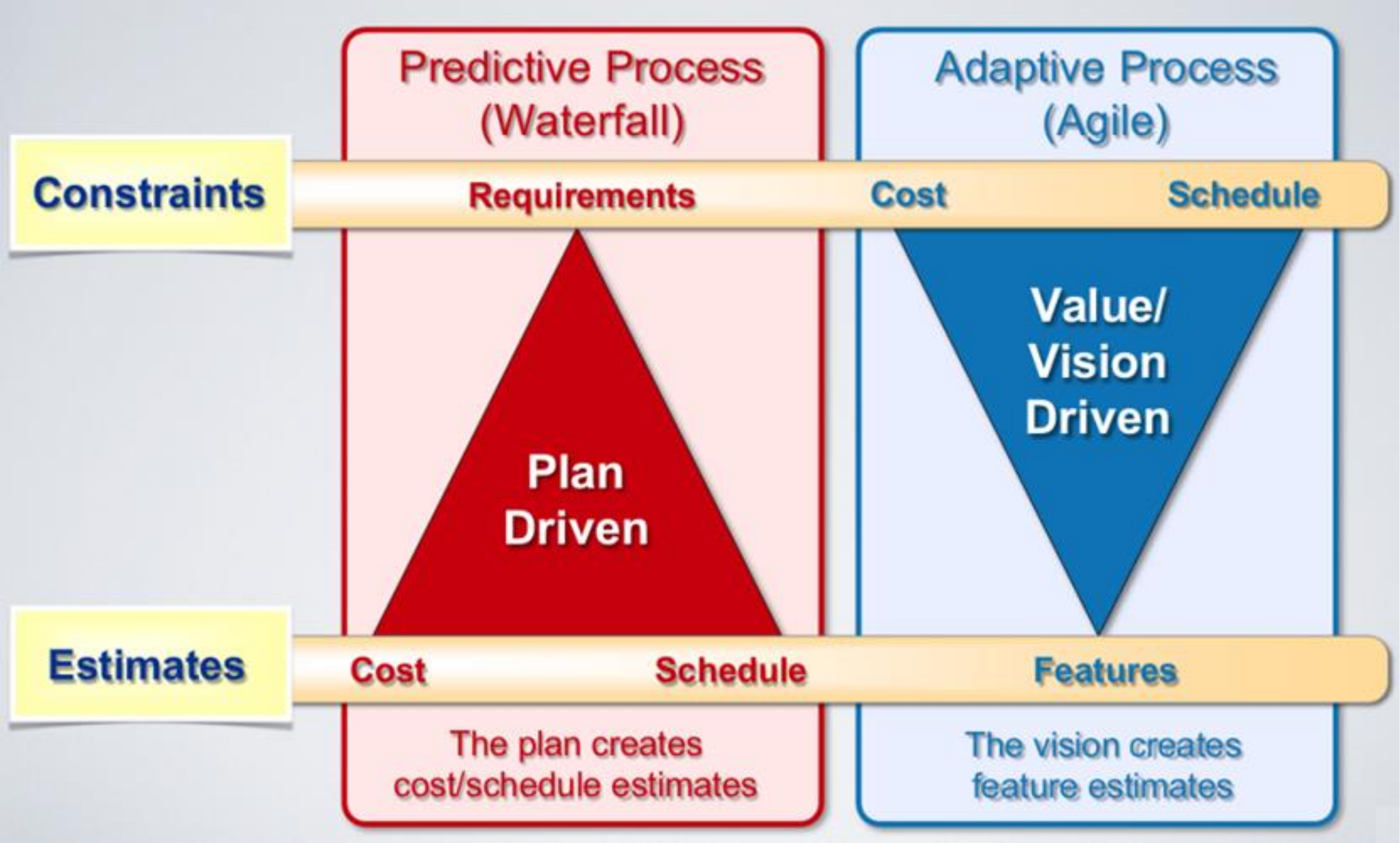
342 pages!!!
Really???

A Comprehensive Guide to Deliver Projects using Scrum

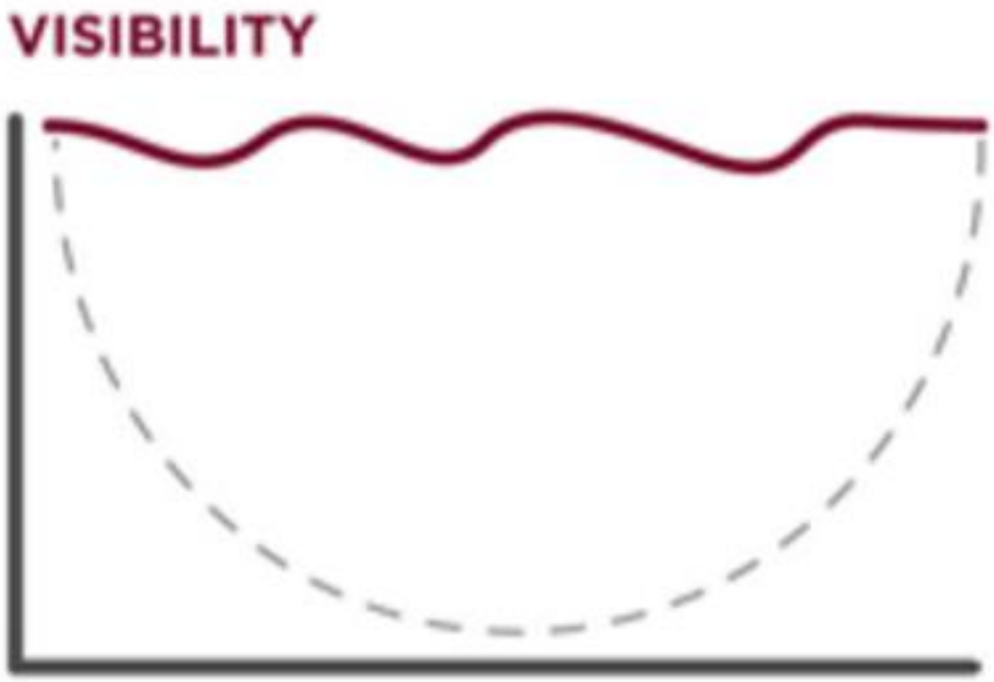




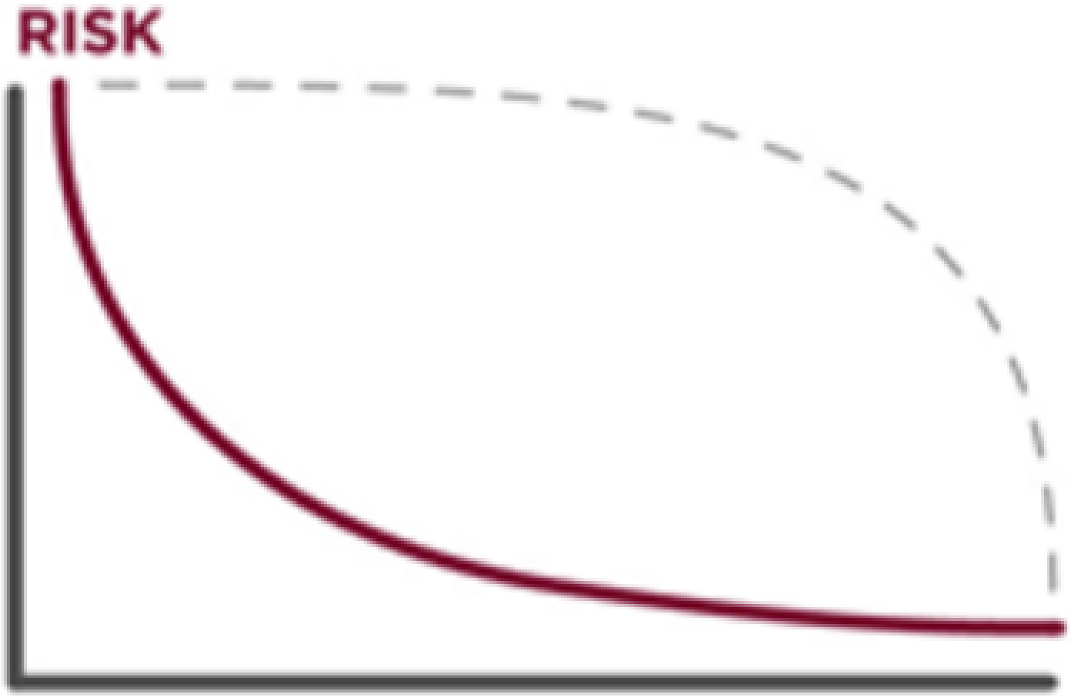
Source: https://umangsoftware.files.wordpress.com/2014/06/agile.png?w=540&h=406



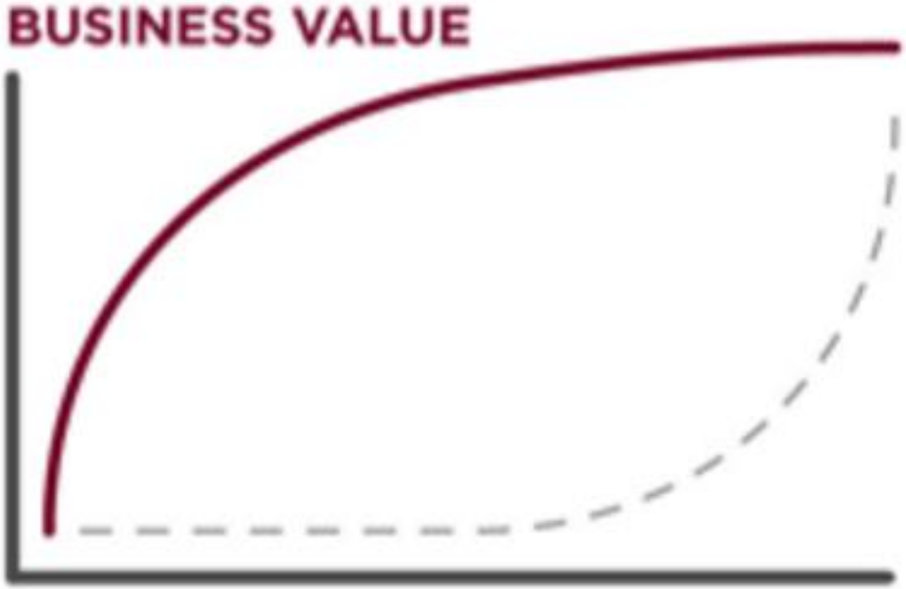
Source: http://tom-sylvester.com/wp-content/uploads/2013/06/agile_waterfall_triple_constraint.png



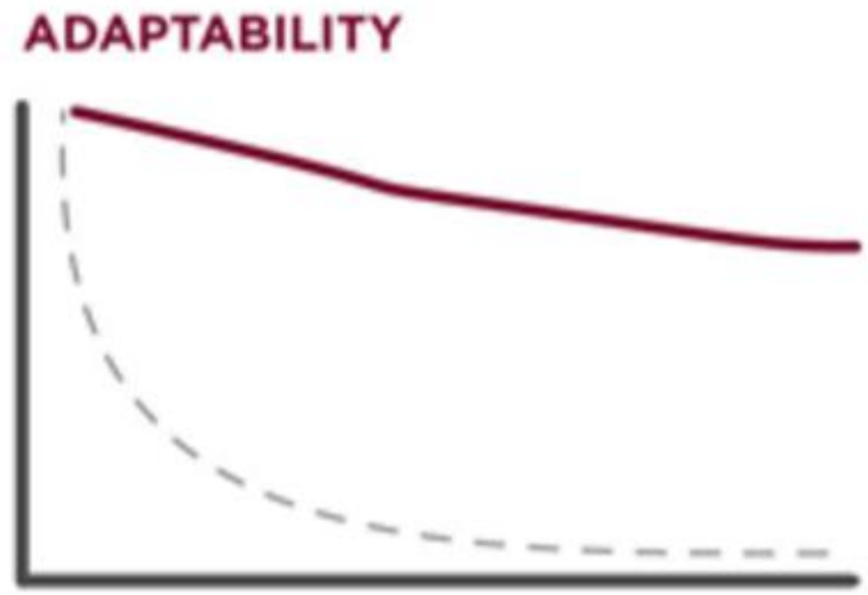
Source: http://image.slidesharecdn.com/ralphjocham-120612034405-phpapp02/95/agile-techniques-as-business-acceleration-tools-20-728.jpg?cb=1339472718



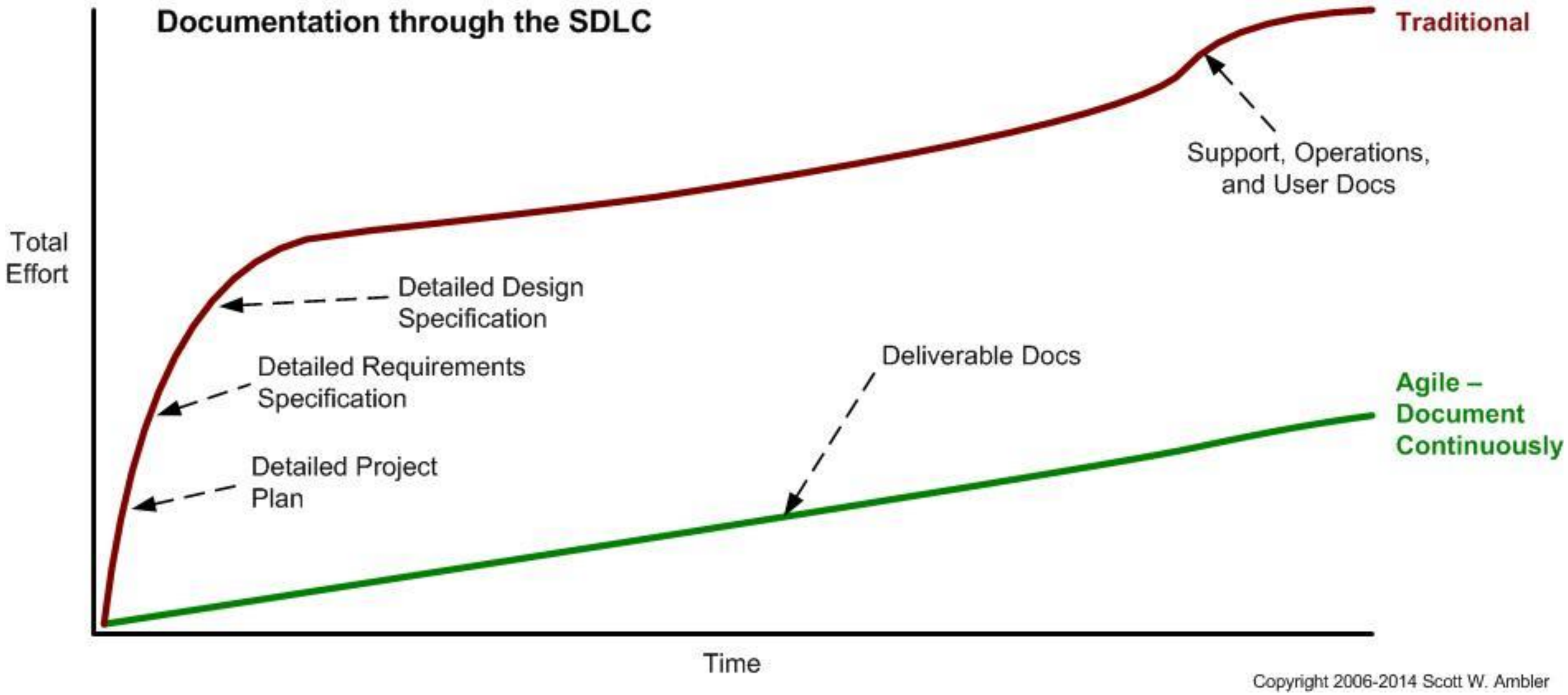
Source: http://image.slidesharecdn.com/ralphjocham-120612034405-phpapp02/95/agile-techniques-as-business-acceleration-tools-20-728.jpg?cb=1339472718



Source: http://image.slidesharecdn.com/ralphjocham-120612034405-phpapp02/95/agile-techniques-as-business-acceleration-tools-20-728.jpg?cb=1339472718



Source: http://image.slidesharecdn.com/ralphjocham-120612034405-phpapp02/95/agile-techniques-as-business-acceleration-tools-20-728.jpg?cb=1339472718



Source: http://www.agilemodeling.com/essays/agileDocumentation.htm

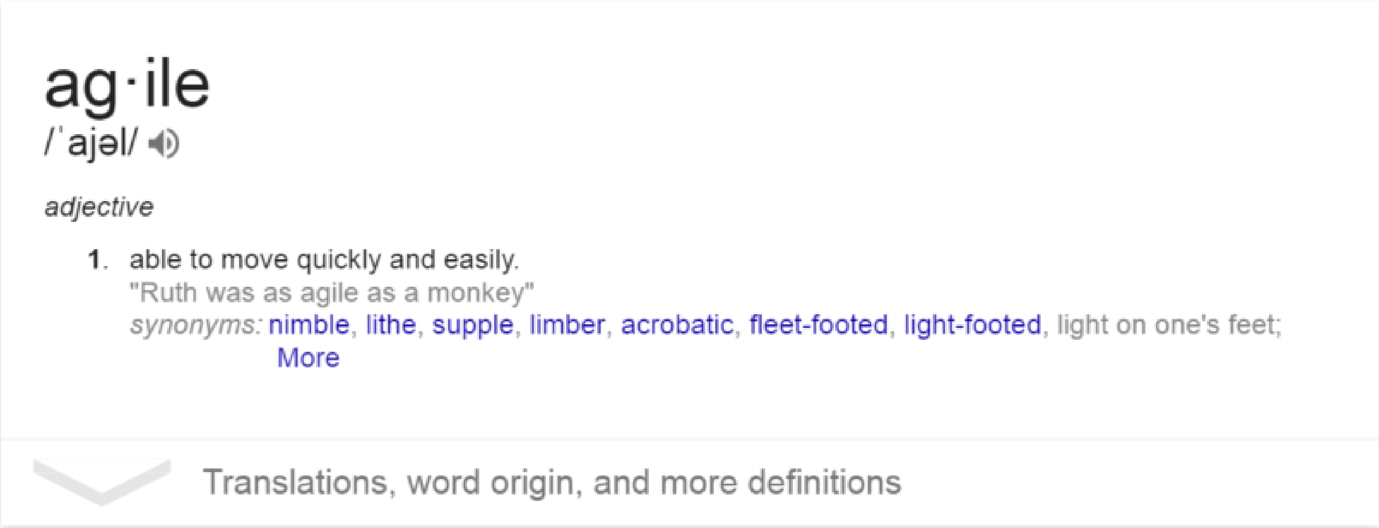
Agile Overview

Agile Overview → Organizational Agility → *What does “Agile” mean?*

What does “Agile” mean?

~~**N O U N**~~ *“You cannot DO Agile”*

A D J E C T I V E *“You can BE Agile”*



ag·ile

/ˈajəl/

adjective

1. able to move quickly and easily.
   "Ruth was as agile as a monkey"
   synonyms: nimble, lithe, supple, limber, acrobatic, fleet-footed, light-footed, light on one's feet;
   More

Translations, word origin, and more definitions

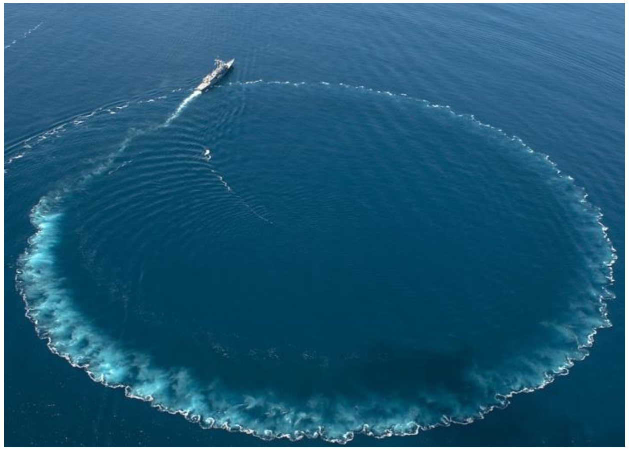
“Ability to turn on a dime for a dime”

C. Larman

What does "Agile" mean?

Image courtesy of Dockstavarvet AB

turning angle / cycle time



What does "Agile" mean?

complexity

vs

simplicity

What does “Agile” mean?

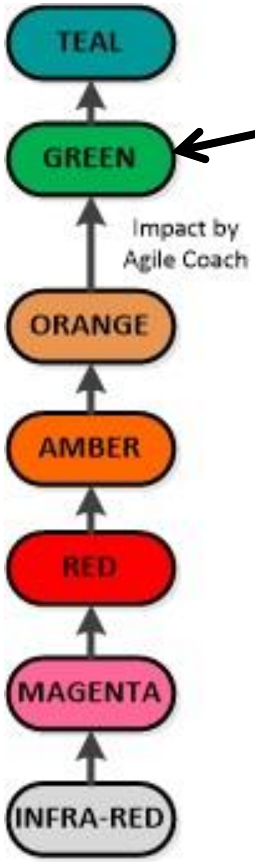
- **Most Common Misconceptions:**
- **Agility = Chaos, Anarchy and Low Discipline**
- **In Agile environments people are slackers and don’t work hard**
- **Agility implies low predictability and no planning**
- **Agile is not suitable for big projects**
- **Agility implies there is no documentation and traceability**
- **“We have installed agile tool. Now we are agile”**
- **“We *attend and report* in Daily Scrum”**

What does "Agile" mean?

Better Perception of Agile:

- **Agile Mindset of Leadership/Executive Management**
- **Waste-conscious Governance & Management**
- **Business Engagement (Product Ownership)**
- **Full Transparency and Collaboration**
- **Frequent Inspection and Adaptation**
- **Personal Dedication and Intrinsic Motivation**
- **Agile Engineering (Internal & External Partners)**

Organizational “Color”



Organizational Agility

Metaphor: LIVING ORGANISM

self management

Metaphor: FAMILY

relationships are more important than outcomes

Metaphor: MACHINE

organizational matrix, bonuses, perks and incentives

Metaphor: ARMY

formal hierarchies command & control

Metaphor: WOLFPACK

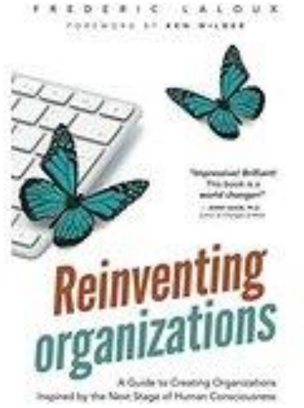
powerful chiefdoms

authority by the Elders

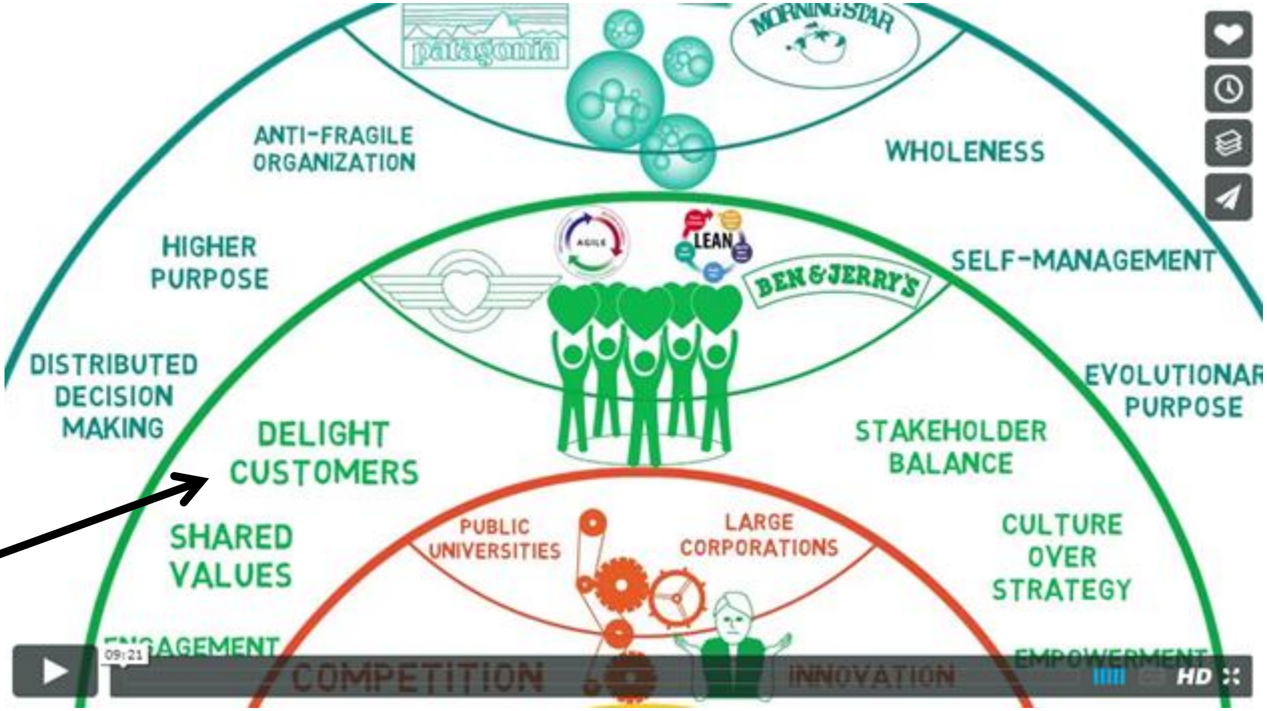
foraging bands

Source: Frederic Laloux’s “Reinventing organizations“

Organizational “Color”


FREDERIC LALOUX
FOREWORD BY KEN WILBER
Reinventing organizations
A Guide to Creating Organizations Inspired by the Next Stage of Human Consciousness

Reinventing Organizations, by Frederic Laloux



Organizational Agility

Source: Lean and Agile Adoption with the Laloux Culture Model, by Peter Green. Source: https://vimeo.com/121517508

Organizational “Tribal Stage”

“Life is Great”: “Less than 2 percent of workplace tribal culture is in this stage, wherein members who have made substantial innovations seek to use their potential to make a global impact.”

“We’re great (and they’re not)”: “The transition from “I’m great” to “we’re great” comes in this stage, wherein the tribe members are excited to work together for the benefit of the entire company.”

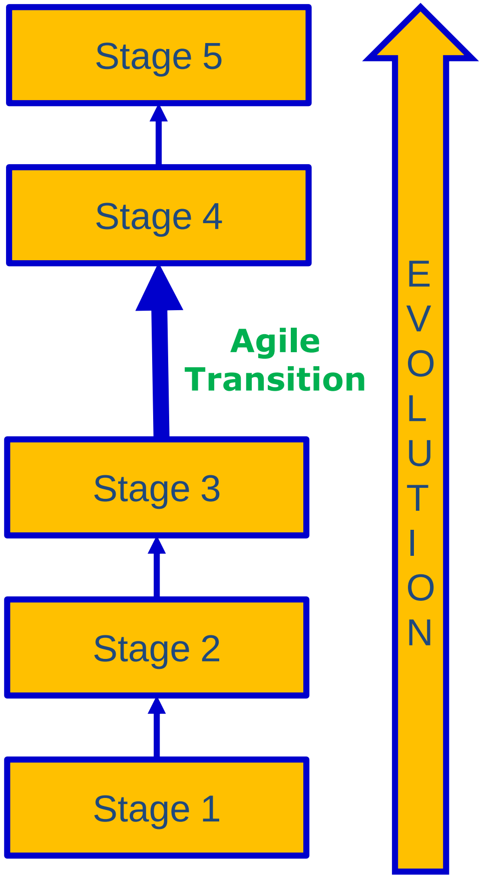
Collective Performers and Team Players

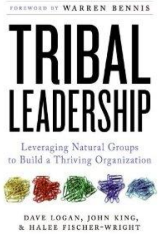
Individual Star Performers and Super Heroes

“I’m great (and you’re not)”: “49 percent of workplace tribes are in this stage, marked by knowledge hoarders who want to outwork and outthink their competitors on an individual basis. They are lone warriors who not only want to win, but need to be the best and brightest.”

“My life Sucks”: “The dominant culture for 25 percent of workplace tribes, this stage includes members who are passively antagonistic, sarcastic, and resistant to new management initiatives..”

“Life Sucks”: “The stage most professionals skip, these are tribes whose members are despairingly hostile—they may create scandals, steal from the company, or even threaten violence.”




FOREWORD BY WARREN BENNIS
TRIBAL LEADERSHIP
Leveraging Natural Groups to Build a Thriving Organization
DAVE LOGAN, JOHN KING, & HALEE FISCHER-WRIGHT

Source: [Tribal Leadership: Leveraging Natural Groups to Build a Thriving Organization Paperback, by Dave Logan and others]

Organizational “Motivation Versioning”

Motivation 1.0

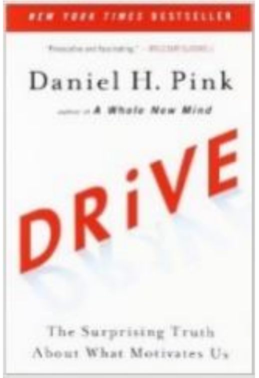
- Presumes that humans were biological creatures, struggling to obtain our basic needs for food, security and sex.
- Good for survival struggles

Motivation 2.0

- Rests on Theory X of Human Motivation, when management assumes that employees are lazy and will avoid work if they can as they dislike work
- Presumes that humans also responded to rewards and punishments in their environment
- Requires Compliance
- Believes in Carrots & Sticks that bring 7 deadly flaws: (1)extinguish intrinsic motivation, (2)diminish performance, (3)crush creativity, (4)crowd out good behavior, (5)encourage cheating, shortcuts, and unethical behavior, (6)become addictive, (7)foster short-term thinking

Motivation 3.0

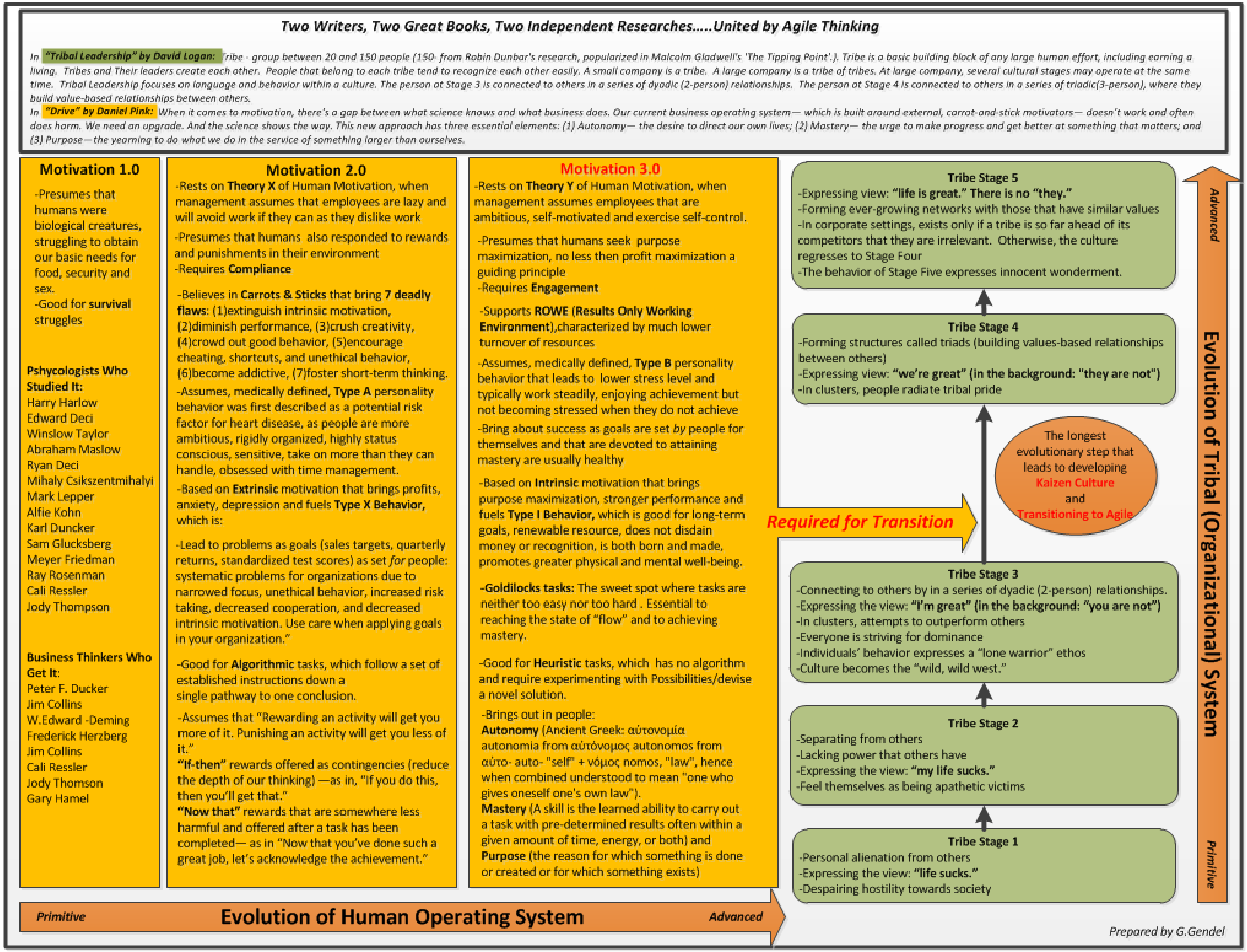
- Rests on Theory Y of Human Motivation, when management assumes employees that are ambitious, self-motivated and exercise self-control
- Presumes that humans seek purpose maximization, no less then profit maximization a guiding principle
- Requires Engagement
- Supports ROWE (Results Only Working Environment),characterized by much lower turnover of resources



(Source: Drive, by Daniel Pink)

Agile Overview

Agile Overview → Organizational Agility → *Organizational "Motivation Versioning"*



Two Writers, Two Great Books, Two Independent Researches.....United by Agile Thinking

*In **"Tribal Leadership" by David Logan:** Tribe - group between 20 and 150 people (150- from Robin Dunbar's research, popularized in Malcolm Gladwell's 'The Tipping Point'.). Tribe is a basic building block of any large human effort, including earning a living. Tribes and Their leaders create each other. People that belong to each tribe tend to recognize each other easily. A small company is a tribe. A large company is a tribe of tribes. At large company, several cultural stages may operate at the same time. Tribal Leadership focuses on language and behavior within a culture. The person at Stage 3 is connected to others in a series of dyadic (2-person) relationships. The person at Stage 4 is connected to others in a series of triadic(3-person), where they build value-based relationships between others.*

*In **"Drive" by Daniel Pink:** When it comes to motivation, there's a gap between what science knows and what business does. Our current business operating system— which is built around external, carrot-and-stick motivators— doesn't work and often does harm. We need an upgrade. And the science shows the way. This new approach has three essential elements: (1) Autonomy— the desire to direct our own lives; (2) Mastery— the urge to make progress and get better at something that matters; and (3) Purpose—the yearning to do what we do in the service of something larger than ourselves.*

Motivation 1.0

-Presumes that humans were biological creatures, struggling to obtain our basic needs for food, security and sex.

-Good for **survival** struggles

Pshycologists Who Studied It:
Harry Harlow
Edward Deci
Winslow Taylor
Abraham Maslow
Ryan Deci
Mihaly Csikszentmihalyi
Mark Lepper
Alfie Kohn
Karl Duncker
Sam Glucksberg
Meyer Friedman
Ray Rosenman
Cali Ressler
Jody Thompson

Business Thinkers Who Get It:
Peter F. Ducker
Jim Collins
W.Edward -Deming
Frederick Herzberg
Jim Collins
Cali Ressler
Jody Thomson
Gary Hamel

Motivation 2.0

-Rests on **Theory X** of Human Motivation, when management assumes that employees are lazy and will avoid work if they can as they dislike work

-Presumes that humans also responded to rewards and punishments in their environment
-Requires **Compliance**

-Believes in **Carrots & Sticks** that bring **7 deadly flaws**: (1)extinguish intrinsic motivation, (2)diminish performance, (3)crush creativity, (4)crowd out good behavior, (5)encourage cheating, shortcuts, and unethical behavior, (6)become addictive, (7)foster short-term thinking.

-Assumes, medically defined, **Type A** personality behavior was first described as a potential risk factor for heart disease, as people are more ambitious, rigidly organized, highly status conscious, sensitive, take on more than they can handle, obsessed with time management.

-Based on **Extrinsic** motivation that brings profits, anxiety, depression and fuels **Type X Behavior,** which is:

-Lead to problems as goals (sales targets, quarterly returns, standardized test scores) as set *for* people: systematic problems for organizations due to narrowed focus, unethical behavior, increased risk taking, decreased cooperation, and decreased intrinsic motivation. Use care when applying goals in your organization."

-Good for **Algorithmic** tasks, which follow a set of established instructions down a single pathway to one conclusion.

-Assumes that "Rewarding an activity will get you more of it. Punishing an activity will get you less of it."
"If-then" rewards offered as contingencies (reduce the depth of our thinking) —as in, "If you do this, then you'll get that."
"Now that" rewards that are somewhere less harmful and offered after a task has been completed— as in "Now that you've done such a great job, let's acknowledge the achievement."

Motivation 3.0

-Rests on **Theory Y** of Human Motivation, when management assumes employees that are ambitious, self-motivated and exercise self-control.

-Presumes that humans seek purpose maximization, no less then profit maximization a guiding principle
-Requires **Engagement**

-Supports **ROWE** (**Results Only Working Environment**),characterized by much lower turnover of resources

-Assumes, medically defined, **Type B** personality behavior that leads to lower stress level and typically work steadily, enjoying achievement but not becoming stressed when they do not achieve

-Bring about success as goals are set *by* people for themselves and that are devoted to attaining mastery are usually healthy

-Based on **Intrinsic** motivation that brings purpose maximization, stronger performance and fuels **Type I Behavior,** which is good for long-term goals, renewable resource, does not disdain money or recognition, is both born and made, promotes greater physical and mental well-being.

-Goldilocks tasks: The sweet spot where tasks are neither too easy nor too hard . Essential to reaching the state of "flow" and to achieving mastery.

-Good for **Heuristic** tasks, which has no algorithm and require experimenting with Possibilities/devise a novel solution.

-Brings out in people:
Autonomy (Ancient Greek: αὐτονομία autonomia from αὐτόνομος autonomos from αὐτο- auto- "self" + νόμος nomos, "law", hence when combined understood to mean "one who gives oneself one's own law").
Mastery (A skill is the learned ability to carry out a task with pre-determined results often within a given amount of time, energy, or both) and
Purpose (the reason for which something is done or created or for which something exists)

Required for Transition

The longest evolutionary step that leads to developing Kaizen Culture and Transitioning to Agile

Tribe Stage 5
-Expressing view: **"life is great." There is no "they."**
-Forming ever-growing networks with those that have similar values
-In corporate settings, exists only if a tribe is so far ahead of its competitors that they are irrelevant. Otherwise, the culture regresses to Stage Four
-The behavior of Stage Five expresses innocent wonderment.

Tribe Stage 4
-Forming structures called triads (building values-based relationships between others)
-Expressing view: **"we're great" (in the background: "they are not")**
-In clusters, people radiate tribal pride

Tribe Stage 3
-Connecting to others by in a series of dyadic (2-person) relationships.
-Expressing the view: **"I'm great" (in the background: "you are not")**
-In clusters, attempts to outperform others
-Everyone is striving for dominance
-Individuals' behavior expresses a "lone warrior" ethos
-Culture becomes the "wild, wild west."

Tribe Stage 2
-Separating from others
-Lacking power that others have
-Expressing the view: **"my life sucks."**
-Feel themselves as being apathetic victims

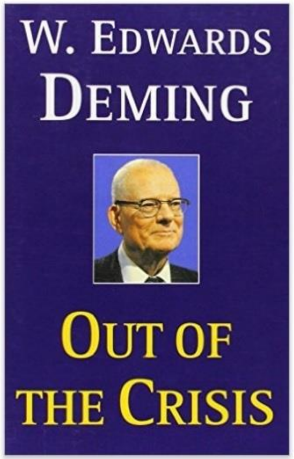
Tribe Stage 1
-Personal alienation from others
-Expressing the view: **"life sucks."**
-Despairing hostility towards society

Primitive **Evolution of Human Operating System** *Advanced*

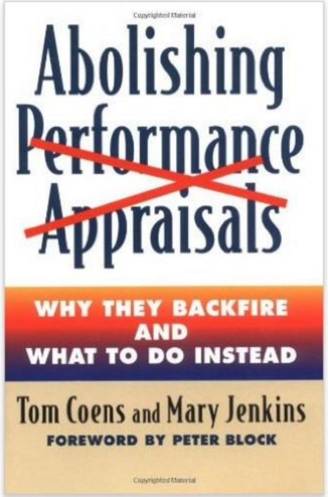
Primitive **Evolution of Tribal (Organizational) System** *Advanced*

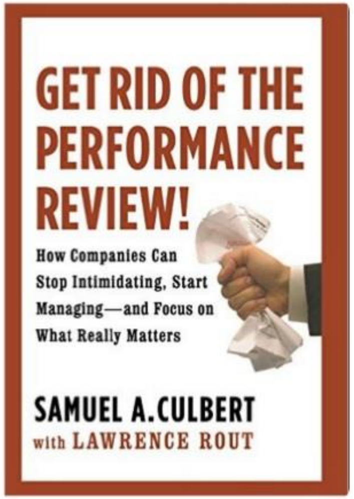
Prepared by G.Gendel

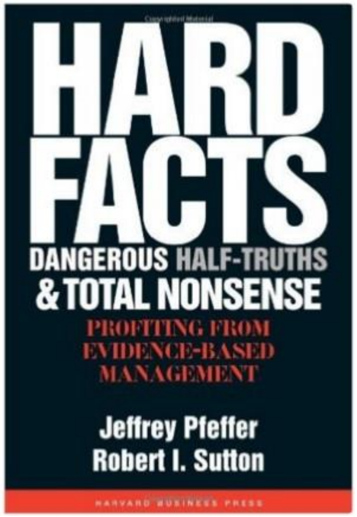
Frequently ignored scientific evidence

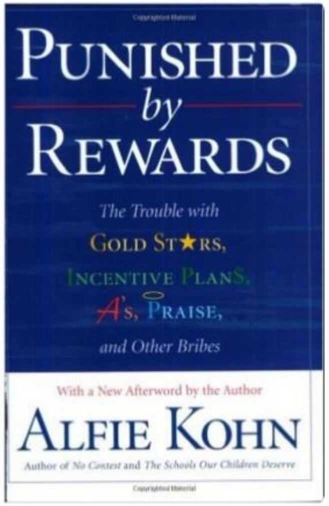


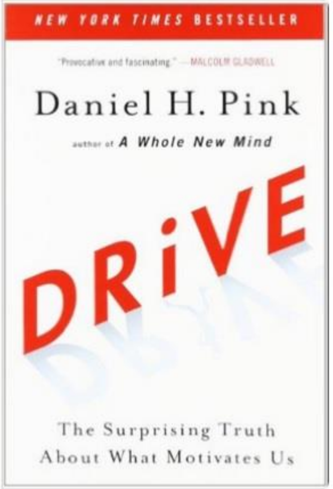
“The idea of a merit rating is alluring. The sound of the words captivates the imagination: pay for what you get; get what you pay for; motivate people to do their best, for their own good. The effect is exactly the opposite of what the words promise.”


Abolishing Performance Appraisals
WHY THEY BACKFIRE AND WHAT TO DO INSTEAD
Tom Coens and Mary Jenkins
FOREWORD BY PETER BLOCK


GET RID OF THE PERFORMANCE REVIEW!
How Companies Can Stop Intimidating, Start Managing—and Focus on What Really Matters
SAMUEL A. CULBERT
with LAWRENCE ROUT


HARD FACTS
DANGEROUS HALF-TRUTHS & TOTAL NONSENSE
PROFITING FROM EVIDENCE-BASED MANAGEMENT
Jeffrey Pfeffer
Robert I. Sutton
HARVARD BUSINESS PRESS


PUNISHED by REWARDS
The Trouble with Gold Stars, Incentive Plans, A's, Praise, and Other Bribes
With a New Afterword by the Author
ALFIE KOHN
Author of No Contest and The Schools Our Children Deserve


NEW YORK TIMES BESTSELLER
"Provocative and fascinating." —MALCOLM GLADWELL
Daniel H. Pink
author of A Whole New Mind
DRiVE
The Surprising Truth About What Motivates Us

What do companies value most?

Its Shareholders?

Primary Focus on:

- Stock prices, dividends
- Quarterly earnings, Budget cuts

Immediate Benefits:

- Happy shareholders
- More market investments

Its Customers?

Primary Focus on:

- Cheaper & higher quality products
- Shorter Cycle Time
- More clients and referrals

Immediate Benefits:

- Higher Revenues and ROI
- Stronger finances/balance sheets

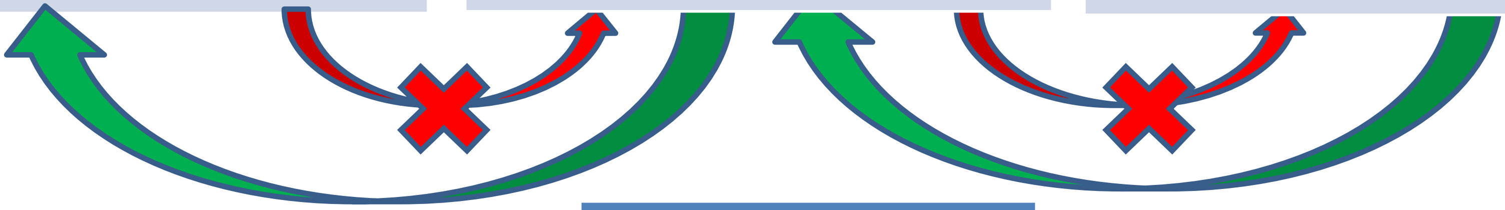
Its Employees?

Primary Focus on:

- Competitive salaries & benefits
- Better working conditions
- Career growth potentials
- Internal learning opportunities
- Job security and safety

Immediate Benefits:

- Happy Employees
- Gaining and retaining talent



3rd Parties?

Primary Focus on:

- SOWs
- SLAs

Immediate Benefits:

- Short term solutions

Daniel Pink
@DanielPink

"The true measure of a company is how they treat their lowest paid employee." -- Scott Scherr, CEO of Ultimate Software

50+ RETWEETS | 21 FAVORITES

What do companies value most?

Roots of Relationships: Organizations & Individuals

Green Zone vs. Red Zone Environments

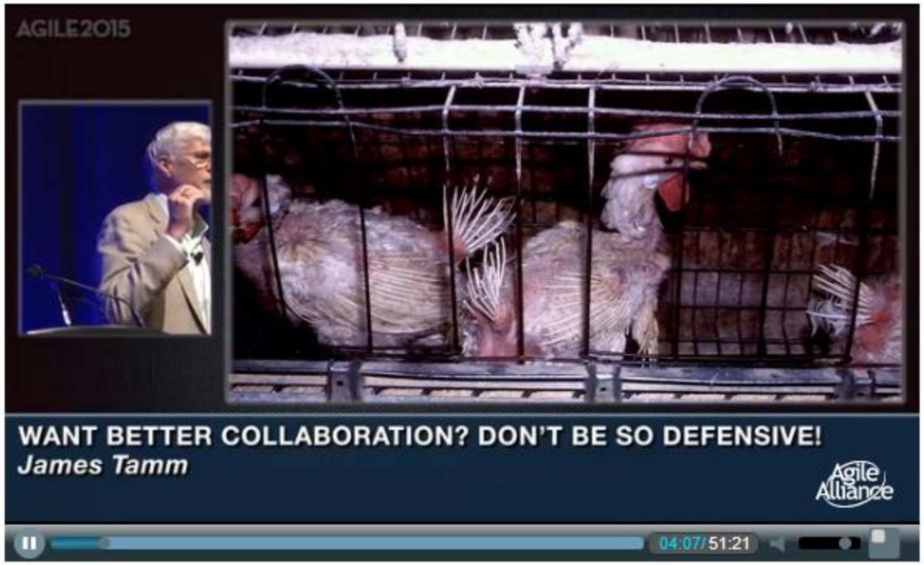
Agile Alliance 2015 (08-2015): Keynote speaker **James Tamm:** “Want Better Collaboration?”

Red Zone Chicken

- Star Performers
- Very aggressive
- Internally competitive
- Lower egg production
- High mortality rate

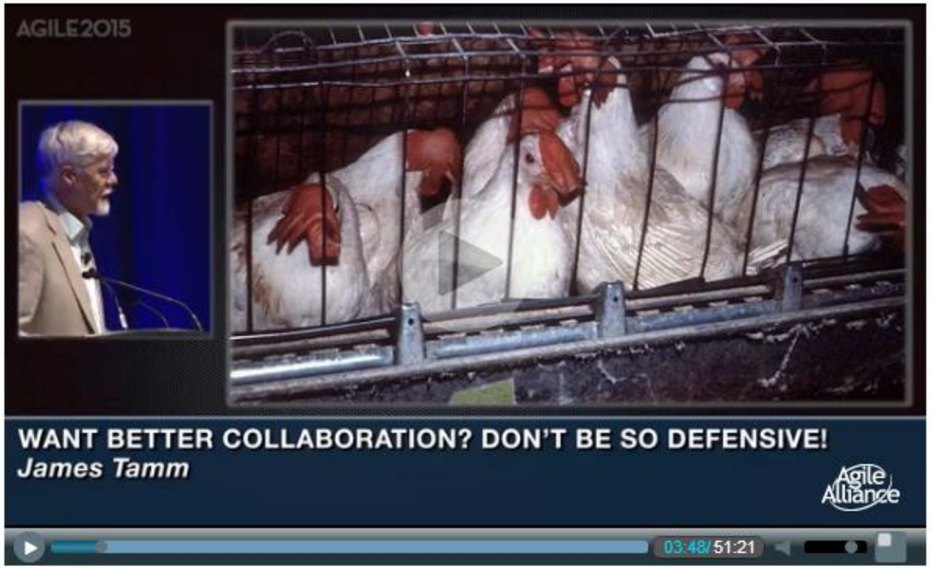
Green Zone Chicken

- Collaborative Performers
- Not aggressive
- Externally competitive
- Higher egg production
- Low mortality rate



WANT BETTER COLLABORATION? DON’T BE SO DEFENSIVE!

James Tamm



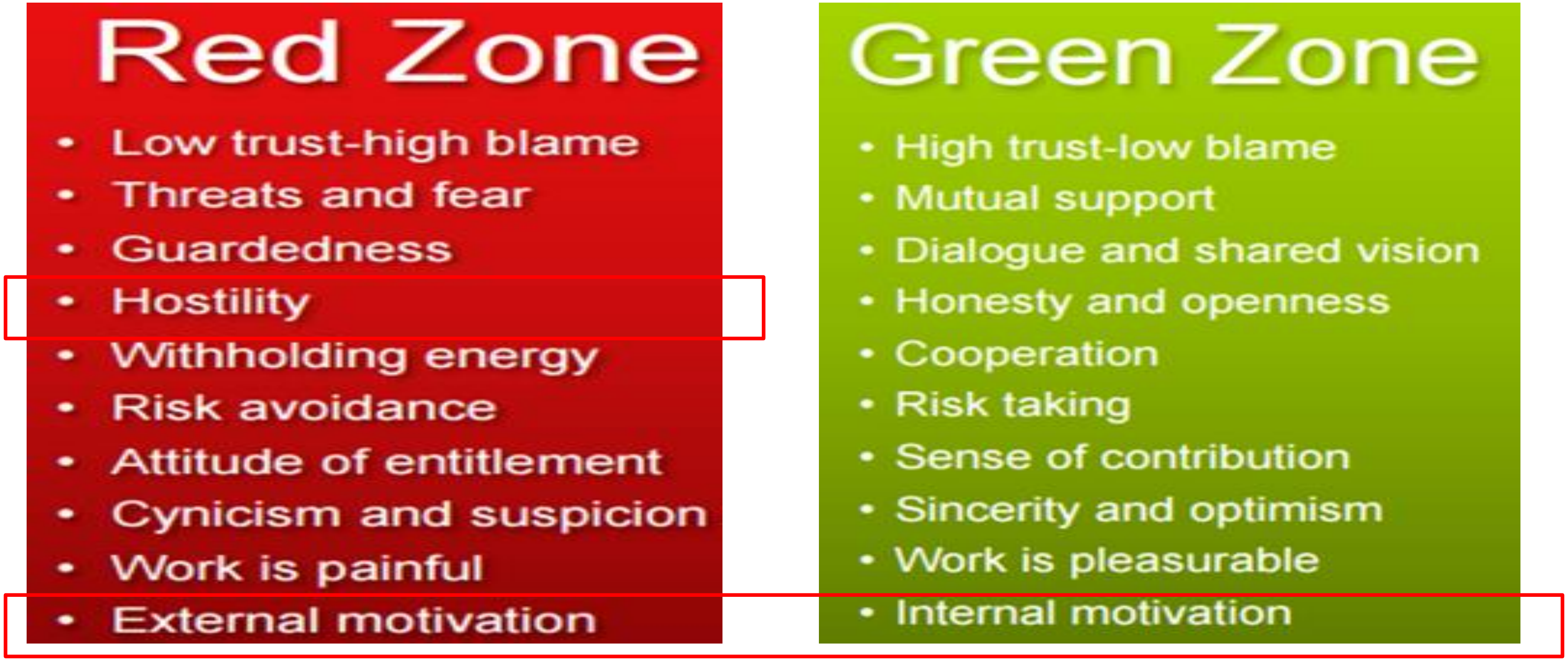
WANT BETTER COLLABORATION? DON’T BE SO DEFENSIVE!

James Tamm

(Source: https://submissions.agilealliance.org/system/attachments/attachments/000/001/674/original/AGILE__Aug_7._2015_for_participants.pdf?143 (page 6)

What do companies value most?

Roots of Relationships: Organizations & Individuals



Red Zone

- Low trust-high blame
- Threats and fear
- Guardedness
- Hostility
- Withholding energy
- Risk avoidance
- Attitude of entitlement
- Cynicism and suspicion
- Work is painful
- External motivation

Green Zone

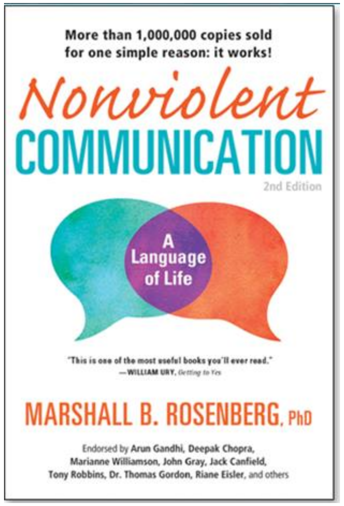
- High trust-low blame
- Mutual support
- Dialogue and shared vision
- Honesty and openness
- Cooperation
- Risk taking
- Sense of contribution
- Sincerity and optimism
- Work is pleasurable
- Internal motivation

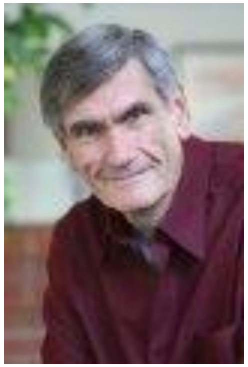
James Tamm:
"In my business Red Zone behaviors lead to LITIGATION , in your business: to SOFTWARE THAT REALLY SUCKS"

(Source: https://submissions.agilealliance.org/system/attachments/attachments/000/001/674/original/AGILE__Aug_7._2015_for_participants.pdf?1438891212, page 6)

What do companies value most?

Roots of Relationships: Organizations & Individuals


More than 1,000,000 copies sold for one simple reason: it works!
Nonviolent
COMMUNICATION
2nd Edition
A Language of Life
"This is one of the most useful books you'll ever read." —WILLIAM URY, Getting to Yes
MARSHALL B. ROSENBERG, PhD
Endorsed by Arun Gandhi, Deepak Chopra, Marianne Williamson, John Gray, Jack Canfield, Tony Robbins, Dr. Thomas Gordon, Riane Eisler, and others



Marshall Rosenberg, Ph.D.

- ➢ Has initiated peace programs in war-torn areas throughout the world including Rwanda, Burundi, Nigeria, Malaysia, Indonesia, Sri Lanka, the Middle East, Serbia, Croatia, and Ireland.
- ➢ Founder and director of educational services for the Center for Nonviolent Communication (CNVC)

Our Focus is on:

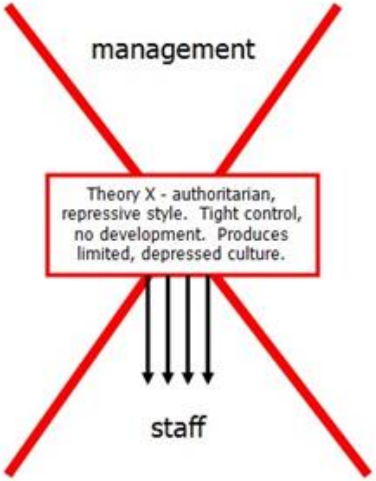
- ➢ How to deal with individual hostility and avoid hostile situations
- ➢ How to handle threats and fear coming from peers and superiors
- ➢ How to gain trust of others
- ➢ How to embrace conflict resolution and peace

(Source: http://www.amazon.com/dp/1892005034/?tag=googhydr-20&hvadid=74866992912&hvpos=1t1&hvexid=&hvnetw=g&hvrand=2511768207107237187&hvpone=&hvptwo=&hvqmt=b&hvdev=c&ref=pd_sl_7xly5lfn9r_b

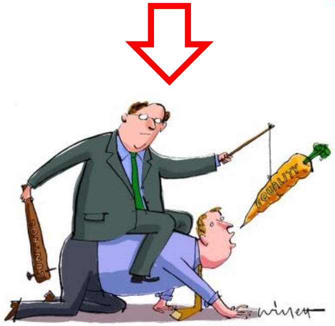
What do companies value most?

Roots of Relationships: Organizations & Individuals

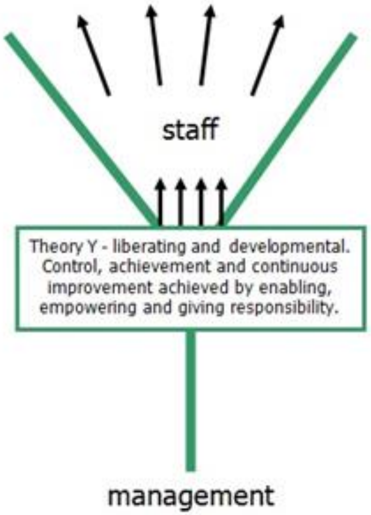
Theory X



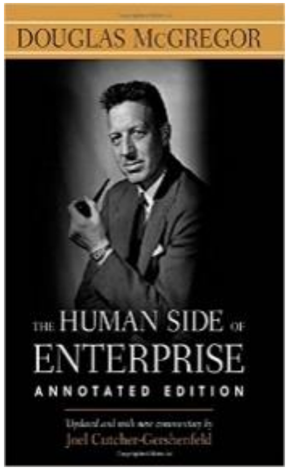
Theory X - authoritarian, repressive style. Tight control, no development. Produces limited, depressed culture.



Theory Y



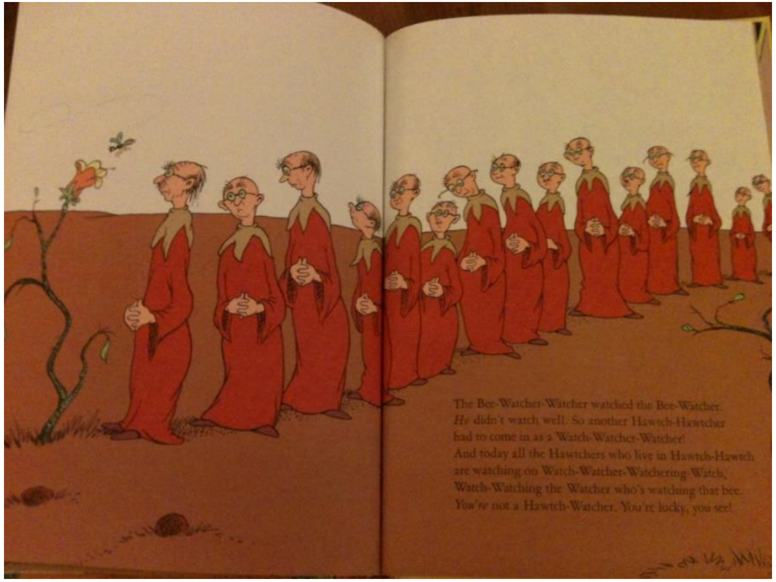
Theory Y - liberating and developmental. Control, achievement and continuous improvement achieved by enabling, empowering and giving responsibility.


DOUGLAS McGREGOR
THE HUMAN SIDE OF ENTERPRISE
ANNOTATED EDITION
Updated and with new commentary by Joel Cutcher-Gershenfeld

(Source: The Human Side of Enterprise, by Douglas McGregor)

McGregor's Theory X vs. Theory Y

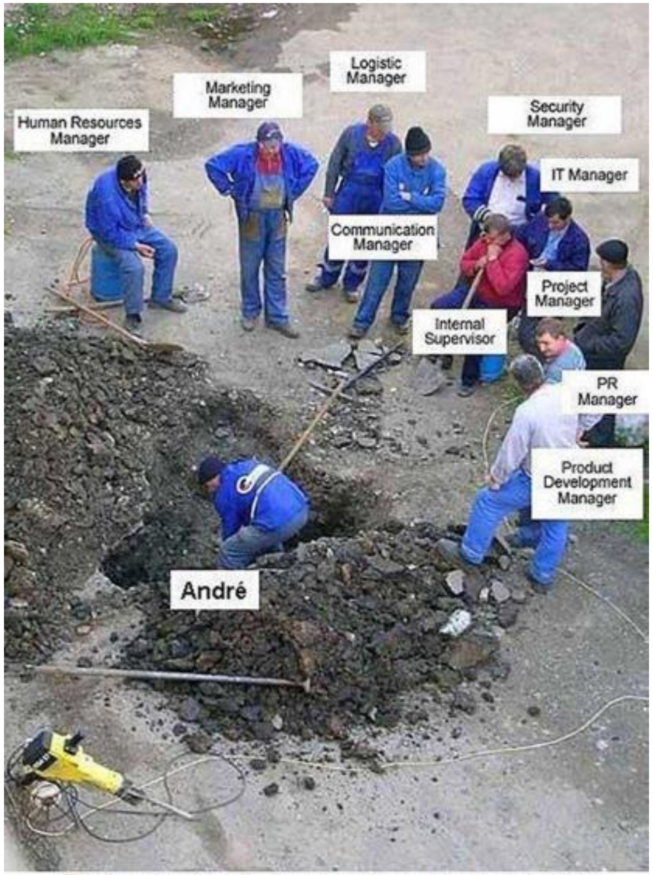
Roots of Relationships: Organizations & Individuals



The Bee-Watcher-Watcher watched the Bee-Watcher.
He didn't watch well. So another Hawtch-Hawtcher
had to come in as a Watch-Watcher-Watcher!
And today all the Hawtchers who live in Hawtch-Hawtch
are watching on Watch-Watcher-Watchering-Watch,
Watch-Watching the Watcher who's watching that bee.
You're not a Hawtch-Watcher. You're lucky, you see!

(Source: http://dutchrush.com/2013/04/pastors-wife/)

Muda?...



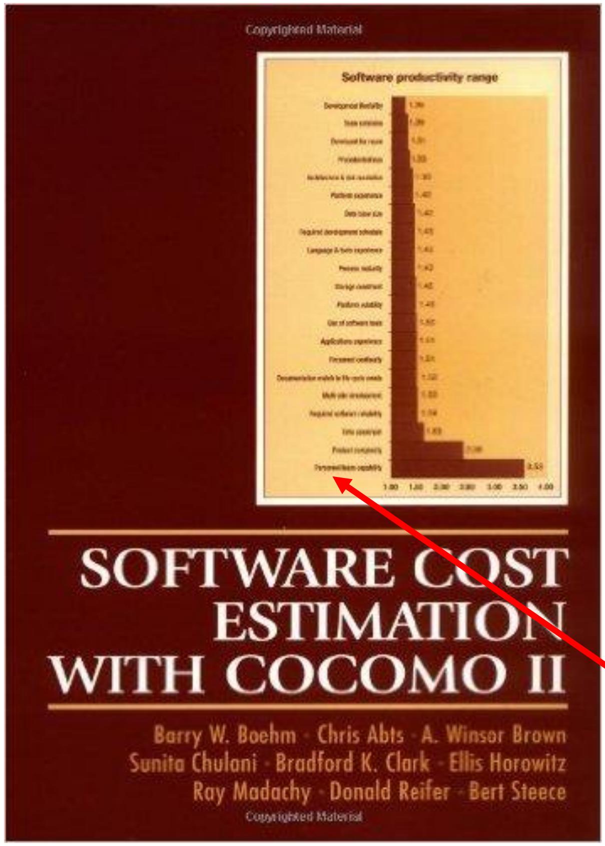
This financial crisis is forcing companies to make tough decisions;

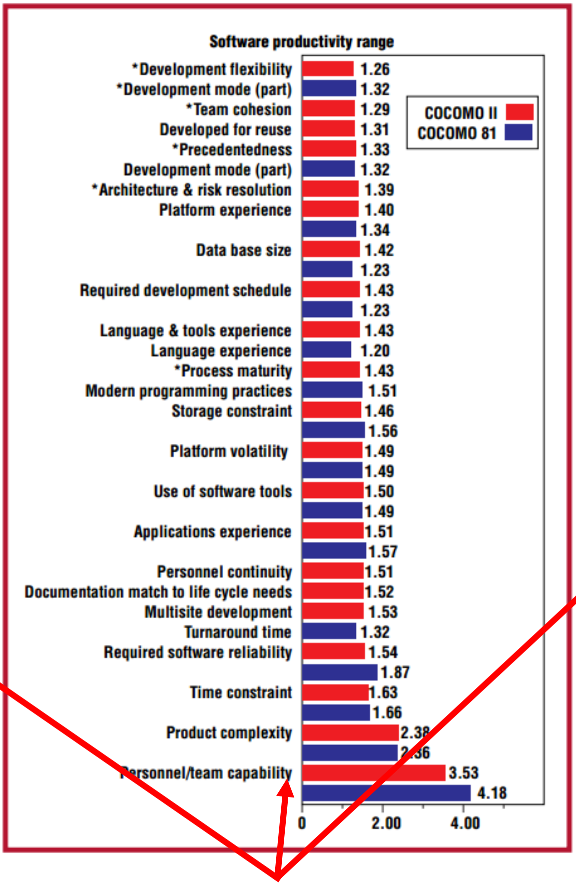
there is a risk that we might need to lay off Andre....

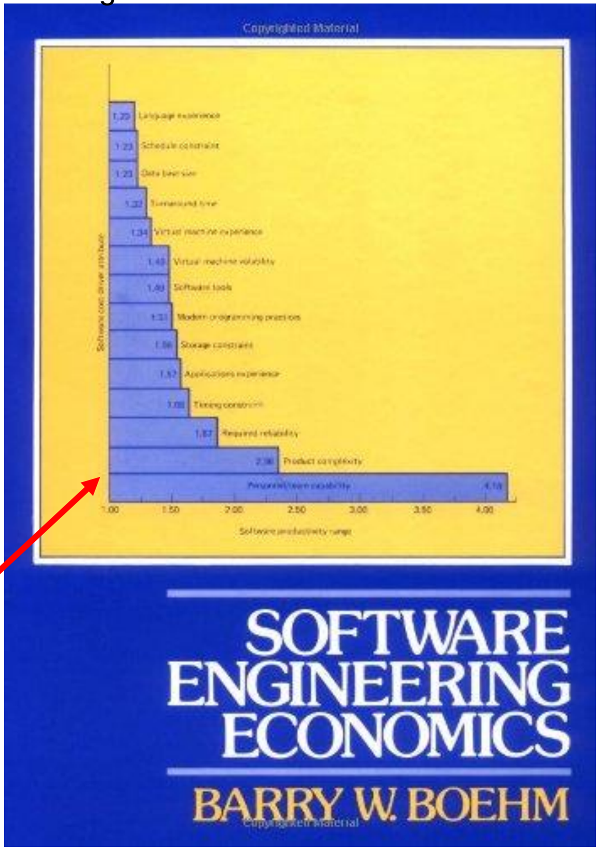
(Source: https://foreverfomo.wordpress.com/tag/dan-pink/)

What do companies value most?

Studies about Productivity Range: What factors carry most weight?


Copyrighted Material
Software productivity range
SOFTWARE COST ESTIMATION WITH COCOMO II
Barry W. Boehm · Chris Abts · A. Winsor Brown
Sunita Chulani · Bradford K. Clark · Ellis Horowitz
Ray Madachy · Donald Reifer · Bert Steece
Copyrighted Material





Humans

COnstructive COst MOdel II (COCOMO® II) - is a model that allows one to estimate the cost, effort, and schedule when planning a new software development activity.

Agile Overview

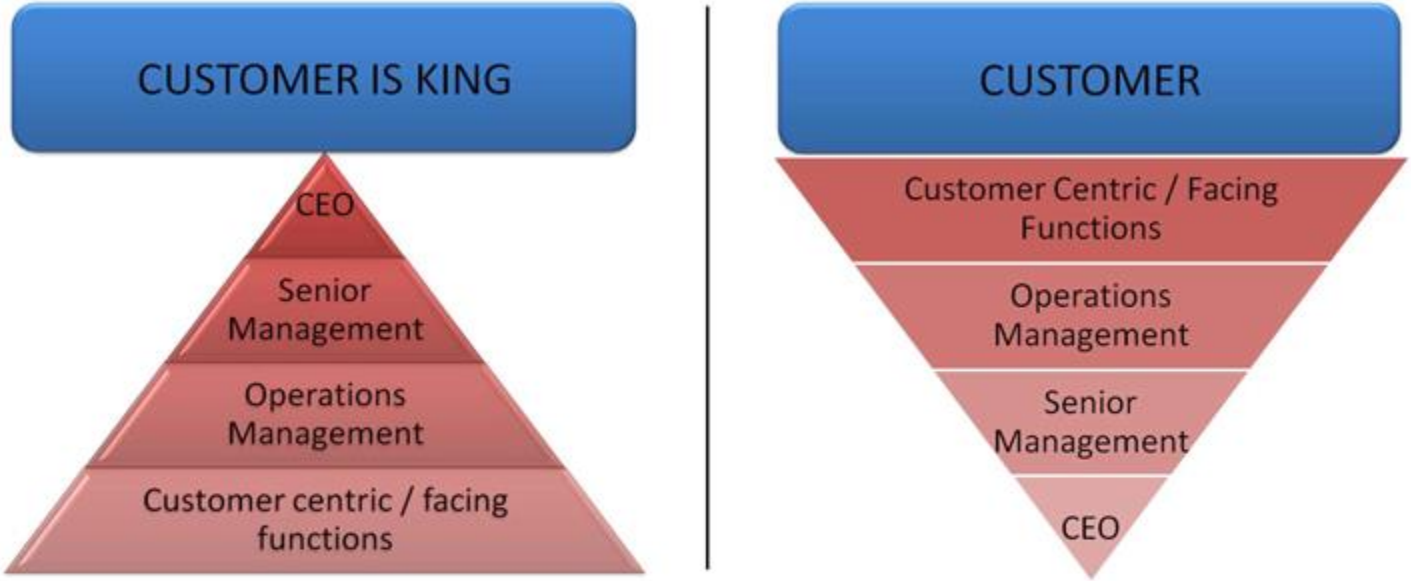
Agile Overview → Thinking Wide & Deep → Lean Thinking → *"Watching the baton, not the runners."*

"Watching the baton, not the runners."

- More focus of Work (workflow)
- Less focus on Worker(s)

Foundation of Lean: Managers as Teachers

- Managers - to do 'Go See' (Gemba)
- Managers, as capability builders and enablers
- Less command & controlling; more empowerment
- Teachers and Mentors
- Inverted organizational structure



Definition and Theory

The Scrum Guide:
“Scrum : A framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value. “

- Characteristics:
  - Lightweight
  - Simple to understand
  - Difficult to master

- Historically, Has been in use since early 90s

- Co-found by: Ken Schwaber and Jeff Sutherland

- Consists of Teams, roles, events, artifacts, bound by rules

- Three (3) pillars of Scrum:
  - transparency
  - inspection
  - adaptation

Scrum

Scrum → Scrum Roles & Responsibilities → *Overview of Scrum Roles*

Overview of Scrum Roles

Product Owner

Development Team

ScrumMaster

- Scrum Team consists of a Product Owner, the Development Team, and a Scrum Master
- Scrum Teams are self-organizing and cross-functional and know how best to accomplish their work
- Scrum Teams have all competencies needed to accomplish the work without depending on others
- Scrum team model in Scrum is designed to optimize flexibility, creativity, and productivity
- Scrum Teams deliver products iteratively and incrementally, maximizing opportunities for feedback
- Incremental deliveries of “Done” product ensure a potentially shippable product increment (PSPI) at the end of each sprint

Product Owner

Product Owner

Responsibility

- Maximizing the value of the product and the work of the Development Team
- Managing Product Backlog and communicating to Development Team
- Ordering Product Backlog items to best achieve strategic goals and missions
- Optimizing the value of work by Development Team
- Making decisions visible in the content and ordering of the Product Backlog

Authority & Empowerment

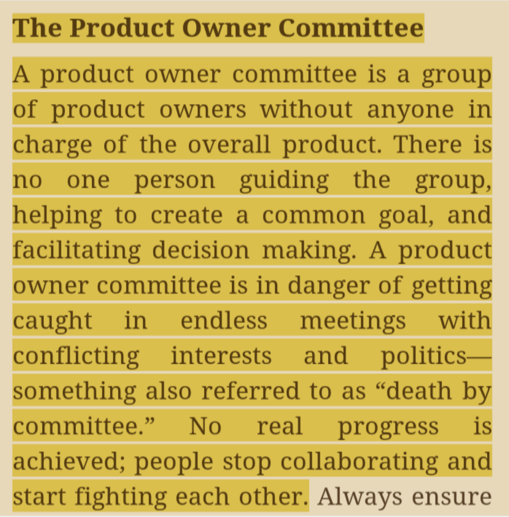
- Making final decisions on strategy
- Changing priorities, based on circumstances
- Protecting Development Team from external distractions

Anti-Patterns

- Instructing Dev. Team on “how” to do work
- Micro-managing Dev. Team or exhibiting Command & Control behaviors
- Lack of Authority/Empowerment
- Lack of Engagement
- “Death by committee”

Product Owner

Challenges with Product Ownership



The Product Owner Committee

A product owner committee is a group of product owners without anyone in charge of the overall product. There is no one person guiding the group, helping to create a common goal, and facilitating decision making. A product owner committee is in danger of getting caught in endless meetings with conflicting interests and politics—something also referred to as "death by committee." No real progress is achieved; people stop collaborating and start fighting each other. Always ensure

ScrumMaster

Responsibility

- Ensuring that Scrum is understood and implemented
- Ensuring that Scrum Team adheres to scrum theory, norms, values, principles
- Acting as servant-leader for the Scrum Team
- Servicing to the Product Owner
  - Suggesting ways to manage Backlog more effectively
  - Facilitating Scrum events

- Servicing to the Development Team
  - Coaching the Development Team in self-organization and cross-functionality
  - Coaching the Development Team in organizational environments in which Scrum is not yet fully adopted and understood
  - Helping the Development Team to create shippable products increments at the end of every sprint
  - Removing impediments to the Development Team's; escalating as needed
  - Facilitating Scrum events

- Servicing to the Organization
  - Leading and coaching the organization in its Scrum adoption
  - Implementing Scrum across an organization
  - Helping everyone understand Scrum values
  - Making changes to increase productivity of the Scrum Team
  - Collaborating with other Scrum Masters to widen and deepen Scrum adoption

ScrumMaster

Authority & Empowerment

- Protecting Development Team from external distractions
- Brokering/arbitrating Team-level impediments and conflicts
- Escalating to senior management impediments and roadblocks that cannot be resolved at team level

Anti-Patterns

- Making final decisions on strategy
- Changing priorities, based on circumstances
- Instructing Dev. Team on “how” to do work
- Micro-managing Dev. Team or exhibiting Command & Control behaviors
- Acting as a line-manager

Scrum

Scrum → Scrum Roles & Responsibilities → Overview of Scrum Roles → *Development Team*

Development Team

Size

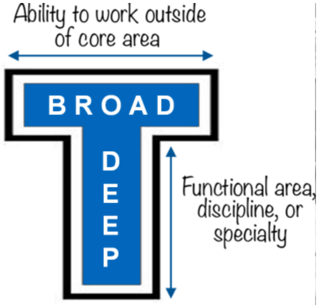
- 7 +/-2 people, but no less than 3
- Small teams may present skillset constraints that will prevent producing PSPI
- Large teams may cause excessive, counter-productive communication and lead to process complexity

Composition

- No sub-teams (component teams) inside Scrum/Feature Team
- Cross-functional team of technologists that are able to work independently
- **T-shaped** individuals that continuously learn from each other

Dynamics

- No titles or ranks, other than Developer
- No sub-teams (component teams) inside Scrum/Feature Team
- May have functional specialties but deliverable belongs to an entire team
- Collective ownership, shared responsibilities, joint delivery


Ability to work outside of core area
BROAD
DEEP
Functional area, discipline, or specialty

Impact on Traditional Roles

Project Manager (PM)

- Becoming Servant/Leaders and Enablers
- Removing Impediments/Obstacles before Teams
- Ensuring that teams have everything they need to get work done

Business Analyst (BA)

- Falling into Development Teams and:
  - Learning other skills
  - Teaching other members analytical skills
- Moving to Business and:
  - becoming SMEs
  - If properly empowered, becoming Product Owners

Manual Tester

- Learning Test Automation
- Learning other development skills

Technical Lead

- Becoming teachers and mentors
- Leading by example, not by command
- Letting junior members pick up more complex tasks

Business Community

- Channeling information/requests through Product Owner
- Helping Product Owner define on Vision and Strategy
- Providing clarifications to Product Owner and Team(s), as requested
- Refraining from giving priorities directly to Team(s)
- Attending Scrum ceremonies, when invited by Product Owner
    - Product Backlog Refinement (PBR) sessions – to provide clarifications and guidance
    - End-of-sprint Showcases – to review Team deliverables and provide feedback
- Actively participating in UAT, when asked by Product Owner
- Collecting and providing to Product Owner and Team(s) end-customer/market feedback
- Providing sponsorship/funding

Scrum Ceremonies

- Scrum framework as a set of prescribed ceremonies
- Scrum ceremonies are held regularity and minimize the need for other, less useful meetings
- All Scrum ceremonies are time-boxed events and limited.
- Sprint's duration is always fixed and cannot be made shorter or longer
- Each Scrum event is an opportunity to inspect and adapt to something new
- Scrum requires full transparency, inspection and adoption

Sprint

- Bounds start and end of a full-term development life cycle
- Rarely is longer than 1 month and is considered a project by itself
- Consistent in duration; scheduled back-to-front with next sprint
- Includes all other Scrum Ceremonies
  - Sprint Planning
  - Daily Scrums
  - Product Backlog Refinements (PBRs)
  - Sprint Review/Showcase
  - Retrospective
- Has a goal/business purpose
- Has fixed scope that should not be changed with exception of critical circumstances and at Product Owner's discretion only
- Can be cancelled only if authorized by Product Owner. This is highly discouraged.
- All incomplete work at sprint-end goes back to Product Backlog and re-estimated

Sprint Planning

- Time-boxed to 4 hours of a bi-weekly Sprint

- Should be always initiated by stating Sprint goal (coming from PO)

- For less mature teams, ScrumMaster is a facilitator; for more mature teams - self-organization might be sufficient

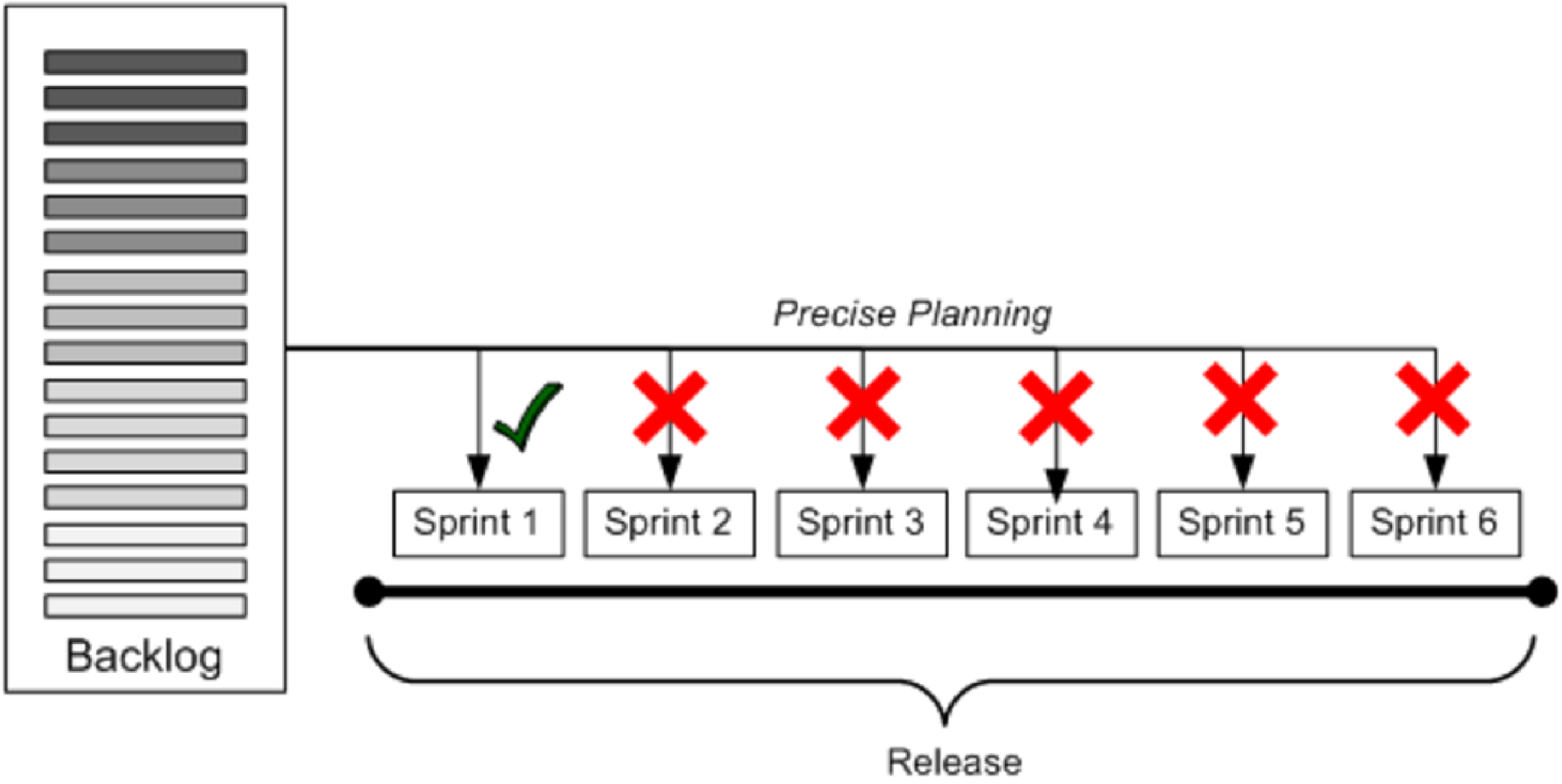
- Planning is frequently broken down into two sessions:

  - **Sprint Planning 1 (SP1)** - to discuss WHAT can be delivered in Sprint (PO is present)
    - Team collaborates and Sprint scope is forecasted/committed
    - Forecasting/commitment is based on:
    - Team's Historical Velocity
    - Team's projected Capacity
    - Backlog priorities

  - **Sprint Planning 2 (SP2)** - to discuss HOW work will get done (PO is not required)
    - Team further collaborates, by breaking down committed work into work smaller items
    - Team ensures that decomposed work does not exceed Team's Capacity
    - Team decides if SP1 commitment was reasonable and/if renegotiation with PO is necessary

Scrum

Sprint Planning



Daily Scrum

- Time-boxed to 15-minutes or less
- For Development Team, to discuss work of the last 24 hours
- Is hallmarked for the following 3 key questions, answered by all Team members:
  - What did I do yesterday that helped the Development Team meet the Sprint Goal?
  - What will I do today to help the Development Team meet the Sprint Goal?
  - Do I see any impediment that prevents me or Team from doing work?
- "Is the time of discovery, not the time of resolution“
- Usually, is followed immediately by more detailed discussions
- Usually, is facilitated by ScrumMaster
- Product Owner presence is optional and at Team's discretion

Sprint Review/Showcase

- Held at the end of every Sprint
- Time-boxed to no more than 2 hours (for 2 weeks sprint)
- Team demonstrates to PO and Stakeholders what was delivered; feedback is solicited
- PO decides what is accepted (based on 'DoD')
- Work shown items that don't fully meet DoD must be clearly identified to PO
- Team and PO decide what work needs to be continued in the next Sprint and what has to go back and remain in Backlog
- Product Backlog gets adjusted, based on results of Sprint Review outcome
- Timeframes, funding/budget and new market demands can be discussed at this time

Sprint Retrospective

- Usually held right after Sprint Showcase
- Time-boxed to no more than 2 hours (for 2 weeks sprint)
- Team's private meeting; others (e.g. PO) can be invited only at Team's discretion
- Usually, facilitated by ScrumMaster if he/she can be completely impartial. If SM has to participate as a team member, another facilitator is recommended
- Purpose of event, to discuss:
  - What went well in the past Sprint?
  - What did not go well in the past Sprint?
  - What can be improved in future Sprints and how to implement it?
- Team decides how to improve Definition of Done (DoD) for next Sprint

Product Backlog Refinement (PBR)

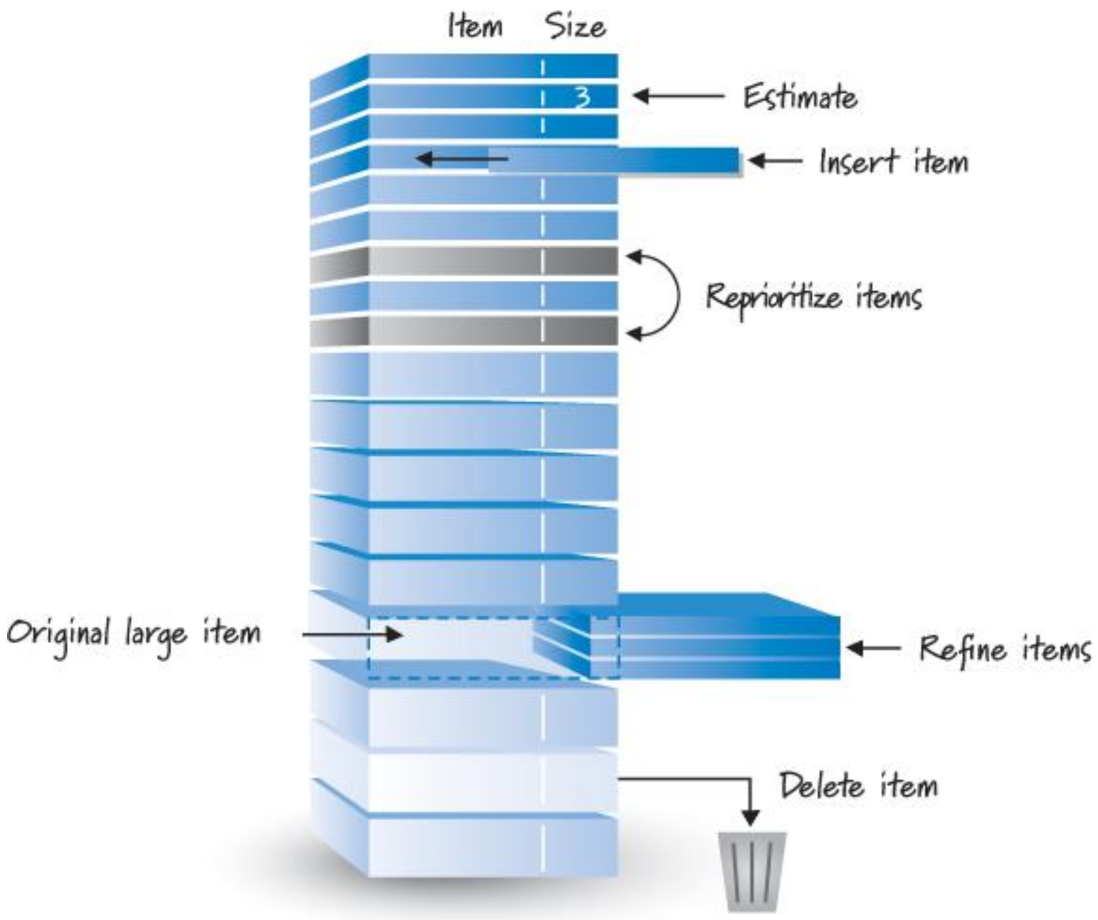


Image source: http://www.informit.com/articles/article.aspx?p=1928232&seqNum=4

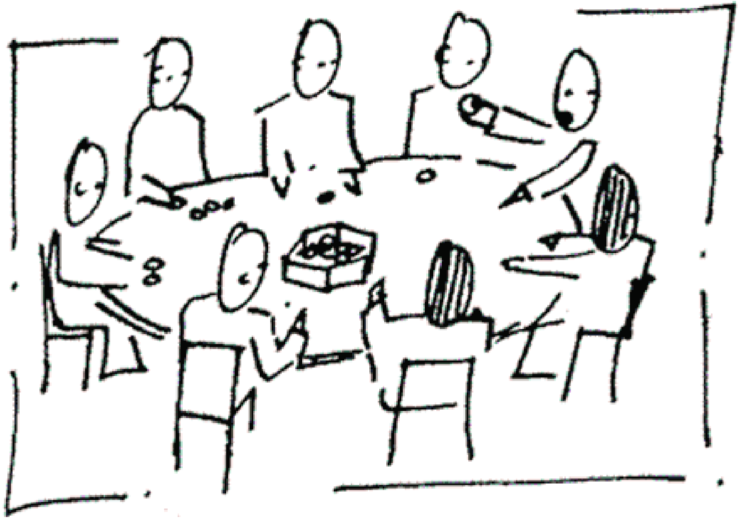
Scrum

Scrum → Scrum Ceremonies → *Product Backlog Refinement (PBR)*

Product Backlog Refinement (PBR)

Highlights:

- A.K.A.: BL Refactoring, BL Maintenance, BL Grooming (be careful with this one!)
- Objective: to prepare stories for future Sprints
- Usually 10% of each sprint (8 hours for a 2-weeks Sprint)
- Shorter & more frequent PBRs are better than on long one
- Product Owner may invite SMEs/Stakeholders to partic
  - Prioritization – by PO
  - Clarification – by PO / Stakeholders



- Breakdown of Epics to User Stories
- Estimation & re-estimation of work items
  - Epics broken into Stories
  - Stories with new discoveries
- Release Planning visualization and re-negotiation

Scrum

Scrum → Scrum Ceremonies → *Product Backlog Refinement (PBR)*

Product Backlog Refinement (PBR)

Anti-Patterns:

- Having inappropriate/uninvited attendees
- Hearing more than "one pair of vocal cords" in prioritization
- Status reporting, individual work assignments, sprint commitments
- Endless discussions, going in circles, no time-boxing
- Deep dives into "How to do work" (technical task decomposition)

Scrum

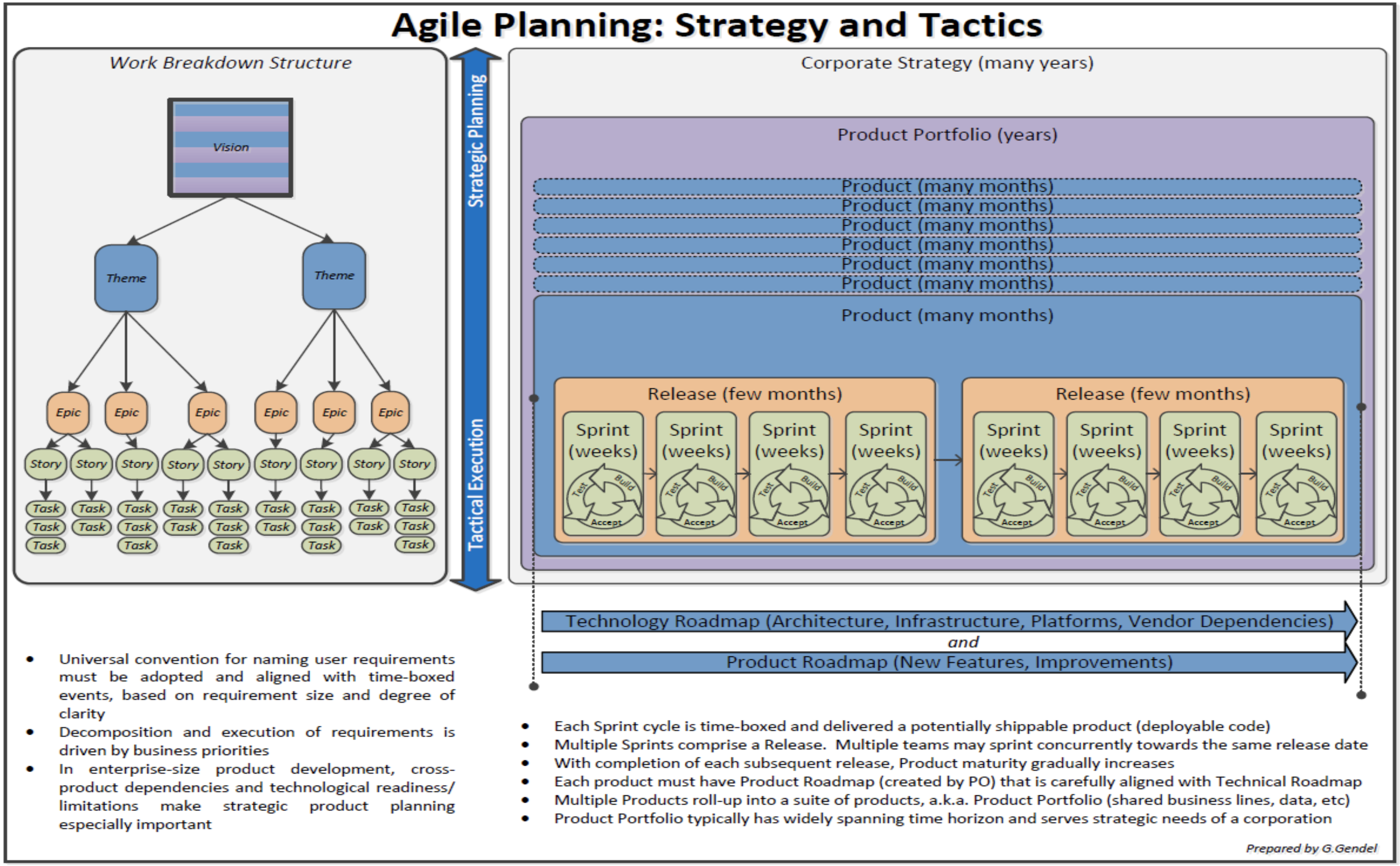
Scrum → Scrum Ceremonies → *Release Planning**

Release Planning*

- Not formally defined Scrum ceremony, since Team expected to release at the end of every Sprint
- Makes more sense to have when Team does not release at the end of every Sprint
- Is based on longer-term, overarching strategy, coming from Customers, SME, Sr. Management
- Format is similar to Sprint Planning but with broader content

Scrum → Scrum Ceremonies → *Release Planning**

Release Planning*



Agile Planning: Strategy and Tactics

- Universal convention for naming user requirements must be adopted and aligned with time-boxed events, based on requirement size and degree of clarity
- Decomposition and execution of requirements is driven by business priorities
- In enterprise-size product development, cross-product dependencies and technological readiness/ limitations make strategic product planning especially important

- Each Sprint cycle is time-boxed and delivered a potentially shippable product (deployable code)
- Multiple Sprints comprise a Release. Multiple teams may sprint concurrently towards the same release date
- With completion of each subsequent release, Product maturity gradually increases
- Each product must have Product Roadmap (created by PO) that is carefully aligned with Technical Roadmap
- Multiple Products roll-up into a suite of products, a.k.a. Product Portfolio (shared business lines, data, etc)
- Product Portfolio typically has widely spanning time horizon and serves strategic needs of a corporation

Prepared by G.Gendel

Scrum

Scrum → Scrum Ceremonies → *Release Retrospective**

Release Retrospective*

- Not formally defined Scrum ceremony, since Team expected to release at the end of every Sprint
- Makes sense to have when Team does not release at the end of every Sprint
- Is based on lessons learned over a longer period of time, during multiple Sprints
- Format is similar to Sprint Retrospective but with broader content

Scrum

Product Backlog

- Single, ordered source of requirements; ever-living, never finalized "document"
- PO is responsible for Backlog (content, availability, and prioritization): anyone can contribute to BL but only PO decides on priorities
- Team is responsible for estimation of Backlog items
- Both, Team an PO are responsible for decomposition/refinement of requirements
- Contains features, functions, requirements, enhancements, and bug fixes
- Items at the top of Product Backlog are better refined and more accurately estimated
- Items at the bottom of Product Backlog are less refined and coarse-estimated
- Creation/Management/Maintenance
- For large Product Backlogs, used by multiple teams, Segmentation/Logical Grouping is used

Sprint Backlog

- Sub-set of Product Backlog items selected for Sprint, by a single Team, based on Sprint Goals
- Forecasted/Committed by Development Team to develop a product increment, at the end of Sprint, as per DoD
- Development Team updates Sprint Backlog during Sprint to reflect work progress
- Development Team can change its Sprint Backlog during a Sprint, PO's discretion
- Completion of scope is monitored by using various techniques:
  - Sprint Burn-up/burn-down charts
  - Cumulative Flow Diagrams (CFDs)

Product Increment

- Quote from "The Scrum Guide" (page 13):

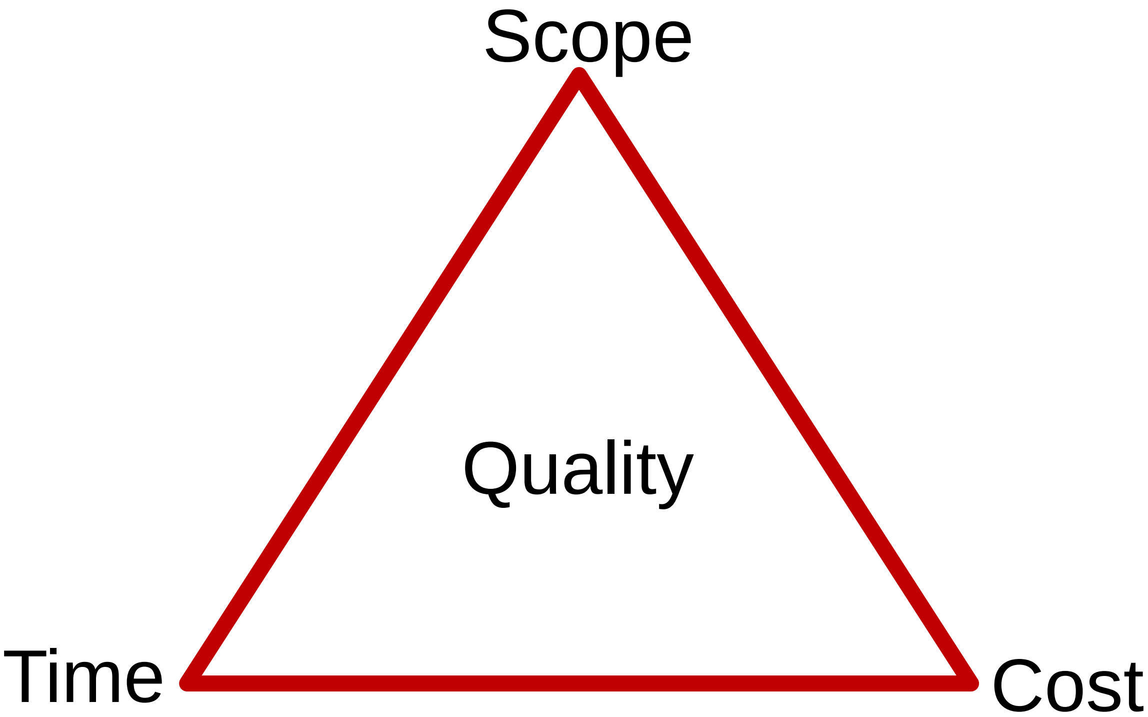
"*The Increment is the sum of all the Product Backlog items completed during a Sprint and the value of the increments of all previous Sprints. At the end of a Sprint, the new Increment must be "Done," which means it must be in useable condition and meet the Scrum Team's definition of "Done." It must be in useable condition regardless of whether the Product Owner decides to actually release it.*"

- Potentially Shippable Product Increment (PSPI)

- Minimal Viable Feature (MVF)

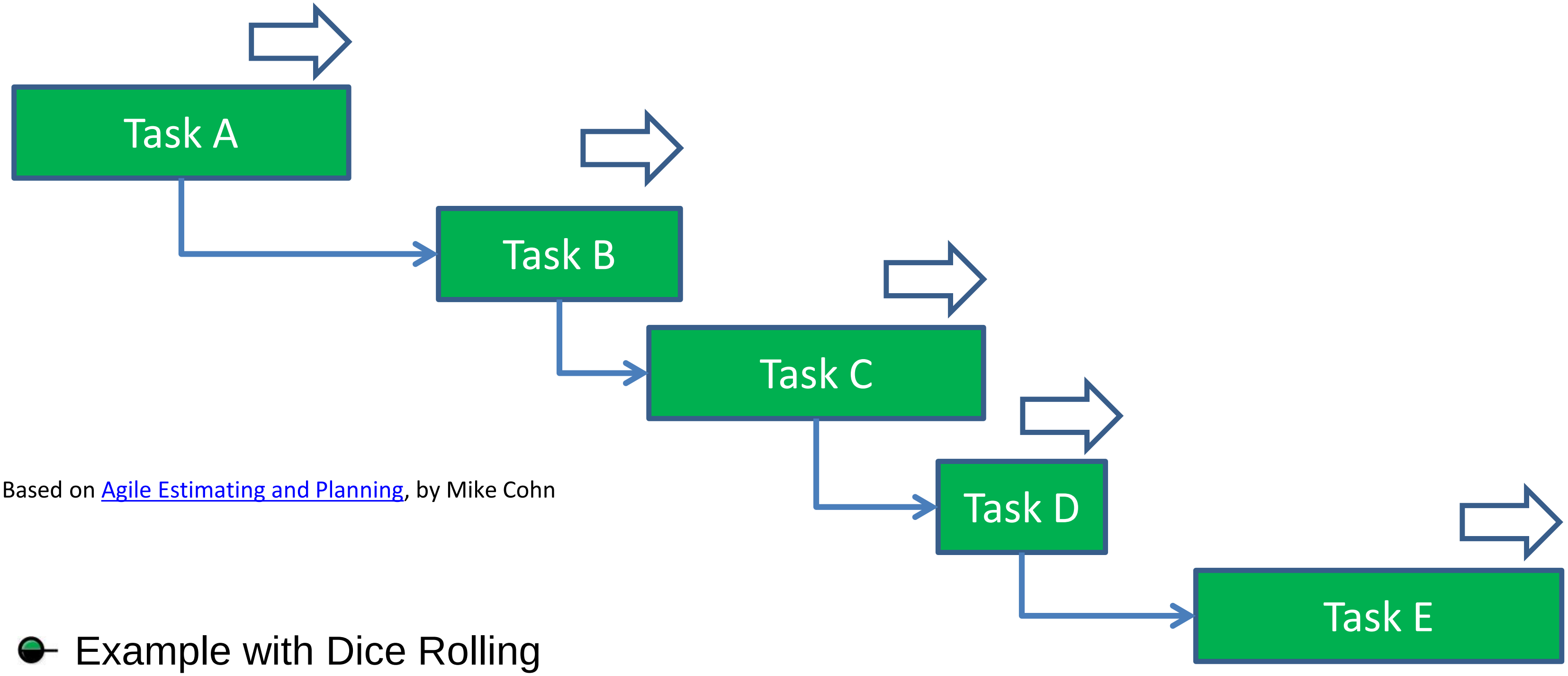
Triple Constraint Triangle of Conventional PM

- Falsely assumes Fixed Scope, Timeline and Cost
- Rigid and inflexible



Ignoring Variability and System Behaviors

Cascading Lateness

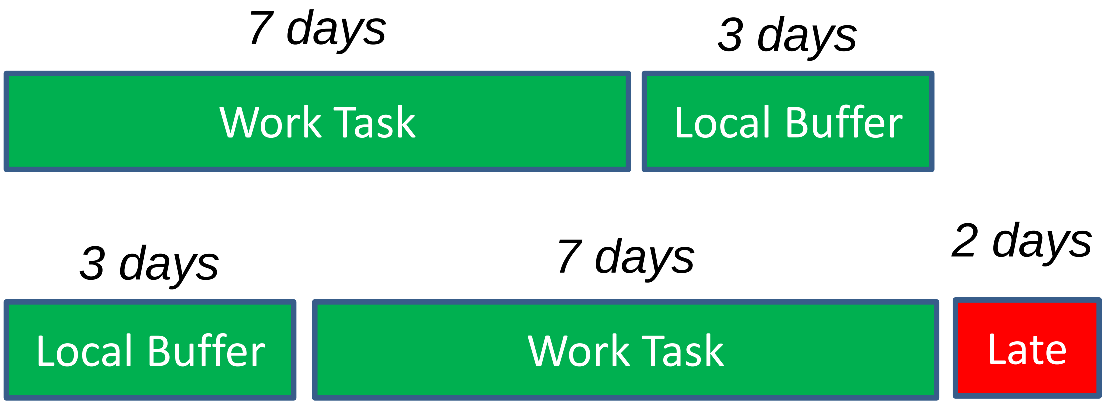


Based on Agile Estimating and Planning, by Mike Cohn

- Example with Dice Rolling
  - Normal Dice: rolling 5 Die will produce Normal Distribution graph between 5 and 30
  - Loaded Dice will produce Normal Distribution graph between 10 and 30
  - Magnetized Dice: e.g. “2” on one die always produces “2” on another die – no Normal Distribution
  - Lateness of “A” will have cascading downstream impact on B,C,D and E

Ignoring Variability and System Behaviors

Student Syndrome



- Example from student life of Mike Cohn
- Local buffer gets always used upfront and, there is no enough time at the end
- Short sprints help resolve this problem

Based on Agile Estimating and Planning, by Mike Cohn

Types of Planning

- **Velocity Driven Planning (Releases)**
  - Historical Velocity is used to determine Team’s Capacity
    - “Yesterday’s Weather”
    - Looking at Historical Average

    - Typical order of events that team goes through during planning:
      - Determine Capacity → Define Commitment → Acquire confidence

        another way to look at this:
      - Story Selection → Task Breakdown → Hour estimation of Tasks

- **Commitment Driven Planning (Iterations)**
  - Determining Capacity in hours for Individuals and Skillset
  - Taking one Story at a time from the top of Backlog and asking yourself: “can we commit to it?”
  - Overly ambitions commitment leads to dropping hours mid-sprint
  - The goal is to understand how many hours of work Team can commit to in one Sprint

Based on Agile Estimating and Planning, by Mike Cohn

User Story writing

What User Stories Are NOT?

- System-centric Functional Specifications
- User-centric Business Requirements
- User –centric Use Cases
  - …There are some exceptions:
    - A single (happy path) Use Case
    - Essential Use Case, stripped of any hidden assumptions about technology or implementation
- Scenarios (usually ~~ multiple stories)
- Screen Design (UI) – It is a Trap!!!

User Story writing

Stories Must be “INVEST”-able

- Independent - as small as possible, without making it dependent on other Stories
- Negotiable - are not meant to be precise, detailed specifications of a feature. They should not be fixed in stone
- Valuable - should be of value to the user
- Estimatable - useful way to measure whether a User Story is good or not
- Small – small enough to be fully developed & tested in one sprint
- Testable - has to do with size and clarity of a story

User Story writing

How to Formulate a Story?

1 - "As <User>, I want to <action>, so that <result>

or

Cucumber format

- **GIVEN** ~~ putting the system in a known state
- **WHEN** ~~ describing the key action the user performs
- **THEN** ~~ observing outcomes

**Cucumber is a tool for running automated acceptance tests written in a behavior driven development (BDD), which is a software development process based on test-driven development (TDD).*

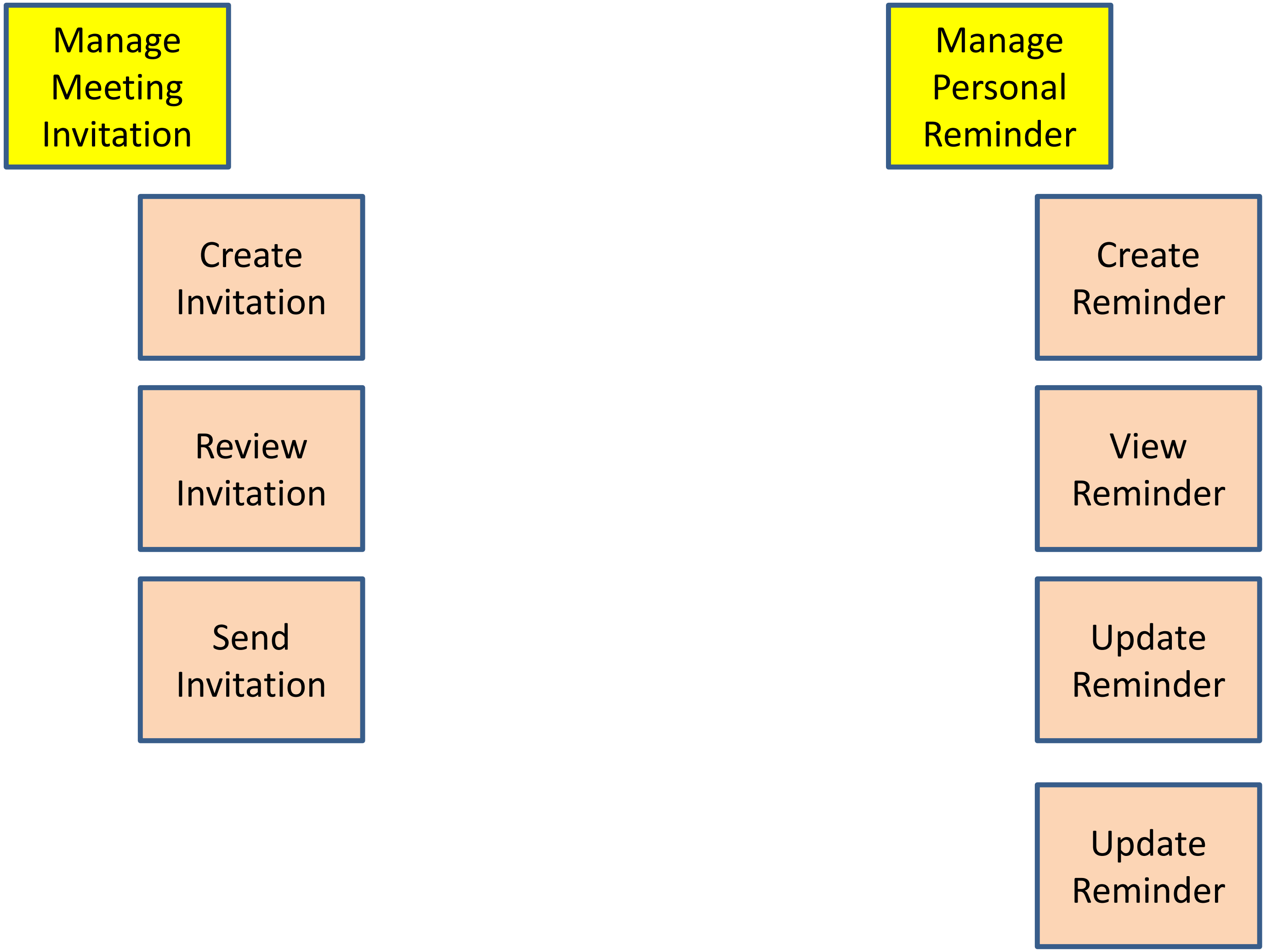
User Story writing

Story Splitting: Done as a part of Backlog Grooming or Sprint Planning

- Large stories (Epics) can be of two types:
  - Compound Story
    - Consists of multiple shorter stories
    - Easily decomposable
    - Characterized by a large volume of work
  - Complex Story
    - Inherently large
    - Cannot be easily disaggregated into smaller stories
    - Can be split into Investigative (spike) and Development stories
    - Characterized by low level of understanding

User Story writing

Story Splitting: Compound Stories



Manage Meeting Invitation

Create Invitation

Review Invitation

Send Invitation

Manage Personal Reminder

Create Reminder

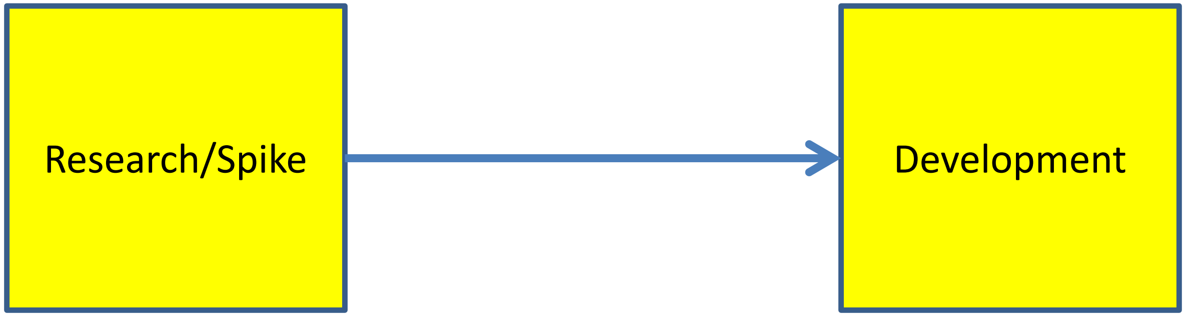
View Reminder

Update Reminder

Update Reminder

User Story writing

Story Splitting: Complex Stories



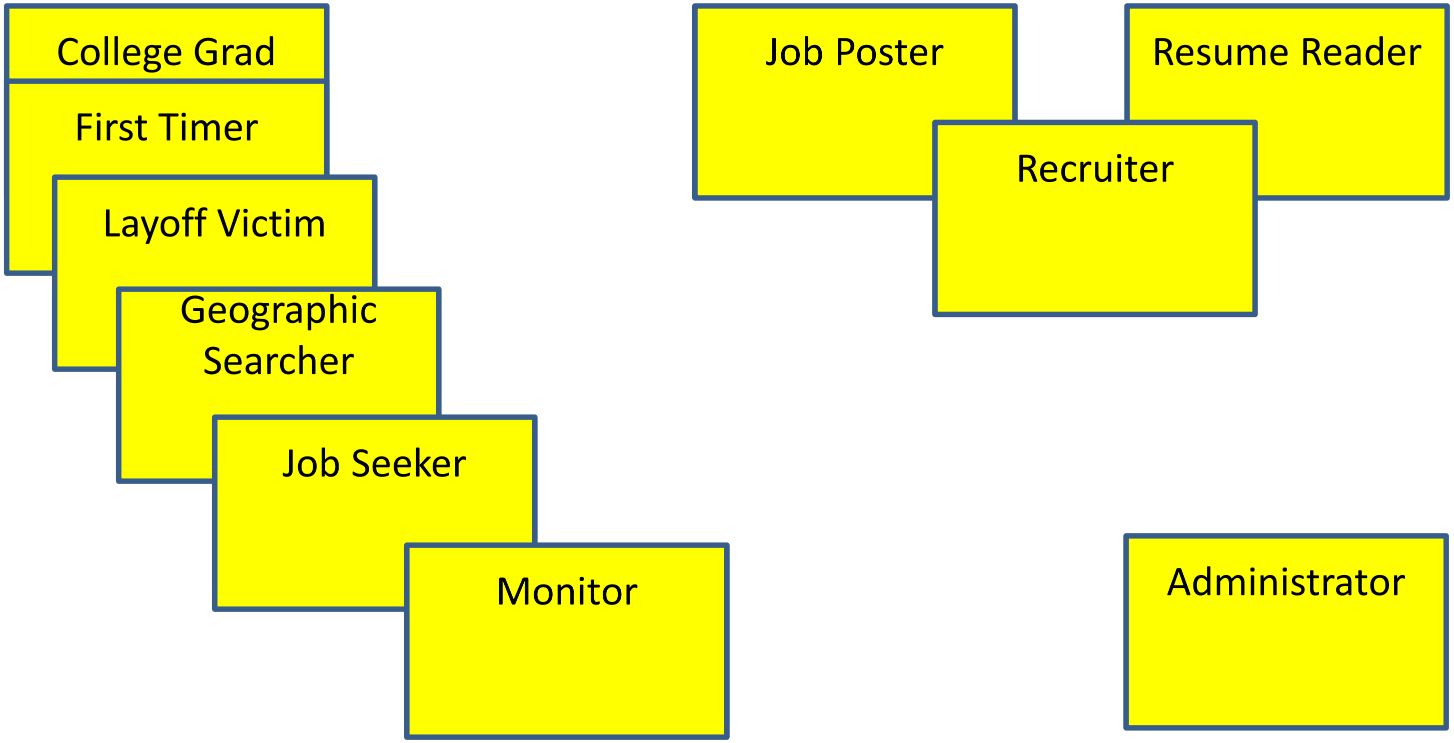
User Story writing

Story Combining: required for more accurate estimation and planning

- Small bug fixes can be combined into user stories
- UI changes
- If technology thinks that stories are too granular, combine them:
  - "A user can add and edit personal contact information"
  - "A user can add and edit education information"
  - "A user can add and edit working history information"
  - "A user can add and edit compensation information"
  - "A user can add and edit hobbies information"
  - "A user can add and edit references information"

User Story writing

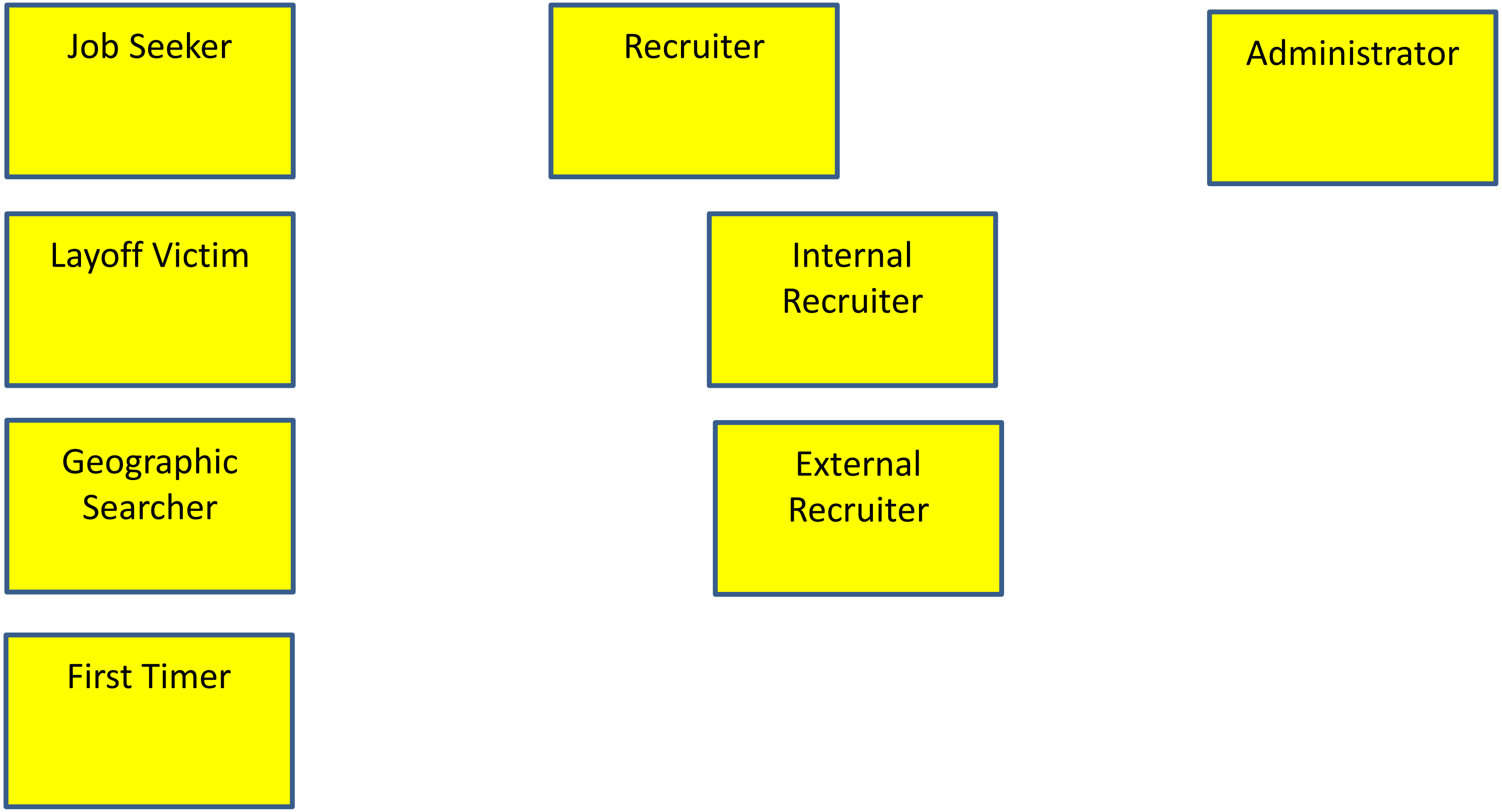
User Role Modelling: organizing User Roles for www.MoneyJos.com . Discover Level of relevance/overlapping



Scrum

User Story writing

User Role Modelling: Group-by, based on relevance/overlapping



Job Seeker

Layoff Victim

Geographic Searcher

First Timer

Recruiter

Internal Recruiter

External Recruiter

Administrator

User Story writing

User Role Modelling: Refining Roles

- Frequency of usage
- User’s level of expertise with domain
- User’s general level of proficiency with PC/software
- User’s general goal for using software

User Story writing

User Role Modelling: Refining Roles

Personas – an imaginary representation of a user role

- Not just providing a name (e.g. John Doe)
- Need explicit description of what this person is and what will be his/her usability pattern from day-to-day
  - "As John Doe…" is better than "As Market Editor…", which is yet better than "As User…"

Extreme Characters

- Very unique system users
- Examples:
  - POPE John
  - Barack Obama
  - Jay Leno (very unique, thanks to his chin!)

User Story writing

User Story Testing:

- Writing Acceptance Criteria before writing Tests
- Writing Tests before Coding
- Customers should specify Tests
- Testing Types
- UI testing
- Usability testing
- Performance testing
- Stress testing

User Story writing

Converting Defects to Stories

- High Priority Functional Defects ~~ keep as defects
- Low Priority Functional Defects ~~ candidates for user story conversion (low business value)
- Any cosmetic defects ~~ combining into one user story
- Prioritize defects along side with user stories
- Always keep track of true defect ratio to understand how much time, on average, a team spends to fix bugs
  - Monitor Defect Density (bug/story) and Defect Trends (open vs. closed)
  - Economics of deprecating Technical Debt are important
    - Tasking out defects to monitor efforts spent

User Story writing

Slicing Stories

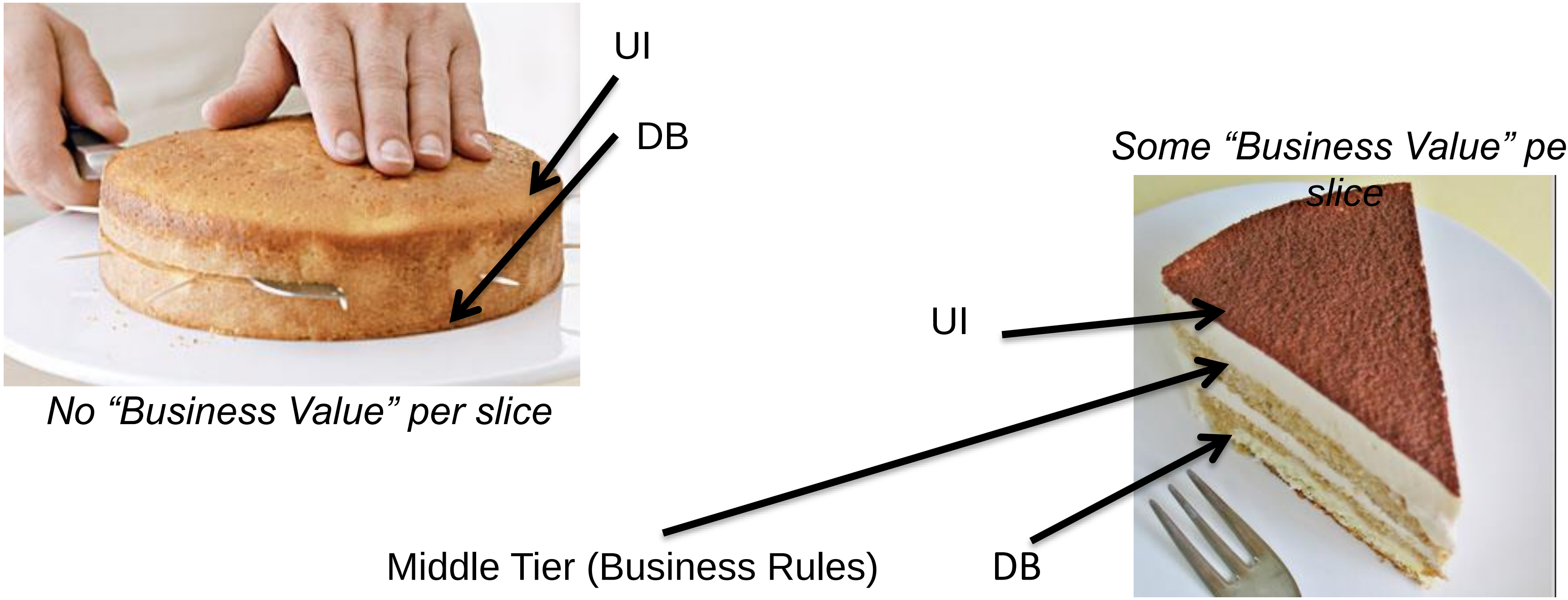
- Starting with Goals ~~ Product Roadmap level
    - Come down to Epics ~~ Large stories
    - Break down to Stories ~~ executable in a single sprint
    - Break down further to Tasks ~~ individual technical work

- Slicing the Cake VERTICALLY, not HORIZONTALLY
    - **Bad** slicing (often dev-driven):
        - “A Job Seeker can fill out a resume form…”
        - “Information on a resume form is saved to DB”

    - **Good** slicing (PO-driven):
        - “A Job Seeker can fill out a resume form that includes only basic information (name, address, education)”
        - “A Job Seeker can fill out a resume form that ALL of the information that an employer wants to see (employment history)”

User Story writing

Slicing Stories



UI

DB

No "Business Value" per slice

Some "Business Value" per slice

UI

Middle Tier (Business Rules)

DB

User Story writing

More guidelines for writing User Stories:

- Story that is **not closed-ended (bad)**:
  - “A recruiter can manage the ads she has placed”

- Stories that are **closed-ended (good)**:
  - “A recruiter can review resumes from applicants to one of her ads”
  - “A recruiter can change the expiration date of an ad”
  - “A recruiter can delete an application that is not a good match for a job”

User Story writing

More guidelines for writing User Stories:

- Put Constraints on Story cards
  - Example: "The system must support peak usage of up to 50 concurrent users"
- Write/size stories based on the Horizon
  - Near horizon – smaller stories
  - Far horizon – larger stories
- Keep the UI Out as Long as Possible
  - UI design is not an independent story it is a part of a story that brings a function
  - Use artifacts other then stories (e.g. capture UI design, e.g. UI guidelines document), if needed
- Include User Roles in Stories
  - "As user…" is not as good as "As Job Seeker…"

User Story writing

More guidelines for writing User Stories:

- Write a story in Active Voice:
  - “A Job Seeker can post a resume” is better then “A resume can be posted by a Job Seeker”

- Write a story for One User for limited set of actions
  - By doing so, avoid confusion and shorten Acceptance Criteria
  - Examples:
    - VAGUE: “Job Seekers can remove resumes from the site”
    - BETTER: “Job Seeker can remove resumes”
    - BEST: “A Job Seeker can remove her own resumes”

User Story writing

More guidelines for writing User Stories:

- Consistency in User Story writing
  - Style of writing
    - Description
    - Acceptance Criteria
    - UI attachments
  - Story Size at the time a team commits
    - Consistency over time between story complexity and size (hours)
    - **Triangulation** during story estimation

User Story writing

More guidelines for writing User Stories:

- Consistency in User Story writing
    - Style of writing
        - Description
        - Acceptance Criteria
        - UI attachments
    - Story Size at the time a team commits
        - Consistency over time between story complexity and size (hours)
        - **Triangulation** during story estimation

User Story writing

User Stories vs. Use Cases

- Use Cases
  - Use cases describe User ←→ System interaction
  - Can be written from End User or System perspective
  - Can be simple or complex
  - Large use cases do not require decomposition to small cases for development
    - Are not always suitable for incremental/iterative development
  - Have happy paths, alternative paths
  - Might be *include*, *extend*, etc, in nature

- User Stories
  - Are End-user centric and written from End User perspective
  - Can be simple or complex
  - Must be broken small for effective incremental development
  - Ideally, should be free of extremal or upstream dependencies

Complex Use Cases can be broken down into multiple User Stories

User Story writing

User Story Mapping

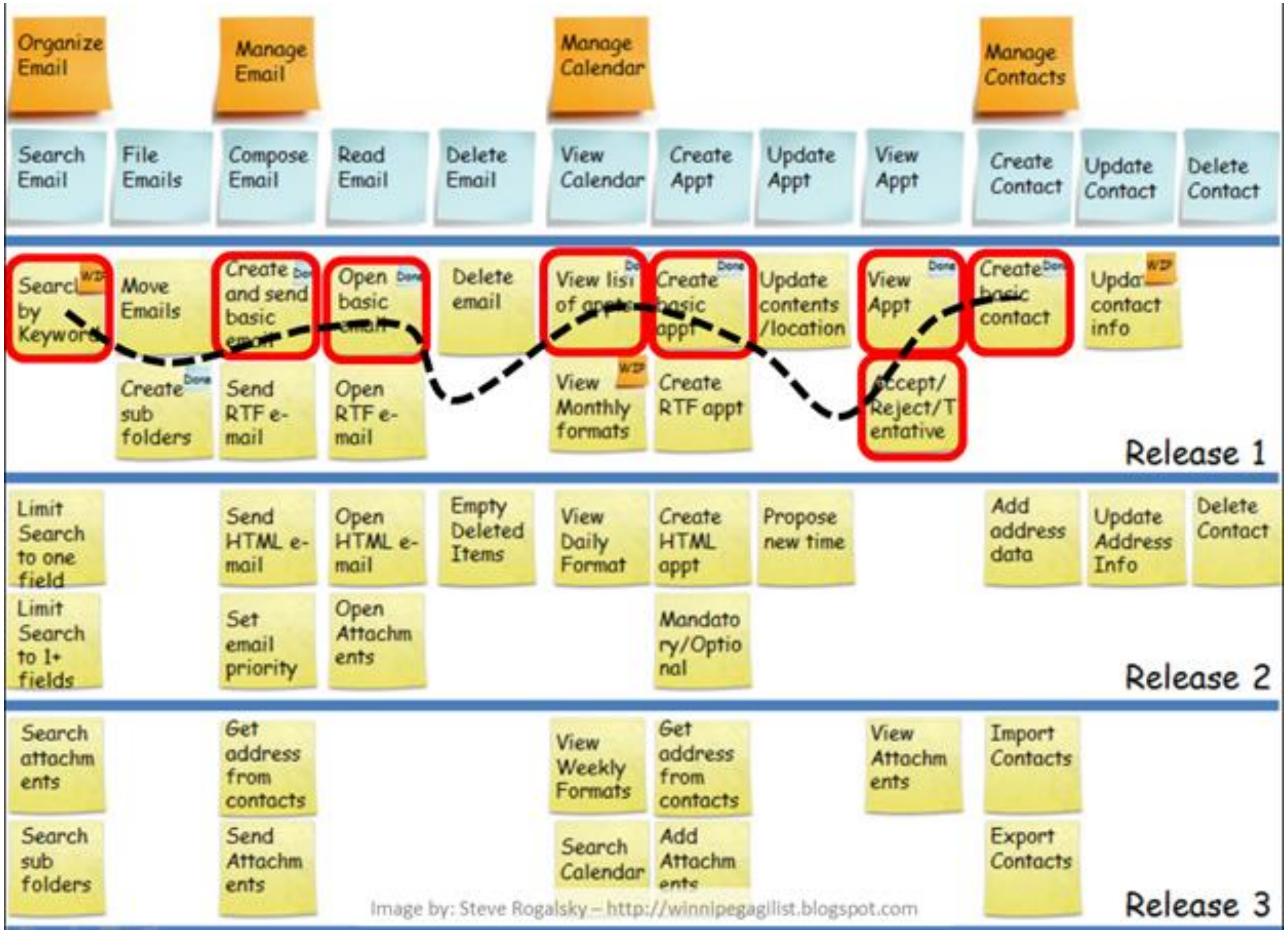
- Effective for brainstorming/JAD sessions
- Frequently used when User Story describe a complex system process (e.g. workflow)
- Effective way to expose dependencies and establish priorities
- Effective way to retain overall, holistic view of the system after story decomposition
- Examples of mapping dimensions
  - Timeline vs. Priority
  - Functionality vs. Priority
  - Functionality vs. Timeline

Scrum

Scrum → Scrum “Under the Hood” → Agile Requirements Management → *User Story writing*

User Story writing

User Story Mapping

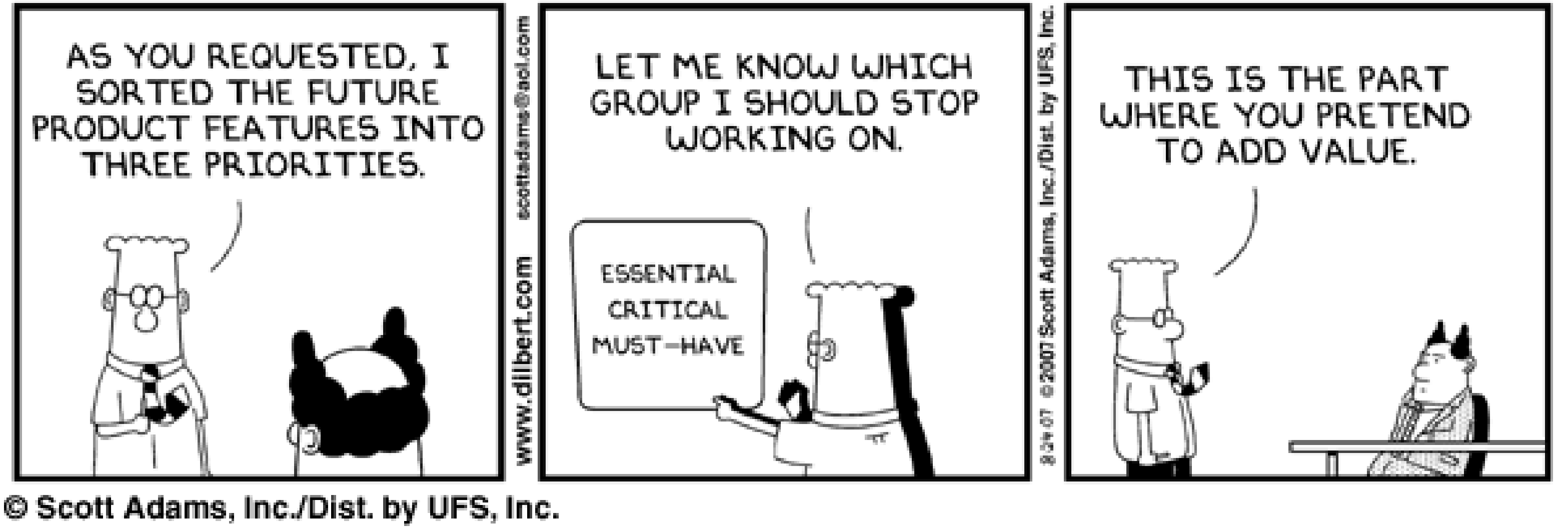


Source: http://winnipegagilist.blogspot.com/2014/11/story-maps-testing-tool-after-all.html

Work Prioritization

- Always relative

- Techniques
  - **MoSCoW: Must** have**, Should** have**, Could** have**, Would have**
  - Business Value: 0-100
  - Stack Ranking: automatic


AS YOU REQUESTED, I SORTED THE FUTURE PRODUCT FEATURES INTO THREE PRIORITIES.
scottadams@aol.com
www.dilbert.com
LET ME KNOW WHICH GROUP I SHOULD STOP WORKING ON.
ESSENTIAL
CRITICAL
MUST-HAVE
8-24-07 ©2007 Scott Adams, Inc./Dist. by UFS, Inc.
THIS IS THE PART WHERE YOU PRETEND TO ADD VALUE.

© Scott Adams, Inc./Dist. by UFS, Inc.

Work Estimation & Planning

- Types of Estimation
    - **By Analogy** – comparing one work item to two others, with one other item being bigger an another one – smaller. (*Example, comparing one Harry Potter book to two others)*
    - **By Calculation** – Determining Size of Work, figuring out Pace (e.g. pages per hour) and then deriving Duration, by dividing Size of Work by Pace
- Base-lining work (using reference samples)
- Order Magnitude (adjusting estimation scale)
- Estimation units
    - Story Points – *relative* units of measure; an effective way to estimate Product Backlog
    - Ideal Days/Hours – *precise* units of measure; a more reliable way to estimate Spring Backlog
        - Even more appropriate for novice Teams (low maturity, low T-shaping)

Work Estimation & Planning

Work Volume vs. Work Complexity

- Both are taken into account when estimating in Story Points:
  - Volume defines size of work
  - Complexity, as ~ Potential "Complication", may influence Team's ability to complete the same volume of work

Work Estimation & Planning

Effects of Anchoring

- Inability to make one's own independent, unbiased, uninfluenced, objective decisions

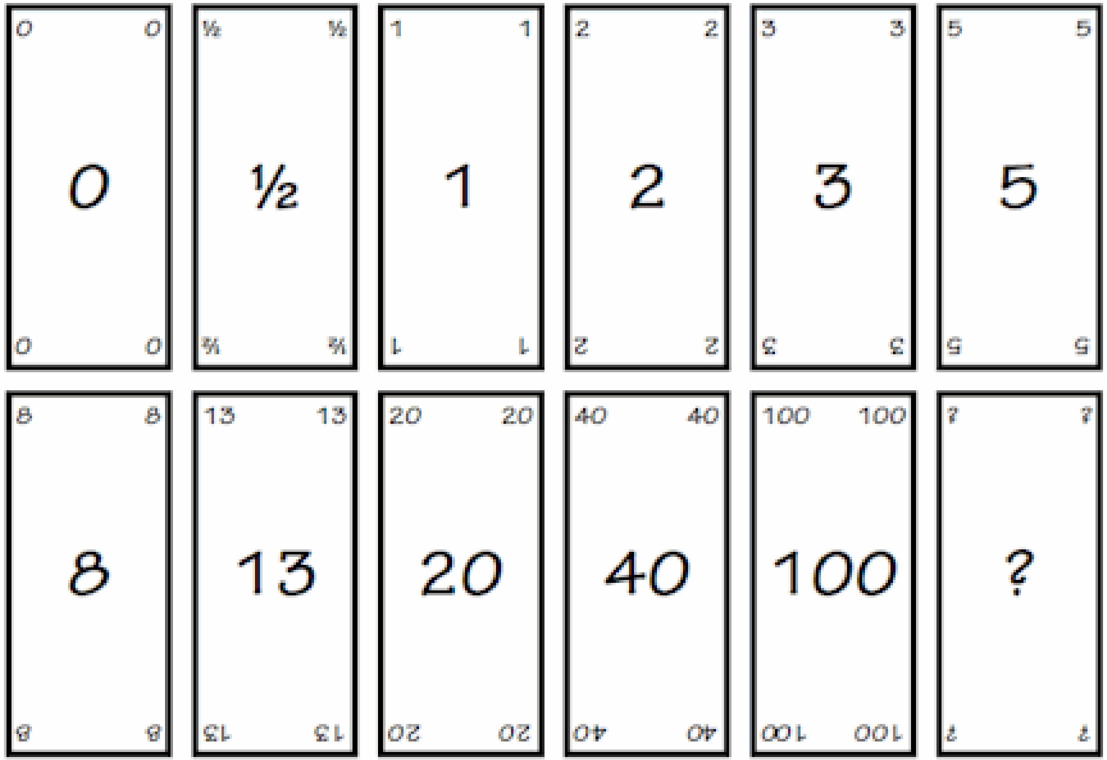
- Caused by:
  - Lack of strong knowledge
  - Pressure: peer, customer, management
  - Status Quo

- Can be avoided by:
  - "Military Democracy" ("privates speak before generals")
  - Using discrete estimation techniques (e.g. Planning Poker)

Work Estimation & Planning

Planning/Agile Poker

- Objective, unbiased; removes anchoring
- Encourages discussions
- Uses various scales:
  - **Fibonacci Sequence**
  - Arithmetic Sequence
  - T-shirt sizes
  - Dog breeds



Scrum

Work Estimation & Planning

Individual Capacity Management

Team Member	Primary Skills	Secondary Skills	Sprint Capacity
John	Java, Flex	SQL, BA, Ruby	60 hours
Jeff	Ruby on Rails, Flex	SQL, VBScript, Java	55 hours
Jill	Java, .Net, Pascal	C++, C#, Java, Flex	65 hours
Jim	Flex, Java, .Net	DHTML, Cold Fusion	55 hours
Julie	Cobol, Fortran	Pascal, SQL	65 hours
Total			**295 hours**

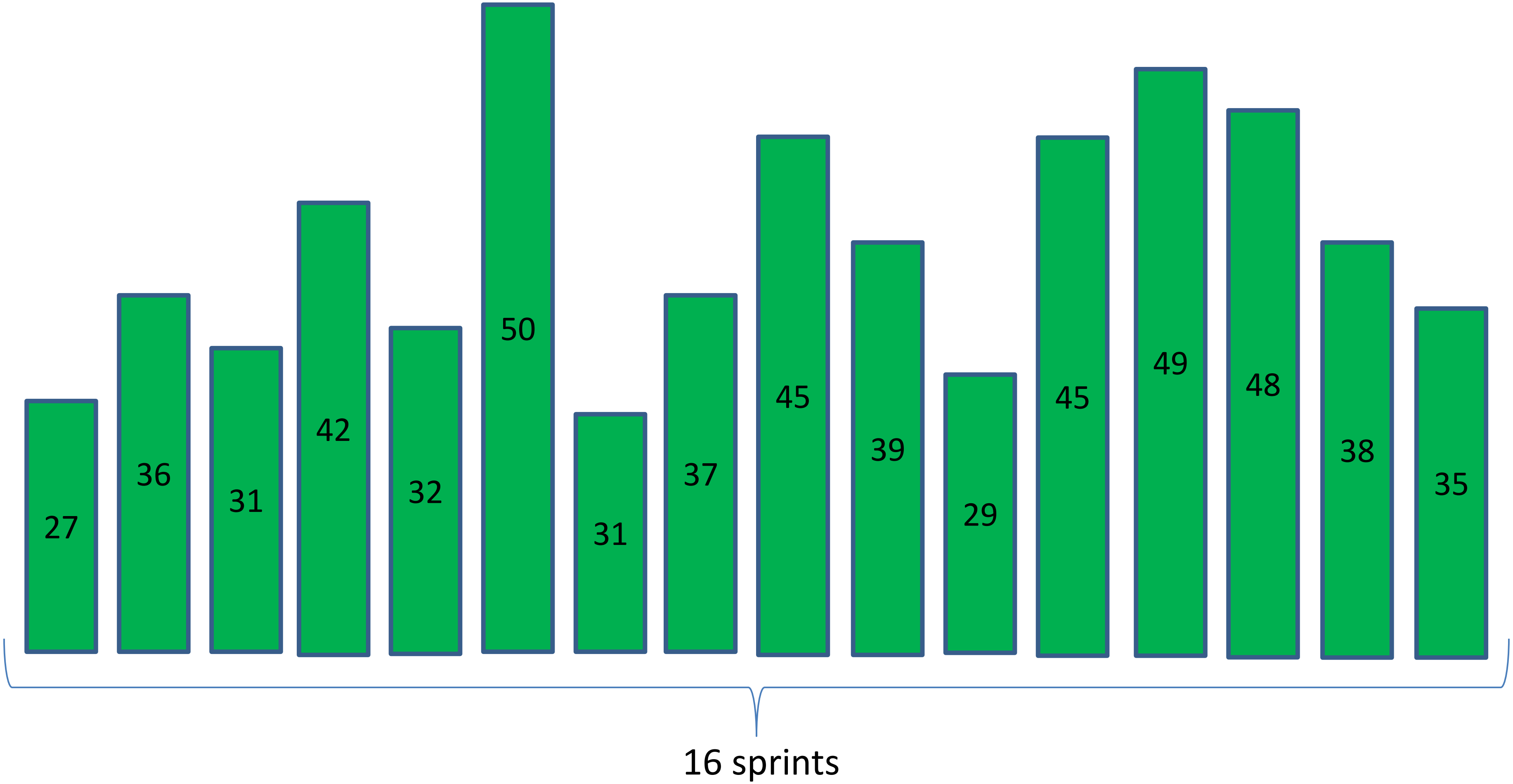
Work Estimation & Planning

Skillset Capacity Management

Skillset	Capacity
Java	150 hours
Flex	50 hours
Ruby	30 hours
.Net	30 hours
Cobol, Fortran, Pascal	35 hours
Total	**295 hours**

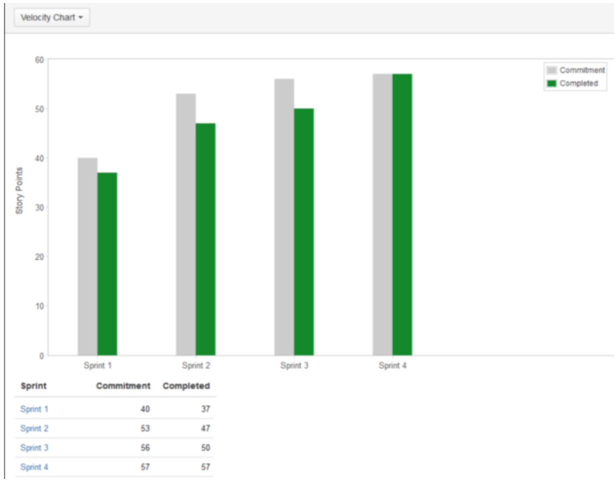
Work Estimation & Planning

Historical Velocity Analysis

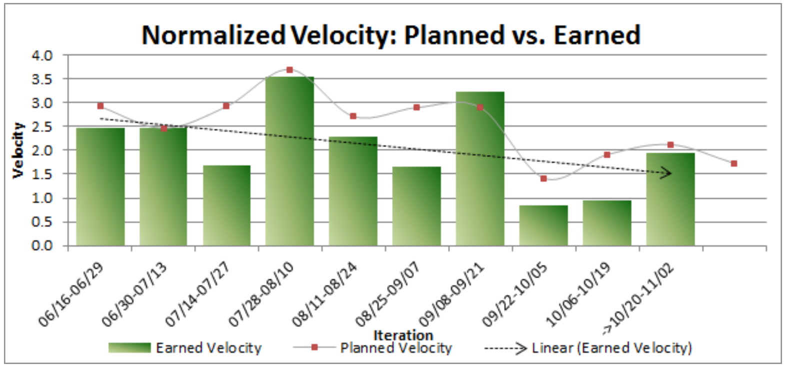


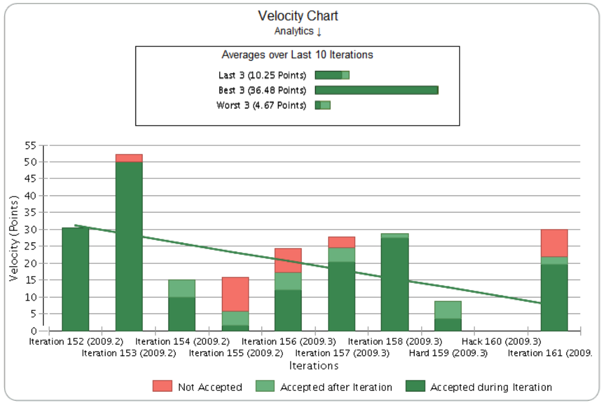
Work Estimation & Planning

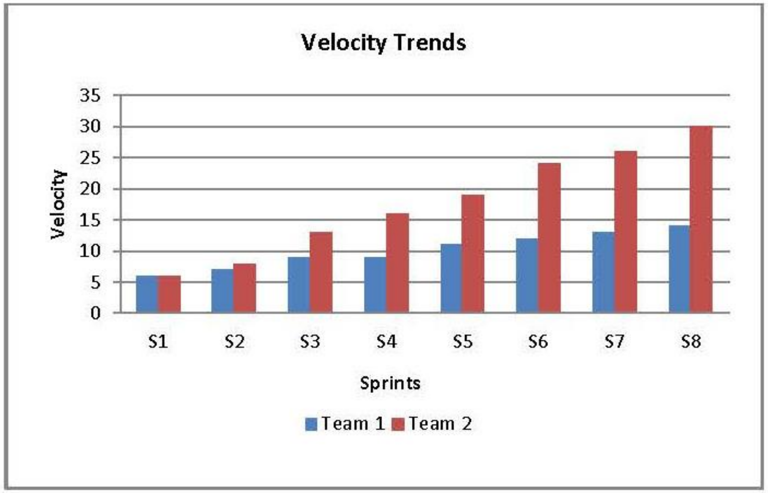
Historical Velocity (Real Life Examples)



Sprint	Commitment	Completed
Sprint 1	40	37
Sprint 2	53	47
Sprint 3	56	50
Sprint 4	57	57

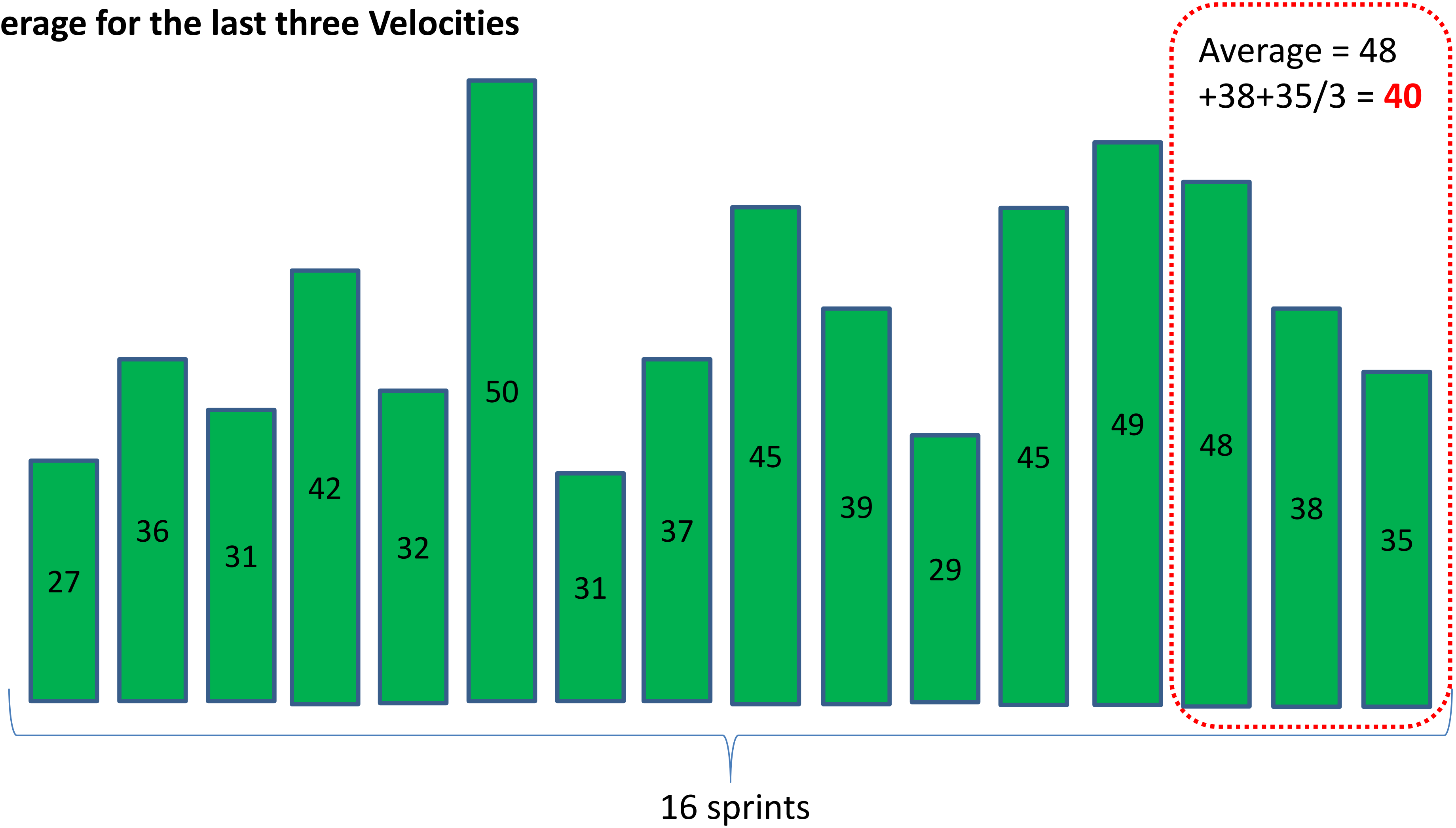






Work Estimation & Planning

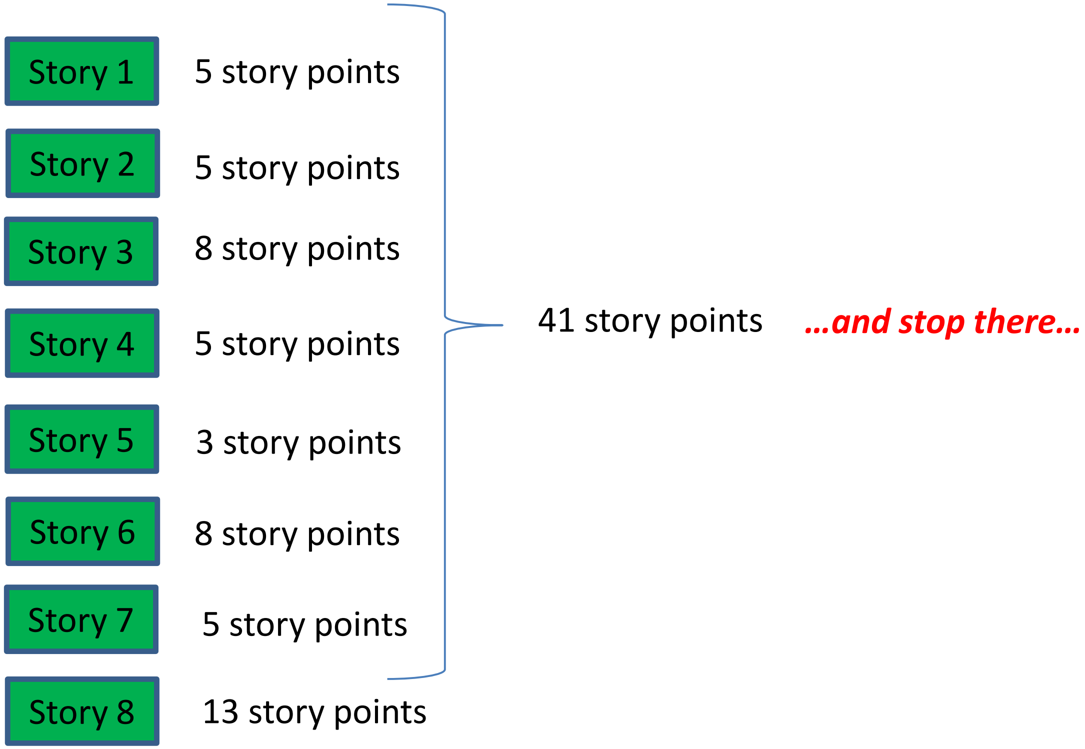
Average for the last three Velocities



Average = 48 +38+35/3 = **40**

Work Estimation & Planning

Identifying Stories-Candidates

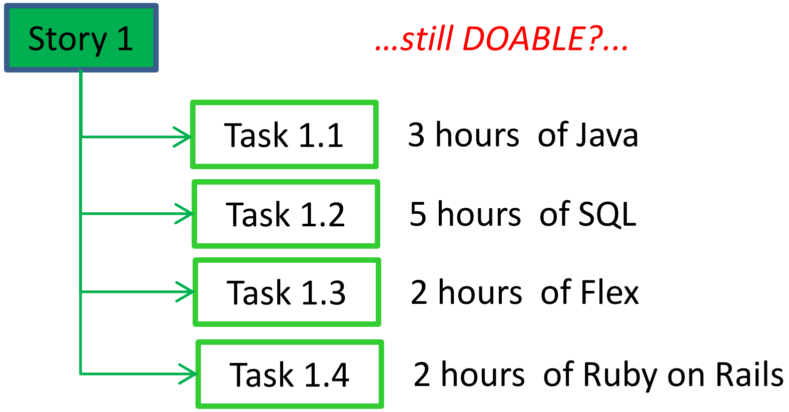


Story 1	5 story points
Story 2	5 story points
Story 3	8 story points
Story 4	5 story points
Story 5	3 story points
Story 6	8 story points
Story 7	5 story points
Story 8	13 story points

41 story points ***…and stop there…***

Work Estimation & Planning

Capacity-Driven Srint Planning: Breaking Stories into Tasks



Story 1

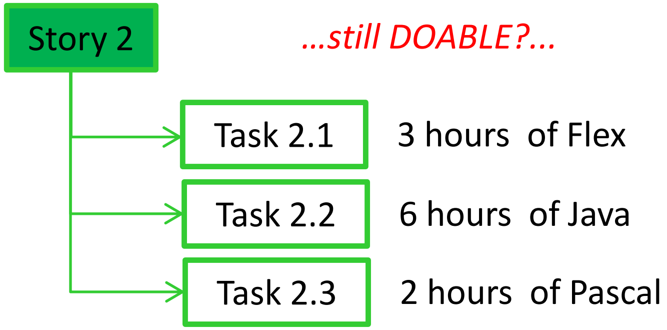
...still DOABLE?...

Task 1.1 3 hours of Java

Task 1.2 5 hours of SQL

Task 1.3 2 hours of Flex

Task 1.4 2 hours of Ruby on Rails



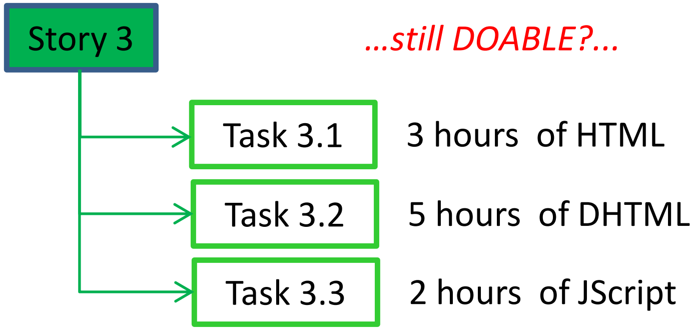
Story 2

...still DOABLE?...

Task 2.1 3 hours of Flex

Task 2.2 6 hours of Java

Task 2.3 2 hours of Pascal



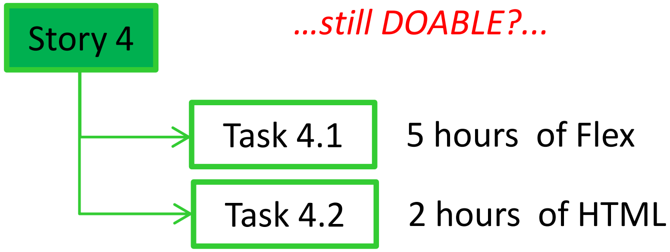
Story 3

...still DOABLE?...

Task 3.1 3 hours of HTML

Task 3.2 5 hours of DHTML

Task 3.3 2 hours of JScript



Story 4

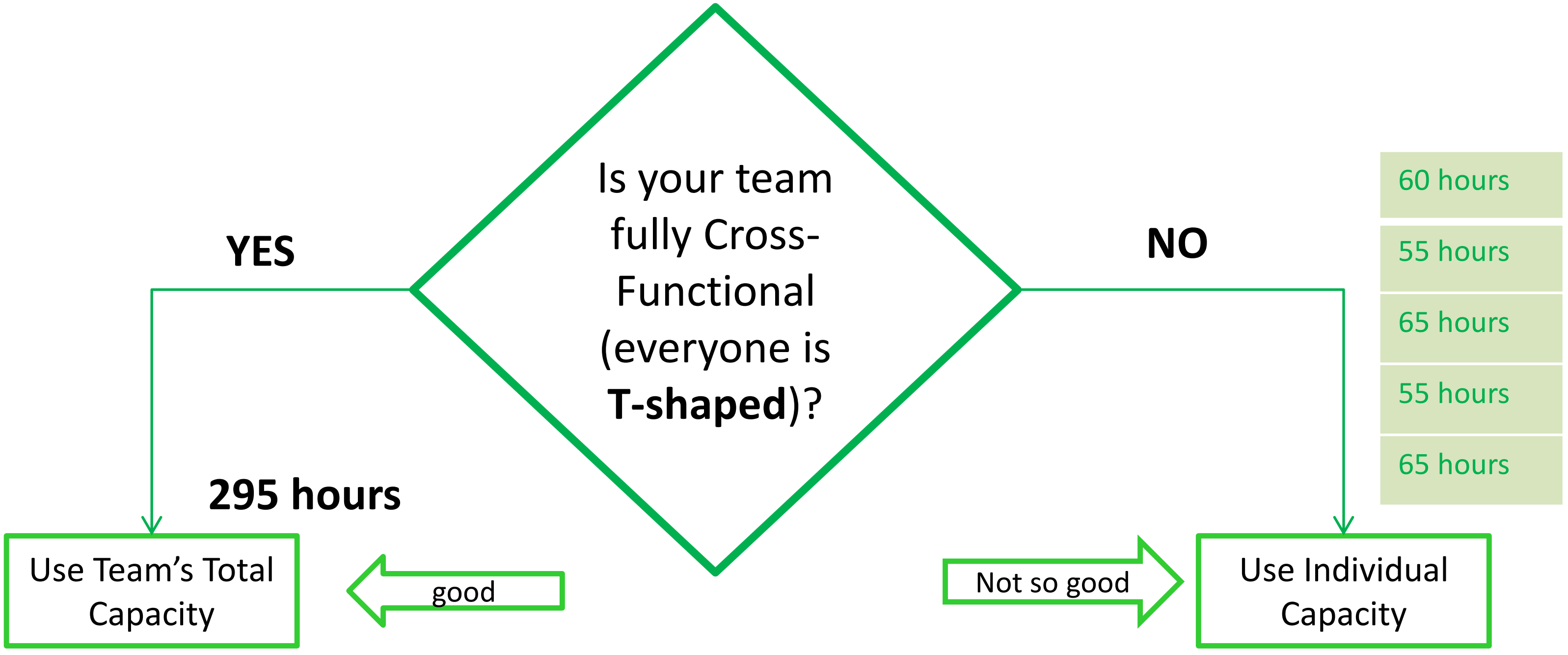
...still DOABLE?...

Task 4.1 5 hours of Flex

Task 4.2 2 hours of HTML

Work Estimation & Planning

Capacity-Driven Planning: two options

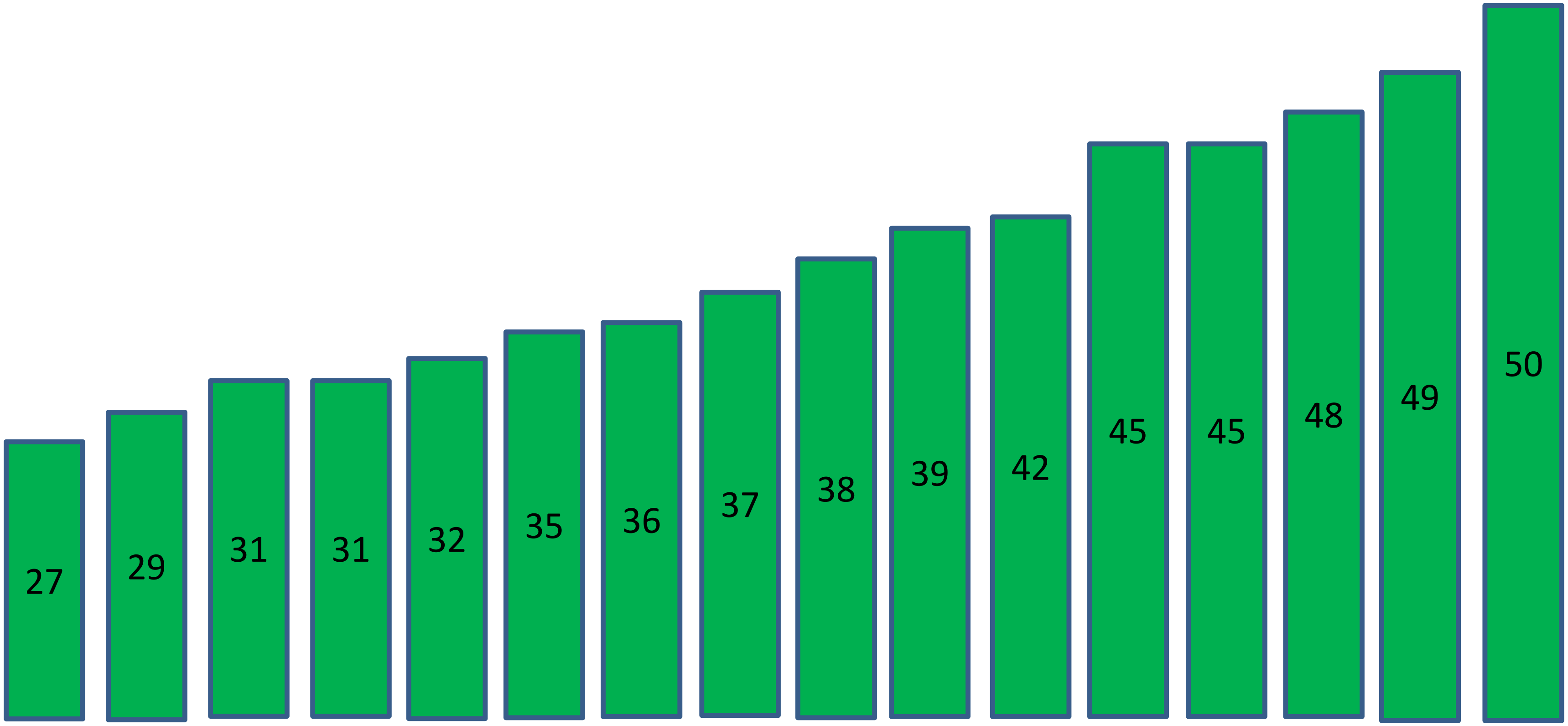


…swarming and collective ownership is expected…

…unfortunately, need to consider who is doing what…

Work Estimation & Planning

Historical Velocity Analysis



Work Estimation & Planning

Discarding Outliers when Data Pool is Large

Number of Data Sets	Number Outliers to Drop on Each End
0-7	0
8-10	1
11-12	2
13-15	3
16-17	4
18-20	5
21-22	6
23-25	7
26+	8

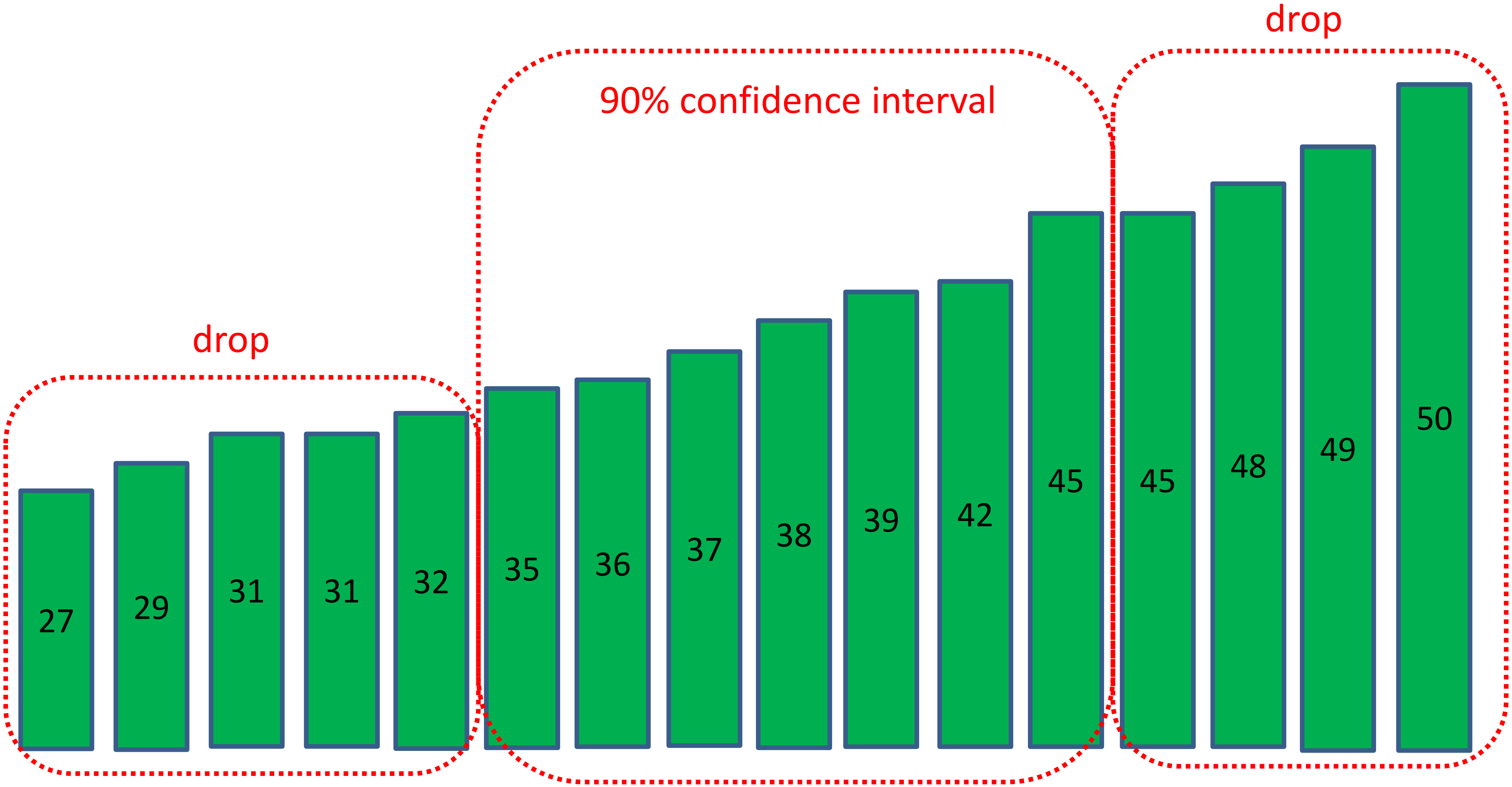
Based on Confidence Interval around the Median using Binomial Distribution

Based on Agile Estimating and Planning, by Mike Cohn

Scrum

Work Estimation & Planning

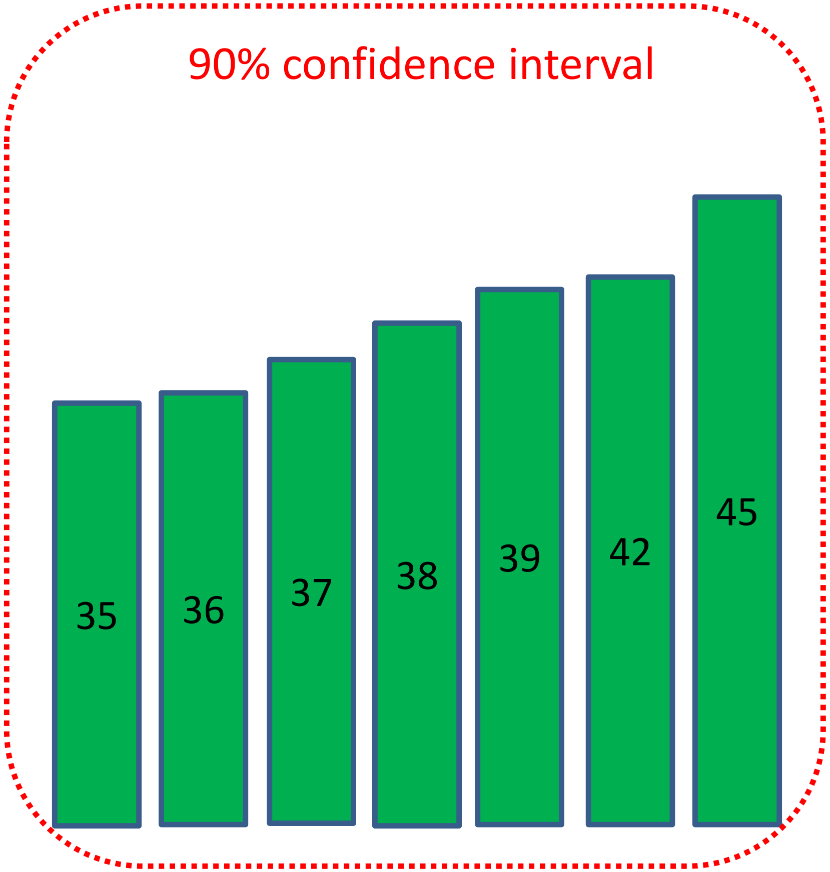
Historical Velocity Analysis – getting rid of Outliers



Work Estimation & Planning

Historical Velocity Analysis – getting rid of Outliers

"Official" Low Velocity = 35



"Official" High Velocity = 45

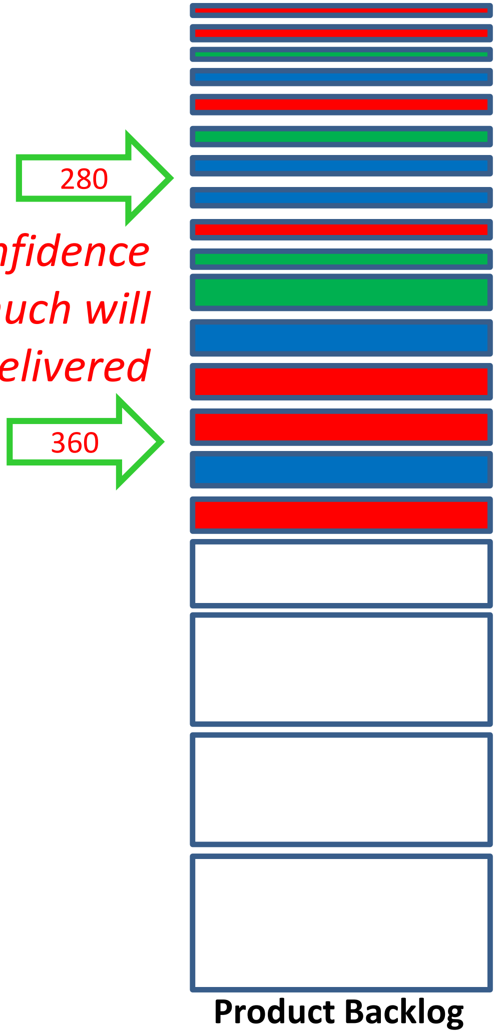
Work Estimation & Planning

Historical Velocity Analysis – forecasting Backlog penetration

If you have a release in 4 months and your sprint is 2 weeks in duration, it means that you have 8 sprints in total

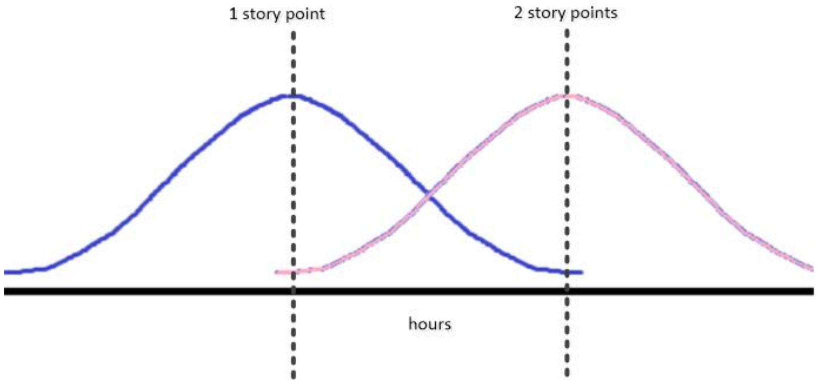
8 Sprints x "Official" Low Velocity of 35 = **280** story points (pessimistic projection)

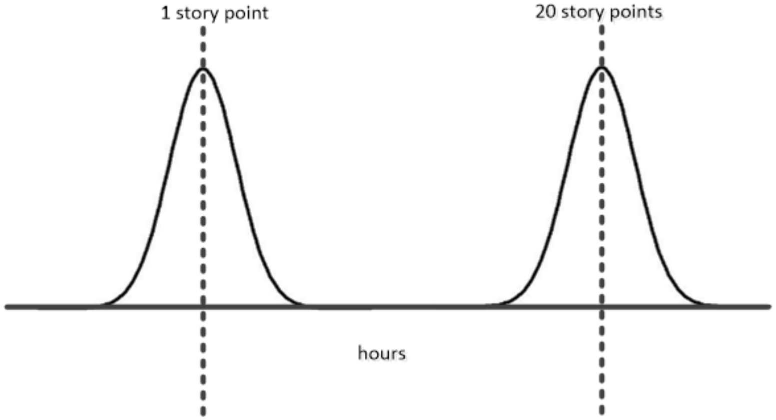
8 Sprints x "Official" High Velocity of 45 = **360** story points (optimistic projection)



Work Estimation & Planning

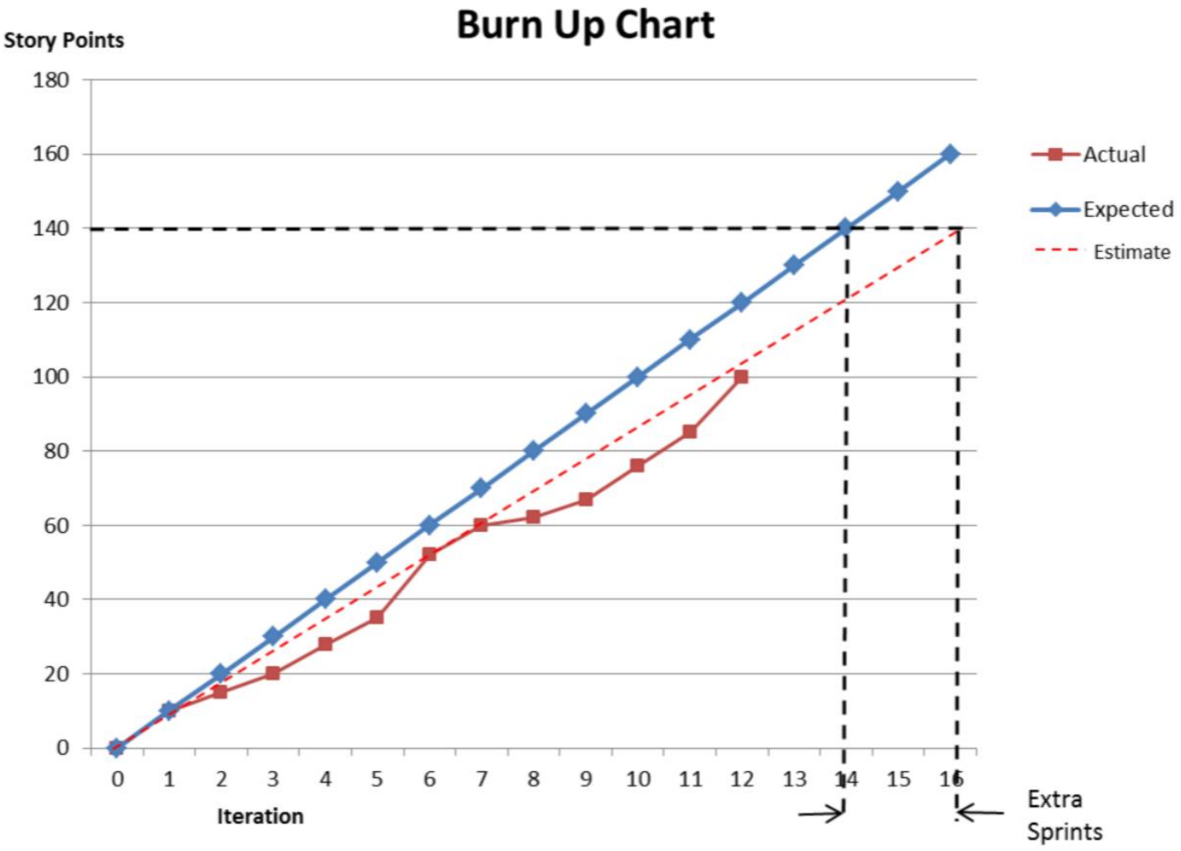
Relating Points to Hours





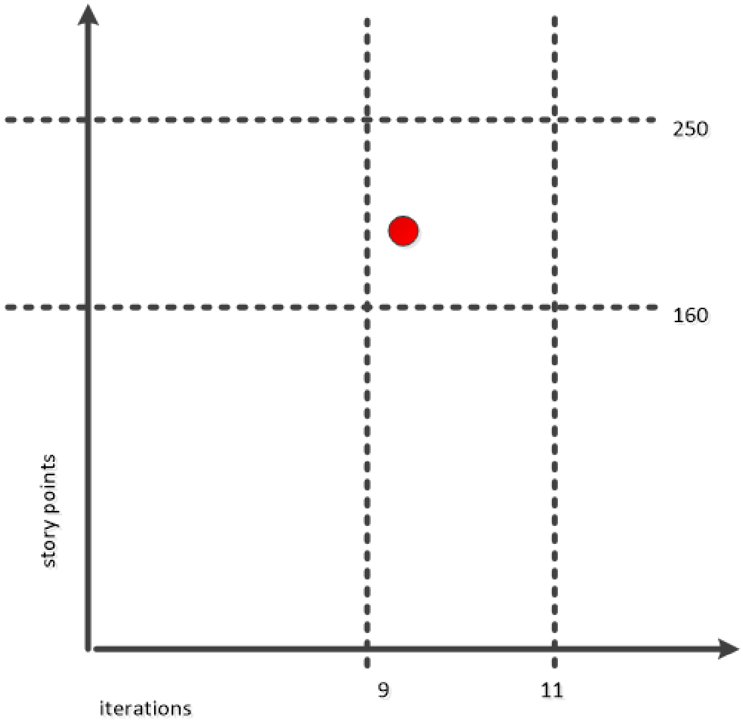
Work Estimation & Planning

Release Planning (release burn-up)



Work Estimation & Planning

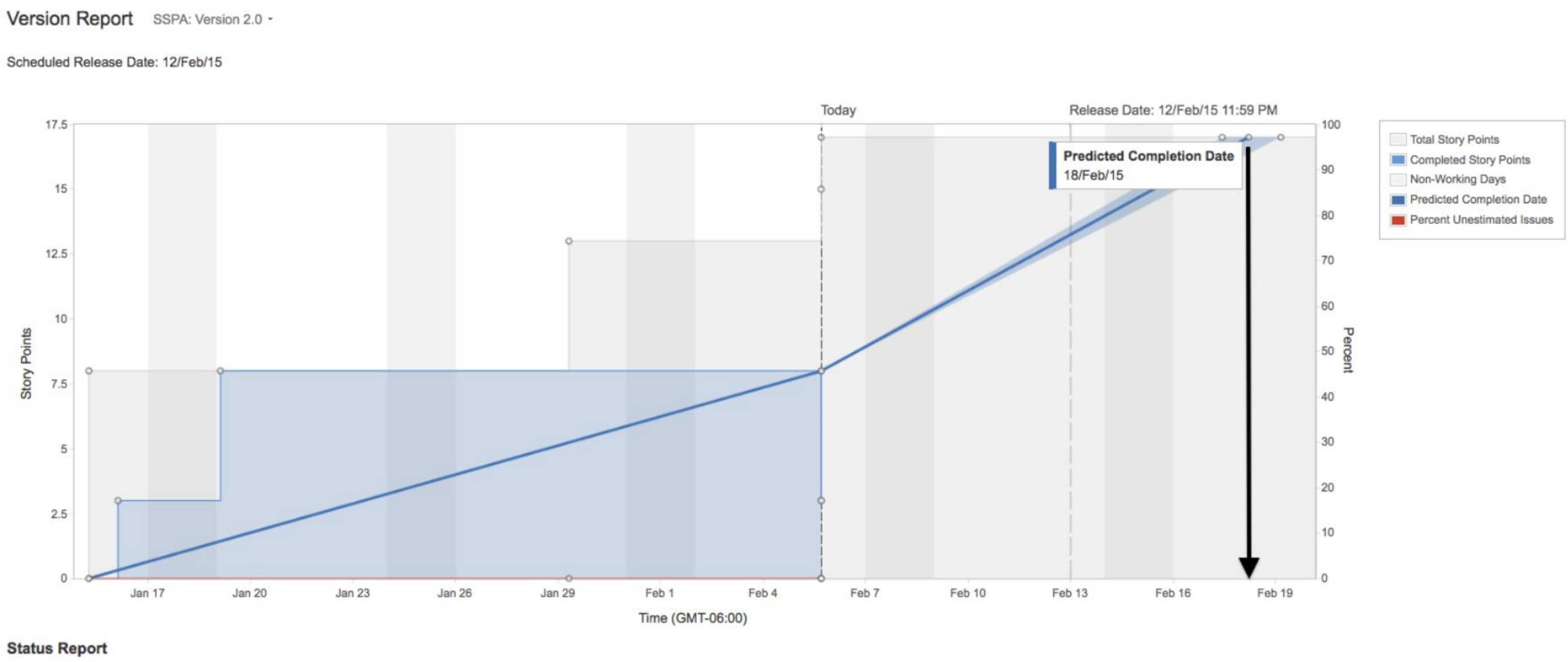
Fixed Price Projects: Time & Scope - Locked



Based on Agile Estimating and Planning, by Mike Cohn

Work Estimation & Planning

Cone of Uncertainly: Optimistic vs. Pessimistic forecasting



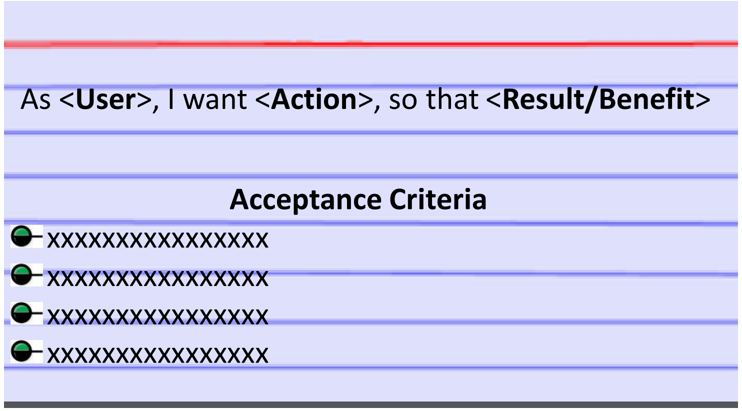
Version Report SSPA: Version 2.0

Scheduled Release Date: 12/Feb/15

Status Report

Important Scrum Elements

Card → Conversation → Confirmation (CCC)



As <**User**>, I want <**Action**>, so that <**Result/Benefit**>

Acceptance Criteria

- xxxxxxxxxxxxxxxx
- xxxxxxxxxxxxxxxx
- xxxxxxxxxxxxxxxx
- xxxxxxxxxxxxxxxx

Source: www.thoughtworks.com

Important Scrum Elements

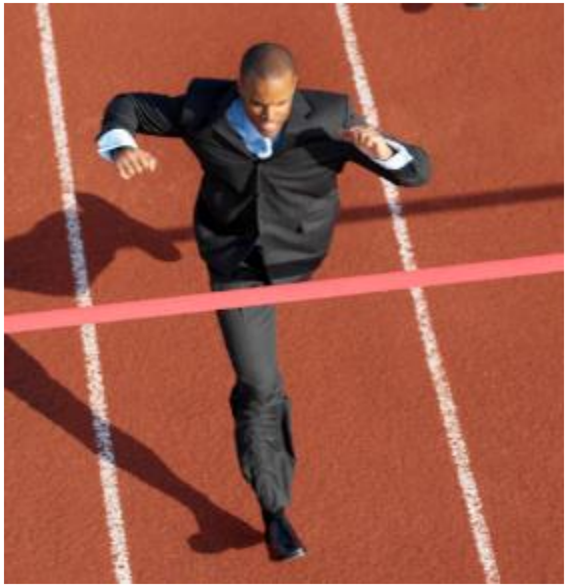
Definition of Ready (DoR)

- User Story is well defined and written
- User Story can be aligned requirements' origin (strategy)
- Acceptance criteria clearly defined
- External/Upstream dependencies are clearly identified
- User Story is estimated by Scrum team (size/complexity)
- Functional, Non-functional and Performance requirements are specified
- End-Customers are identified

Important Scrum Elements

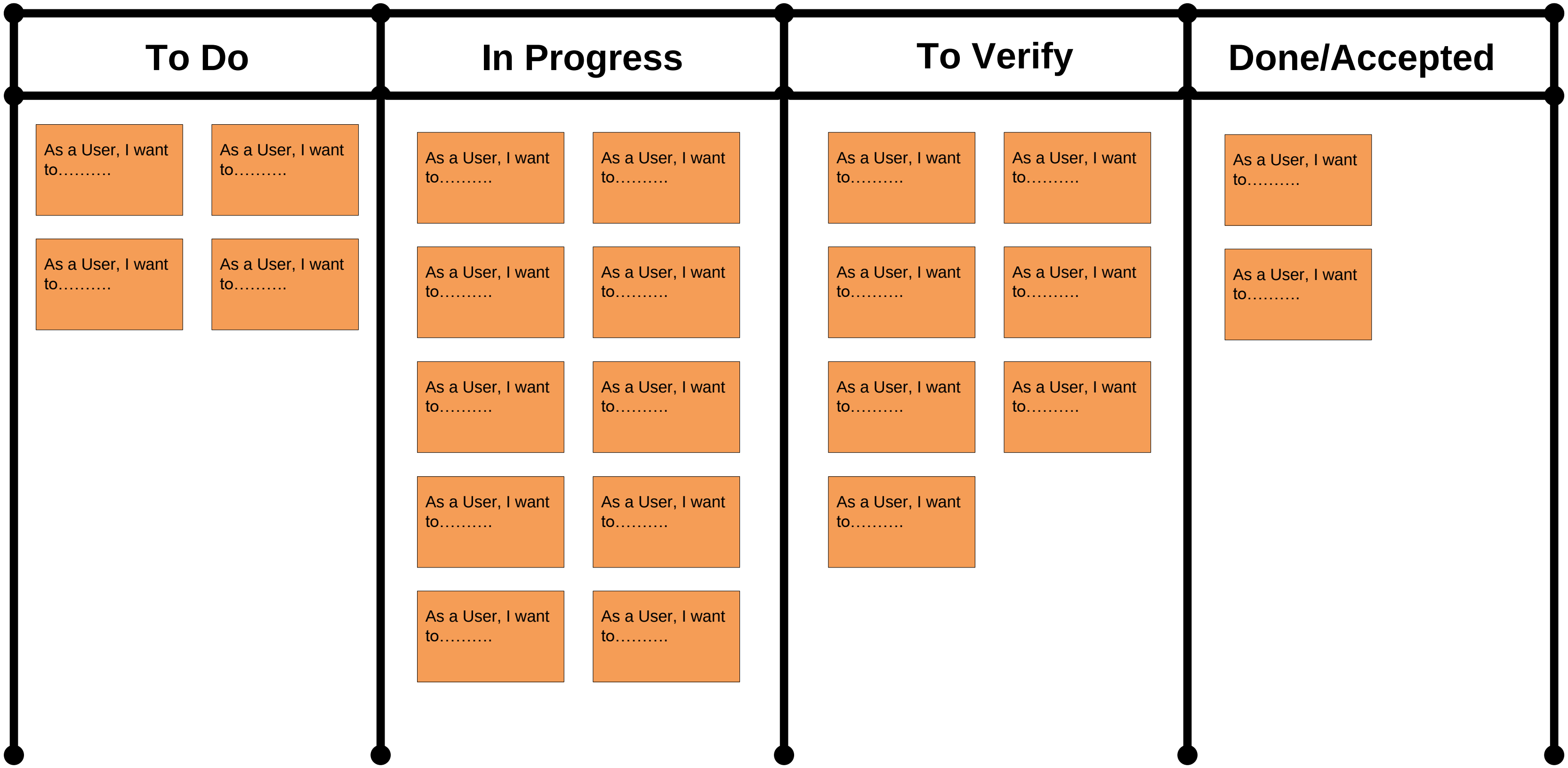
Definition of Done (DoD)

- All unit tests are is complete and passed
- Scrum team has peer reviewed its code
- All Integration tests are complete and passed
- All User Stories were presented to Product Owner and accepted
- UAT is complete and passed
- Functionality developed is potentially shippable (PSPI)
- End-User documentation (user guide, instructions) are complete
- Product Backlog is updated with any "undone" work
- etc



Important Scrum Elements

Visualization in Scrum: Scrum Story Board



To Do

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

In Progress

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

To Verify

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

As a User, I want to……….

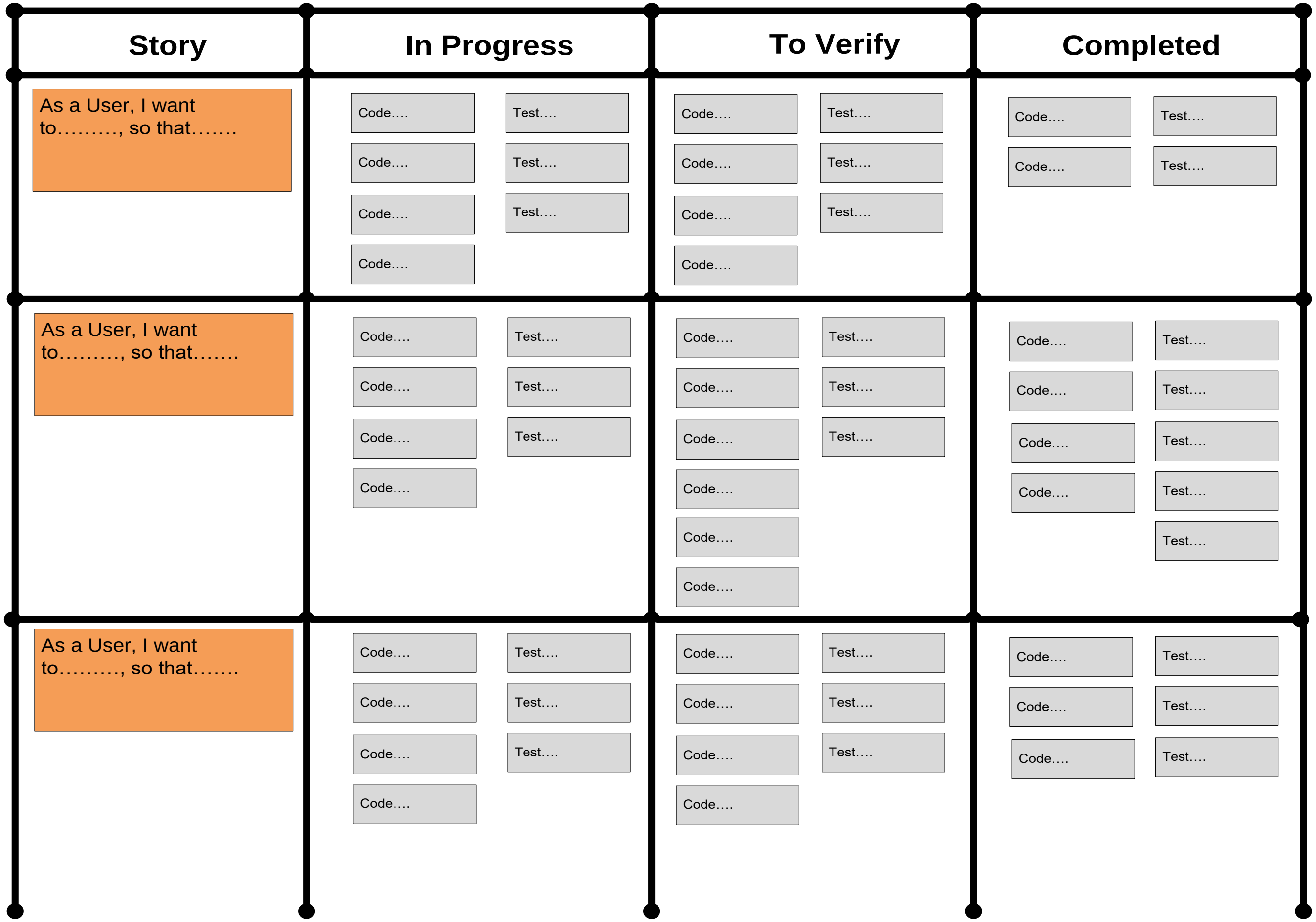
Done/Accepted

As a User, I want to……….

As a User, I want to……….

Important Scrum Elements

Visualization in Scrum: Scrum Task Board



Story	In Progress	To Verify	Completed
As a User, I want to………, so that…….	Code…. Test…. Code…. Test…. Code…. Test…. Code….	Code…. Test…. Code…. Test…. Code…. Test…. Code….	Code…. Test…. Code…. Test….
As a User, I want to………, so that…….	Code…. Test…. Code…. Test…. Code…. Test…. Code….	Code…. Test…. Code…. Test…. Code…. Test…. Code…. Code…. Code….	Code…. Test…. Code…. Test…. Code…. Test…. Code…. Test…. Test….
As a User, I want to………, so that…….	Code…. Test…. Code…. Test…. Code…. Test…. Code….	Code…. Test…. Code…. Test…. Code…. Test…. Code….	Code…. Test…. Code…. Test…. Code…. Test….

Scrum → Scrum “Under the Hood” → ***Engineering in Scrum***

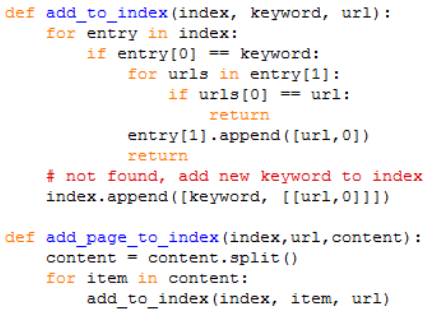
Engineering in Scrum

- Refactoring

- Technical Debt
  - Duplicate Code & Complexity,
  - Potential Bugs
  - Design Spaghetti
  - Comments

Impacts: maintainability, changeability and testability, Security & Reliability

REFACTOR EARLY, REFACTOR OFTEN
Or make damn sure you can support that legacy code
for years to come...



```
def add_to_index(index, keyword, url):
    for entry in index:
        if entry[0] == keyword:
            for urls in entry[1]:
                if urls[0] == url:
                    return
            entry[1].append([url,0])
            return
    # not found, add new keyword to index
    index.append([keyword, [[url,0]]])

def add_page_to_index(index,url,content):
    content = content.split()
    for item in content:
        add_to_index(index, item, url)
```

Kanban

History of Kanban

Wikipedia: "Kanban (看板?) (literally signboard or billboard in Japanese) is a scheduling system for lean manufacturing and just-in-time manufacturing.[2] Kanban is an inventory-control system to control the supply chain. Taiichi Ohno, an industrial engineer at Toyota, developed kanban to improve manufacturing efficiency. Kanban is one method to achieve JIT.“ https://en.wikipedia.org/wiki/Kanban

- KANBAN:
  - KAN = visual
  - BAN = card

- Toyota Production System (TPS)

- Lean tool that reduces idle time in a production process
  - Effective method to achieve Just In Time (JIT)
  - Increases efficiency and decreases waste by receiving workflow items only as when they are needed in production process
  - Reduces inventory (costs)
  - Relies on accurate forecast of demand

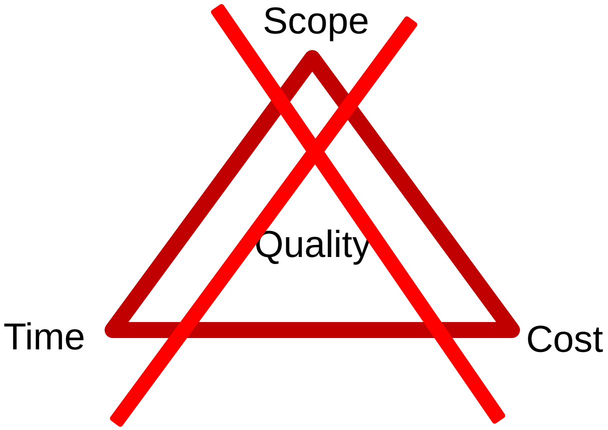
Taiichi Ohno

Empire State Building

- Year built: 1930
- Constraints:
  - 2 Aches of Land
  - $35 Million in Budget
  - Laws of Physics
- Architectural Firm: Sheve, Lamb & Harmon
- First Steel Column: April 7, 1930
- Rate of Growth: 20 stories per month
- 4 “Pacemakers”, pacing 15 stories apart:
  - Structural Steel
  - Concrete Floors
  - Metal Window Columns
  - Limestone Exterior

Empire State Building

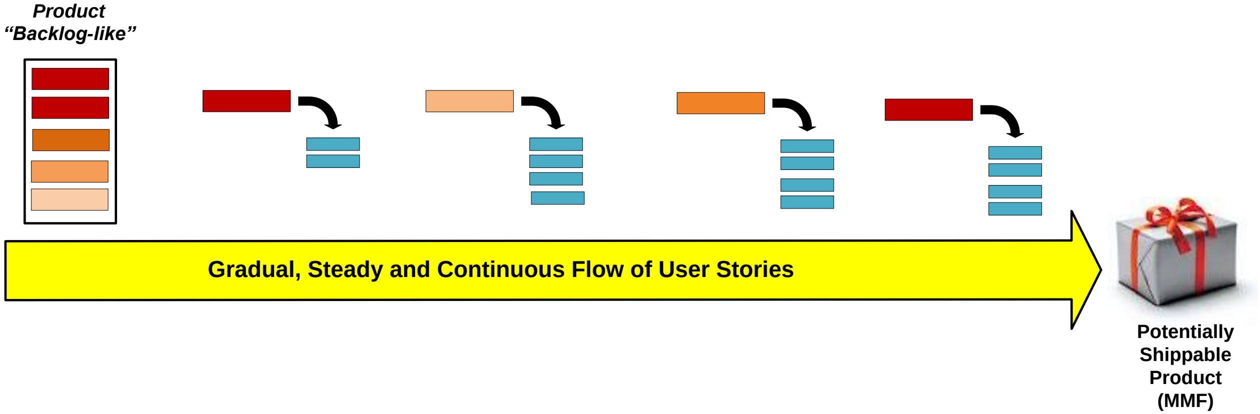
- Breaking Triple Constraint Triangle
- Focus on workflow, instead of Task Scheduling
- PULL scheduled used to meet immovable deadlines (not PUSH)
- Repetitions Efforts of:
  - Material Purchase
  - Transport of materials to the site
  - Placing materials in the same position over an over
- Avoidance of Turbulence in workflow (peaks and valleys)



Continuous Workflow

- Backlog is not prescribed
- Prioritization is not prescribed
- Estimation & Planning not prescribed
- Task Breakdown is not prescribed

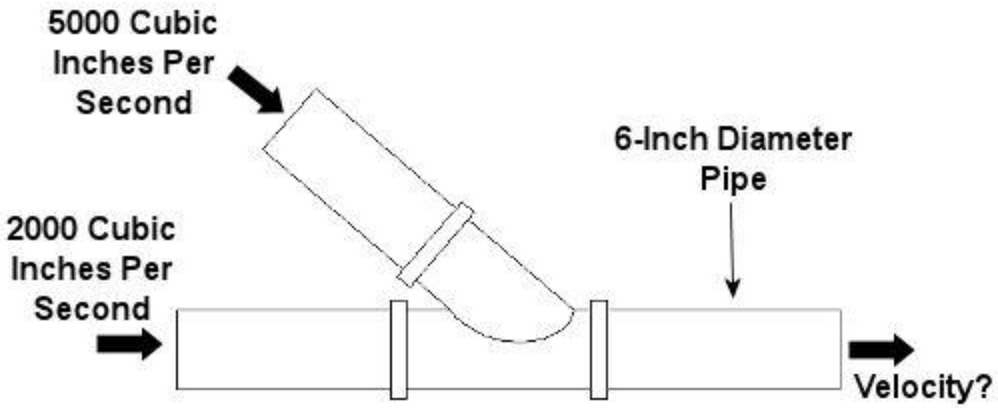
- One-Piece Flow
- PO can adjust Kanban Board
- Each Story == MMF
- WIP is controlled at State level


Product “Backlog-like”
Gradual, Steady and Continuous Flow of User Stories
Potentially Shippable Product (MMF)

Throughput

Throughput– is amount of measurable work that can be done in a given period of time

- Measures Gradual Flow of work through the system
- Tells "how much water flows through a cross-section of a pipe" at any given moment in time
- Assumes that work is consistently sized (low Variation)
- Becomes more reliable than Velocity (Scrum) since it is "point in time" specific



Work in Progress (WIP)

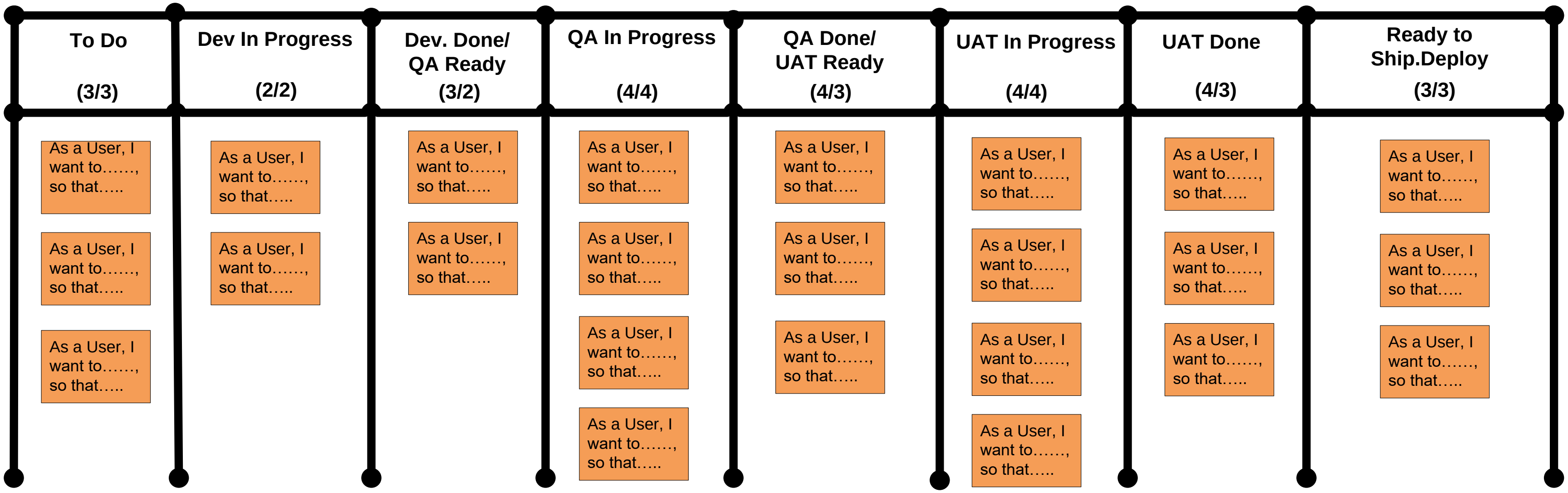
Work in Progress (WIP) – materials in the production process, at various stages, but are not yet a finished product.

- Excludes inventory of raw materials at the start of the production cycle
- Excludes finished products inventory at the end of the production cycle

Kanban → Continuous Workflow →WIP → *WIP Limits*

WIP Limits

- WIP Limits on active States & Buffer Queues are not violated
- No bottlenecks and bubbles
- Downstream vacancy signals Upstream to “pull” in more work
- Gradual Flow from Left to Right - Cadence

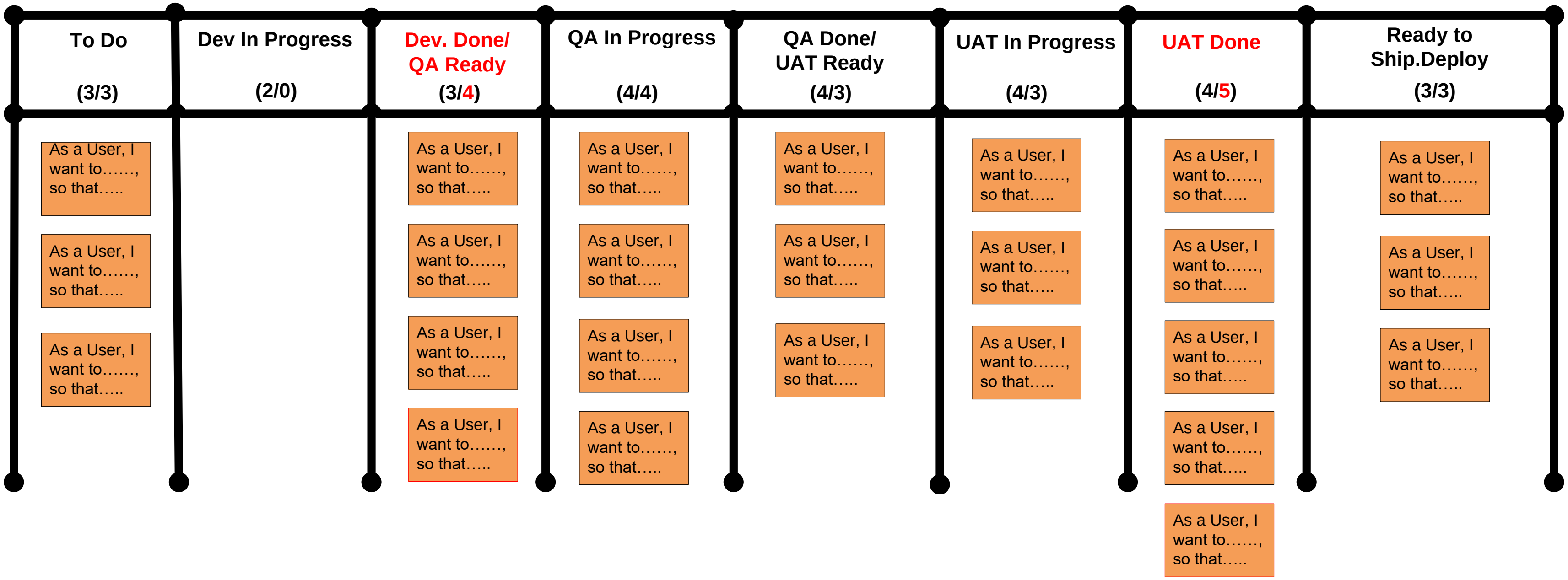


To Do (3/3)	Dev In Progress (2/2)	Dev. Done/ QA Ready (3/2)	QA In Progress (4/4)	QA Done/ UAT Ready (4/3)	UAT In Progress (4/4)	UAT Done (4/3)	Ready to Ship.Deploy (3/3)
As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..
As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..
As a User, I want to……, so that…..			As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..
			As a User, I want to……, so that…..		As a User, I want to……, so that…..		

Kanban → Continuous Workflow →WIP → *WIP Limits*

WIP Limits

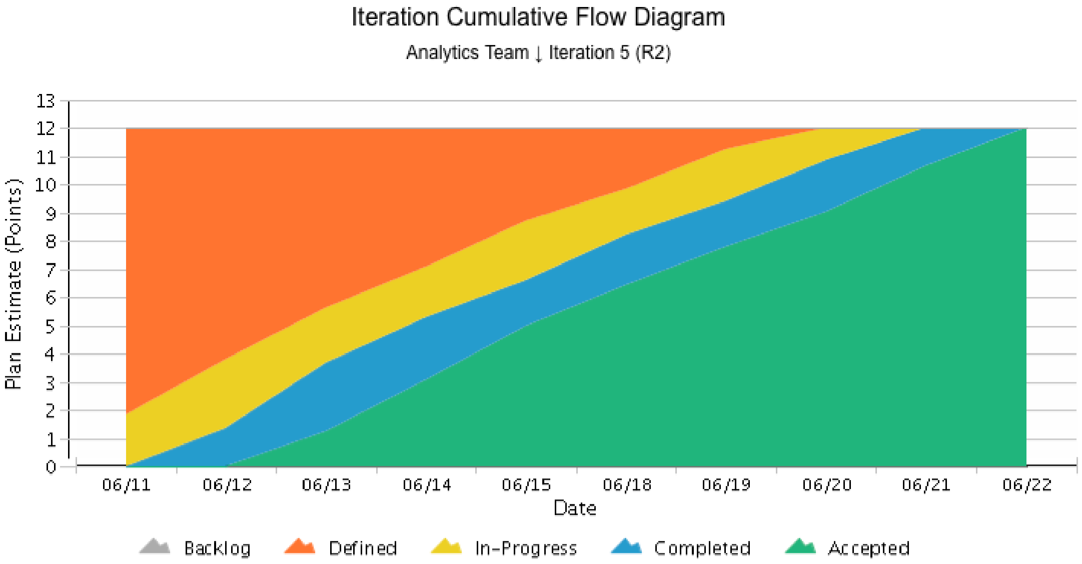
- WIP Limits on active States & Buffer Queues are violated
- Bottlenecks and bubbles form around “congested” states



To Do	Dev In Progress	Dev. Done/ QA Ready	QA In Progress	QA Done/ UAT Ready	UAT In Progress	UAT Done	Ready to Ship.Deploy
(3/3)	(2/0)	(3/4)	(4/4)	(4/3)	(4/3)	(4/5)	(3/3)
As a User, I want to……, so that…..		As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..
As a User, I want to……, so that…..		As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..
As a User, I want to……, so that…..		As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..	As a User, I want to……, so that…..
		As a User, I want to……, so that…..	As a User, I want to……, so that…..			As a User, I want to……, so that…..	
						As a User, I want to……, so that…..	

Cumulative Flow Diagram (CFD)

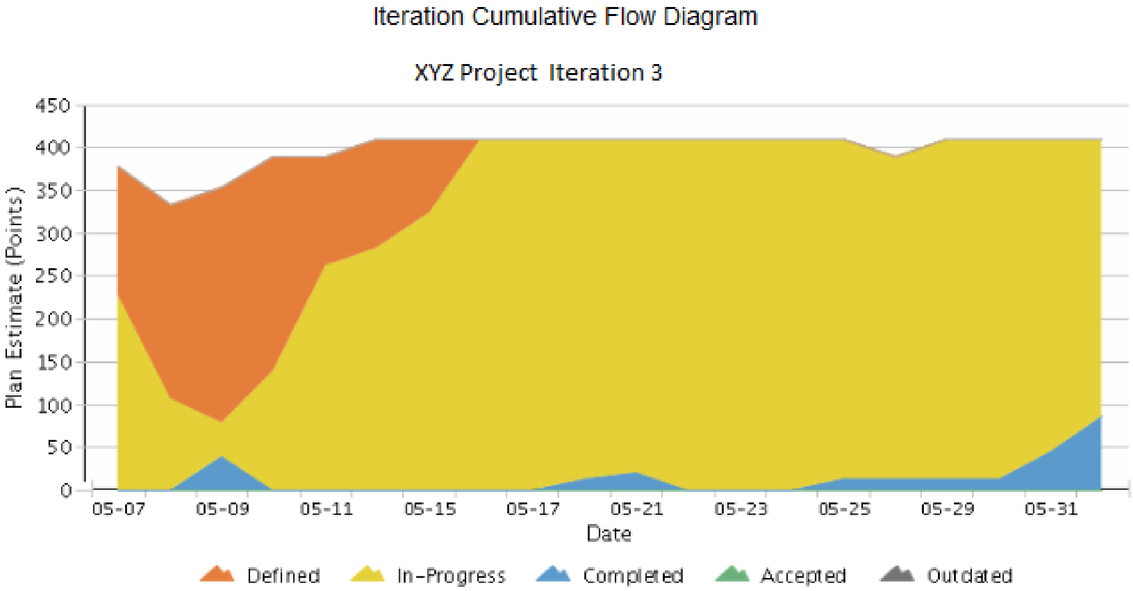
- States of work gradually change as Iteration progresses
- Percentage of work products with “downstream states” is gradually increasing over time
- Work gets Accepted early



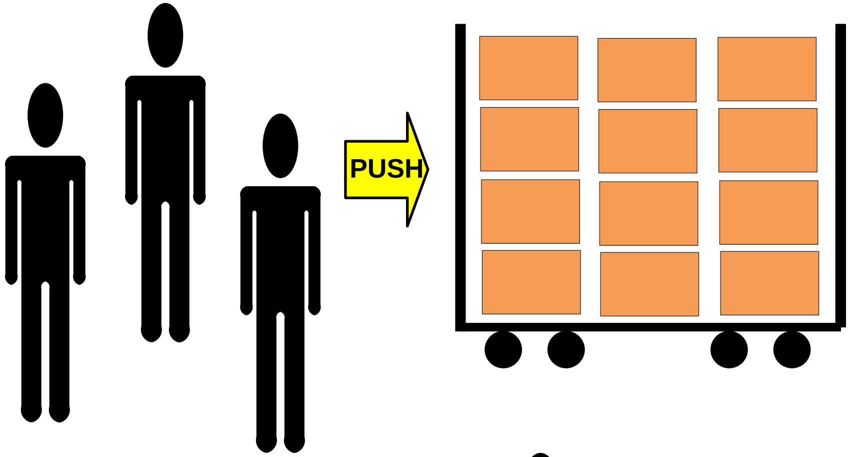
Courtesy of: http://www.rallydev.com/help/progress-reports

Cumulative Flow Diagram (CFD)

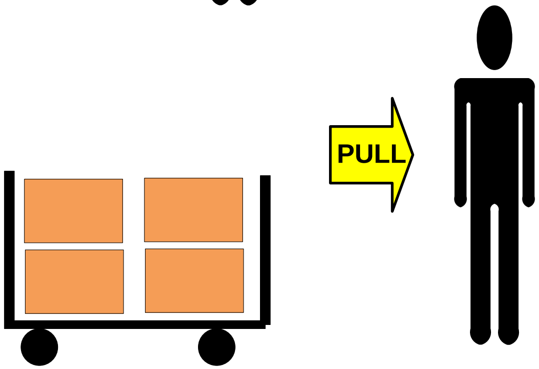
- States of work DON'T gradually change as Iteration progresses
- Percentage of work products with "downstream states" is NOT gradually increasing over time
- Work DOES NOT gets Accepted early
- Too much work In-Progress for a long time



Pull vs. Push



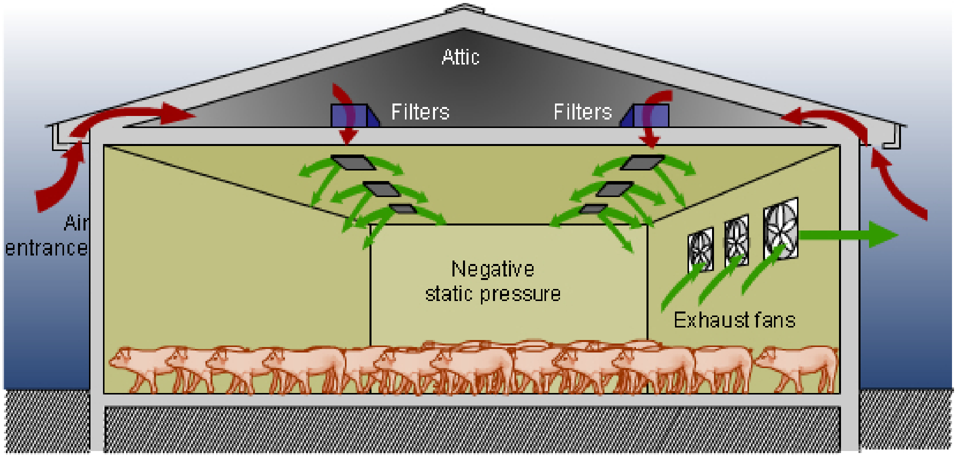
Anticipated Demand
Approximated Usage
High Quantities
Voluminous Storage/Inventory
High Waste of resources (time & money)
Heavy Weight Shipments
Longer turn around
Bottlenecks & Bubbles (Vacuum)



Driven by precise Demand
Smaller, more digestible quantities
Low Waste of resources (time & money)
Light Weight Shipments
Cost effectiveness
Quicker turn around
Smoother Flow

Pull System

- “Negative pressure” effect
- Downstream availability sends a signal Upstream
- Works gets ‘pulled’, not ‘pushed’
- WIP limits define Throughout
- WIP limit size of each State determines if merging of work streams will produce plausible results
- Consistency of Work Product size ensures accuracy (reliability of Kanban Throughout is somewhere higher then Velocity of Scrum)



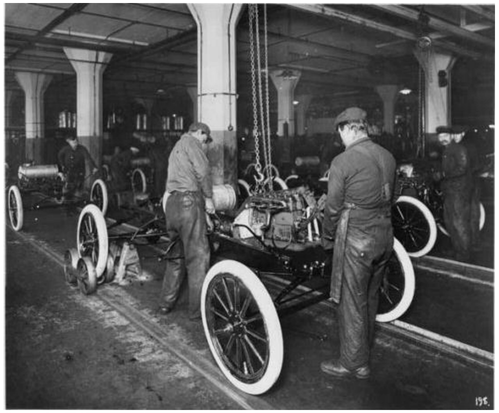
Courtesy of: http://www.pig333.com/what_the_experts_say/filtration-systems-on-commercial-sow-farms-definitions-and-strategies_4960/

KANBAN

One Piece Flow

- Used in Assembly/Production Line
- Historically, working on one “PIECE” at a time had proved to be move effective and less error-prone…..

Manual Production Line



Courtesy of: http://people.hofstra.edu

Semi-Automatic Production Line

Courtesy of: http://www.carpages.co.uk/mini/mini-sales-04-04-07.asp

Fully Automatic Production Line

Courtesy of: http://www.telegraph.co.uk

One Piece Flow

- Even Workload Distribution
- WIP per “State” is limited
- Gradual Flow
- No gaps b/w “States”

One Piece Flow

Order, Pay & Pick Up...Fight….all at once...

Courtesy Of: http://rbth.ru/articles/2011/02/25/from_big_macs_to_the_circus_12505.html

One Piece Flow

Starbucks Model: Gridlock Removal Practice

- Regular Hour
  - Waiting Line
  - Placing Order
  - Making Payment
  - Picking Up Order
  - Getting Sugar/Milk/Half & Half

- Rush Hour
  - Waiting Line
  - **In-Line Order Taking**
  - Placing Order
  - Making Payment
  - Picking Up Order
  - Getting Sugar/Milk/Half & Half

One Piece Flow

Optimal WIP Limits

WIP = 1 Brick

Courtesy of: http://www.lafollette.wisc.edu/alumnifriends/newsletter/2010/spring.html

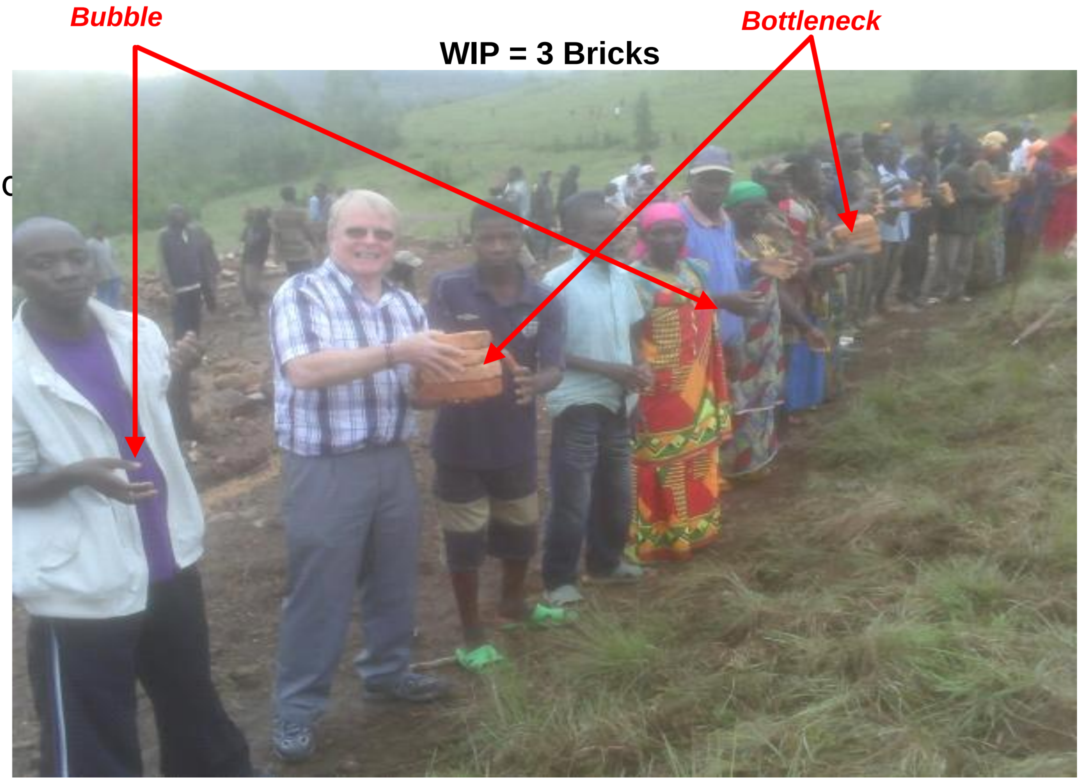
WIP = 2 Bricks

Courtesy of: http://hungeree.com/society/passing-the-bricks-in-jaipur-india/

One Piece Flow

Sub-Optimal WIP Limits

- WIP is too high
- Multiple WIP violations in the same “proc
- Creation of multiple Bubbles



Courtesy of: http://leecadden.wordpress.com/2010/06/page/2/

One Piece Flow

Strong Violation of WIP Limits

- Exceeding WIP Limits
- Creating an ultimate bottleneck & halting “production line”

…..Exceeding WIP Limit…..

Courtesy of: http://www.12ozprophet.com/forum/showthread.php?t=130723&page=2

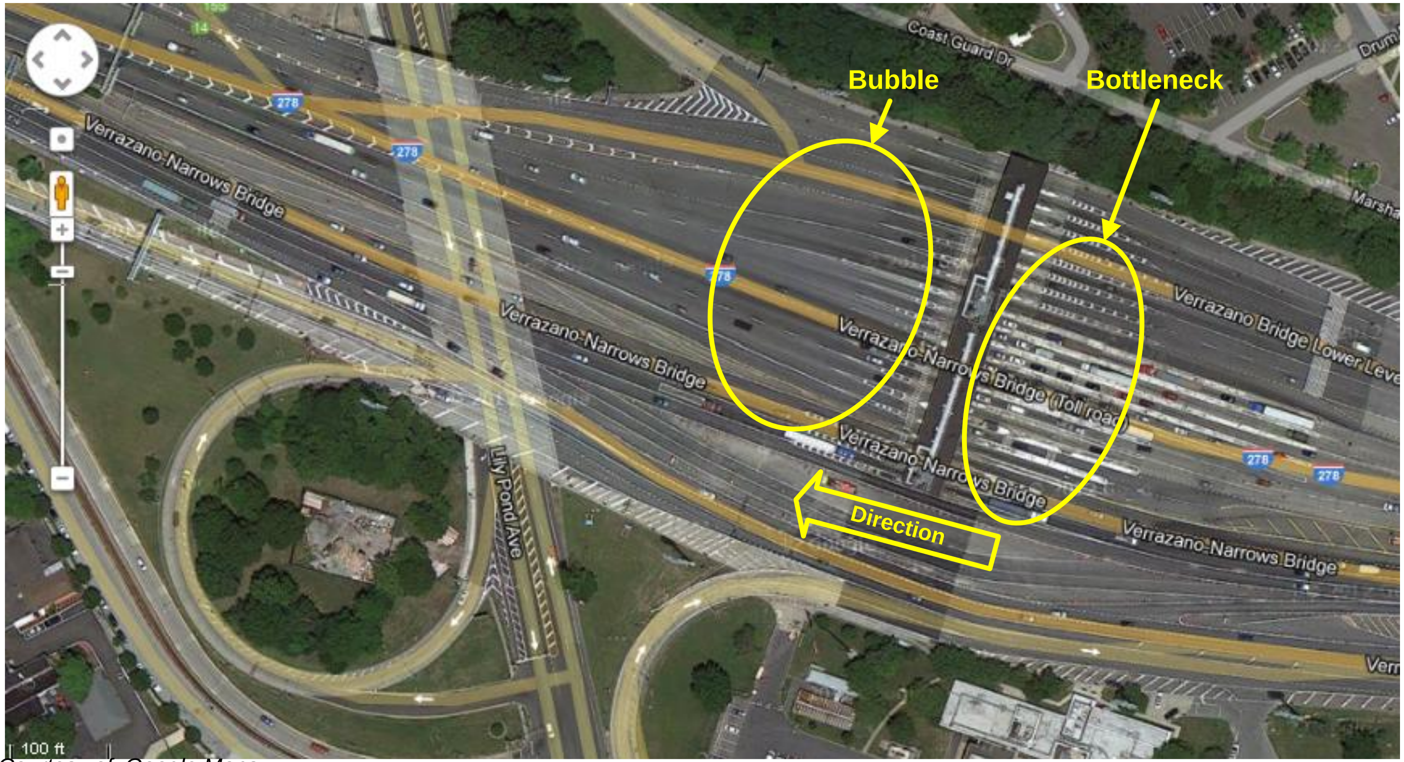
…..Clogging & Overflow…..

Courtesy of: http://thatblueyak.blogspot.com/2011/04/time-toilet-overflowed-at-work-nooo.html

One Piece Flow

Bottlenecks & Bubbles

- Exceeding WIP Limits
- Creating an ultimate bottleneck & halting “production line”



Courtesy of: Google Maps

One Piece Flow

Bottlenecks & Bubbles

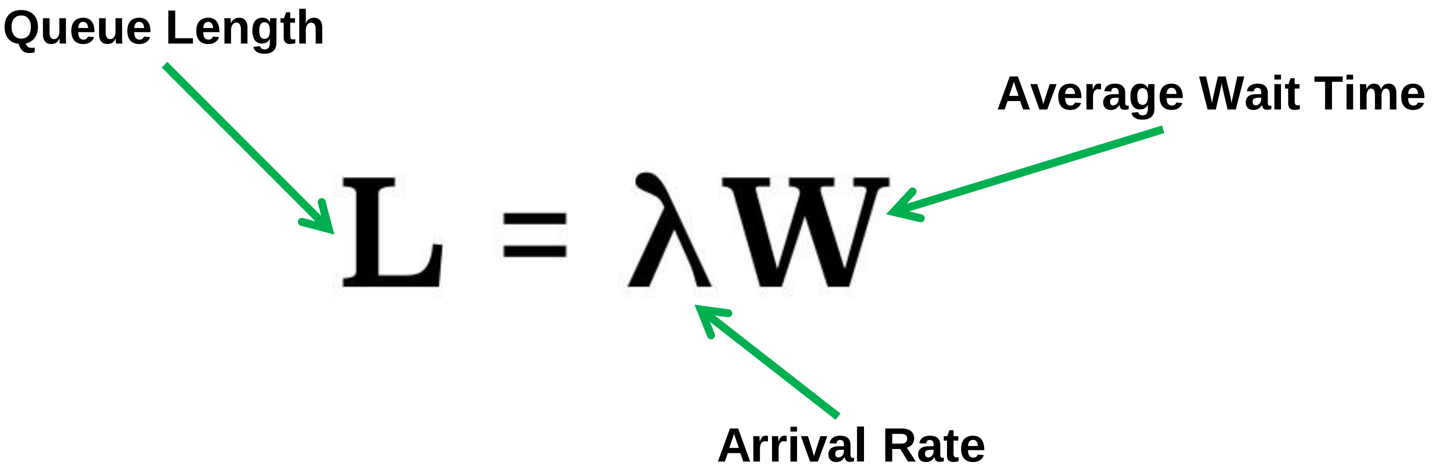
- Violation of WIP with Merging
- Creation of Bottleneck behind the merge
- Creation of Bubble in front of the merge

Courtesy of: http://www.thejakartapost.com/files/images/P-30-TRAFFIC.jpg

Principles of Product Development Flow

Little’s Law

Wikipedia: "The long-term average number of customers in a stable system L is equal to the long-term average effective arrival rate, λ, multiplied by the (Palm-)average time a customer spends in the system, W; or expressed algebraically: L = λW."



Queue Length

Average Wait Time

$$L = \lambda W$$

Arrival Rate

Principles of Product Development Flow

Costs: Holding vs. Transactional

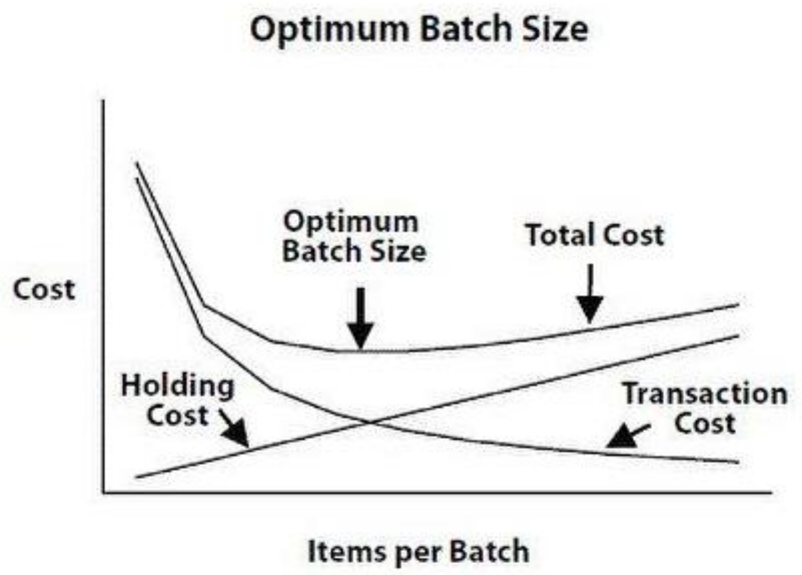
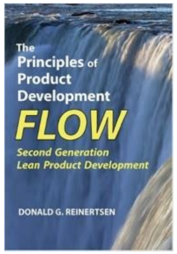


Figure 5-5 Optimum batch size is another U-curve optimization. Higher transaction costs shift optimum batch size higher. Higher holding costs shift it lower.


The Principles of Product Development
FLOW
Second Generation Lean Product Development
DONALD G. REINERTSEN

The Principles of Product Development Flow: Second Generation Lean Product Development, by Donald G. Reinertsen

Principles of Product Development Flow

Managing Capacity of Part-Time Workers

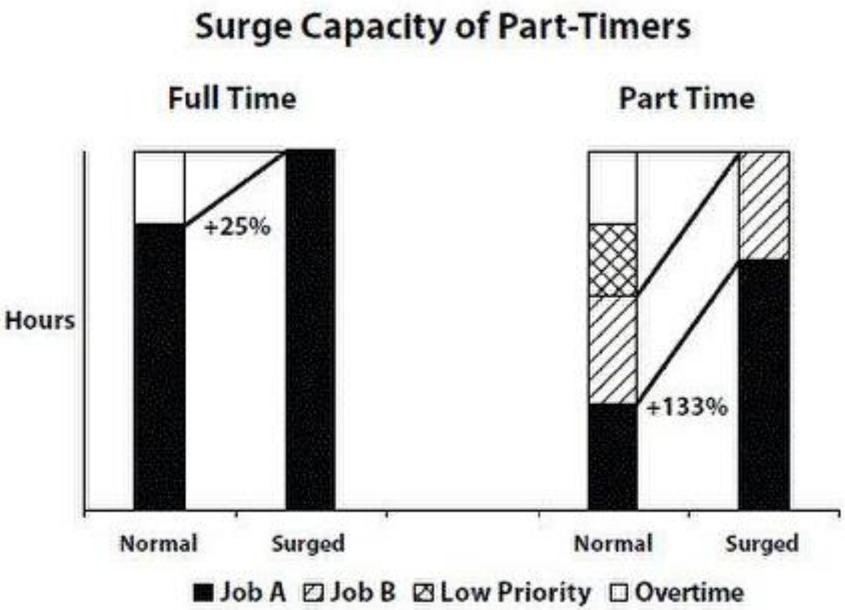
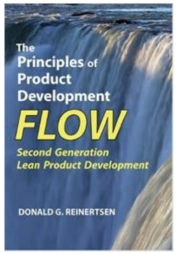


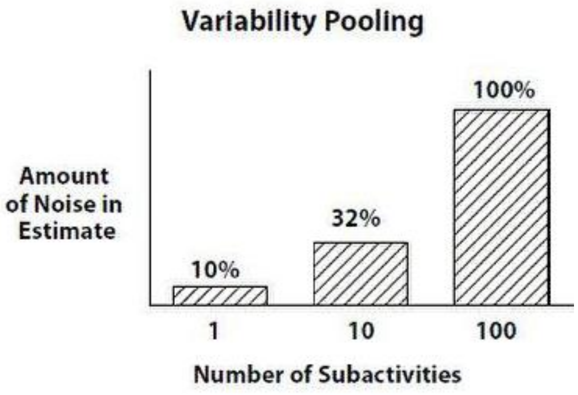
Figure 6-4 Part-time workers have significantly more surge capacity than full-time workers on the same project.



The Principles of Product Development Flow: Second Generation Lean Product Development, by Donald G. Reinertsen

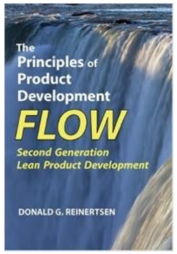
Principles of Product Development Flow

Reducing System Variability



$$\boxed{\sigma_T = \sqrt{\sigma_1^2 + \sigma_2^2 + \sigma_3^2 \ldots.. + \sigma_n^2}}$$

Figure 4-7 When we combine activities with uncorrelated variability, the total standard deviation grows as a square root function. The more we subdivide activities, the higher the coefficient of variation in each subactivity. We can reduce variability by pooling activities together.

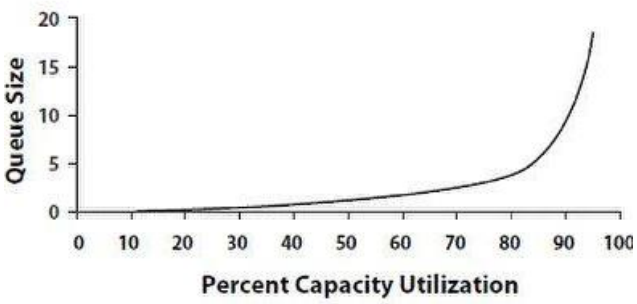

The Principles of Product Development
FLOW
Second Generation Lean Product Development
DONALD G. REINERTSEN

The Principles of Product Development Flow: Second Generation Lean Product Development, by Donald G. Reinertsen

Principles of Product Development Flow

Impact of Variation on Capacity Utilization and Queue Size

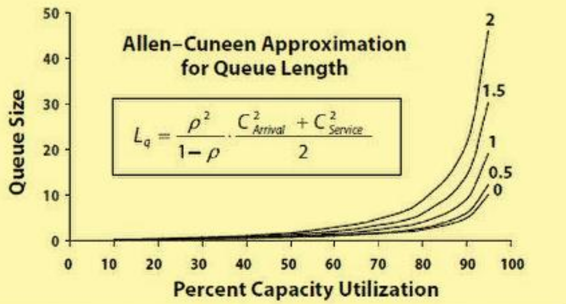
Queue Size vs. Capacity Utilization



Note: Assumes M/M/1/∞ Queue

Figure 3-2 Queue size increases rapidly with capacity utilization.

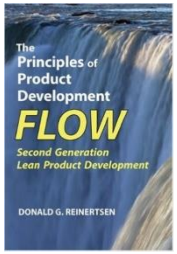
Queue Size with Different Coefficients of Variation



$$L_q = \frac{\rho^2}{1-\rho} \cdot \frac{C^2_{Arrival} + C^2_{Service}}{2}$$

Note: Coefficient of Variation = Standard Deviation/Mean
An exponential distribution has a coefficient of variation equal to one.

Figure 3-5 Queue size is linearly proportional to variability. As a result, reducing variability has much less effect on queue size than lowering capacity utilization.


The Principles of Product Development FLOW
Second Generation Lean Product Development
DONALD G. REINERTSEN

The Principles of Product Development Flow: Second Generation Lean Product Development, by Donald G. Reinertsen

Principles of Product Development Flow

Throughput and System Utilization

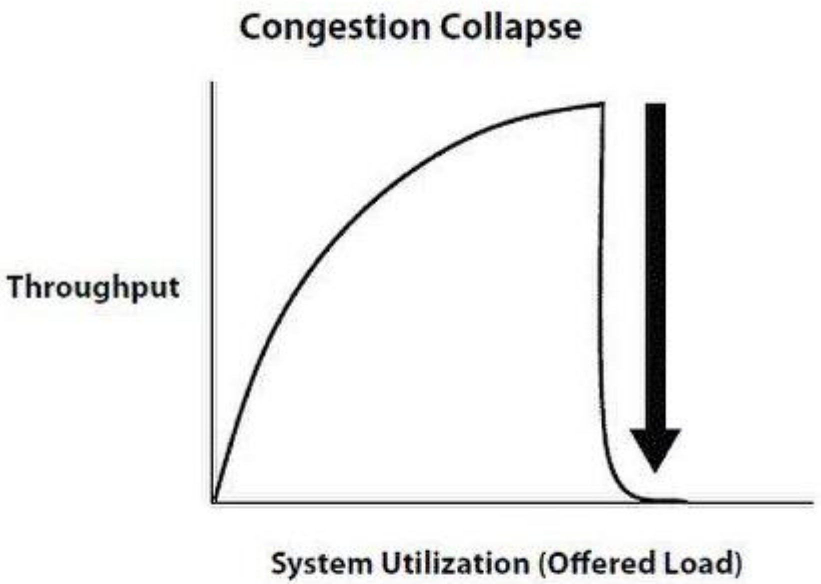
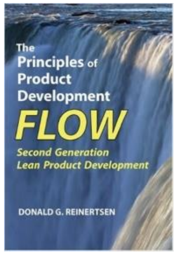


Figure 7-4 Congestion collapse is an abrupt drop in throughput at high levels of system utilization. It results in an undesirable state of high utilization and low throughput.


The Principles of Product Development FLOW
Second Generation Lean Product Development
DONALD G. REINERTSEN

The Principles of Product Development Flow: Second Generation Lean Product Development, by Donald G. Reinertsen

Principles of Product Development Flow

Queues and Servers

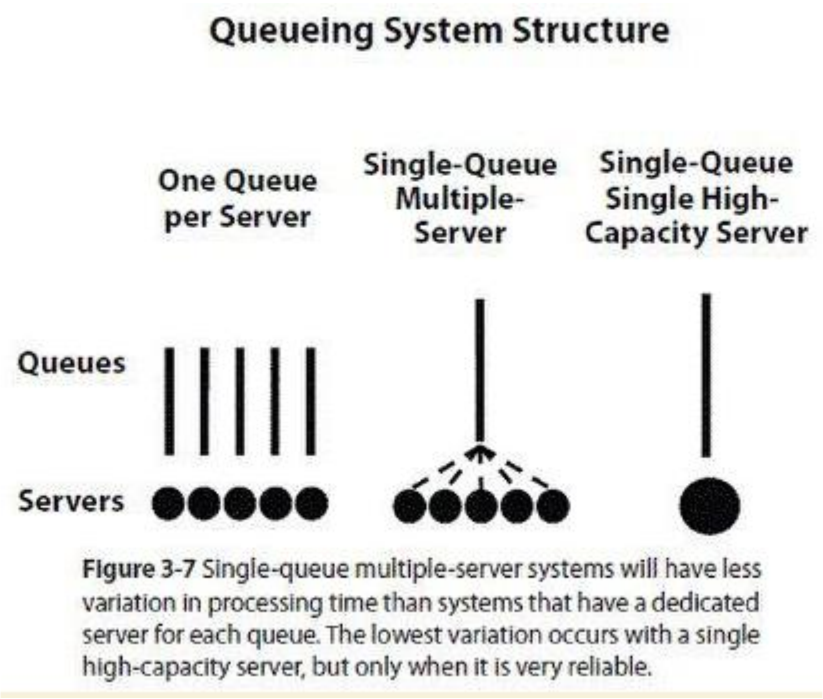
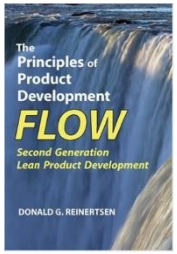

Queueing System Structure
One Queue per Server
Single-Queue Multiple-Server
Single-Queue Single High-Capacity Server
Queues
Servers

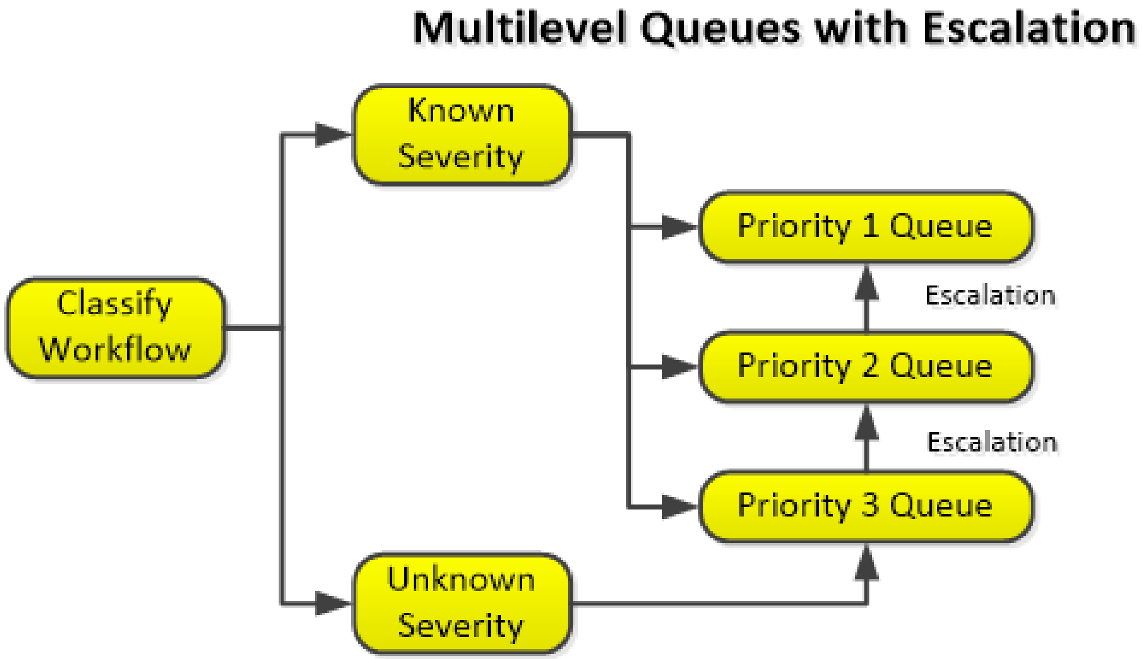
Figure 3-7 Single-queue multiple-server systems will have less variation in processing time than systems that have a dedicated server for each queue. The lowest variation occurs with a single high-capacity server, but only when it is very reliable.


The Principles of Product Development FLOW
Second Generation Lean Product Development
DONALD G. REINERTSEN

The Principles of Product Development Flow: Second Generation Lean Product Development, by Donald G. Reinertsen

Escalation in Kanban

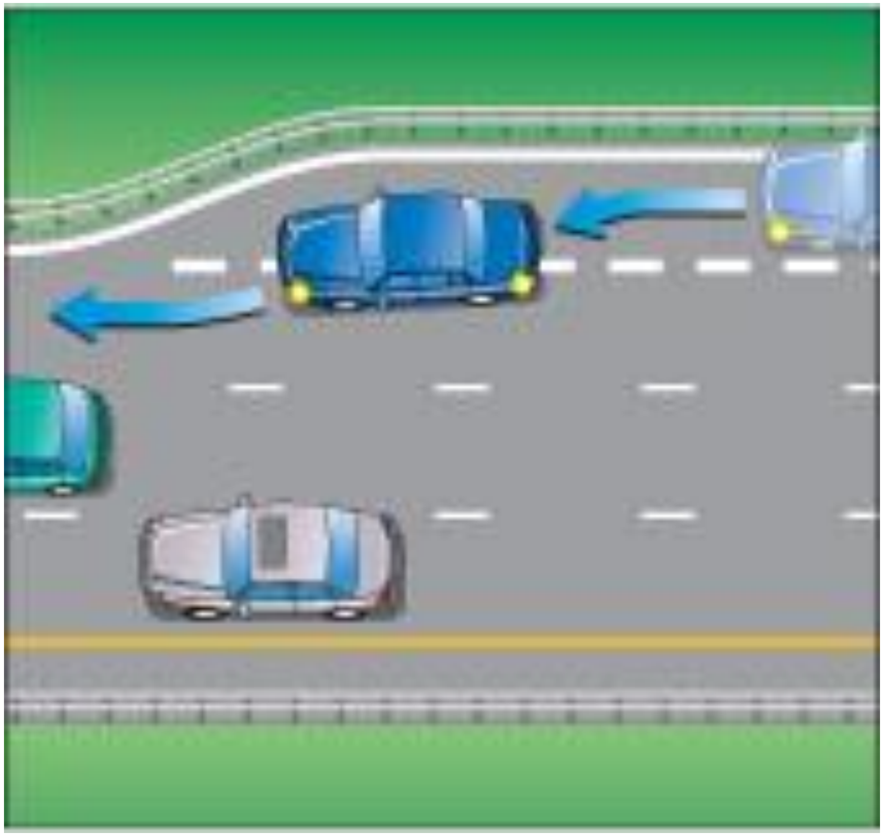
- Each queue is allocated a portion of capacity
- Use RR scheduling within each queue
- Jobs stuck in queue move to a higher priority queue



Jobs with known severity can be allocated to queues based on their priority. Jobs with unknown severity will start at lowest-level queue and will be escalated if they exceed a certain time limit in the queue.

The Principles of Product Development Flow: Second Generation Lean Product Development, by Donald G. Reinertsen

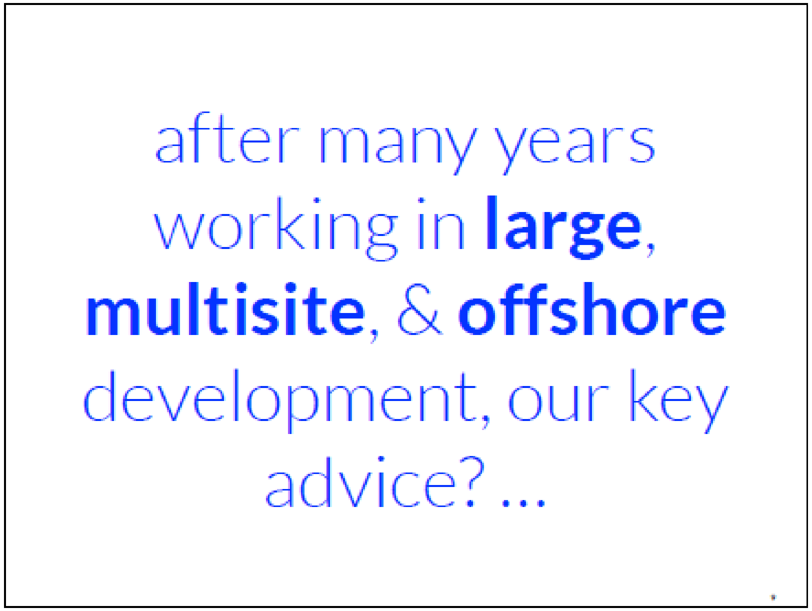
Enterprise Kanban



Converting work streams

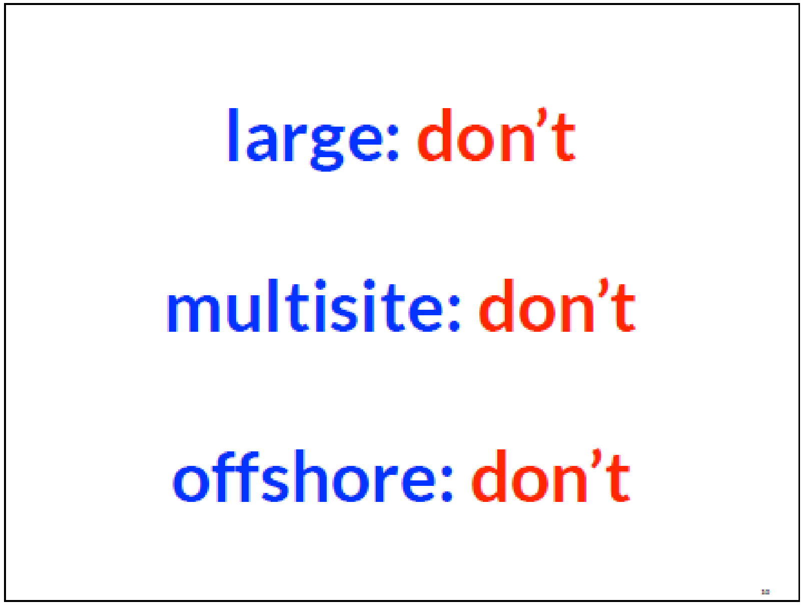
Scale or NOT to Scale?

Source: LeSS Training by Craig Larman



after many years working in **large**, **multisite**, & **offshore** development, our key advice? …

9



large: don’t

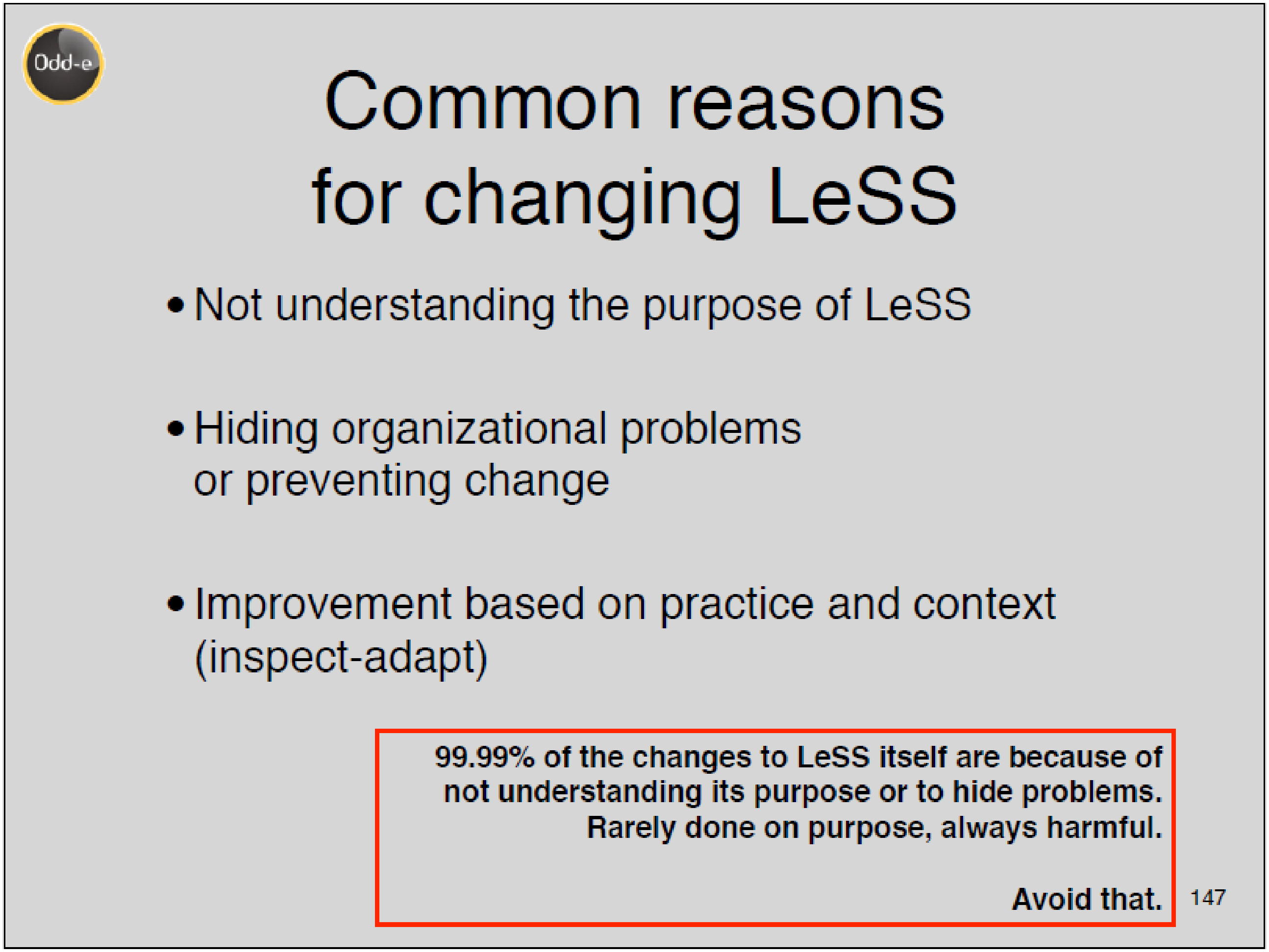
multisite: don’t

offshore: don’t

10

Common Reasons for "Customizing" LeSS

Source: LeSS Training by Bas Vodde



Odd-e

Common reasons for changing LeSS

- Not understanding the purpose of LeSS
- Hiding organizational problems or preventing change
- Improvement based on practice and context (inspect-adapt)

99.99% of the changes to LeSS itself are because of not understanding its purpose or to hide problems. Rarely done on purpose, always harmful.

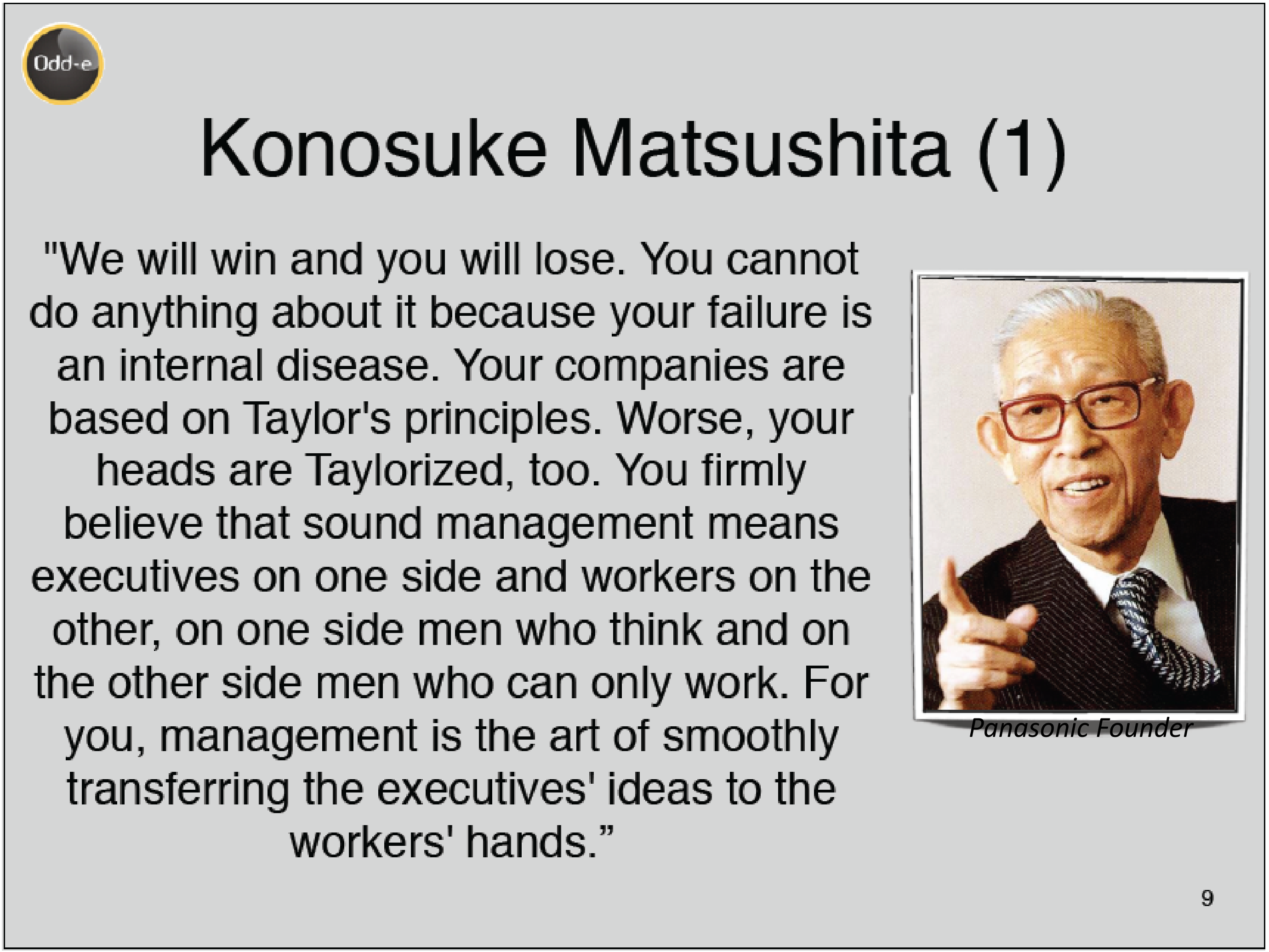
Avoid that.

147

Agile Scaling

Voices of "Elders"

Source: LeSS Training by Bas Vodde



Konosuke Matsushita (1)

"We will win and you will lose. You cannot do anything about it because your failure is an internal disease. Your companies are based on Taylor's principles. Worse, your heads are Taylorized, too. You firmly believe that sound management means executives on one side and workers on the other, on one side men who think and on the other side men who can only work. For you, management is the art of smoothly transferring the executives' ideas to the workers' hands."

Panasonic Founder

9

Voices of “Elders”

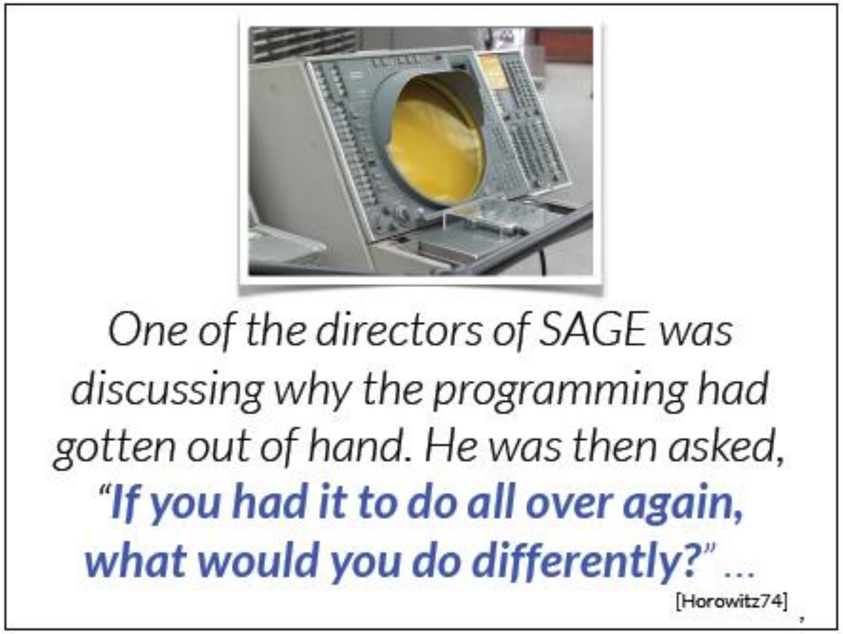
*"…**The leadership needs to become Agile**. They need to support the teams, remove impediments, and coach the organizations to be agile. There are so few managers trained and good as this today that **many companies will be driven out of business by their agile competitors**…"*

Jeff Sutherland

(Source: http://www.versionone.com/pdf/2013-state-of-agile-survey.pdf)

Voices of “Elders”

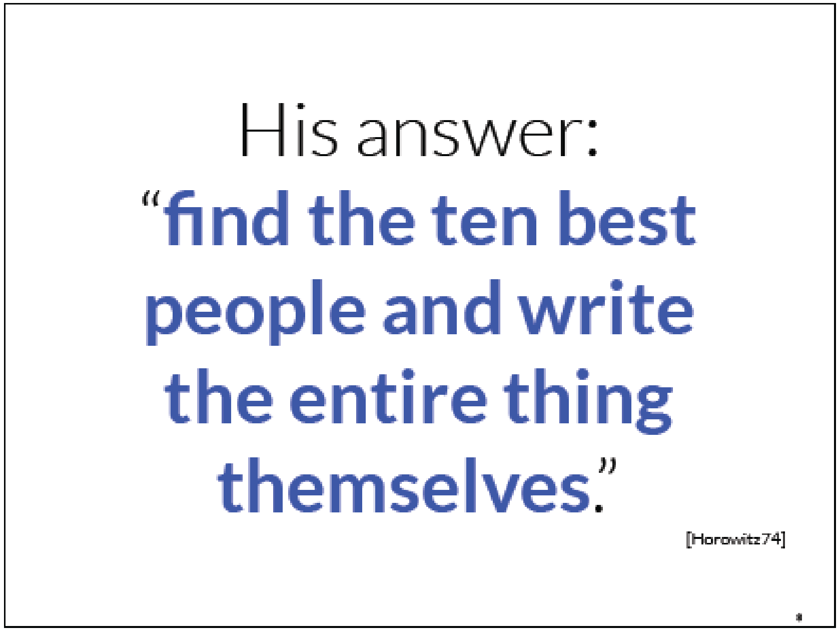
Source: LeSS Training by Craig Larman



One of the directors of SAGE was discussing why the programming had gotten out of hand. He was then asked, ***“If you had it to do all over again, what would you do differently?”*** …

[Horowitz74]

7



His answer:

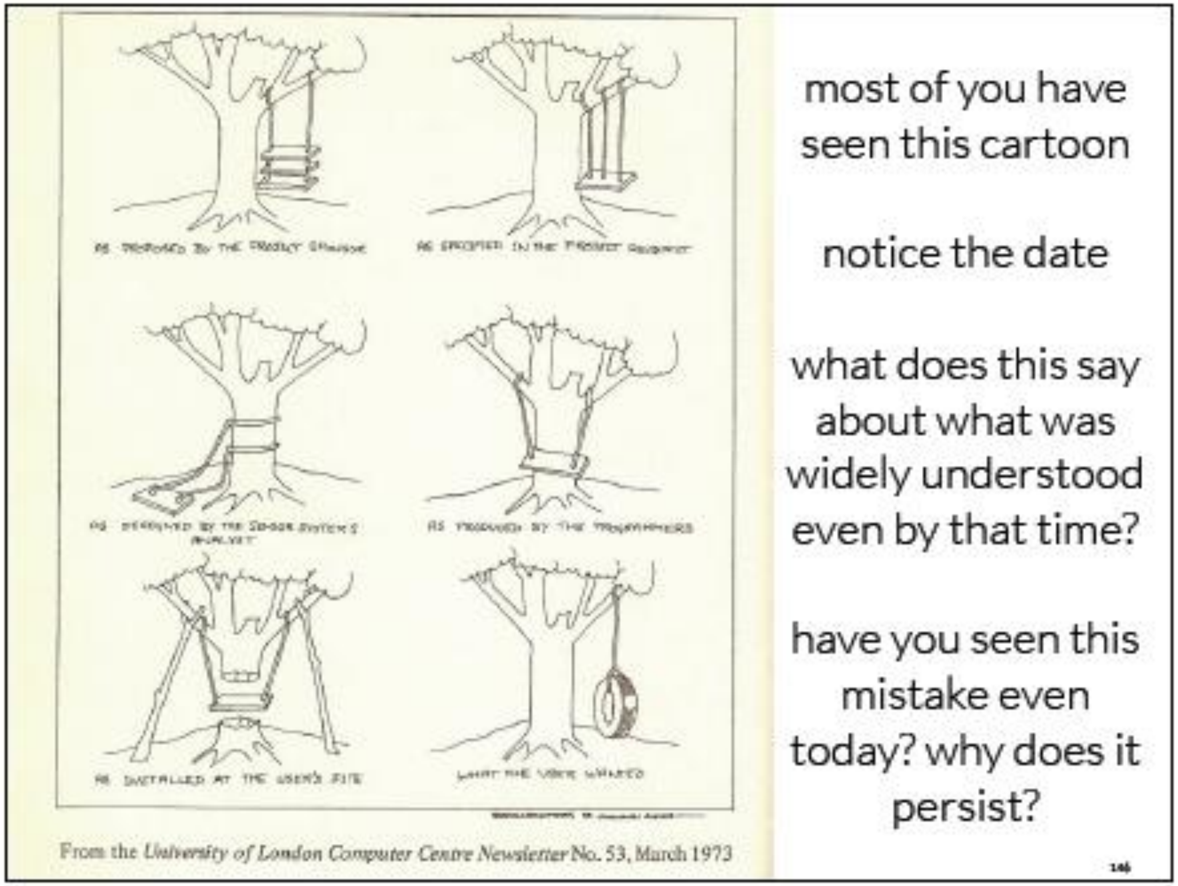
“find the ten best people and write the entire thing themselves.”

[Horowitz74]

8

Types of Product Development

Source: LeSS Training by Bas Vodde


AS PROPOSED BY THE PROJECT SPONSOR
AS SPECIFIED IN THE PROJECT REQUEST
AS DESIGNED BY THE SENIOR SYSTEMS ANALYST
AS PRODUCED BY THE PROGRAMMERS
AS INSTALLED AT THE USER'S SITE
WHAT THE USER WANTED

From the *University of London Computer Centre Newsletter* No. 53, March 1973

most of you have seen this cartoon

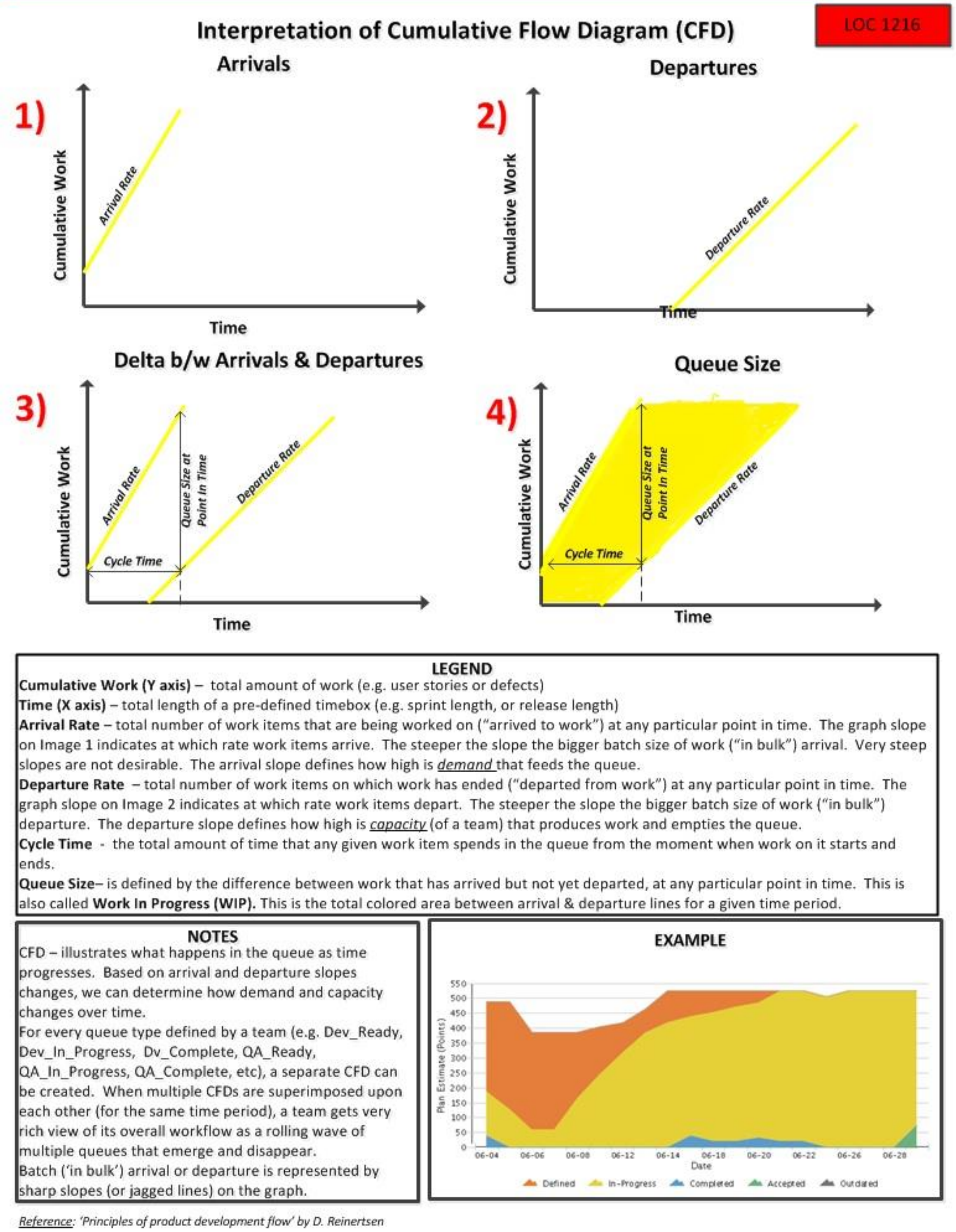
notice the date

what does this say about what was widely understood even by that time?

have you seen this mistake even today? why does it persist?

146

Appendix



Interpretation of Cumulative Flow Diagram (CFD)

LEGEND

Cumulative Work (Y axis) – total amount of work (e.g. user stories or defects)

Time (X axis) – total length of a pre-defined timebox (e.g. sprint length, or release length)

Arrival Rate – total number of work items that are being worked on ("arrived to work") at any particular point in time. The graph slope on Image 1 indicates at which rate work items arrive. The steeper the slope the bigger batch size of work ("in bulk") arrival. Very steep slopes are not desirable. The arrival slope defines how high is *demand* that feeds the queue.

Departure Rate – total number of work items on which work has ended ("departed from work") at any particular point in time. The graph slope on Image 2 indicates at which rate work items depart. The steeper the slope the bigger batch size of work ("in bulk") departure. The departure slope defines how high is *capacity* (of a team) that produces work and empties the queue.

Cycle Time - the total amount of time that any given work item spends in the queue from the moment when work on it starts and ends.

Queue Size– is defined by the difference between work that has arrived but not yet departed, at any particular point in time. This is also called **Work In Progress (WIP).** This is the total colored area between arrival & departure lines for a given time period.

NOTES

CFD – illustrates what happens in the queue as time progresses. Based on arrival and departure slopes changes, we can determine how demand and capacity changes over time.

For every queue type defined by a team (e.g. Dev_Ready, Dev_In_Progress, Dv_Complete, QA_Ready, QA_In_Progress, QA_Complete, etc), a separate CFD can be created. When multiple CFDs are superimposed upon each other (for the same time period), a team gets very rich view of its overall workflow as a rolling wave of multiple queues that emerge and disappear.

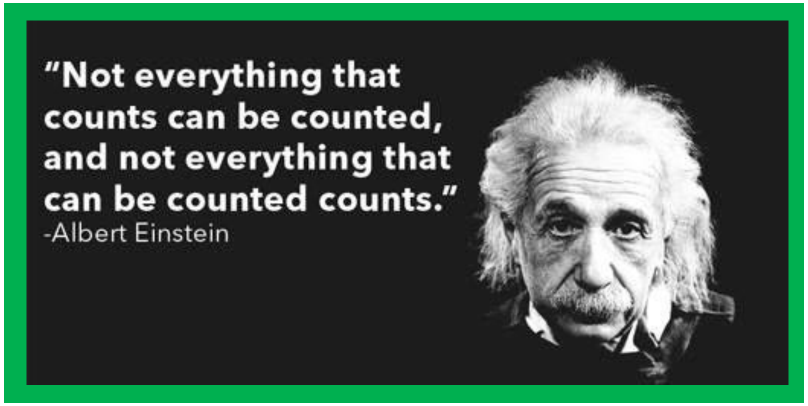
Batch ('in bulk') arrival or departure is represented by sharp slopes (or jagged lines) on the graph.

Reference: 'Principles of product development flow' by D. Reinertsen

Metrics

Examples of *Metrics & Progress Reports*, generated from electronic tools, by Agile Teams

Note*: Information inside is collected from Jira (real reports, by real teams). However, the guidelines listed on the next page and observations on the pages that follow, apply to* <u>*any*</u> *issue tracking commercial electronic tool (e.g. Rally, Version1, TFS/VSTS, Pivotal Tracker, LeanKit, Trello, etc) or any metrics and reporting, collected and produced manually.*

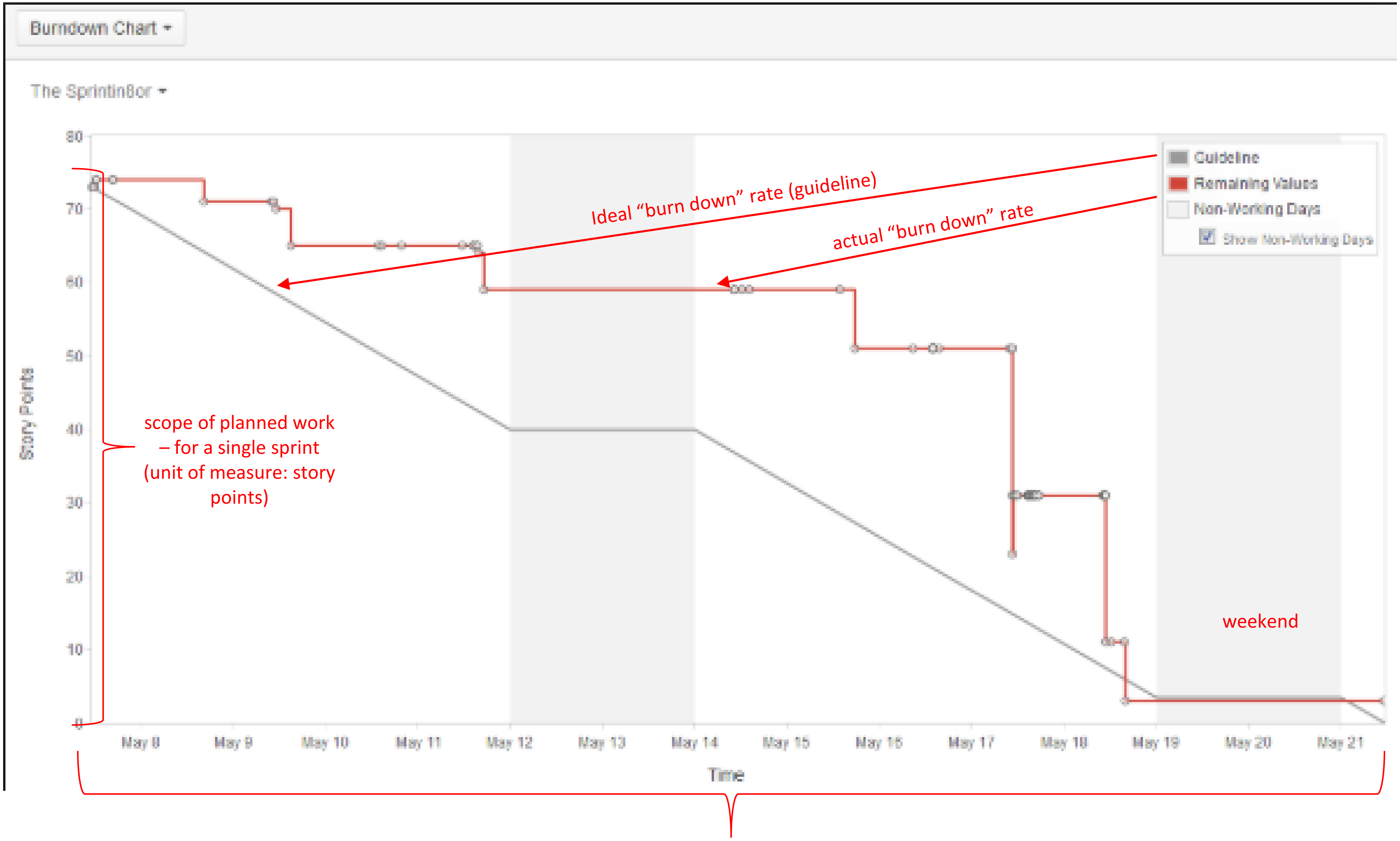


"Not everything that counts can be counted, and not everything that can be counted counts."
-Albert Einstein

Please, make a note of the following Guidelines:

- Understand that Accuracy != Precision
- Understand that Reliability of Forecasting depends on:
  - Accuracy of underlying data
  - Number of available datasets
  - Time frames (time period covered by historical data)
- **Understand the process** (Scrum, Kanban, ScrumBan, etc)
- Understand mechanics of the tool you use (why it works the way it works)
- Understand that Accuracy of estimation/sizing depends on degree of: individuals' participation, T-shaping, swarming, personal safety
- **Try always relate metrics/numbers to underlining root causes: events, behaviors, dynamics, team & organizational design.**
- **Try always apply System Thinking. Don't think linearly and look at superficial indicators only.**

Sprint Burn-Down Chart - *Looking OK*



Ideal "burn down" rate (guideline)

actual "burn down" rate

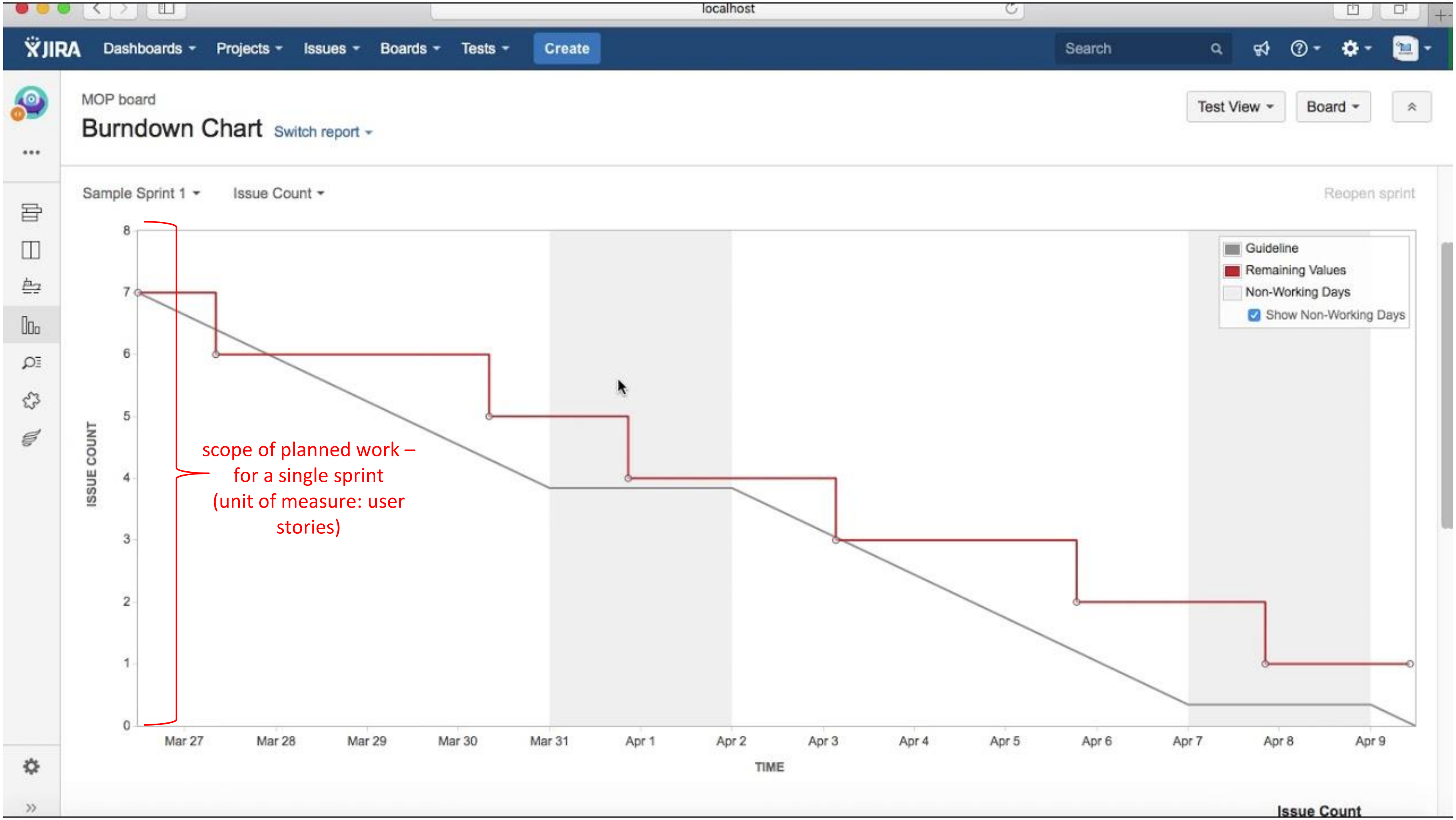
scope of planned work – for a single sprint (unit of measure: story points)

weekend

sprint length

Graph annotations (in red) may require interpretation.

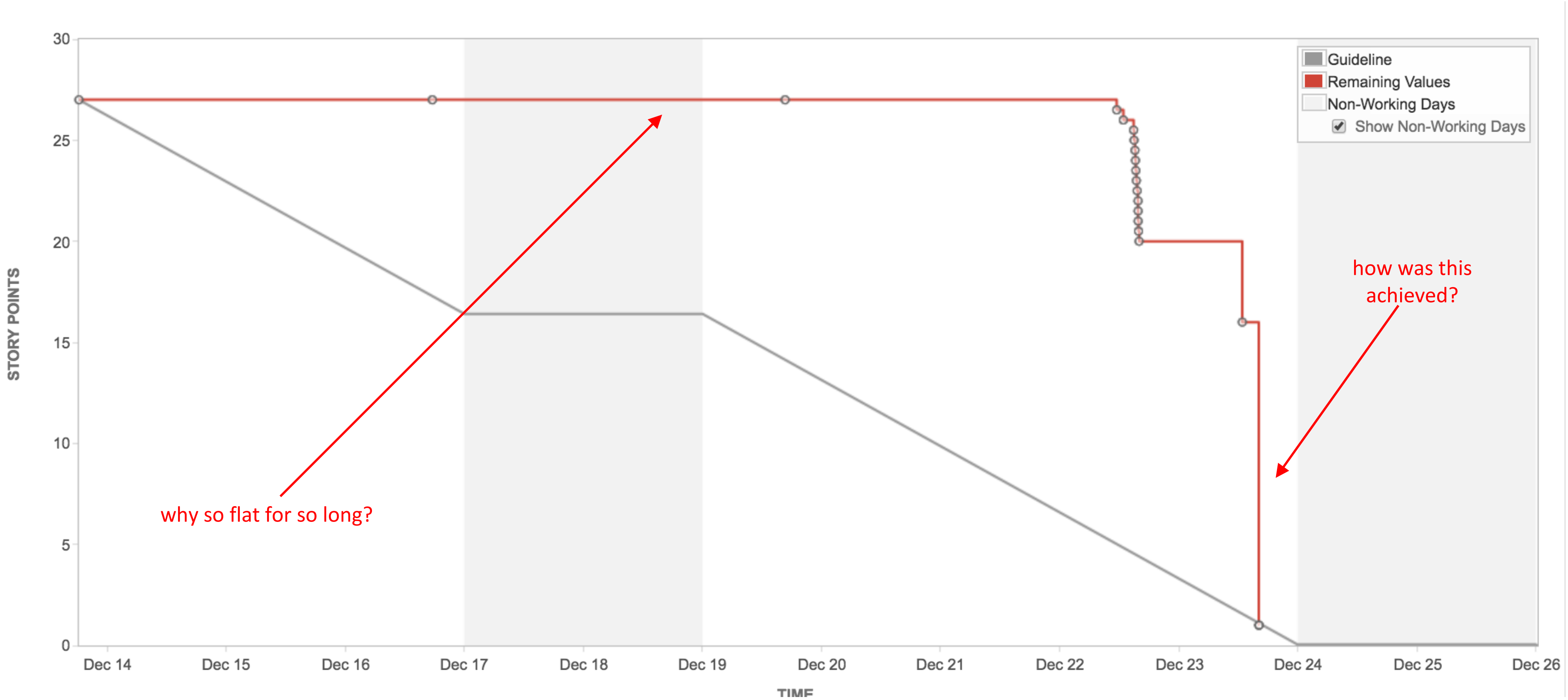
Sprint Burn-Down Chart – *Looking Really Nice*



scope of planned work – for a single sprint (unit of measure: user stories)

Graph annotations (in red) may require interpretation.

Sprint Burn-Down Chart – *Looking Worry*

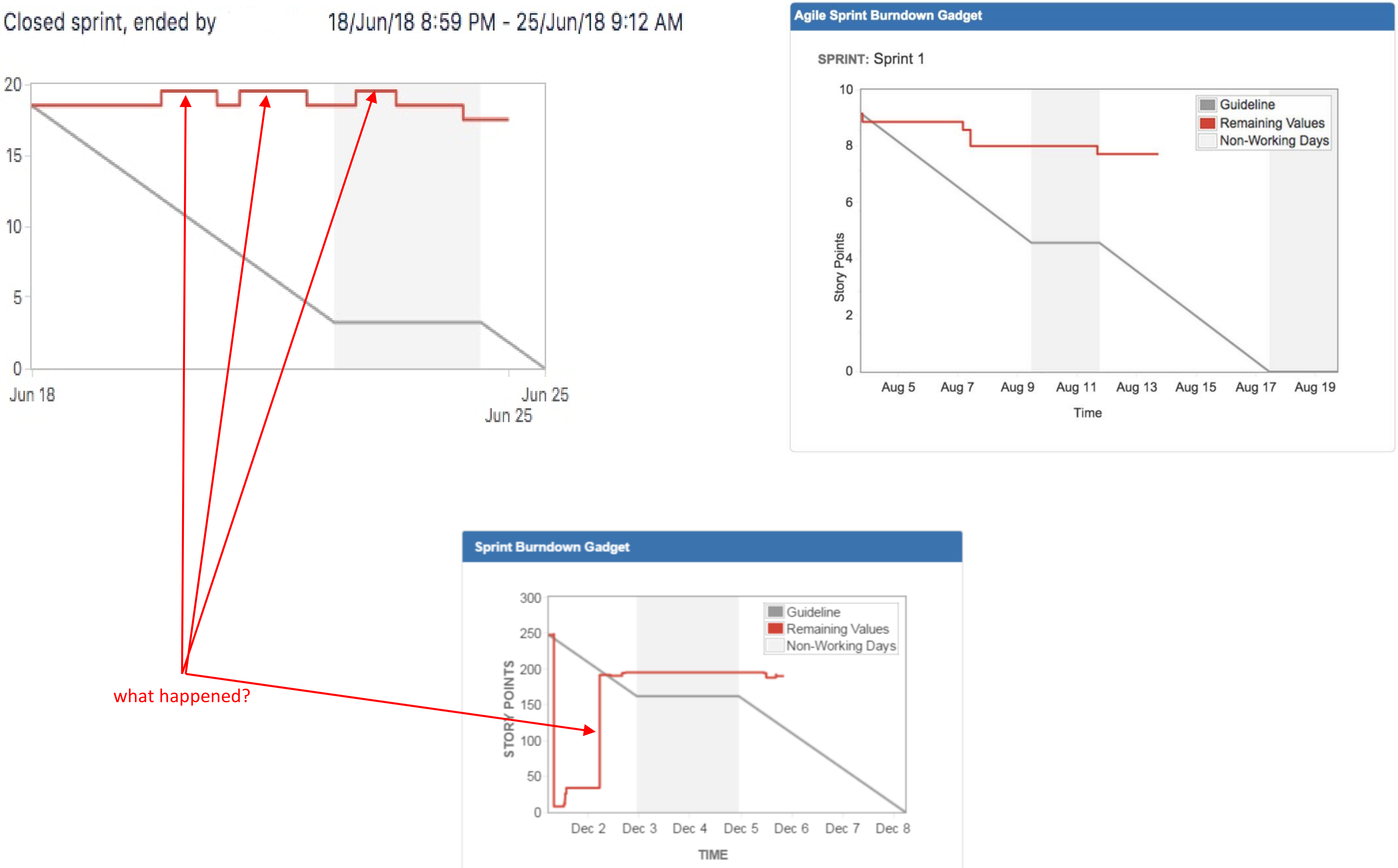


why so flat for so long?

how was this achieved?

Graph annotations (in red) may require interpretation.

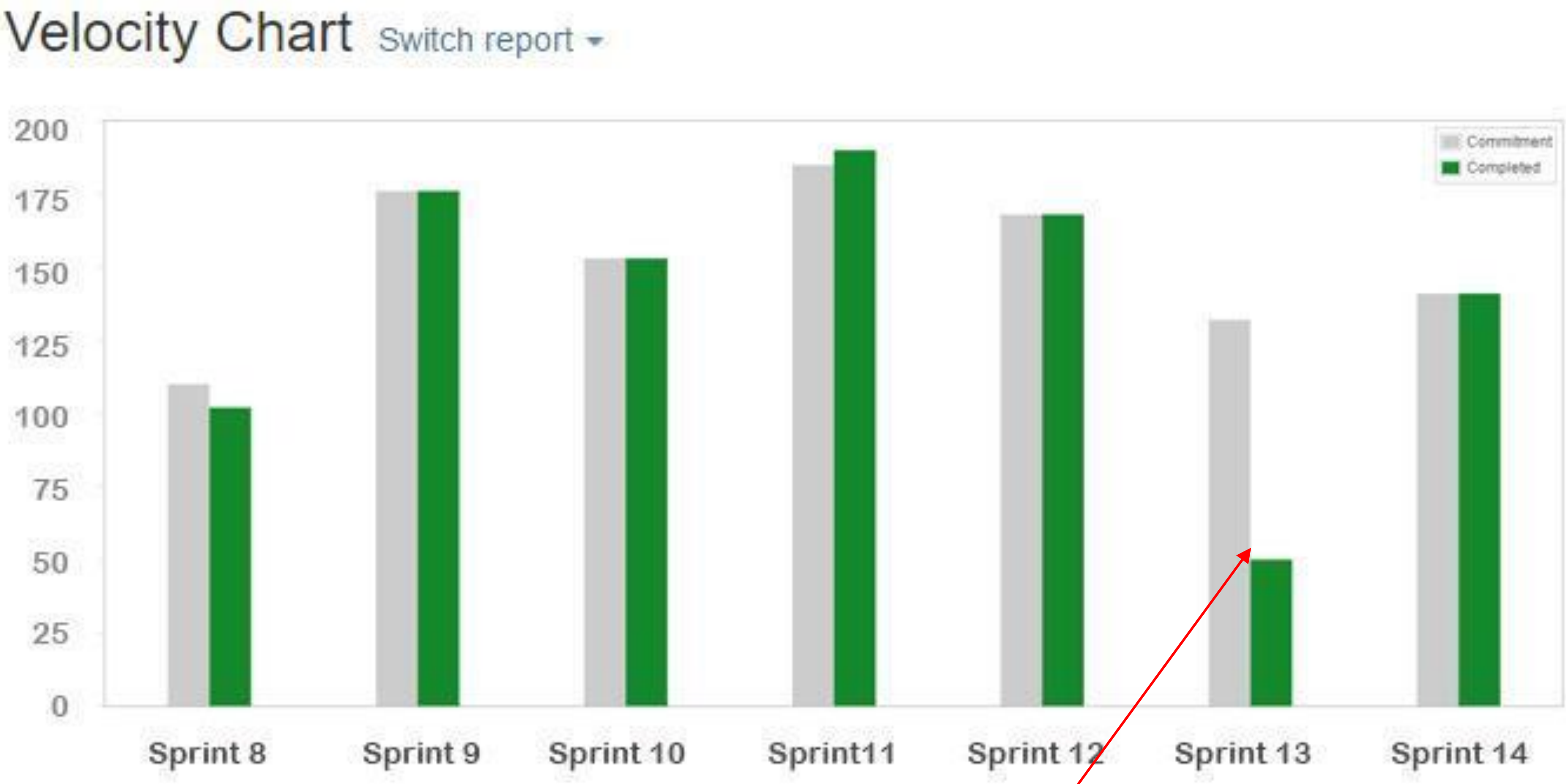
Sprint Burn-Down Chart – *Looking Troubling*



Closed sprint, ended by 18/Jun/18 8:59 PM - 25/Jun/18 9:12 AM

Graph annotations (in red) may require interpretation.

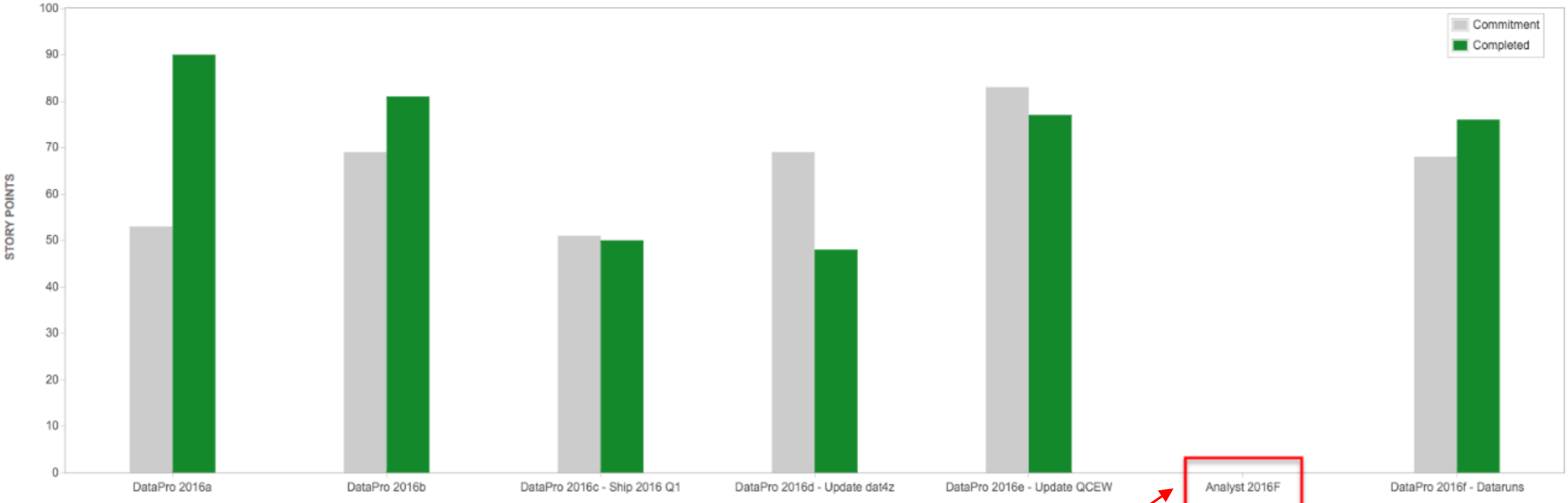
Historical Velocity Report – *Looking Nice*

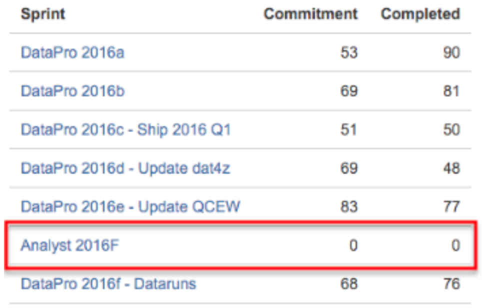


what
happened?

Graph annotations (in red) may require interpretation.

Historical Velocity Report – *Looking Questionable*



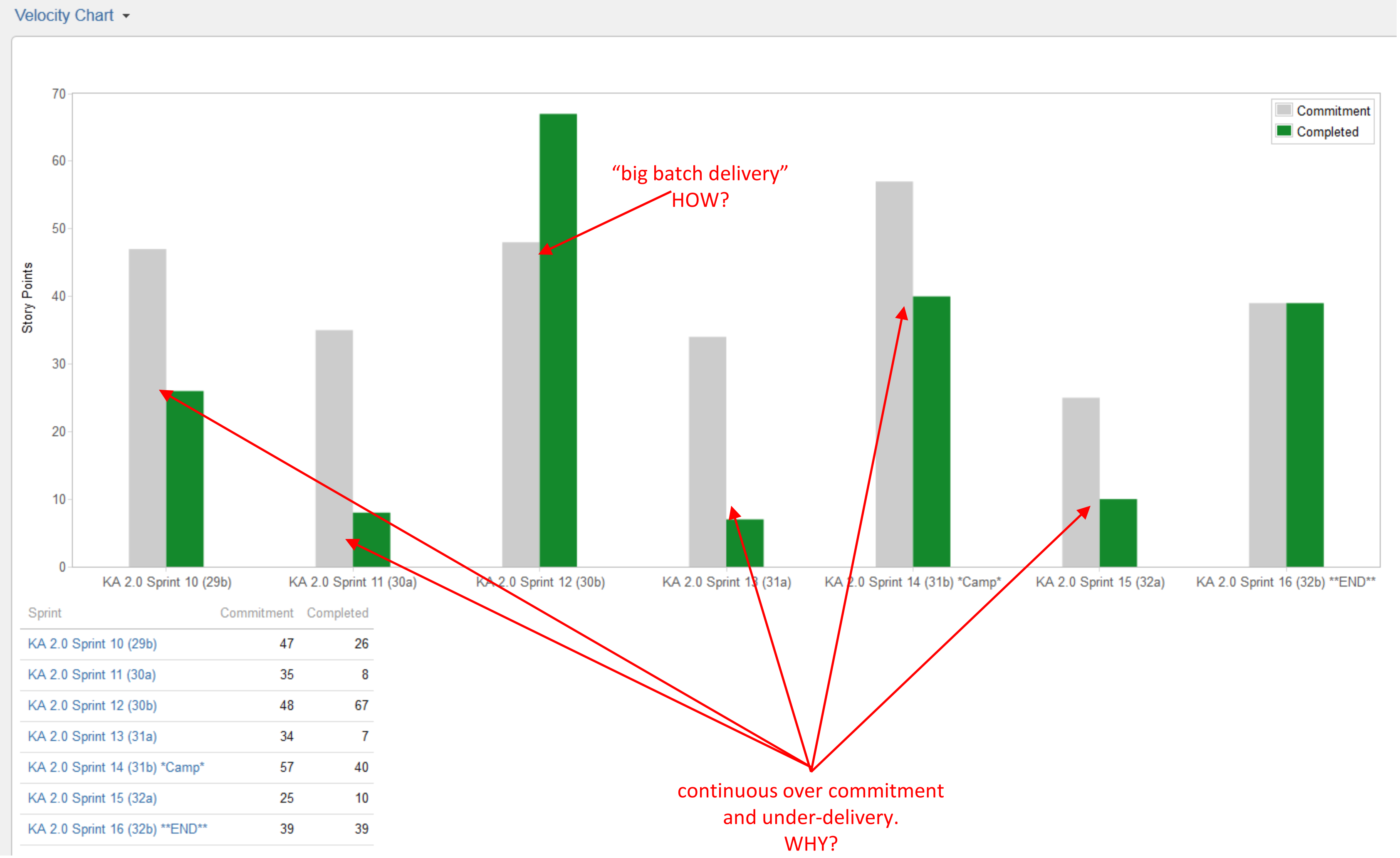


Sprint	Commitment	Completed
DataPro 2016a	53	90
DataPro 2016b	69	81
DataPro 2016c - Ship 2016 Q1	51	50
DataPro 2016d - Update dat4z	69	48
DataPro 2016e - Update QCEW	83	77
Analyst 2016F	0	0
DataPro 2016f - Dataruns	68	76

what
happened?

Graph annotations (in red) may require interpretation.

Historical Velocity Report – *Looking Troubling*



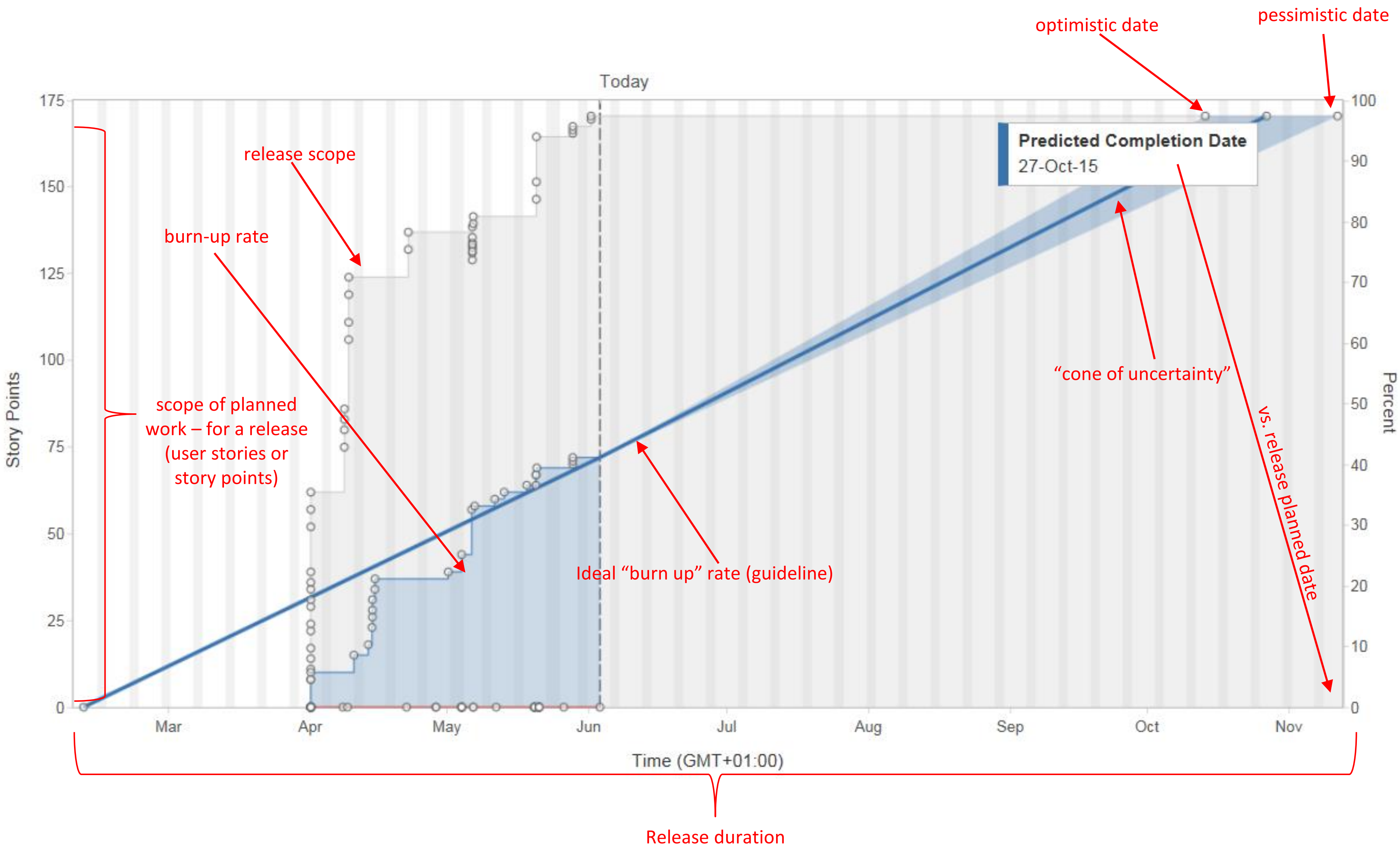
Sprint	Commitment	Completed
KA 2.0 Sprint 10 (29b)	47	26
KA 2.0 Sprint 11 (30a)	35	8
KA 2.0 Sprint 12 (30b)	48	67
KA 2.0 Sprint 13 (31a)	34	7
KA 2.0 Sprint 14 (31b) *Camp*	57	40
KA 2.0 Sprint 15 (32a)	25	10
KA 2.0 Sprint 16 (32b) **END**	39	39

"big batch delivery"
HOW?

continuous over commitment
and under-delivery.
WHY?

Graph annotations (in red) may require interpretation.

Version (Release) Report - *Looking OK*



optimistic date

pessimistic date

release scope

burn-up rate

scope of planned work – for a release (user stories or story points)

“cone of uncertainty”

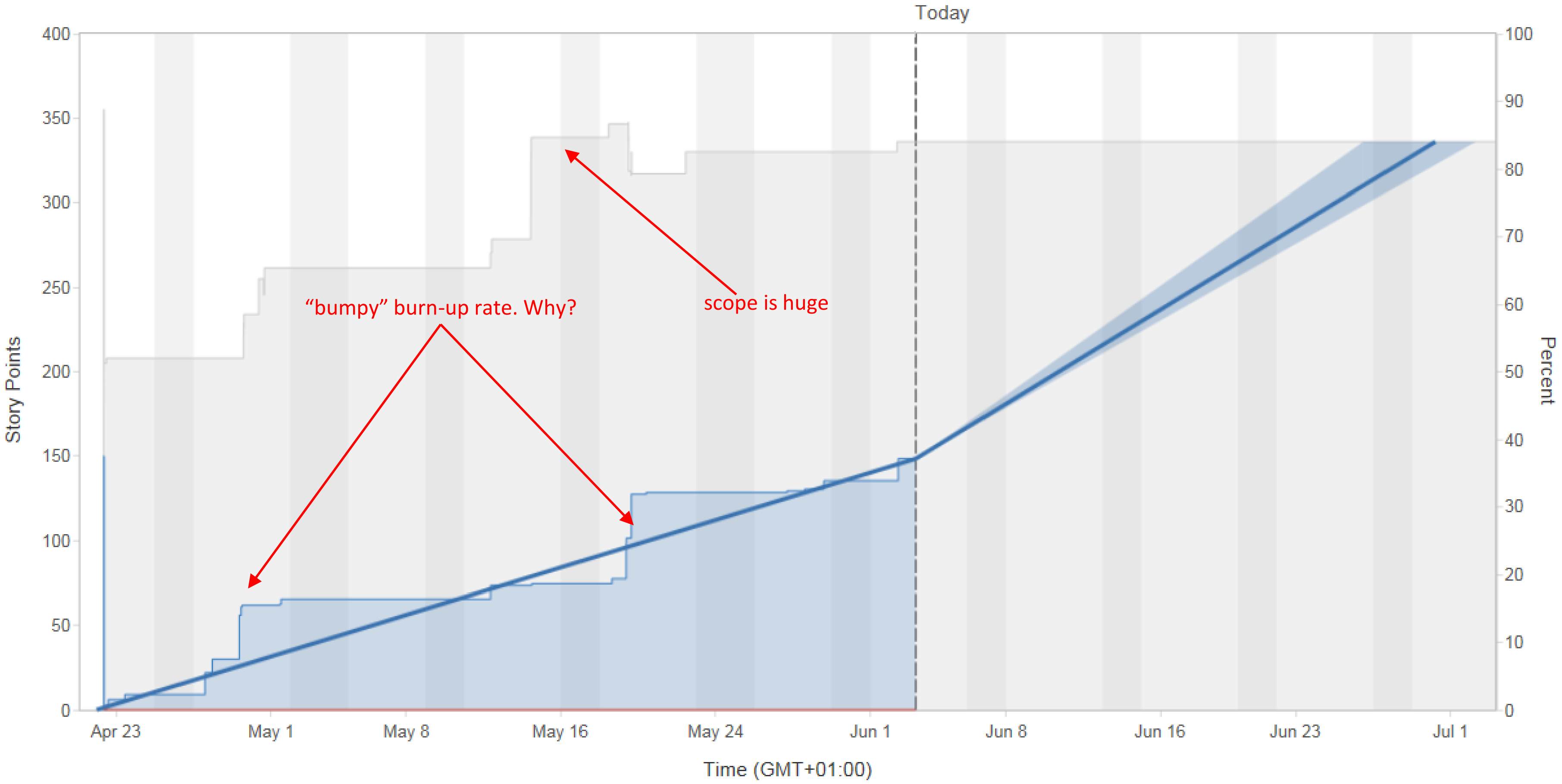
vs. release planned date

Ideal “burn up” rate (guideline)

Release duration

Graph annotations (in red) may require interpretation.

Version (Release) Report – *Looking Questionable*

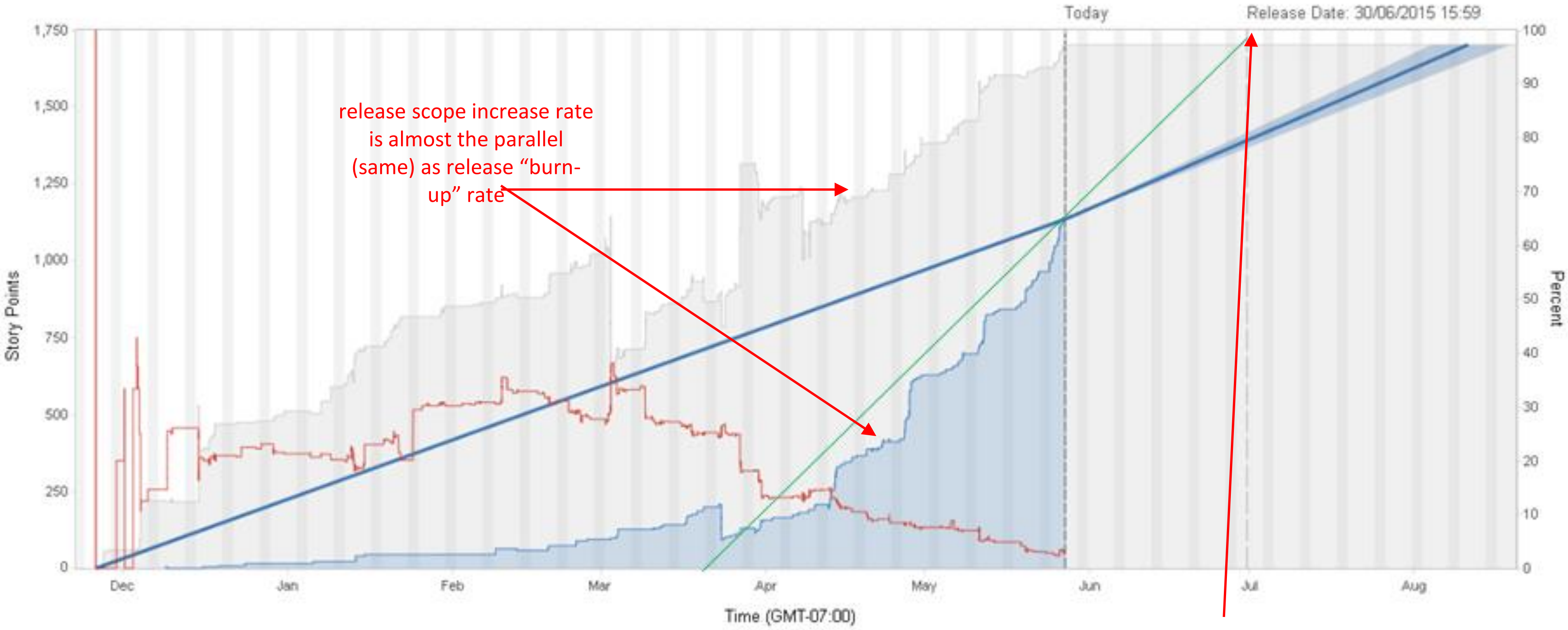


“bumpy” burn-up rate. Why?

scope is huge

Graph annotations (in red) may require interpretation.

Version (Release) Report – *Looking Good*

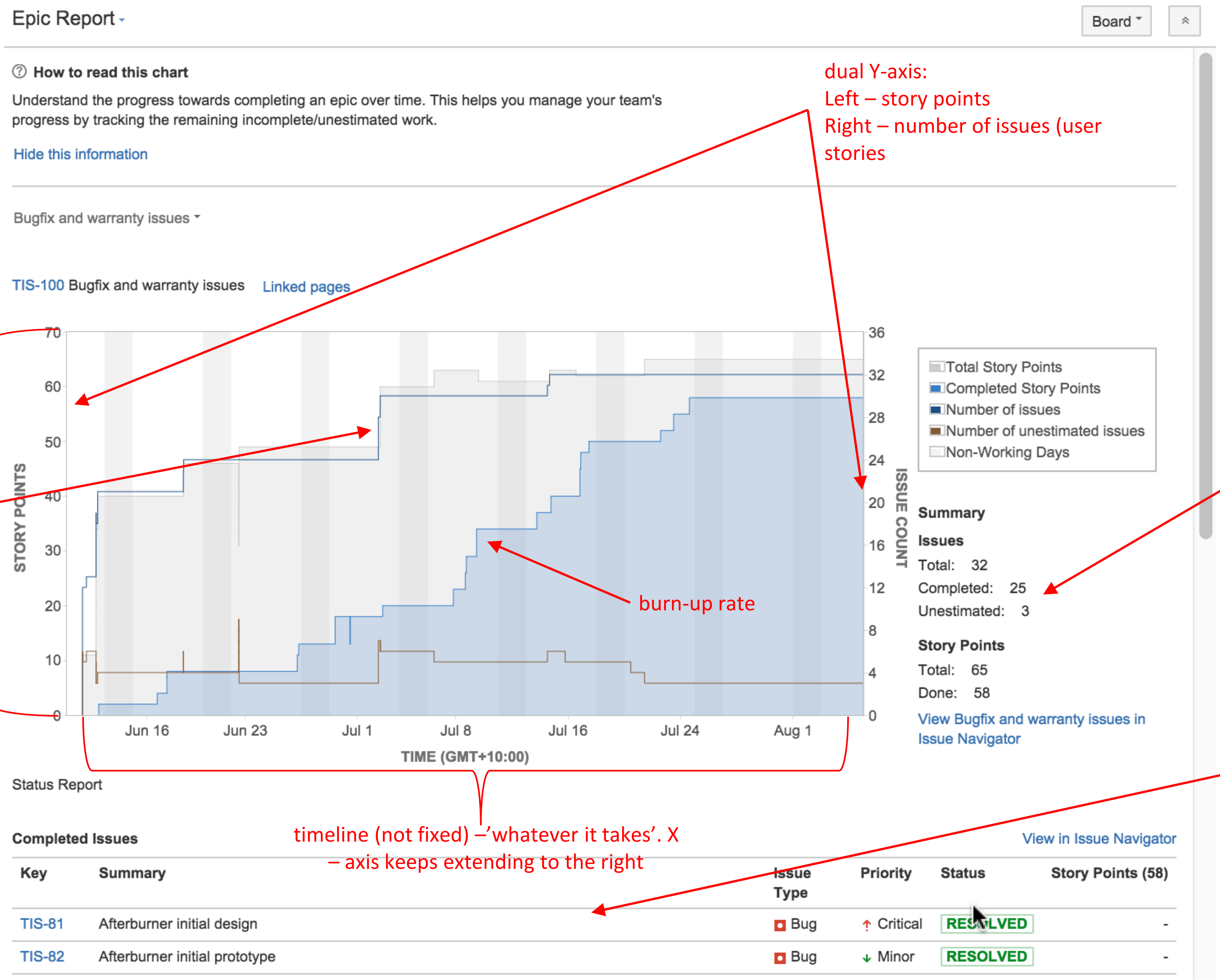


release scope increase rate is almost the parallel (same) as release "burn-up" rate

Resetting a forecasted date, based on changed 'burn up rate' (not every tool allows it)

Graph annotations (in red) may require interpretation.

Epic Report – *Looking OK*



Epic Report ▾ Board ▾

How to read this chart

Understand the progress towards completing an epic over time. This helps you manage your team's progress by tracking the remaining incomplete/unestimated work.

Hide this information

Bugfix and warranty issues ▾

TIS-100 Bugfix and warranty issues Linked pages

Summary

Issues

Total: 32

Completed: 25

Unestimated: 3

Story Points

Total: 65

Done: 58

View Bugfix and warranty issues in Issue Navigator

Status Report

Completed Issues View in Issue Navigator

Key	Summary	Issue Type	Priority	Status	Story Points (58)
TIS-81	Afterburner initial design	Bug	↑ Critical	RESOLVED	-
TIS-82	Afterburner initial prototype	Bug	↓ Minor	RESOLVED	-

dual Y-axis:
Left – story points
Right – number of issues (user stories

scope of epic (roll-up collection of all underlying work items)

burn-up rate

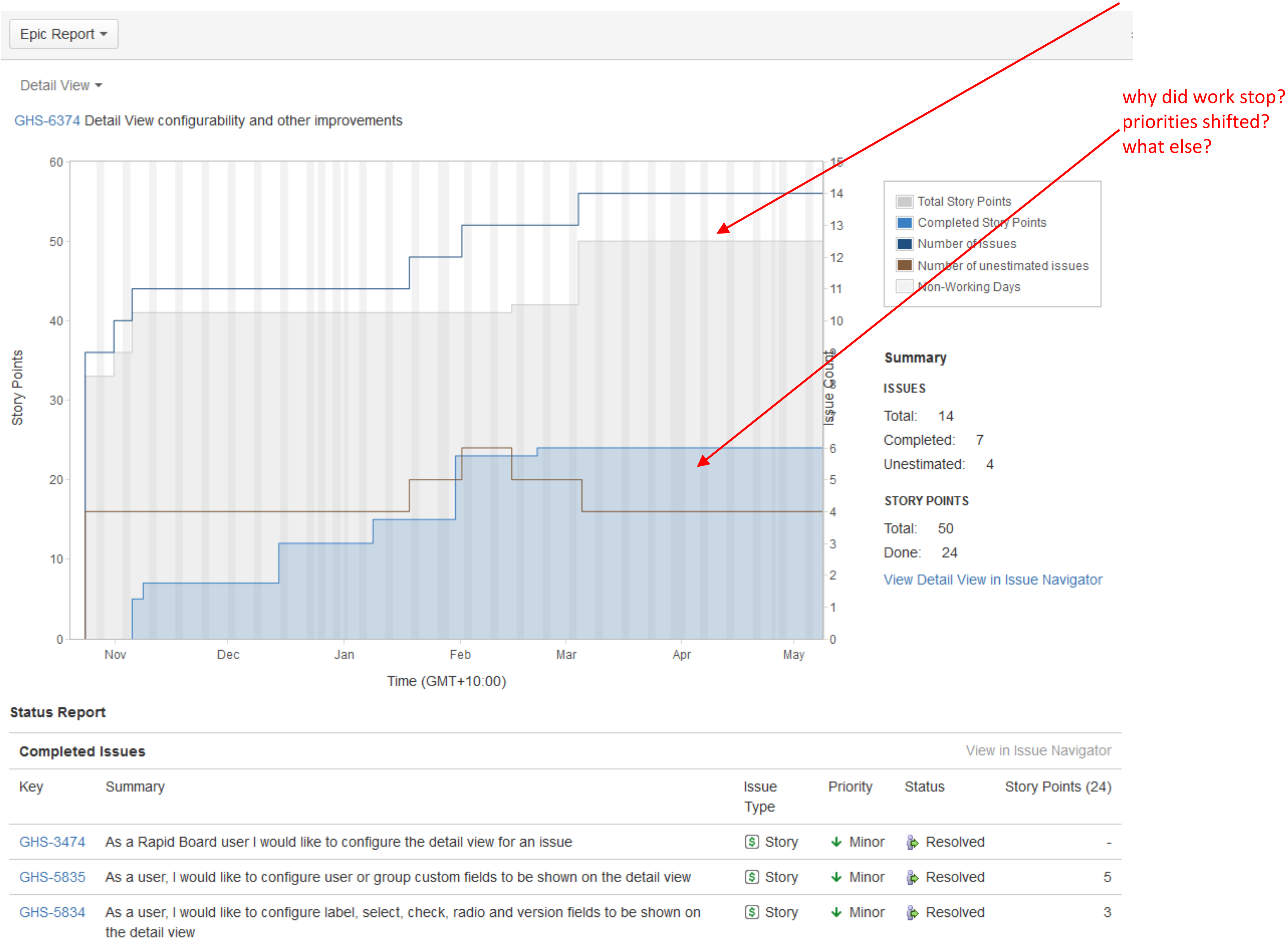
additional (sliced) roll-up data

detailed history

timeline (not fixed) –'whatever it takes'. X – axis keeps extending to the right

Graph annotations (in red) may require interpretation.

Epic Report – *Looking Odd. Requires Explanation*



GHS-6374 Detail View configurability and other improvements

Summary

ISSUES

Total: 14

Completed: 7

Unestimated: 4

STORY POINTS

Total: 50

Done: 24

Status Report

Completed Issues

Key	Summary	Issue Type	Priority	Status	Story Points (24)
GHS-3474	As a Rapid Board user I would like to configure the detail view for an issue	Story	↓ Minor	Resolved	-
GHS-5835	As a user, I would like to configure user or group custom fields to be shown on the detail view	Story	↓ Minor	Resolved	5
GHS-5834	As a user, I would like to configure label, select, check, radio and version fields to be shown on the detail view	Story	↓ Minor	Resolved	3

is this scope still valid?

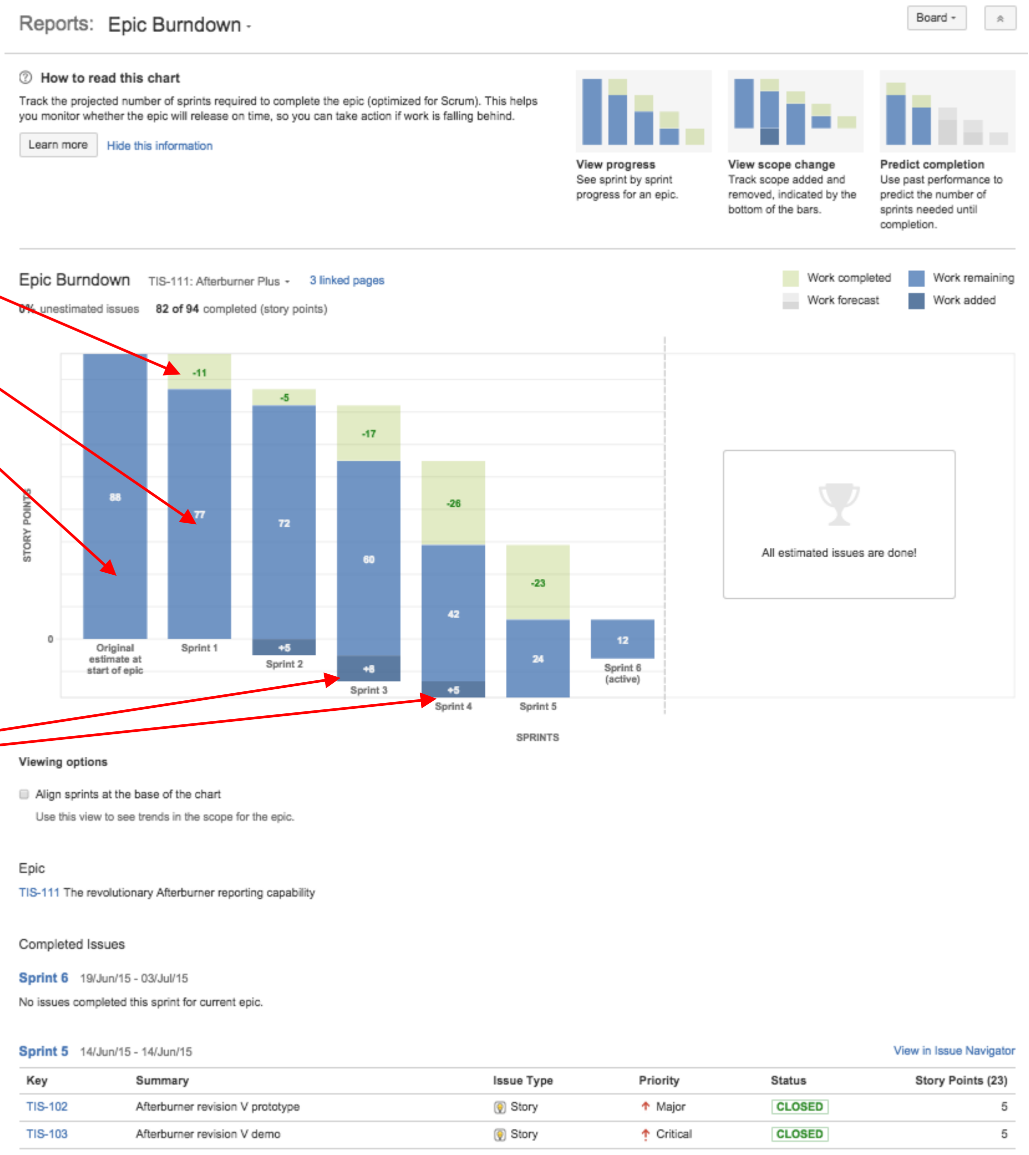
why did work stop?
priorities shifted?
what else?

Graph annotations (in red) may require interpretation.

Epic Burn Down – *Looking Nice*

As a team starts sprinting, initial Epic scope starts decreasing, whereas amount of completed work (user stories) that belong to an epic, starts growing.

Expanding epic scope will require more sprints to complete work and achieve a strategic goal



Reports: Epic Burndown ▾

Board ▾

How to read this chart

Track the projected number of sprints required to complete the epic (optimized for Scrum). This helps you monitor whether the epic will release on time, so you can take action if work is falling behind.

Learn more | Hide this information

View progress
See sprint by sprint progress for an epic.

View scope change
Track scope added and removed, indicated by the bottom of the bars.

Predict completion
Use past performance to predict the number of sprints needed until completion.

Epic Burndown TIS-111: Afterburner Plus ▾ 3 linked pages

0% unestimated issues **82 of 94** completed (story points)

Viewing options

☐ Align sprints at the base of the chart

Use this view to see trends in the scope for the epic.

Epic

TIS-111 The revolutionary Afterburner reporting capability

Completed Issues

Sprint 6 19/Jun/15 - 03/Jul/15

No issues completed this sprint for current epic.

Sprint 5 14/Jun/15 - 14/Jun/15

View in Issue Navigator

Key	Summary	Issue Type	Priority	Status	Story Points (23)
TIS-102	Afterburner revision V prototype	Story	↑ Major	CLOSED	5
TIS-103	Afterburner revision V demo	Story	↑ Critical	CLOSED	5

Graph annotations (in red) may require interpretation.

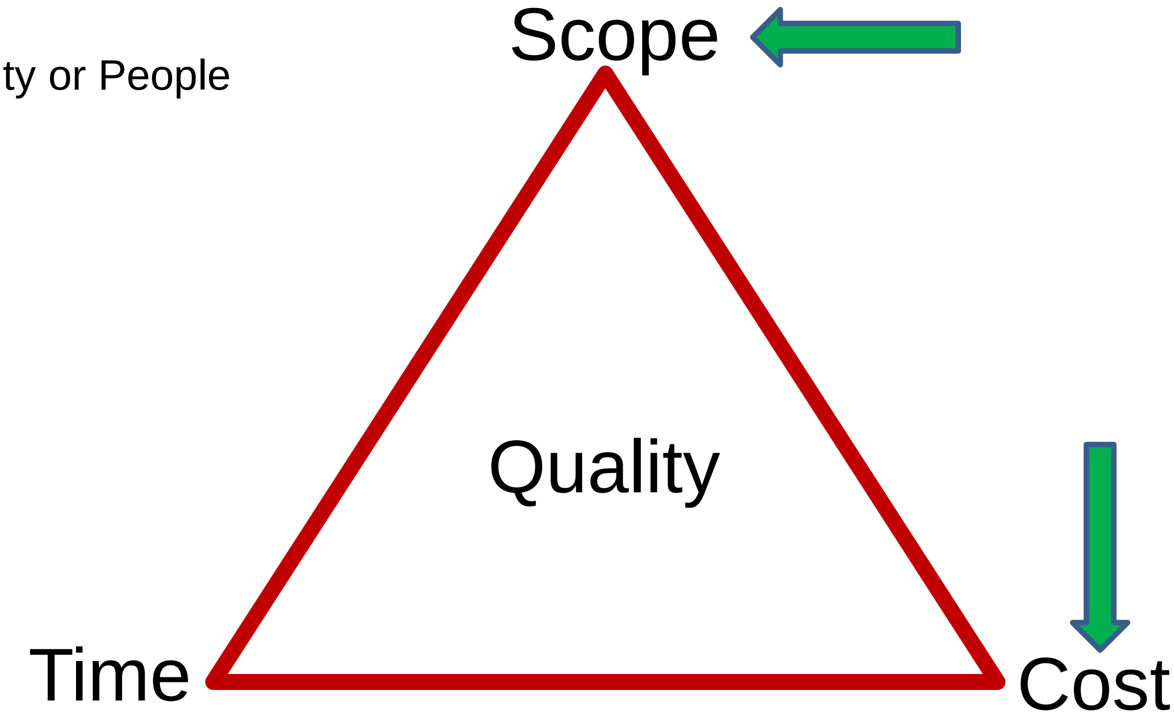
Agile Budgeting & Finances:
unveiling conventional management mistakes

Inspired, Fueled and Driven by:
“Implementing Beyond Budgeting: Unlocking the Performance Potential”,
by Bjarte Bogsnes

Agile Budgeting

Triple Constraint Triangle of Conventional Management

- Fixed Scope, Timeline and Cost
- Rigid and inflexible
- Sacrifices either Quality or People



Scope

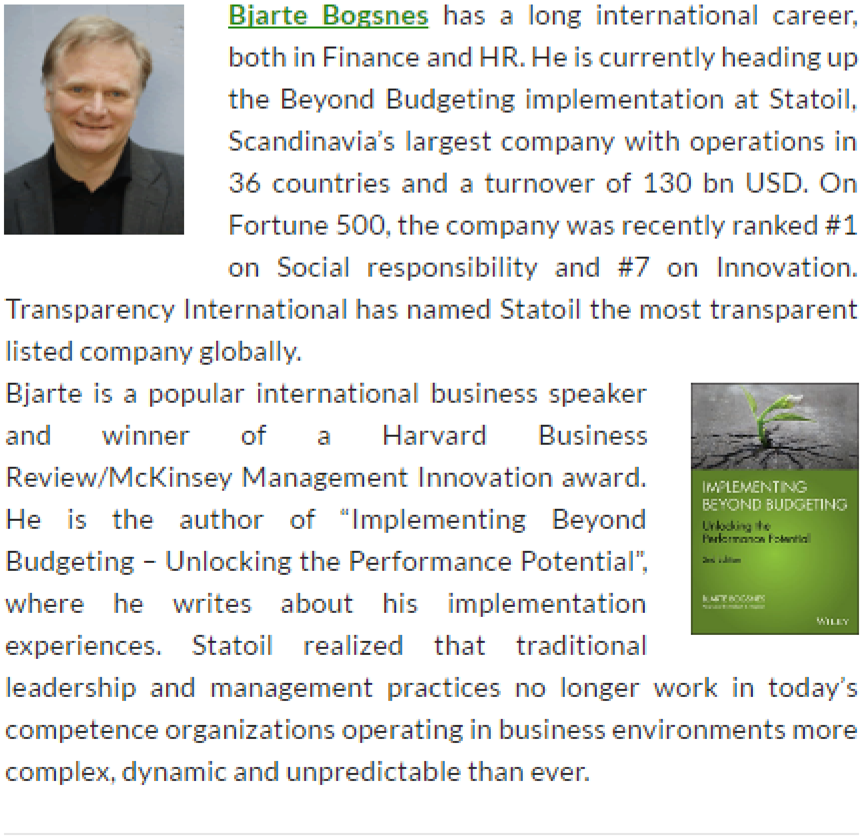
Quality

Time

Cost

Agile Budgeting

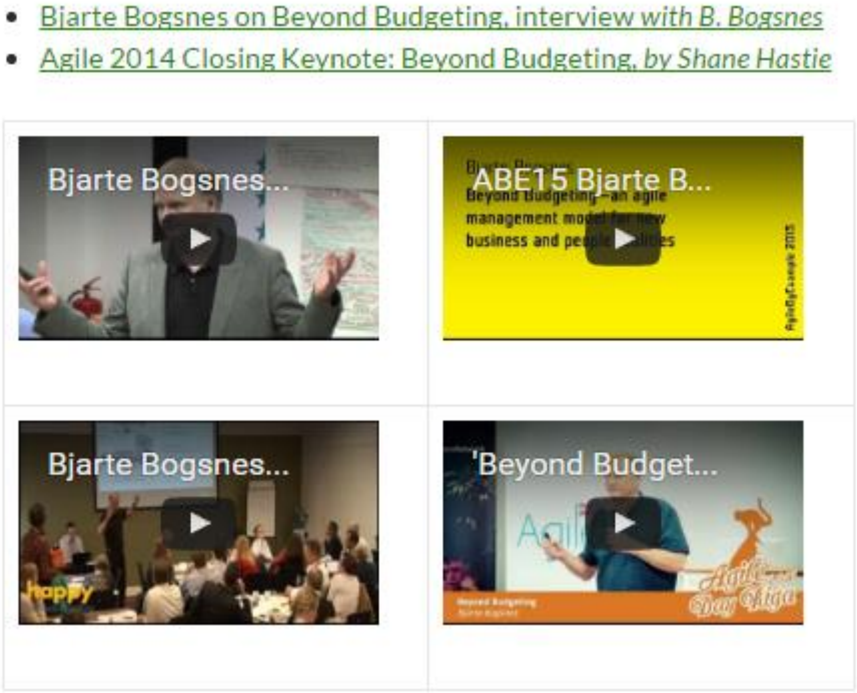
About Bjarte Bogsnes



Bjarte Bogsnes has a long international career, both in Finance and HR. He is currently heading up the Beyond Budgeting implementation at Statoil, Scandinavia's largest company with operations in 36 countries and a turnover of 130 bn USD. On Fortune 500, the company was recently ranked #1 on Social responsibility and #7 on Innovation. Transparency International has named Statoil the most transparent listed company globally.

Bjarte is a popular international business speaker and winner of a Harvard Business Review/McKinsey Management Innovation award. He is the author of "Implementing Beyond Budgeting – Unlocking the Performance Potential", where he writes about his implementation experiences. Statoil realized that traditional leadership and management practices no longer work in today's competence organizations operating in business environments more complex, dynamic and unpredictable than ever.

IMPLEMENTING BEYOND BUDGETING
Unlocking the Performance Potential
2nd edition
BJARTE BOGSNES
WILEY



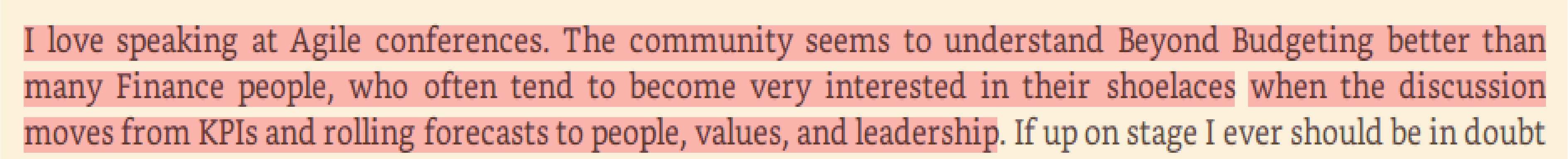
- Bjarte Bogsnes on Beyond Budgeting, interview *with B. Bogsnes*
- Agile 2014 Closing Keynote: Beyond Budgeting, *by Shane Hastie*

Bjarte Bogsnes...
ABE15 Bjarte B...
Beyond Budgeting—an agile management model for new business and people realities
Bjarte Bogsnes...
'Beyond Budget...
Agile Day Riga

QUOTES FROM: IMPLEMENTING BEYOND BUDGETING AT:
HTTP://WWW.KEYSTEPSTOSUCCESS.COM/2016/08/IMPLEMENTING-BEYOND-BUDGETING/

Agile Budgeting

Why Bjarte turns to Agile Community?

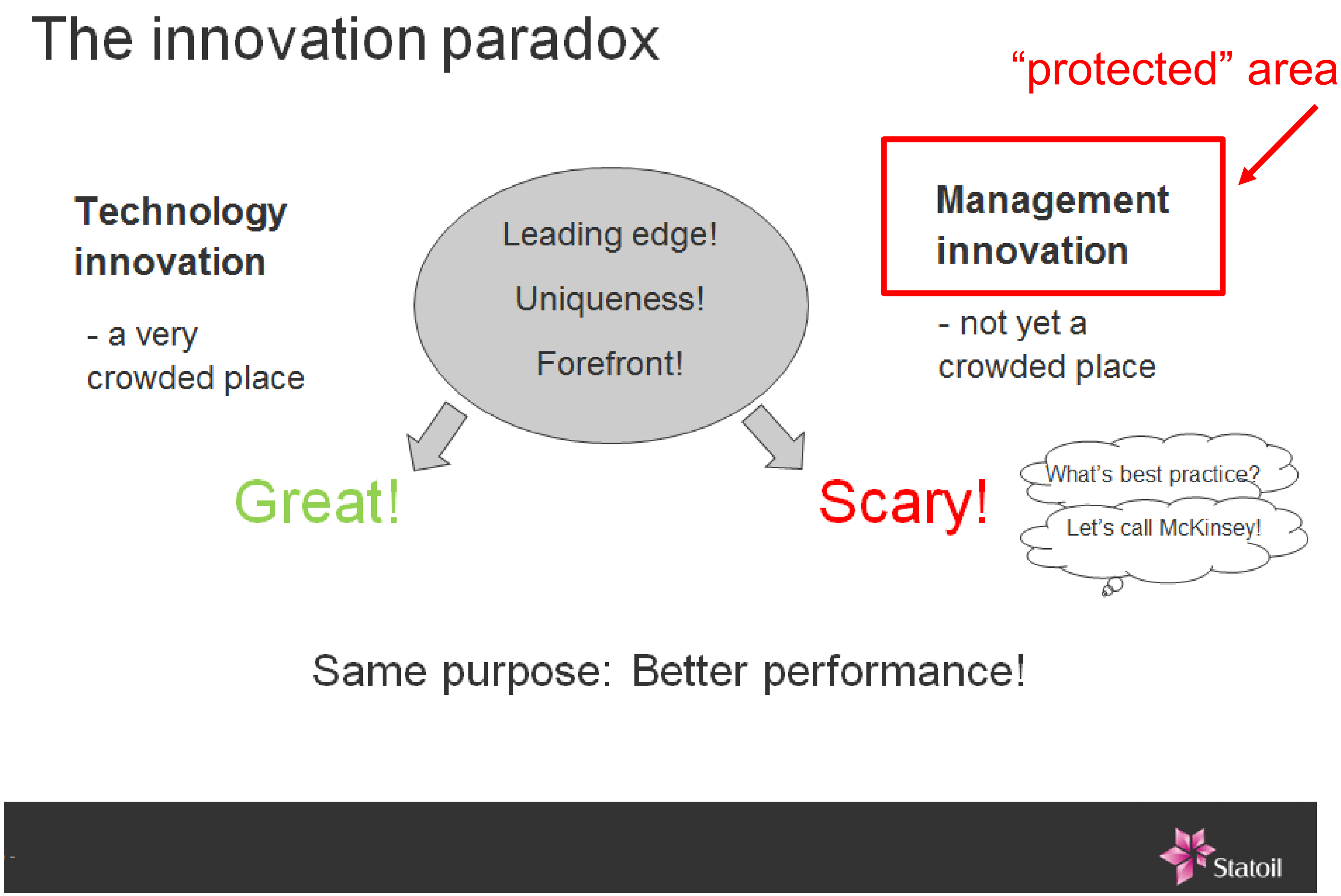


I love speaking at Agile conferences. The community seems to understand Beyond Budgeting better than many Finance people, who often tend to become very interested in their shoelaces when the discussion moves from KPIs and rolling forecasts to people, values, and leadership. If up on stage I ever should be in doubt

Quote from “Implementing Beyond Budgeting ” by Bjarte Bogsnes

Agile Budgeting

“Untapped” Area for Improvements



The innovation paradox

Technology innovation

- a very crowded place

Leading edge!

Uniqueness!

Forefront!

Management innovation

- not yet a crowded place

Great!

Scary!

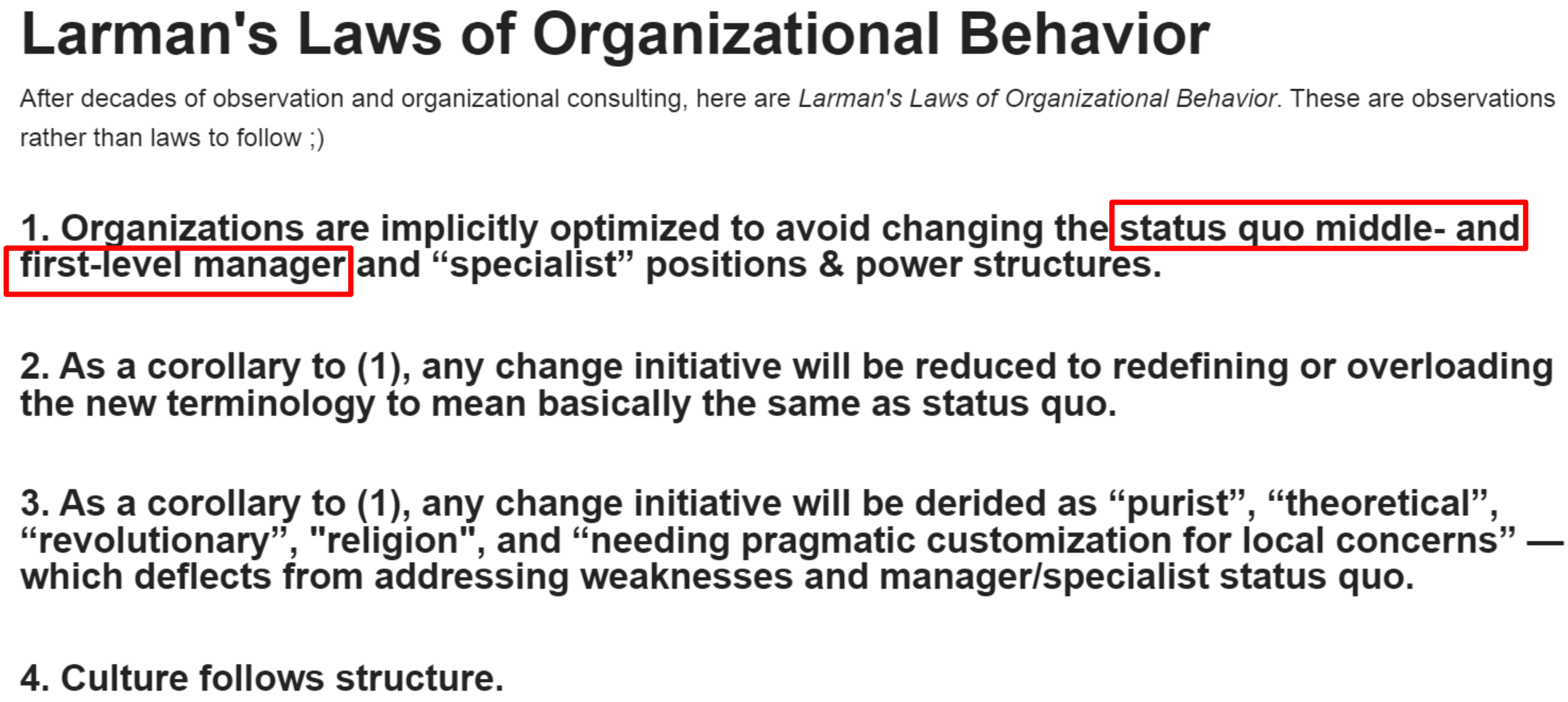
Same purpose: Better performance!

“protected” area??

From: Beyond Budgeting - an agile management model for new business and people realities. Ambition to Action - the Statoil journey. Presented by Bjarte Bogsnes

Agile Budgeting

Why is Management Space So Vacant?



Larman's Laws of Organizational Behavior

After decades of observation and organizational consulting, here are *Larman's Laws of Organizational Behavior*. These are observations rather than laws to follow ;)

1. Organizations are implicitly optimized to avoid changing the status quo middle- and first-level manager and "specialist" positions & power structures.

2. As a corollary to (1), any change initiative will be reduced to redefining or overloading the new terminology to mean basically the same as status quo.

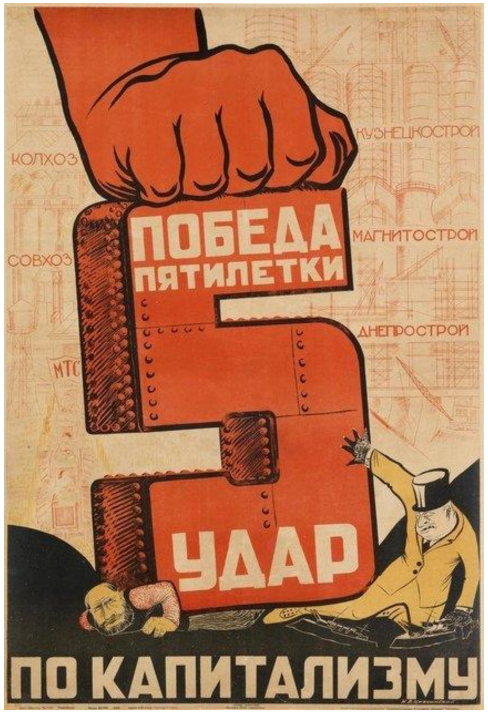
3. As a corollary to (1), any change initiative will be derided as "purist", "theoretical", "revolutionary", "religion", and "needing pragmatic customization for local concerns" — which deflects from addressing weaknesses and manager/specialist status quo.

4. Culture follows structure.

Agile Budgeting

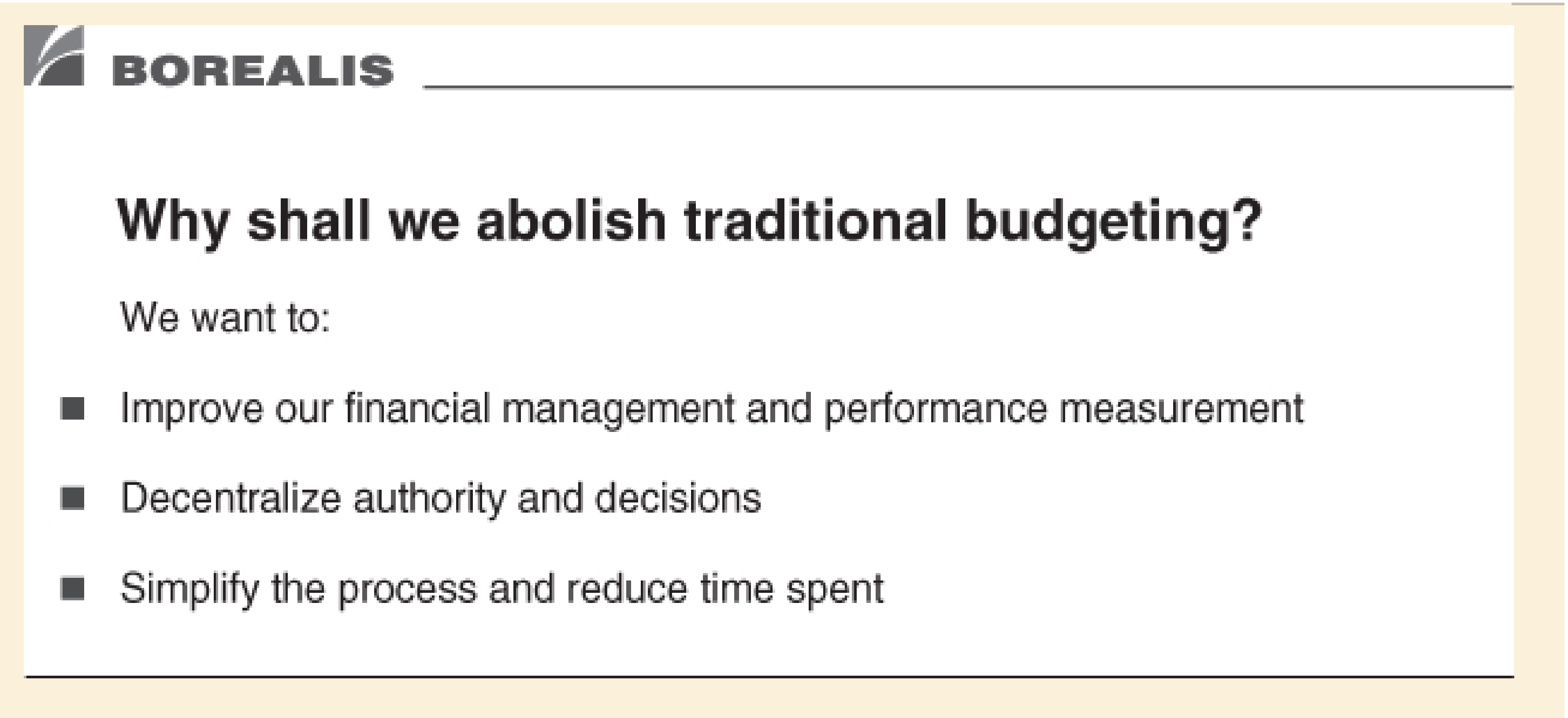
What does Classic Budgeting Remind Us Of?

- Classing Budgeting is like **5-year Soviet Economy Plan**
  - “Traditional management has more in common with how the Soviet Union was run than with the principles and beliefs of a true democracy.”
  - Traditional management fears transparency because it threatens control



Agile Budgeting

Why did Borealis Abolish Traditional Budgeting?



BOREALIS

Why shall we abolish traditional budgeting?

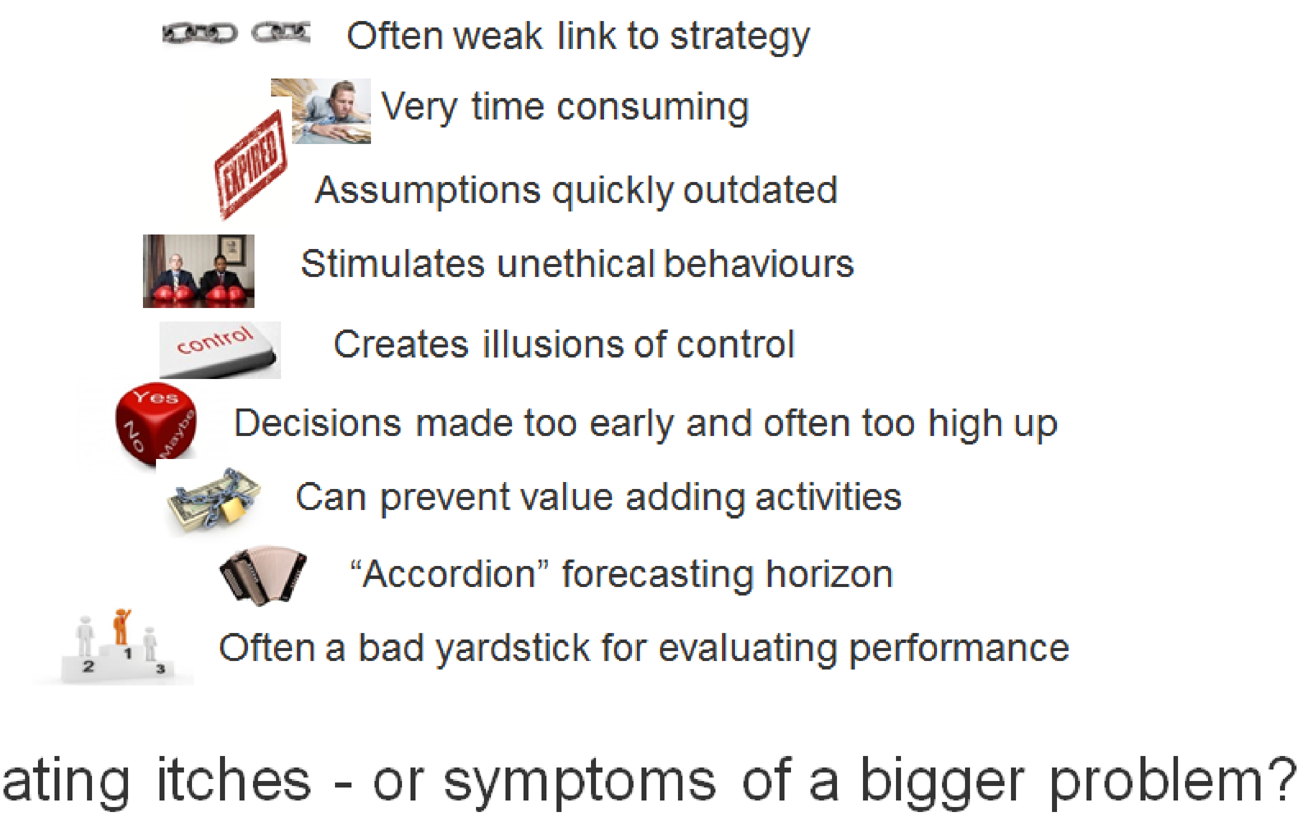
We want to:

- Improve our financial management and performance measurement
- Decentralize authority and decisions
- Simplify the process and reduce time spent

From: Beyond Budgeting - an agile management model for new business and people realities. Ambition to Action - the Statoil journey. Presented by Bjarte Bogsnes

Agile Budgeting

What Were Adverse Effects of Traditional Budgeting?



- Often weak link to strategy
- Very time consuming
- Assumptions quickly outdated
- Stimulates unethical behaviours
- Creates illusions of control
- Decisions made too early and often too high up
- Can prevent value adding activities
- “Accordion” forecasting horizon
- Often a bad yardstick for evaluating performance

Irritating itches - or symptoms of a bigger problem?

From: Beyond Budgeting - an agile management model for new business and people realities. Ambition to Action - the Statoil journey. Presented by Bjarte Bogsnes

Agile Budgeting

Decentralize Authority and Decisions

There is no
traffic cop inside

Screenshot(s) from "Implementing Beyond Budgeting " by Bjarte Bogsnes

Agile Budgeting

Floor is also Ceiling (Unfortunately)

- Cost budgets are spent even when initial assumptions are lowered
- Managers do not cheat intentionally: system makes them to do so

Agile Budgeting

Senseless Cost Reduction

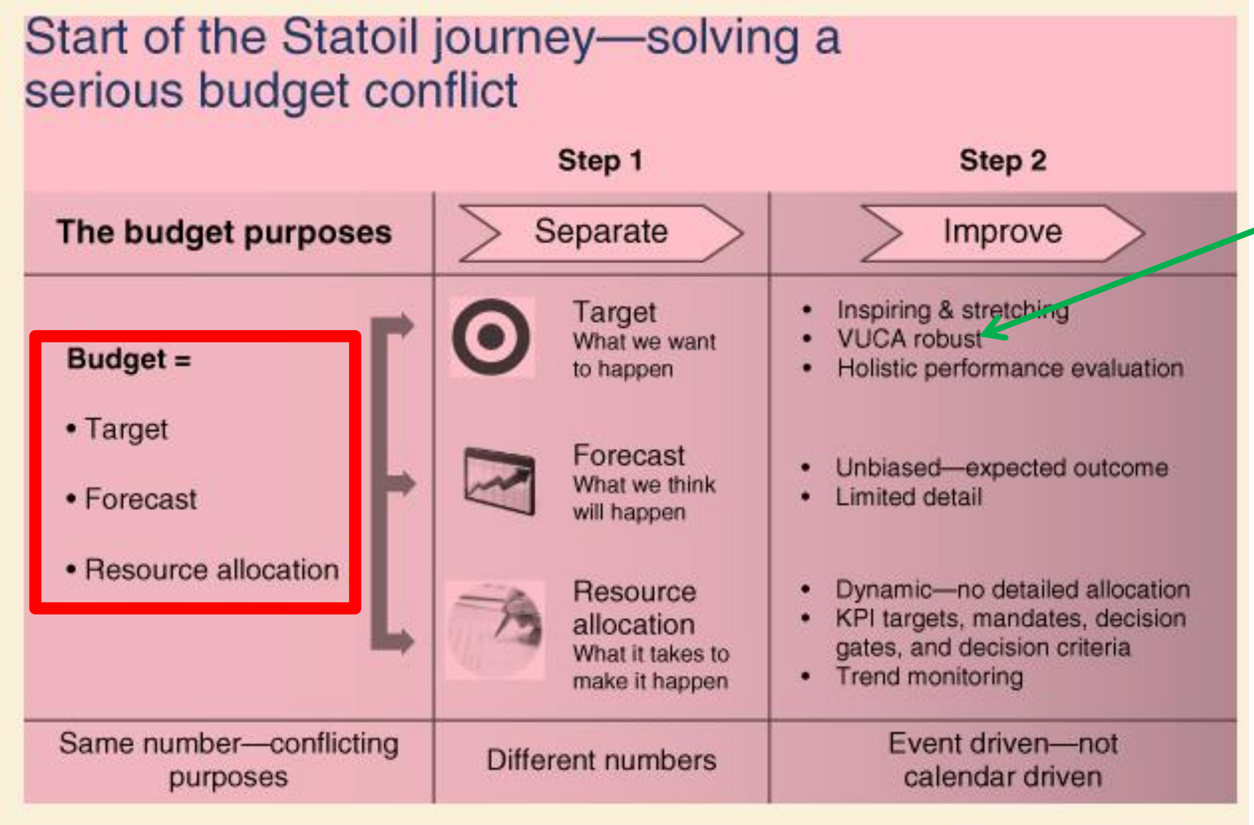
- Lowering Costs:
  - Not the “lowest possible” level
  - But optimal for value creation

__NB__: People are smart, and any system can be gamed if people want to

- Finance people = Controllers
  - Types of Control:
    - Good:
      - Transparency
      - Values-based culture
    - Bad:
      - detailed budgets
      - tight mandates
      - detailed job descriptions
      - rigid organizational structures
      - smartly constructed bonus schemes
      - other Theory X– driven control mechanisms

Agile Budgeting

Resolving Serious Budget Conflict



Start of the Statoil journey—solving a serious budget conflict

	Step 1	Step 2
The budget purposes	Separate	Improve
Budget = • Target • Forecast • Resource allocation	Target What we want to happen	• Inspiring & stretching • VUCA robust • Holistic performance evaluation
	Forecast What we think will happen	• Unbiased—expected outcome • Limited detail
	Resource allocation What it takes to make it happen	• Dynamic—no detailed allocation • KPI targets, mandates, decision gates, and decision criteria • Trend monitoring
Same number—conflicting purposes	Different numbers	Event driven—not calendar driven

VUCA = Volatility, Uncertainty, Complexity, Ambiguity

Screenshot(s) from "Implementing Beyond Budgeting " by Bjarte Bogsnes

- When Forecasts "travel upstairs", they become Targets
- When Targets "travel downstairs", they drive Bonuses

- Low-balling Sales Forecasts
- High-balling Costs Forecasts

Agile Budgeting

Example of System Gaming

- Project **Estimate** (e.g. **1,000**) is the expected cost estimate used in the profitability analysis of the project. Being merely a forecast, this number will continue to live throughout the project.

- The more ambitious **Target** cost (e.g. **900**) is the cost level the project team aims for.

- The **Resource Allocation** estimate (e.g. **1,100**) or the mandate to spend is set higher than the project estimate, *to avoid on average every second project having to come back and ask for more money*. – Ugggh!!!!!!

Agile Budgeting

Forecast vs. Target

Agile Budgeting

Dynamic Forecasting: Who needs It Most?

- Who has a bigger momentum?
- Who is less agile “by size”?
- Who is less maneuverable ?
- Who is more dependent on radar system?
- BTW: “Dynamic” Forecast !=“Rolling” Forecast

<u>NB: </u> *“What is the point of having the world's largest radar screen and the ability to sense and respond instantly, if there is no dynamic resource allocation ensuring that the necessary resources also can be instantly accessed or reallocated, instead of being locked up in a detailed annual budget?”*

Quote from “Implementing Beyond Budgeting ” by Bjarte Bogsnes

Agile Budgeting

“The World Stops on December 31”. Really?

- Forecasting “against the wall,” (accordion forecasting)
- What is better: Business Cycles vs. Calendar Cycles?
- Some Forecasting Tips:
  - Forecasting should be ‘locally owned’
  - Forecasting should be actionable
  - Don’t confuse Forecasts (= EXPECTATION) with Targets (= INSPIRATION)
  - Don’t give a gap between Forecast and Target a negative connotation
  - Base it on trust and the belief that people are mature and responsible
    - *“If we can't be trusted to manage our own travel cost, how can we be trusted when we advise and recommend on million- and billion-dollar projects?”*

Quote from “Implementing Beyond Budgeting ” by Bjarte Bogsnes

Agile Budgeting

KPIs

- Good KPI checklist:
  - Do they measure progress toward strategic objectives?
  - Do they measure real performance?
  - Do they address areas where we want change or improve?
  - Are the KPIs perceived as meaningful at the level they are used?
  - Can data be collected easily?

Agile Budgeting

KPIs

- **Illusion**: "If we don't manage performance, there will be no performance. If we don't develop people, there will be no development."
- **KPI != KPT (Key Performance Truths!)**
- Precision != Relevance (avoid chasing decimals)
- **Albert Einstein': *"Not everything that can be counted counts, and not everything that counts can be counted."***
- KPI "RAGs":
  - When OK?: when used as team-status sharing
  - When Dangerous?: "*when perceived as part of a top-down control-and-reward regime, sometimes triggering gaming and unethical behaviors to change reds to greens*

Quote from "Implementing Beyond Budgeting " by Bjarte Bogsnes

Agile Budgeting

S.M.A.R.T. or S.M.A.R.T. “ER”?

- S.M.A.R.T.
  - Specific
  - Measurable
  - Attainable
  - Relevant
  - Timely

+

- E.R.
  - Ethical
  - Reasonable

© 2010 by Randy Glasbergen / glasbergen.com

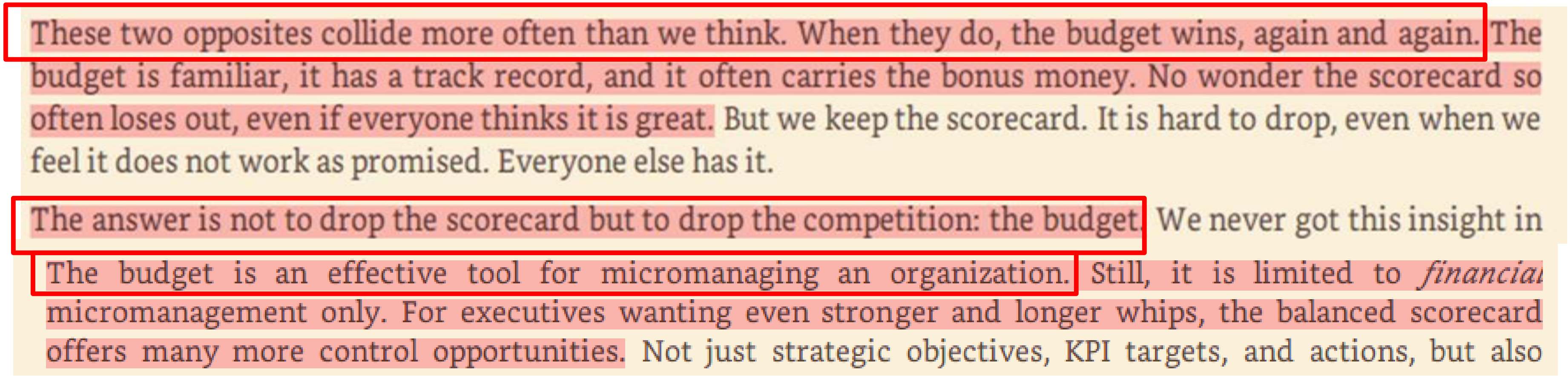
“If you write down your Top 10 goals 1,000 times a day, you can accomplish *anything*...if you have any time left over.”

Agile Budgeting

Balanced Scorecards vs. Budgets

- Balanced Scorecards – rooted in strategy and longer time perspectives. Focus on non-financial drivers that lead to financial performance and value creation

- Budgets – have weak strategy links and short time horizons and about money only



These two opposites collide more often than we think. When they do, the budget wins, again and again. The budget is familiar, it has a track record, and it often carries the bonus money. No wonder the scorecard so often loses out, even if everyone thinks it is great. But we keep the scorecard. It is hard to drop, even when we feel it does not work as promised. Everyone else has it.

The answer is not to drop the scorecard but to drop the competition: the budget. We never got this insight in

The budget is an effective tool for micromanaging an organization. Still, it is limited to *financial* micromanagement only. For executives wanting even stronger and longer whips, the balanced scorecard offers many more control opportunities. Not just strategic objectives, KPI targets, and actions, but also

Quotes from "Implementing Beyond Budgeting " by Bjarte Bogsnes

- Dangers with Scorecards:
  - Connecting it to bonus system: "the greener the KPIs, the higher the bonus"
  - Using it to protect and reinforce a command-and-control regime

Agile Budgeting

Scorecards: Cascading vs. Translation

- Autopilot: “If there is KPI, there always must be a target”. Not True. If common sense exists we know what good looks like why do we need to spell out decimals?
- We want best performance but, given the circumstances
- Reasons why mergers and acquisitions often fail is not financial surprises. Rather, it is lack of **Cultural Fit**

__NB__: Most Finance people believe that all target numbers must add up exactly to the corporate target, and that this can be achieved only through top-down __cascading__.

Quote from “Implementing Beyond Budgeting ” by Bjarte Bogsnes

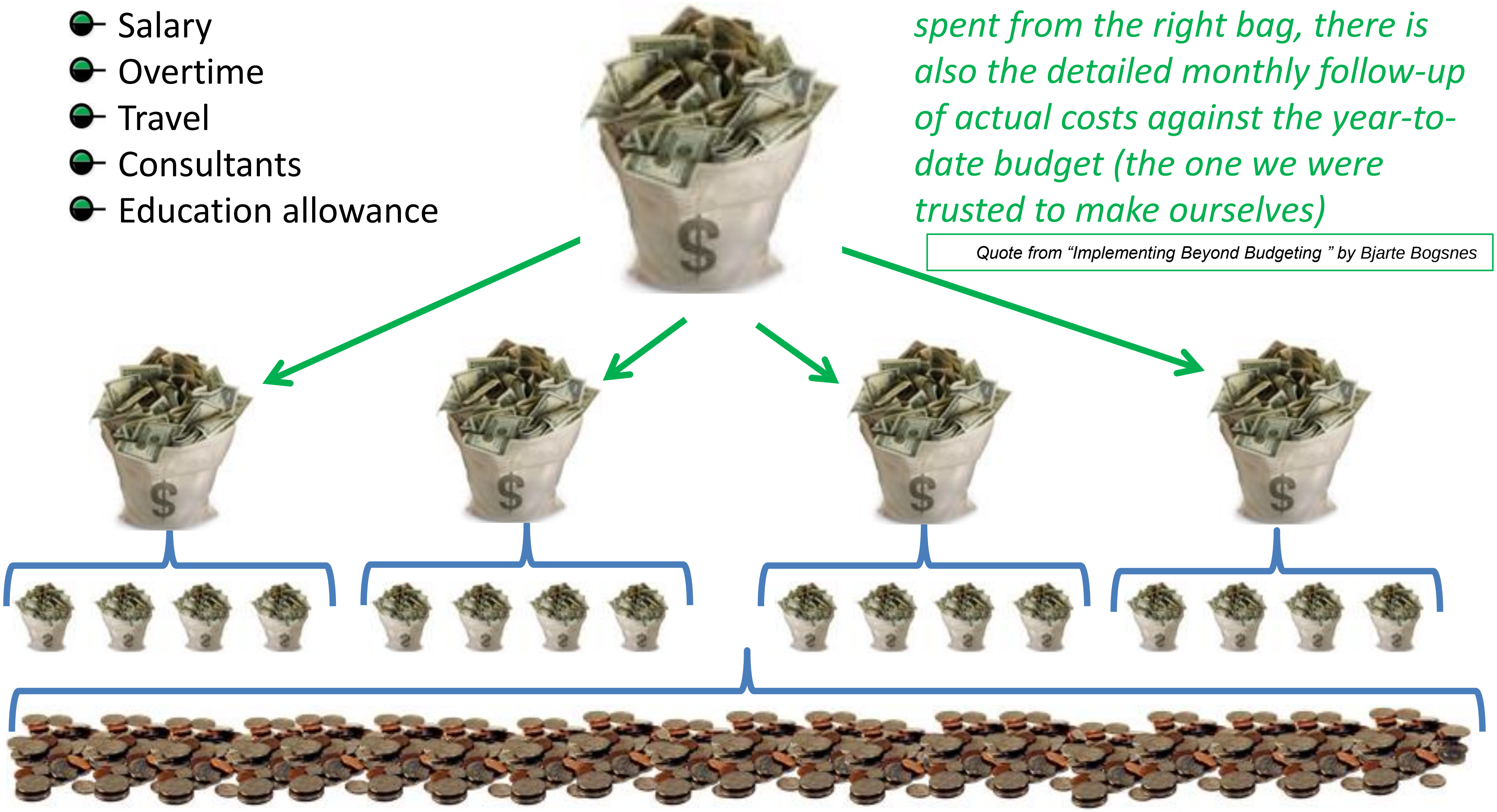
Agile Budgeting

Splitting a Bag of Cash

- Many small bags, labeled:
  - Salary
  - Overtime
  - Travel
  - Consultants
  - Education allowance

***NB**: To make sure that money is spent from the right bag, there is also the detailed monthly follow-up of actual costs against the year-to-date budget (the one we were trusted to make ourselves)*

Quote from "Implementing Beyond Budgeting " by Bjarte Bogsnes



Agile Budgeting

Monetary Incentives for Doing Good Deeds?

- Donating Blood vs. Selling Bodily Fluid

- Quote from John Cryan, co-CEO at Deutsche Bank: *“I have no idea why I was offered a contract with a bonus in it because I promise you I will not work any harder or any less hard in any year, in any day because someone is going to pay me more or less*.”

- There are some alternative views By Skinnerian Behaviorists (American psychologist B.F. Skinner strongly advocated extrinsic motivation) ….but…
  - Skinner's supporting studies and experiments were conducted on *mice, rats, and pigeons*
  - Studies were about *simple, mechanical, and repetitive tasks* where individual results are easily measured— not exactly what life in today's knowledge organizations is about.

Agile Budgeting

Does Meeting a Budget Drive Individual Performance?

- Main reason for appraisal documentation: “**legal trail for rainy day**”

- The best appraisal – is an open dialogue (open, honest, and constructive feedback)

- Performance !=Results
  - Result - is a measured outcome
  - Performance is the behavior and effort behind

- Forced Ranking = *“hopeless management practice” (Bjarte)*

- **Washington Post**: “***10 percent of the Fortune 500 companies have now abolished the traditional annual performance appraisal, including Microsoft, Accenture, Deloitte, and Expedia. Even GE is experimenting with alternatives***.”

Quote from “Implementing Beyond Budgeting ” by Bjarte Bogsnes

Agile Budgeting

What do Monetary Incentives to do People?

- Costs of appraisals
  - Tons of management time and resources wasted
  - Making everybody unhappy
- Alfie Kohn: "*Pay people fairly, and then do whatever possible to make them forget everything related to pay and money*."
- "Atlassian abandons their sales bonus system"
- "2013: the pharmaceutical company GSK (GlaxoSmithKline) announced a new compensation program abolishing individual targets within sales"
- Alternatives:
  - Individual bonuses vs. Collective bonuses ←
  - Collective profit-sharing system
  - Peer-to-peer bonuses or non-financial rewards (experiments by Google, HCL, Zappos)

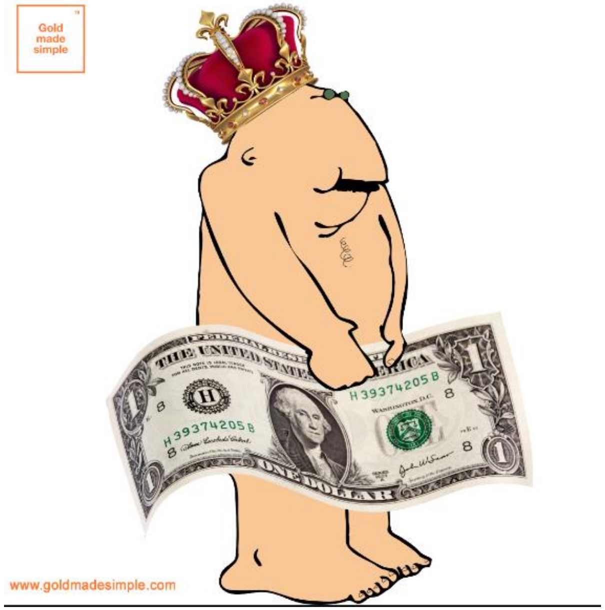
N.B.: *Miles [company name] does not operate with individual bonuses, but employees get a share in two different ways. There is a provision system where the employee gets a cut of the revenue he or she generates. They can choose the risk profile that best fits their private situation: high fixed or high variable. In addition, if the annual profit margin for their unit exceeds 10 percent, all employees with partners/ spouses are invited for a weekend trip abroad.*

Quote from "Implementing Beyond Budgeting " by Bjarte Bogsnes

Agile Budgeting

What do Monetary Incentives to do People?

- Who has the guts to say:
  - "The emperor has no clothes on"?
  - "Bonuses will be driven out of town, shamed and undressed."



N.B.: *Individual bonus can be a very effective motivational mechanism for simple work where there is little motivation in the job itself, where the link between individual efforts and outcomes is easy to measure, and where quantity is more important than quality. So for picking fruit, catching rats, and similar simple, repetitive work, individual bonus definitely works. But when moving to more complex tasks, where more cognitive skills and teamwork are required, research shows that individual bonus loses its power.*

Quote from "Implementing Beyond Budgeting " by Bjarte Bogsnes

Agile Budgeting

What do Skeptics say about ”Beyond Budgeting?

- European Audience: “How cost can be managed without a budget?”
- United States Audience: “What drives bonus if there is no budget?”

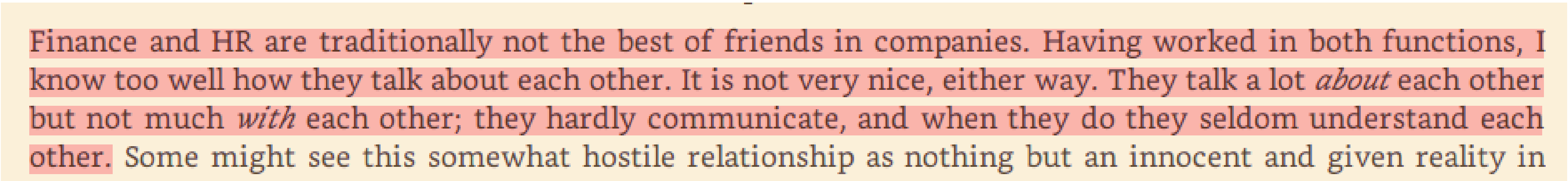
N.B.: *The smallest problem with bonuses is that they often are tied to delivery of budget numbers, which as we have discussed is a language quite ill-suited for performance evaluation. A much more serious problem is the* ***negative effect on motivation*** *and performance…... I have totally lost my belief in individual bonus systems. I am convinced they do much more harm than good.*

Quote from “Implementing Beyond Budgeting ” by Bjarte Bogsnes

Agile Budgeting

Partnership between HR and Finance

- Does **Finance** and **HR** interface and cooperate?
- Does HR present itself as a true “Business Partner”?
  - This will become true when HR starts caring about Performance Management process **BEFORE** it hits usual HR territory

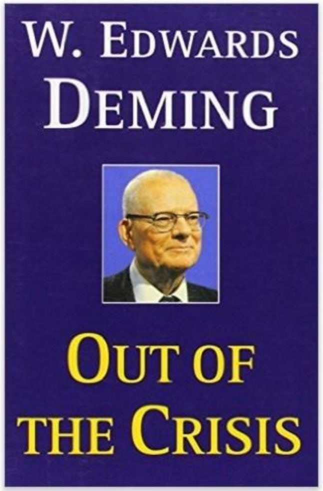


Finance and HR are traditionally not the best of friends in companies. Having worked in both functions, I know too well how they talk about each other. It is not very nice, either way. They talk a lot *about* each other but not much *with* each other; they hardly communicate, and when they do they seldom understand each other. Some might see this somewhat hostile relationship as nothing but an innocent and given reality in

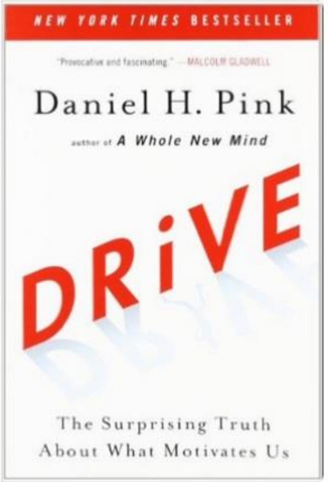
Quote from “Implementing Beyond Budgeting ” by Bjarte Bogsnes

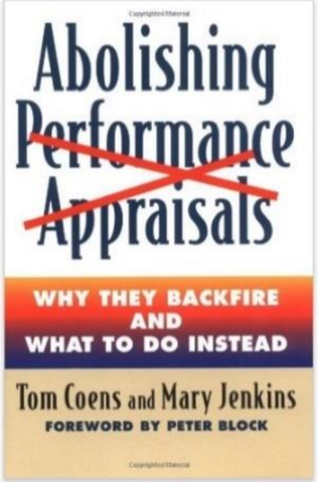
Agile Budgeting

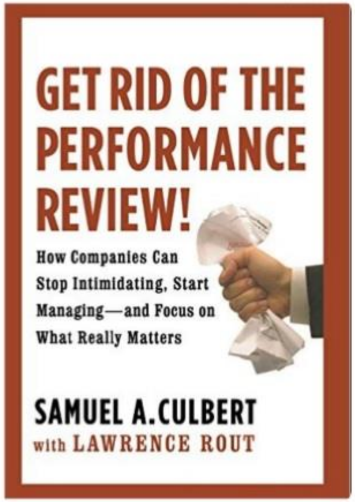
Frequently ignored scientific evidence

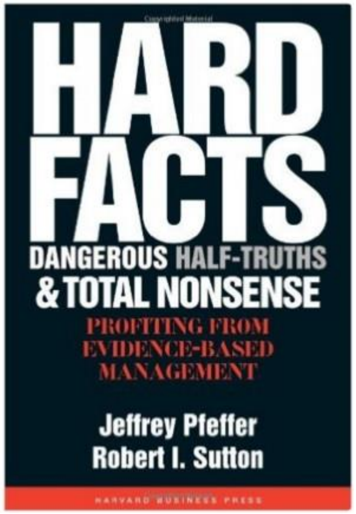


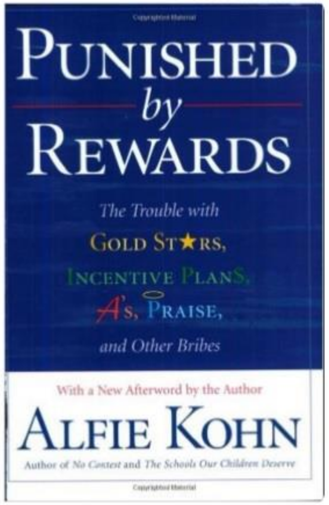
“The idea of a merit rating is alluring. The sound of the words captivates the imagination: pay for what you get; get what you pay for; motivate people to do their best, for their own good. The effect is exactly the opposite of what the words promise.”


NEW YORK TIMES BESTSELLER
"Provocative and fascinating." —MALCOLM GLADWELL
Daniel H. Pink
author of A Whole New Mind
DRIVE
The Surprising Truth About What Motivates Us


Abolishing
Performance
Appraisals
WHY THEY BACKFIRE AND WHAT TO DO INSTEAD
Tom Coens and Mary Jenkins
FOREWORD BY PETER BLOCK


GET RID OF THE PERFORMANCE REVIEW!
How Companies Can Stop Intimidating, Start Managing—and Focus on What Really Matters
SAMUEL A. CULBERT
with LAWRENCE ROUT


HARD FACTS
DANGEROUS HALF-TRUTHS & TOTAL NONSENSE
PROFITING FROM EVIDENCE-BASED MANAGEMENT
Jeffrey Pfeffer
Robert I. Sutton
HARVARD BUSINESS PRESS


PUNISHED by REWARDS
The Trouble with Gold Stars, Incentive Plans, A's, Praise, and Other Bribes
With a New Afterword by the Author
ALFIE KOHN
Author of No Contest and The Schools Our Children Deserve

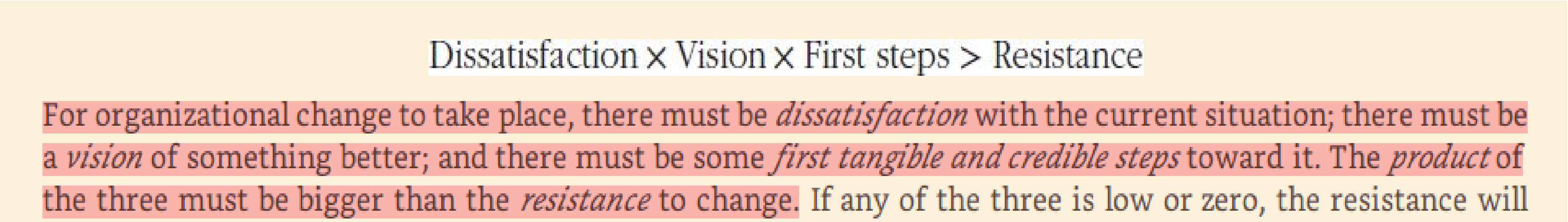
Agile Budgeting

How to Overcome Resistance?

- Today, Agile folks are still a minority (Freedom fighters), fighting a guerrilla war. Most likely, it will remain so until academia (colleges and universities) pick this up or there will be a cataclysmic event (e.g. Lehman Brothers)

- Two types of sceptics:
  - Skeptical because genuinely confused
  - Skeptical because they are Theory X lovers and Theory Y haters

<u>Advice</u>: Focus on the first type. The second type will slowly become extinct



Dissatisfaction × Vision × First steps > Resistance

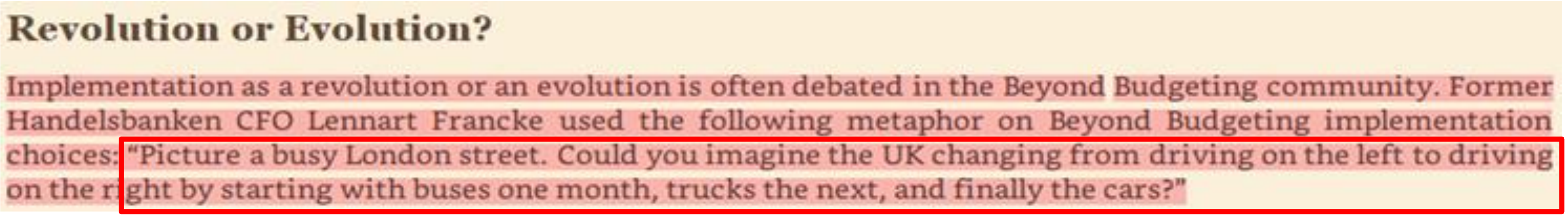
For organizational change to take place, there must be *dissatisfaction* with the current situation; there must be a *vision* of something better; and there must be some *first tangible and credible steps* toward it. The *product* of the three must be bigger than the *resistance* to change. If any of the three is low or zero, the resistance will

Quote from "Implementing Beyond Budgeting " by Bjarte Bogsnes

Agile Budgeting

Evolution vs. Revolution?

- Beyond Budgeting: Evolution vs Revolution?
    - Evolution through many step-wise mini-revolutions
    - Should not prevent us from thinking as revolutionaries
    - Be cautious of Big Bang counter-revolution by “dark forces”



Revolution or Evolution?

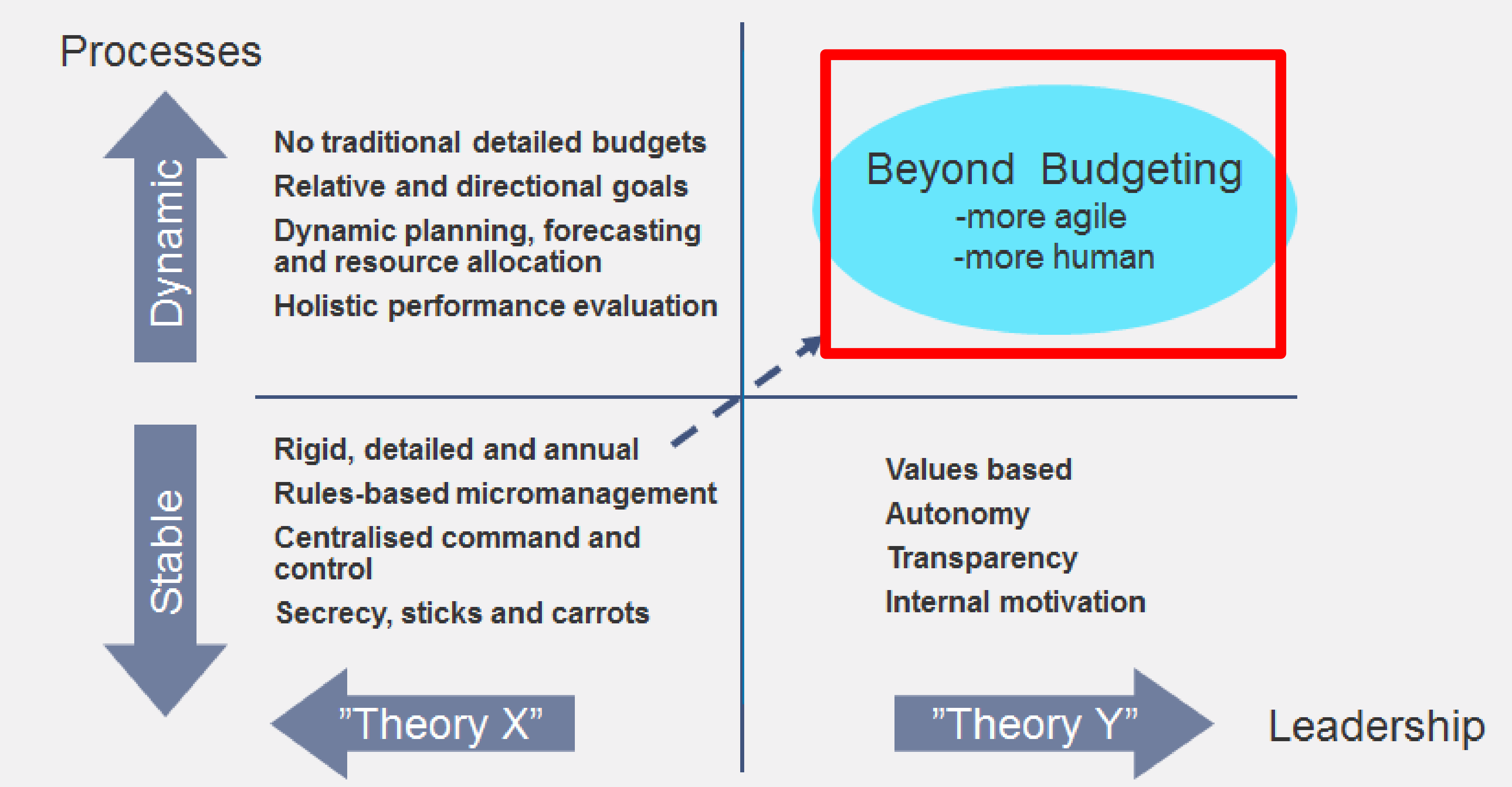
Implementation as a revolution or an evolution is often debated in the Beyond Budgeting community. Former Handelsbanken CFO Lennart Francke used the following metaphor on Beyond Budgeting implementation choices: “Picture a busy London street. Could you imagine the UK changing from driving on the left to driving on the right by starting with buses one month, trucks the next, and finally the cars?”

Quote from “Implementing Beyond Budgeting ” by Bjarte Bogsnes

Gene Gendel, Certified Enterprise Coach

Agile Budgeting

Where Should Smart Companies be Heading?

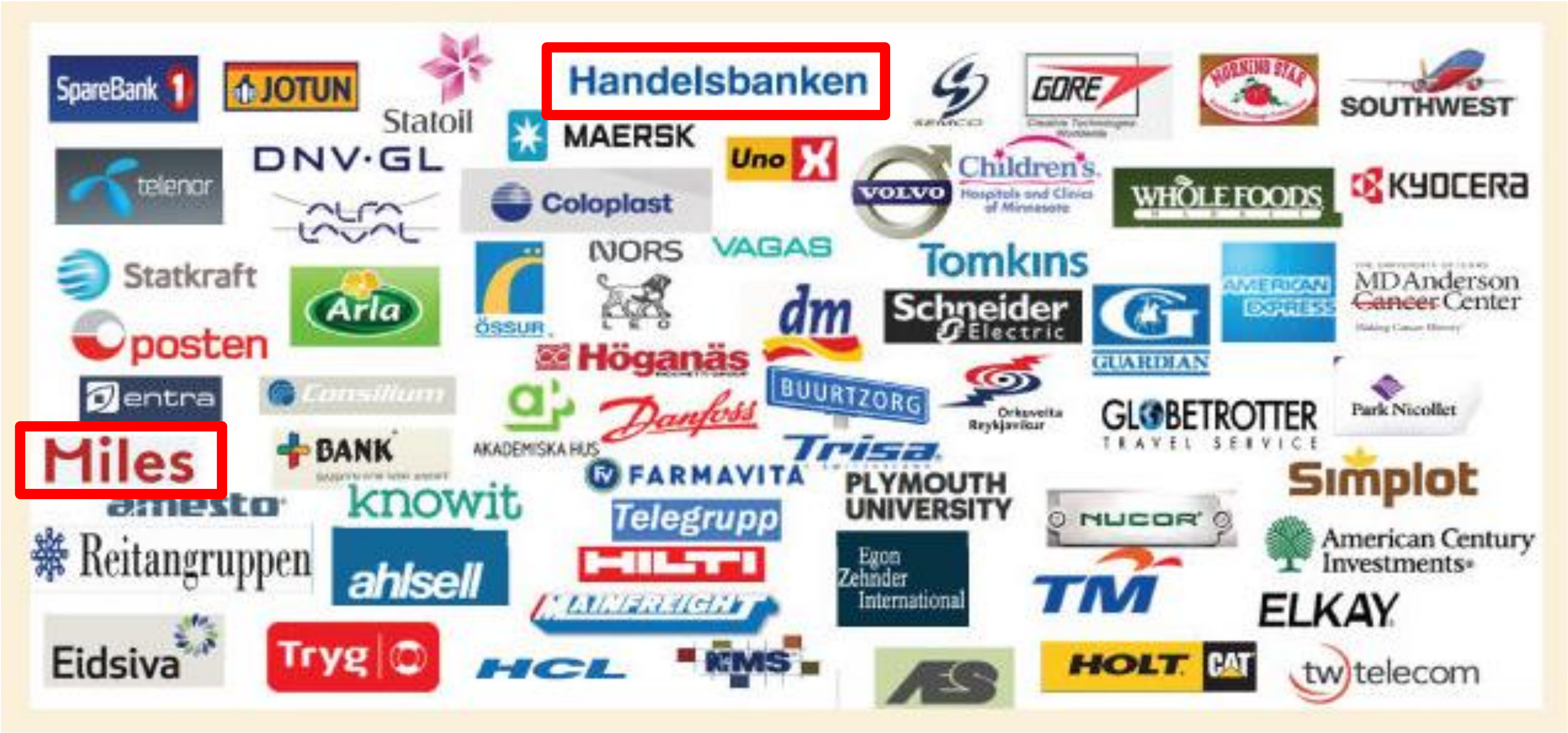

Processes
Dynamic
No traditional detailed budgets
Relative and directional goals
Dynamic planning, forecasting and resource allocation
Holistic performance evaluation
Beyond Budgeting
-more agile
-more human
Stable
Rigid, detailed and annual
Rules-based micromanagement
Centralised command and control
Secrecy, sticks and carrots
Values based
Autonomy
Transparency
Internal motivation
"Theory X"
"Theory Y"
Leadership

From: Beyond Budgeting - an agile management model for new business and people realities. Ambition to Action - the Statoil journey. Presented by Bjarte Bogsnes

Agile Budgeting

Who Went Down This Crazy ☺ Path?

Scandinavian *European*



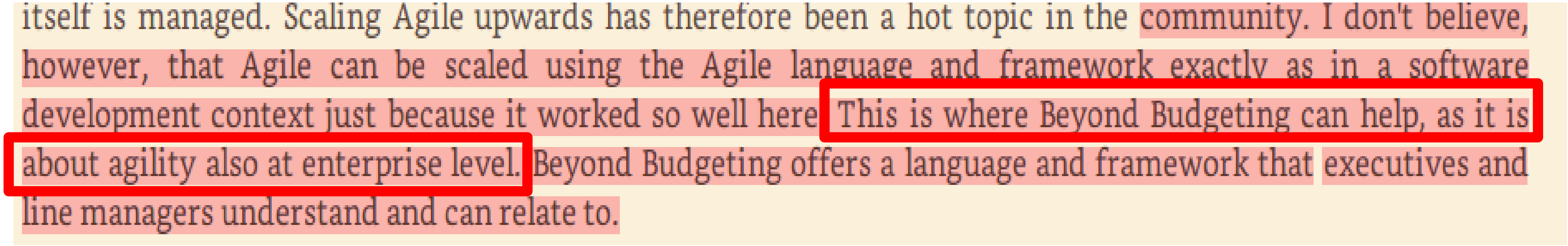
From: Beyond Budgeting - an agile management model for new business and people realities. Ambition to Action - the Statoil journey. Presented by Bjarte Bogsnes

Agile Budgeting

What is Required for Agile Scaling?

- Agile Scaling is not just “IT thing”

- Agile Scaling requires engaging an entire organization or its cross-cutting slice

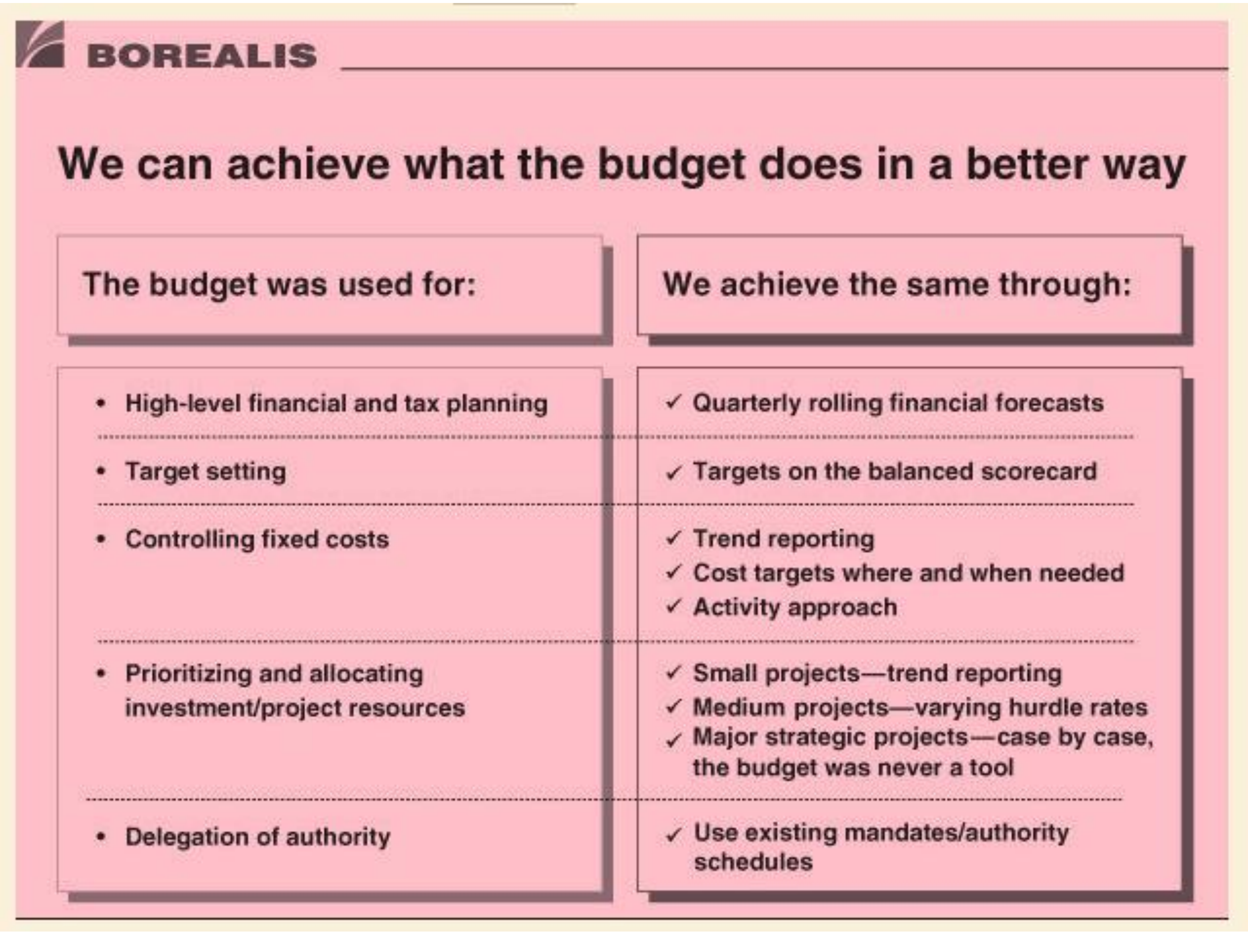


itself is managed. Scaling Agile upwards has therefore been a hot topic in the community. I don't believe, however, that Agile can be scaled using the Agile language and framework exactly as in a software development context just because it worked so well here. This is where Beyond Budgeting can help, as it is about agility also at enterprise level. Beyond Budgeting offers a language and framework that executives and line managers understand and can relate to.

Quote from “Implementing Beyond Budgeting ” by Bjarte Bogsnes

Agile Budgeting

Secrets of Borealis Successful Journey



BOREALIS

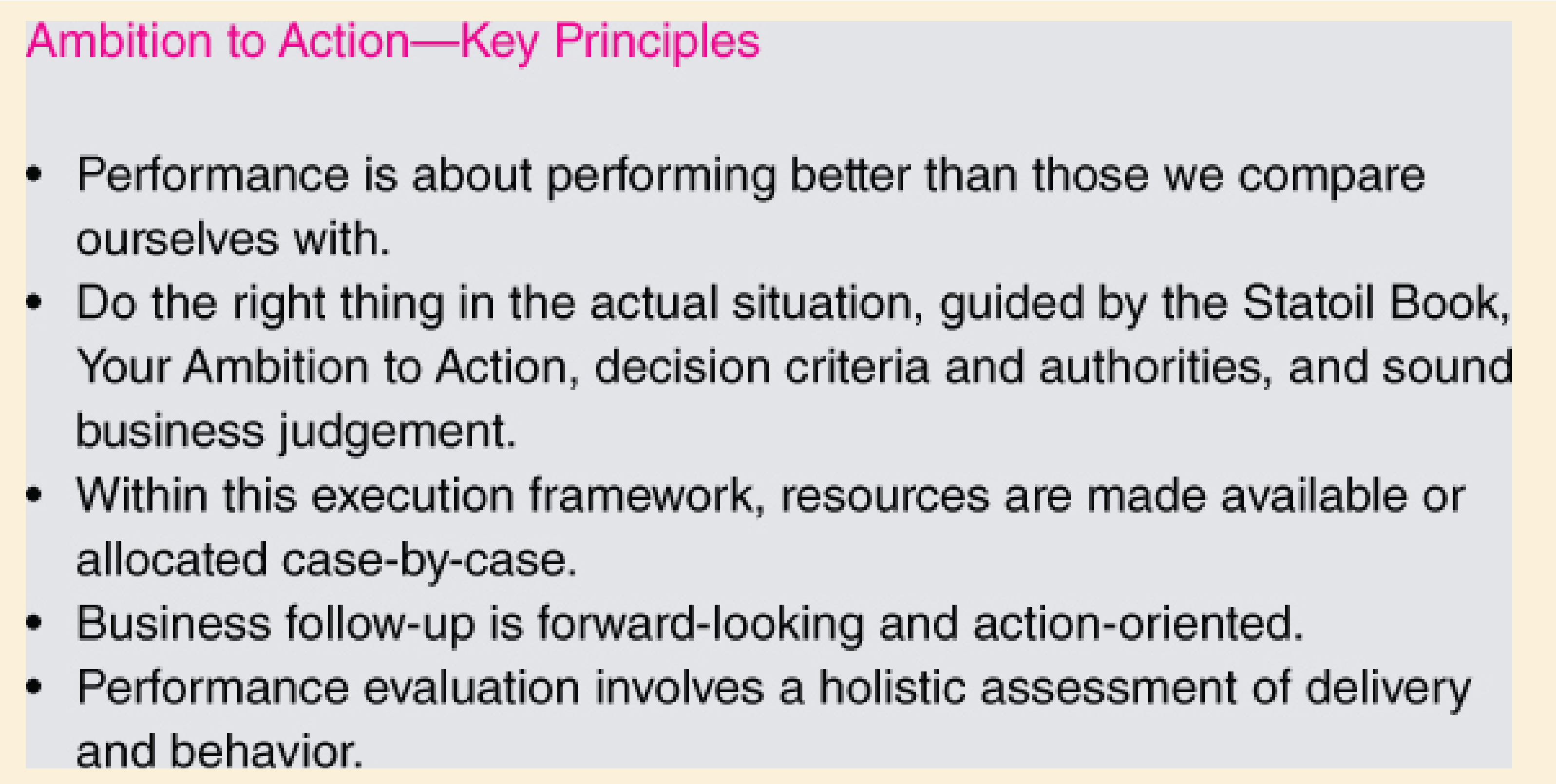
We can achieve what the budget does in a better way

The budget was used for:	We achieve the same through:
• High-level financial and tax planning	✓ Quarterly rolling financial forecasts
• Target setting	✓ Targets on the balanced scorecard
• Controlling fixed costs	✓ Trend reporting ✓ Cost targets where and when needed ✓ Activity approach
• Prioritizing and allocating investment/project resources	✓ Small projects—trend reporting ✓ Medium projects—varying hurdle rates ✓ Major strategic projects—case by case, the budget was never a tool
• Delegation of authority	✓ Use existing mandates/authority schedules

Quote from "Implementing Beyond Budgeting " by Bjarte Bogsnes

Agile Budgeting

Secrets of Borealis Successful Journey



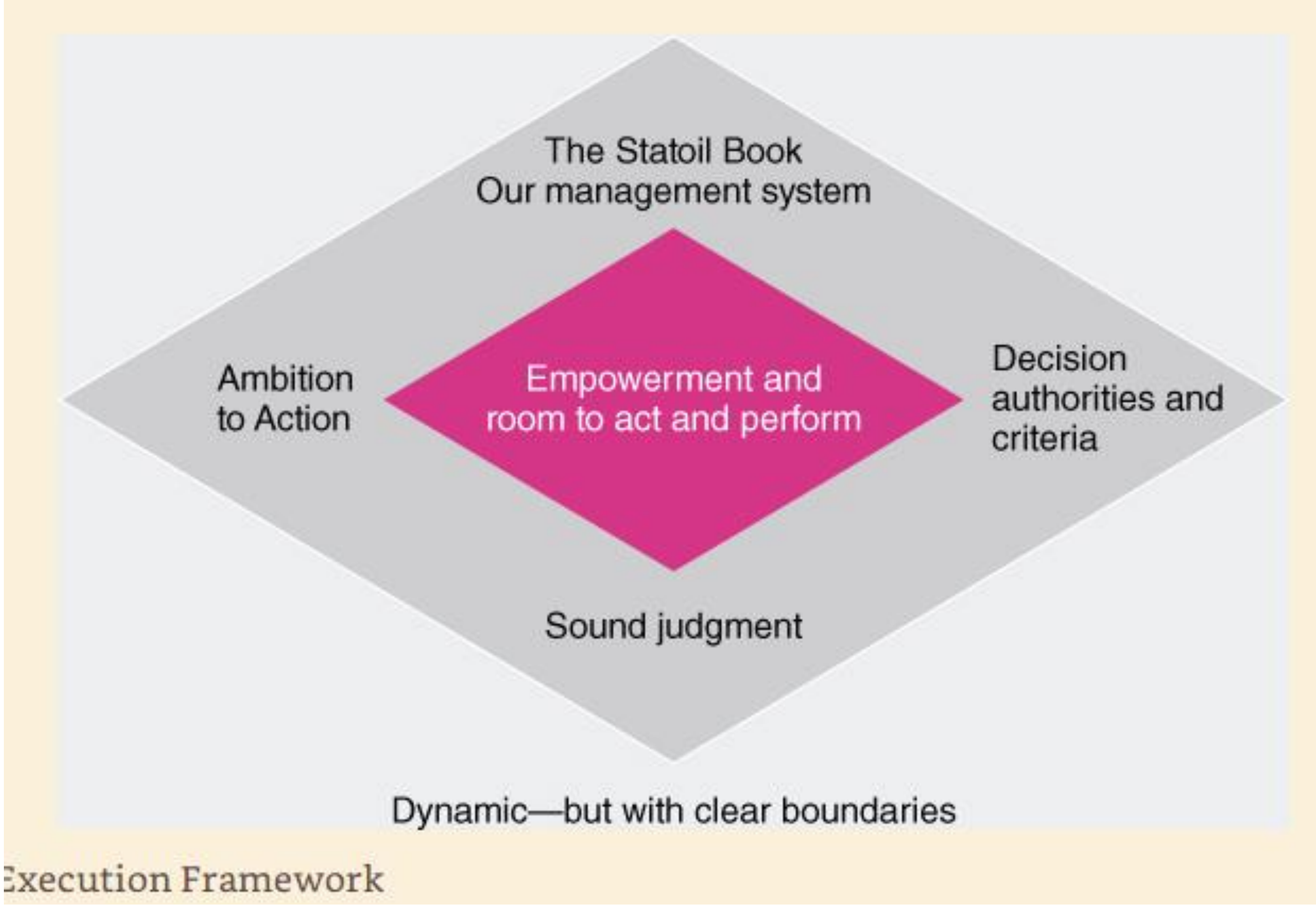
Ambition to Action—Key Principles

- Performance is about performing better than those we compare ourselves with.
- Do the right thing in the actual situation, guided by the Statoil Book, Your Ambition to Action, decision criteria and authorities, and sound business judgement.
- Within this execution framework, resources are made available or allocated case-by-case.
- Business follow-up is forward-looking and action-oriented.
- Performance evaluation involves a holistic assessment of delivery and behavior.

Quote from "Implementing Beyond Budgeting " by Bjarte Bogsnes

Agile Budgeting

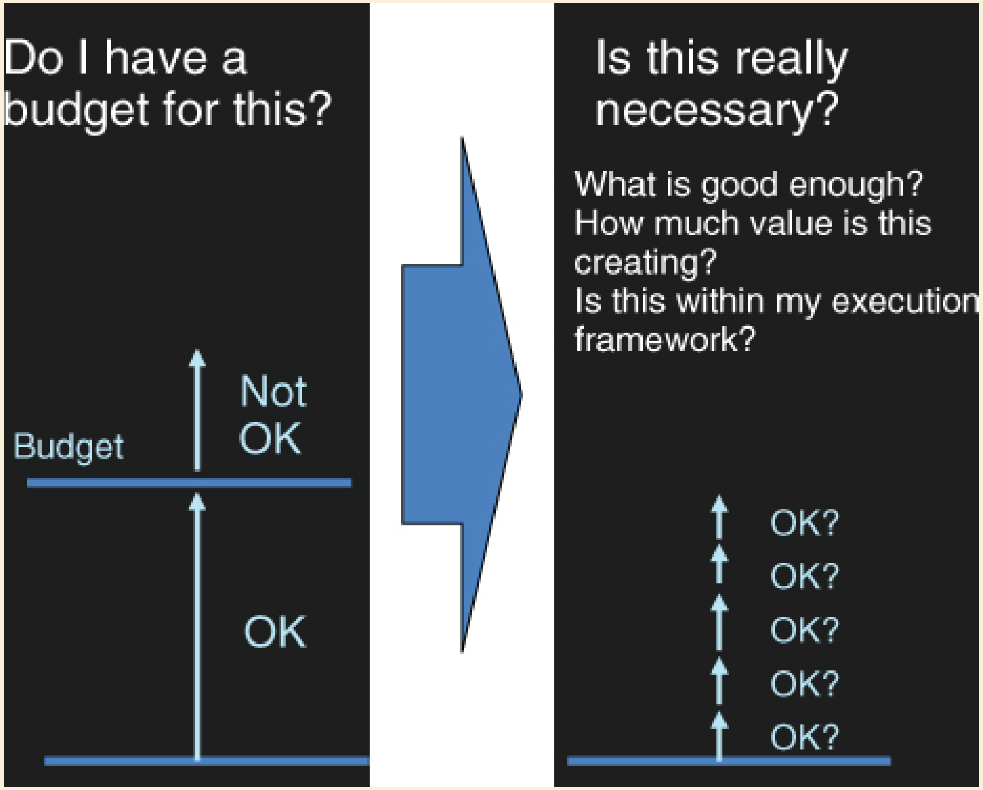
Secrets of Borealis Successful Journey



Execution Framework

Agile Budgeting

Secrets of Borealis Successful Journey



Do I have a budget for this?

Is this really necessary?

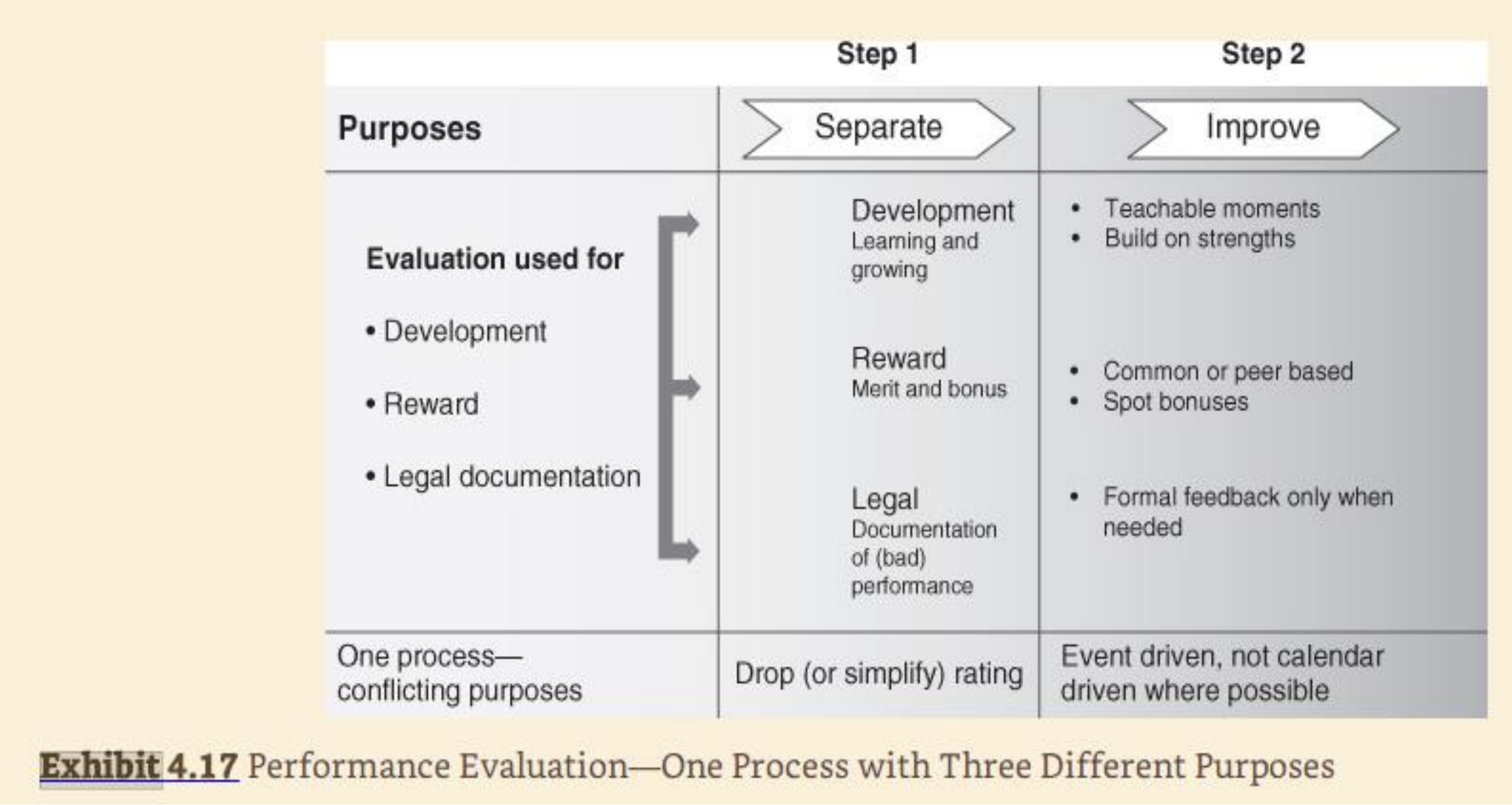
What is good enough?
How much value is this creating?
Is this within my execution framework?

Exhibit 4.12 The Mindset Required—Cost-Conscious from the First Cent

Quote from "Implementing Beyond Budgeting " by Bjarte Bogsnes

Agile Budgeting

Secrets of Borealis Successful Journey



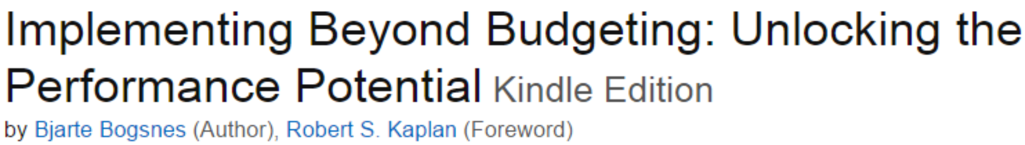
	Step 1	Step 2
Purposes	Separate	Improve
Evaluation used for • Development	Development Learning and growing	• Teachable moments • Build on strengths
• Reward	Reward Merit and bonus	• Common or peer based • Spot bonuses
• Legal documentation	Legal Documentation of (bad) performance	• Formal feedback only when needed
One process—conflicting purposes	Drop (or simplify) rating	Event driven, not calendar driven where possible

Exhibit 4.17 Performance Evaluation—One Process with Three Different Purposes

Quote from “Implementing Beyond Budgeting ” by Bjarte Bogsnes

Agile Budgeting

What is Next?

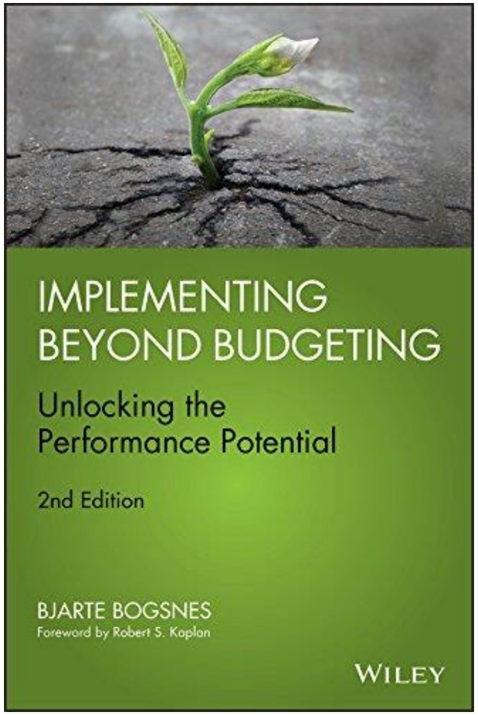


Implementing Beyond Budgeting: Unlocking the Performance Potential Kindle Edition

by Bjarte Bogsnes (Author), Robert S. Kaplan (Foreword)

- As an organizational and agile coach, I see immense value in Bjartes' work, for the following reasons:
- His book, boldly and unambiguously debunks archaic and outdated beliefs that are still shared by many people, whose understanding of management is based on the turn-of-last-century Taylorian Principles, worship of McGregor's Theory X management and Skinnerian Behaviorism.
- The book very skillfully leverages today's abundant research and conclusive scientific evidence on Intrinsic Motivation what that has been taught and re-taught, on numerous occasions, by so many other experts: economists, social scientists and behavioral sociologists. As such, the book definitely belongs on the same book-shelf with publications by Samuel Culbert, Tom Coens, Alfie Kohn, Jeffrey Pfeffer, and Ewards Demming himself. At least, in my library.
- The book paints a vivid picture of fallacies and misconceptions of conventional management in the area of budgeting and finance. Bjarte provides strong evidence why lumping Targets, Forecasts and Resource Allocation in the same KPI number produces false, misleading and harmful results.
- The Key benefit (HUGE, imo) for agile coaches and trainers: in our work we often refer to problems that are caused by the "Iron Triangle" of conventional Project Management. There, our main argument is that in agile/iterative development, where Timeline and Budget are fixed (two corners of the triangle) Scope corner must always remain fluid to "relief" system constraints. In agile product development, Scope corner is usually "clipped". Well, Bjarte's teaching generously offers another great way to alleviate system constraints – it is by "clipping" Budget corner. Effectively, Beyond Budgeting puts another dent on "rusty iron" and makes it more agile. As such, Beyond Budgeting teaching is a very powerful tool in the toolbox of agile coaches and organizational change agents.
- Everything that Bjarte offers is not theoretical dogma only. Throughout the book, he provides strong evidence that is based on his personal experience and research data provided by other companies, most of which are having worldwide-known names. This, in turn, is also very helpful for agile coaches and change agents: we are frequently being asked if we can "provide examples" of our teachings being implemented elsewhere. Well, now we have these examples in abundance, thanks to Bjarte's publication.
- With enthusiasm and excitement, I am adding the learning received from Implementing "Beyond Budgeting" to my training materials.

Gene Gendel (Amazon Recommendation)


IMPLEMENTING BEYOND BUDGETING
Unlocking the Performance Potential
2nd Edition
BJARTE BOGSNES
Foreword by Robert S. Kaplan
WILEY

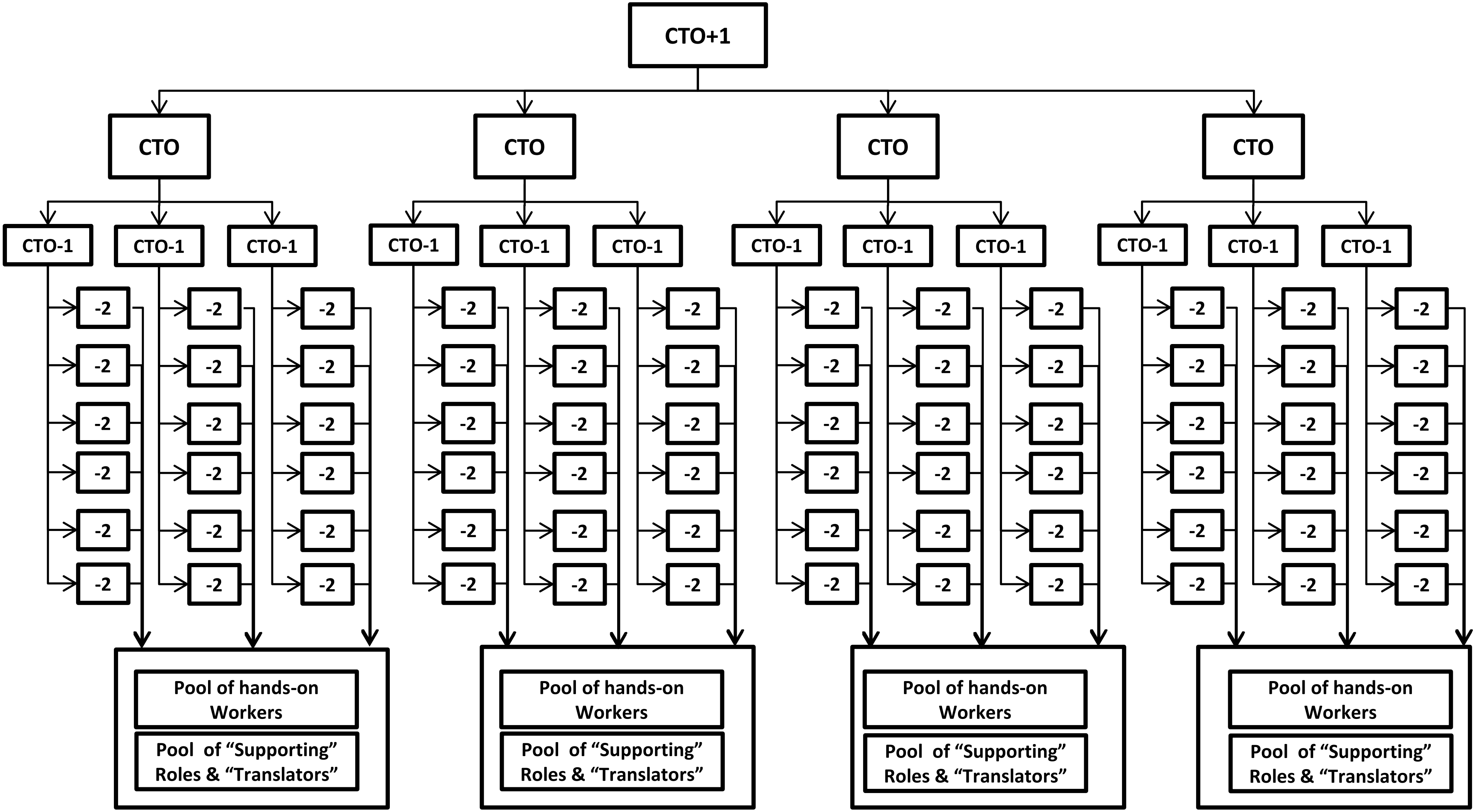
De-Scaling & Cross-Cutting Organization for Agile Transformation?

Common Mistakes with Agile Transformation (& Education)

- Senior Leadership can be exempt (they are too busy, this is too "beneath" them)
- HR can be exempt – they are "above" all of this ("the Untouchables")
- Only hands-on developers must be trained. Agile is IT thing.
- Business wants to "agilize" on their own, away from Technology.
- Products/Services are not properly analyzed and aligned, to maximize "front-to-back" changes. *Not enough* of an organization is involved in education/transformation. Results are too "narrow/local" and not sustainable.
- Opposite to the above, *too much* of an organization tries to change at once ("bite a piece that is too big to swallow"). Too much resistance. Not enough thrust to overcome inertia.
- Education is delivered to intended recipients but not by qualified parties:
  - Through unfit translation layers (PMO, first-line management or other "power" structures)"

    OR
  - By self-proclaimed experts (internal & external), only with theoretical knowledge that is based on various home-bred BoKs

Typical Organizational Structure (Technology)



CTO+1

CTO

CTO-1 CTO-1 CTO-1

-2 -2 -2 -2 -2 -2

-2 -2 -2 -2 -2 -2

-2 -2 -2 -2 -2 -2

Pool of hands-on Workers

Pool of “Supporting” Roles & “Translators”

CTO

CTO-1 CTO-1 CTO-1

-2 -2 -2 -2 -2 -2

-2 -2 -2 -2 -2 -2

-2 -2 -2 -2 -2 -2

Pool of hands-on Workers

Pool of “Supporting” Roles & “Translators”

CTO

CTO-1 CTO-1 CTO-1

-2 -2 -2 -2 -2 -2

-2 -2 -2 -2 -2 -2

-2 -2 -2 -2 -2 -2

Pool of hands-on Workers

Pool of “Supporting” Roles & “Translators”

CTO

CTO-1 CTO-1 CTO-1

-2 -2 -2 -2 -2 -2

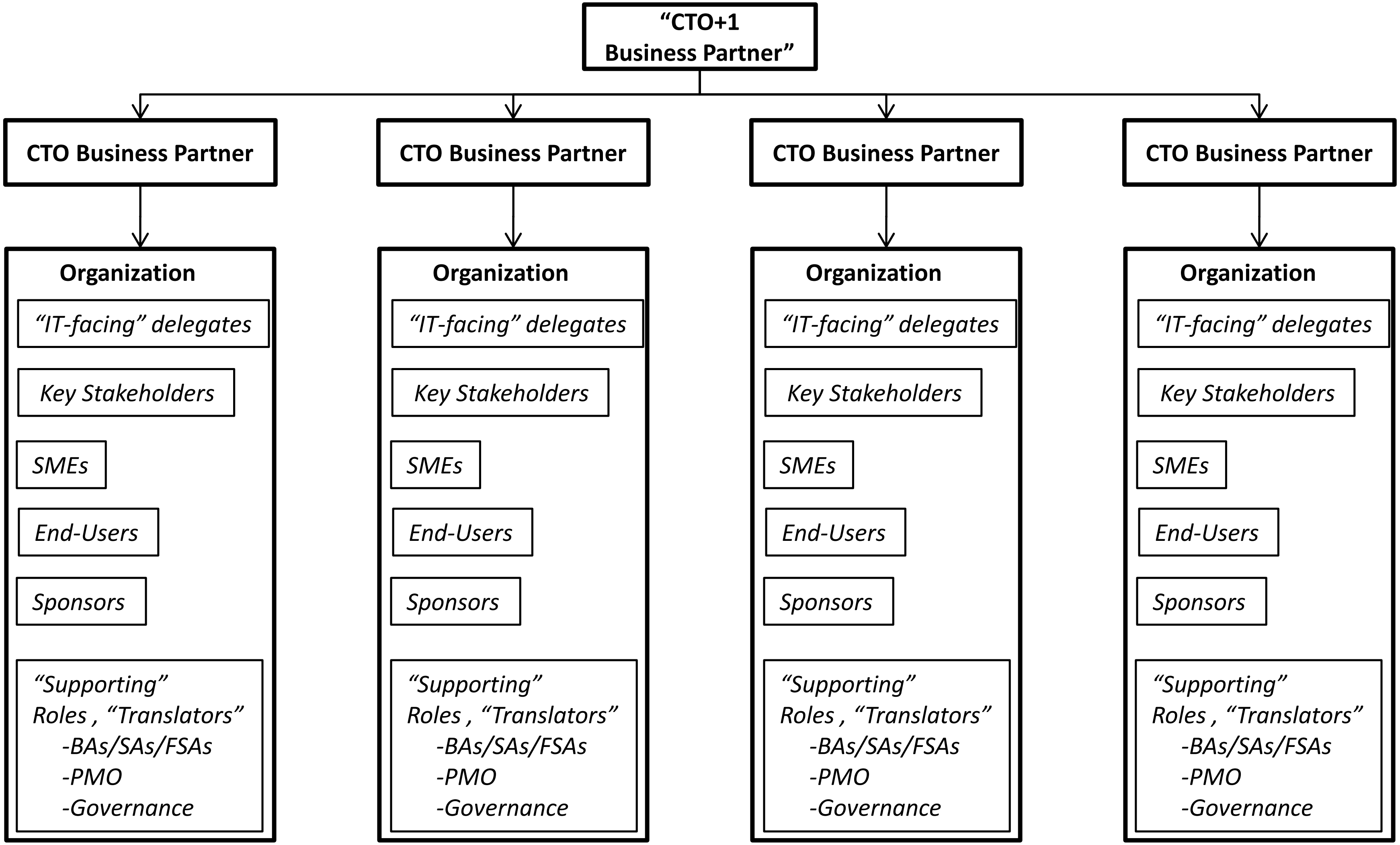
-2 -2 -2 -2 -2 -2

-2 -2 -2 -2 -2 -2

Pool of hands-on Workers

Pool of “Supporting” Roles & “Translators”

Typical Organizational Structure (Business)



"CTO+1
Business Partner"

CTO Business Partner

Organization

"IT-facing" delegates

Key Stakeholders

SMEs

End-Users

Sponsors

"Supporting" Roles , "Translators"
-BAs/SAs/FSAs
-PMO
-Governance

CTO Business Partner

Organization

"IT-facing" delegates

Key Stakeholders

SMEs

End-Users

Sponsors

"Supporting" Roles , "Translators"
-BAs/SAs/FSAs
-PMO
-Governance

CTO Business Partner

Organization

"IT-facing" delegates

Key Stakeholders

SMEs

End-Users

Sponsors

"Supporting" Roles , "Translators"
-BAs/SAs/FSAs
-PMO
-Governance

CTO Business Partner

Organization

"IT-facing" delegates

Key Stakeholders

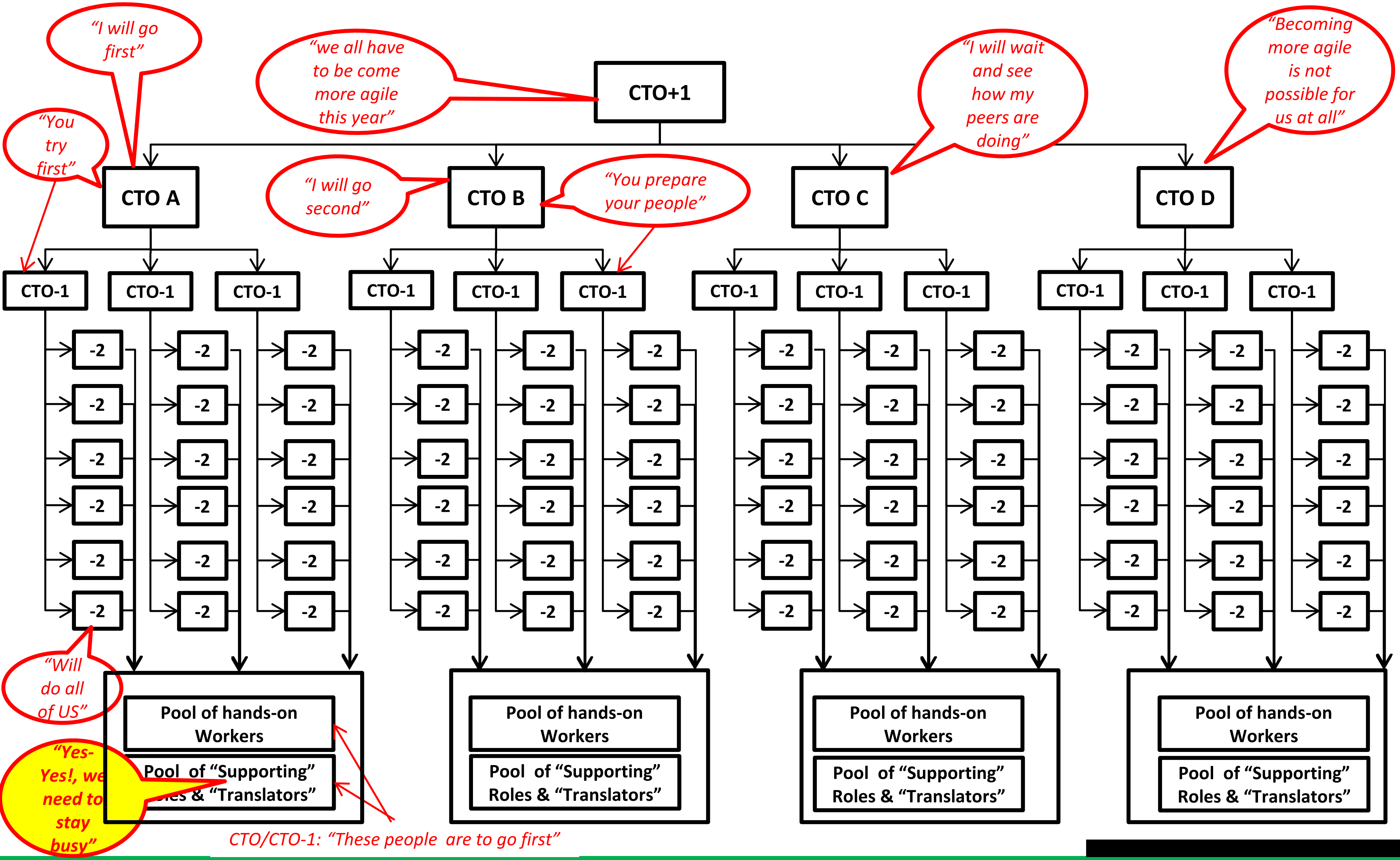
SMEs

End-Users

Sponsors

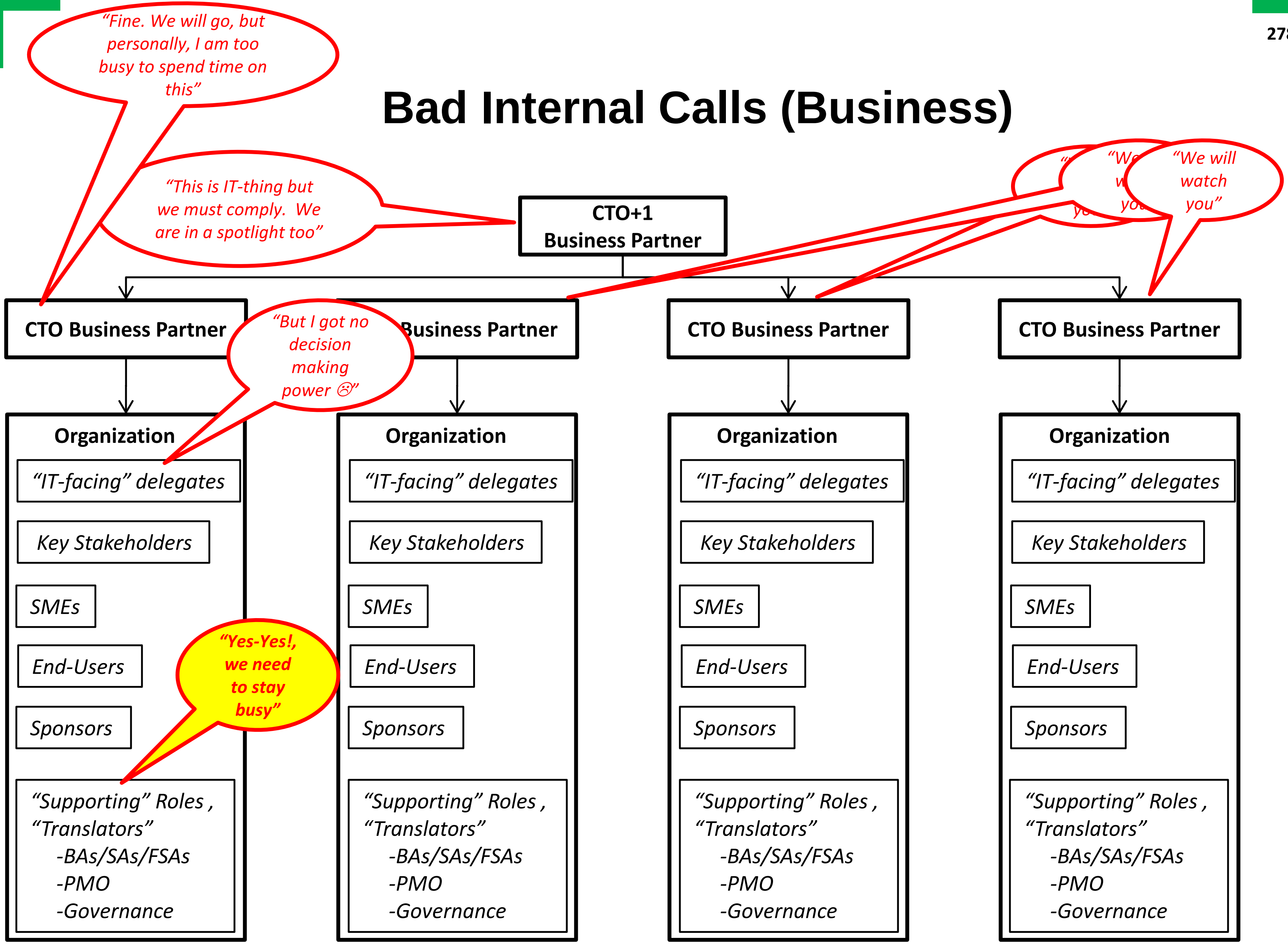
"Supporting" Roles , "Translators"
-BAs/SAs/FSAs
-PMO
-Governance

Bad Internal Calls (Technology)


"I will go first"
"You try first"
"we all have to be come more agile this year"
CTO+1
"I will wait and see how my peers are doing"
"Becoming more agile is not possible for us at all"
CTO A
"I will go second"
CTO B
"You prepare your people"
CTO C
CTO D
CTO-1
CTO-1
CTO-1
CTO-1
CTO-1
CTO-1
CTO-1
CTO-1
CTO-1
CTO-1
CTO-1
CTO-1
-2
"Will do all of US"
Pool of hands-on Workers
Pool of "Supporting" Roles & "Translators"
"Yes- Yes!, we need to stay busy"
Pool of hands-on Workers
Pool of "Supporting" Roles & "Translators"
Pool of hands-on Workers
Pool of "Supporting" Roles & "Translators"
Pool of hands-on Workers
Pool of "Supporting" Roles & "Translators"

CTO/CTO-1: "These people are to go first"

Bad Internal Calls (Business)



CTO+1 Business Partner

CTO Business Partner

Business Partner

CTO Business Partner

CTO Business Partner

Organization

"IT-facing" delegates

Key Stakeholders

SMEs

End-Users

Sponsors

"Supporting" Roles , "Translators"
-BAs/SAs/FSAs
-PMO
-Governance

"Fine. We will go, but personally, I am too busy to spend time on this"

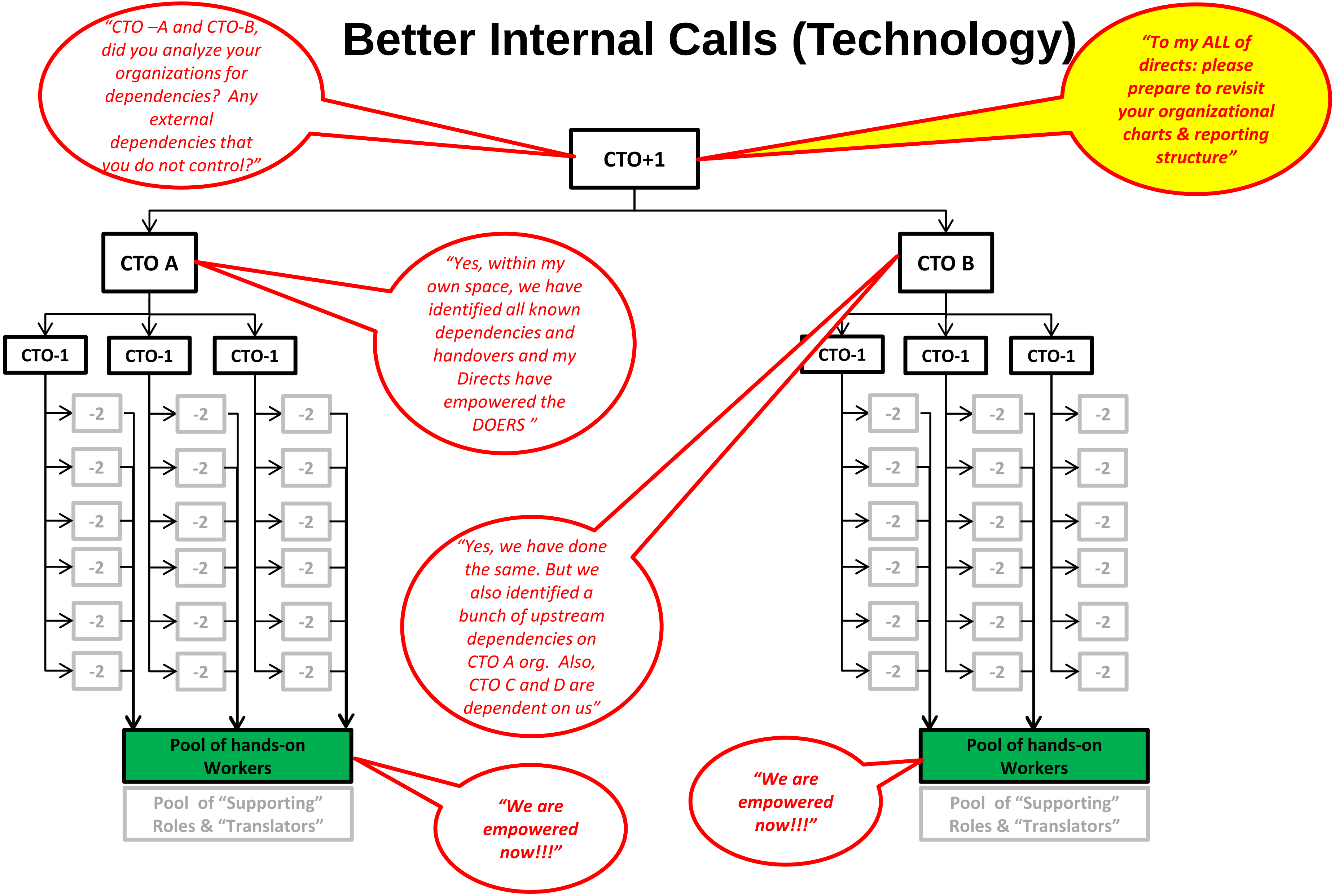
"This is IT-thing but we must comply. We are in a spotlight too"

"But I got no decision making power ☹"

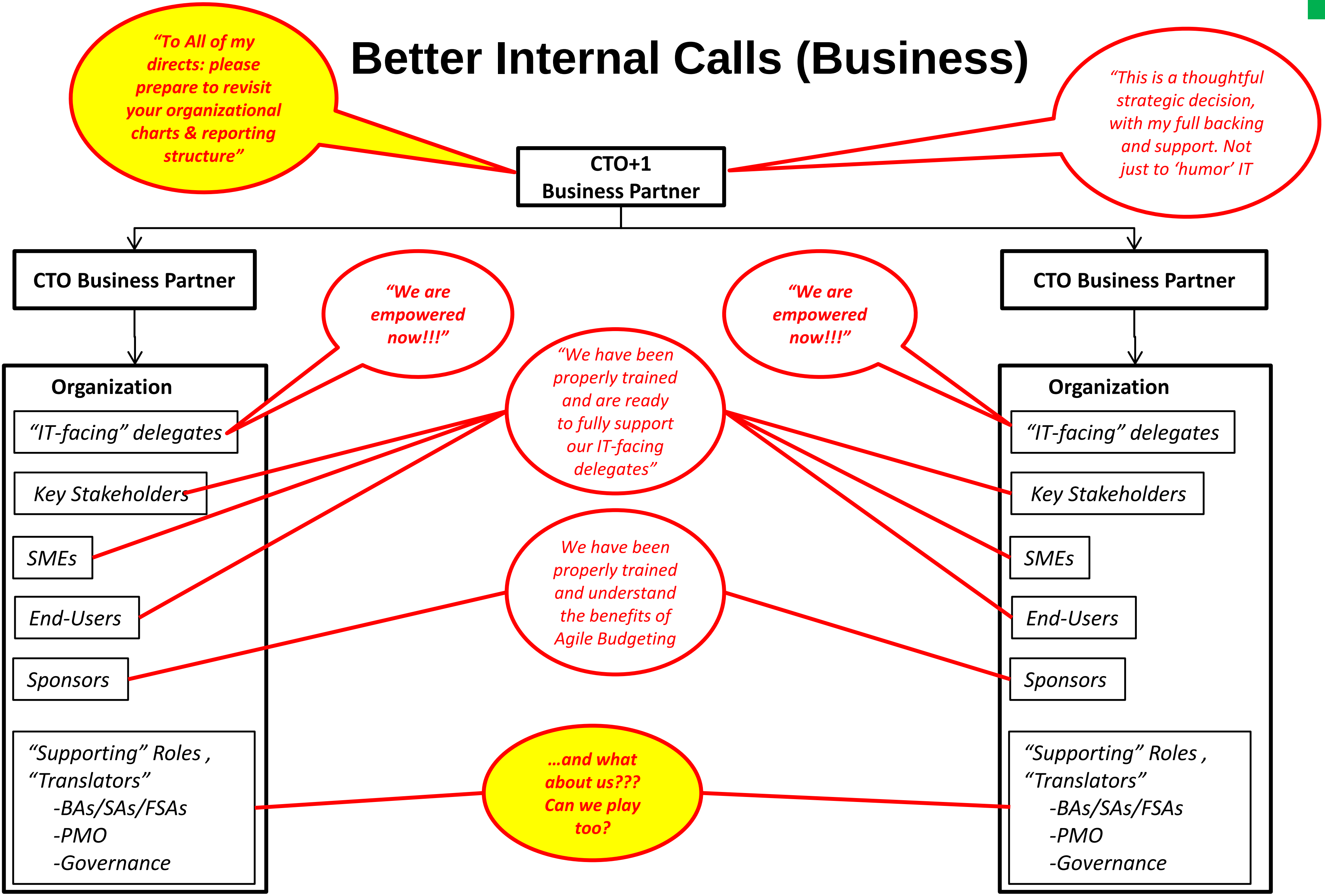
"Yes-Yes!, we need to stay busy"

"We will watch you"

Better Internal Calls (Technology)


"CTO –A and CTO-B, did you analyze your organizations for dependencies? Any external dependencies that you do not control?"
"To my ALL of directs: please prepare to revisit your organizational charts & reporting structure"
CTO+1
CTO A
CTO B
"Yes, within my own space, we have identified all known dependencies and handovers and my Directs have empowered the DOERS "
CTO-1
CTO-1
CTO-1
CTO-1
CTO-1
CTO-1
-2
"Yes, we have done the same. But we also identified a bunch of upstream dependencies on CTO A org. Also, CTO C and D are dependent on us"
Pool of hands-on Workers
Pool of "Supporting" Roles & "Translators"
"We are empowered now!!!"
"We are empowered now!!!"
Pool of hands-on Workers
Pool of "Supporting" Roles & "Translators"

Better Internal Calls (Business)



"To All of my directs: please prepare to revisit your organizational charts & reporting structure"

CTO+1
Business Partner

"This is a thoughtful strategic decision, with my full backing and support. Not just to 'humor' IT

CTO Business Partner

Organization

"IT-facing" delegates

Key Stakeholders

SMEs

End-Users

Sponsors

"Supporting" Roles ,
"Translators"
-BAs/SAs/FSAs
-PMO
-Governance

"We are empowered now!!!"

"We have been properly trained and are ready to fully support our IT-facing delegates"

We have been properly trained and understand the benefits of Agile Budgeting

...and what about us??? Can we play too?

"We are empowered now!!!"

CTO Business Partner

Organization

"IT-facing" delegates

Key Stakeholders

SMEs

End-Users

Sponsors

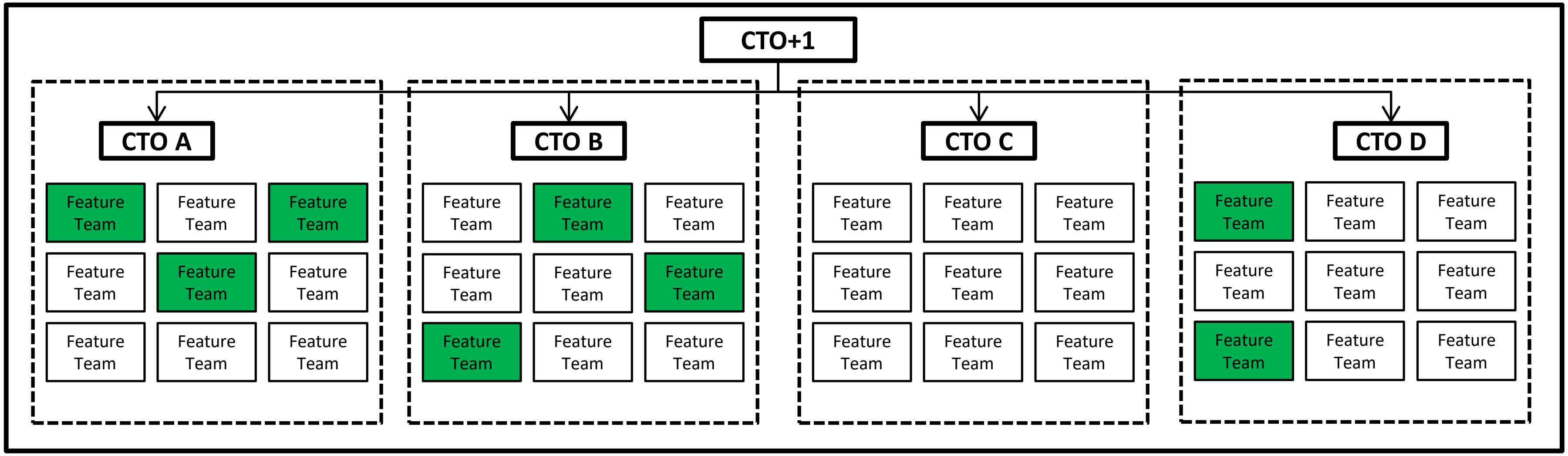
"Supporting" Roles ,
"Translators"
-BAs/SAs/FSAs
-PMO
-Governance

Avoid Local Optimization!!!

<JOB SECURITY> != <ROLE SECUTIRY>

/note: one of key teachings of Large Scale Scrum/

Better Internal Design (Technology)



CTO+1

CTO A

Feature Team Feature Team Feature Team
Feature Team Feature Team Feature Team
Feature Team Feature Team Feature Team

CTO B

Feature Team Feature Team Feature Team
Feature Team Feature Team Feature Team
Feature Team Feature Team Feature Team

CTO C

Feature Team Feature Team Feature Team
Feature Team Feature Team Feature Team
Feature Team Feature Team Feature Team

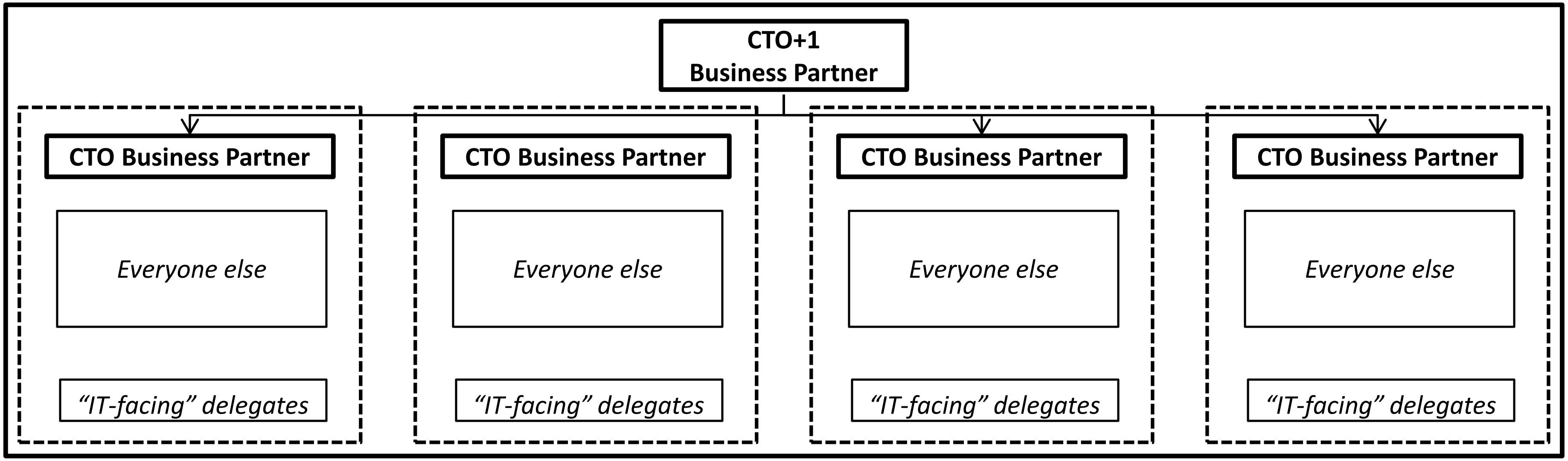
CTO D

Feature Team Feature Team Feature Team
Feature Team Feature Team Feature Team
Feature Team Feature Team Feature Team

What must be considered?

- Dependencies (internal) *within* CTOs' towers?
- Dependencies *across* CTOs' towers?
- Any shared services? Applications? Technologies? Platforms?
- Any shared technical resources?
- Any shared Business Partners' resources?

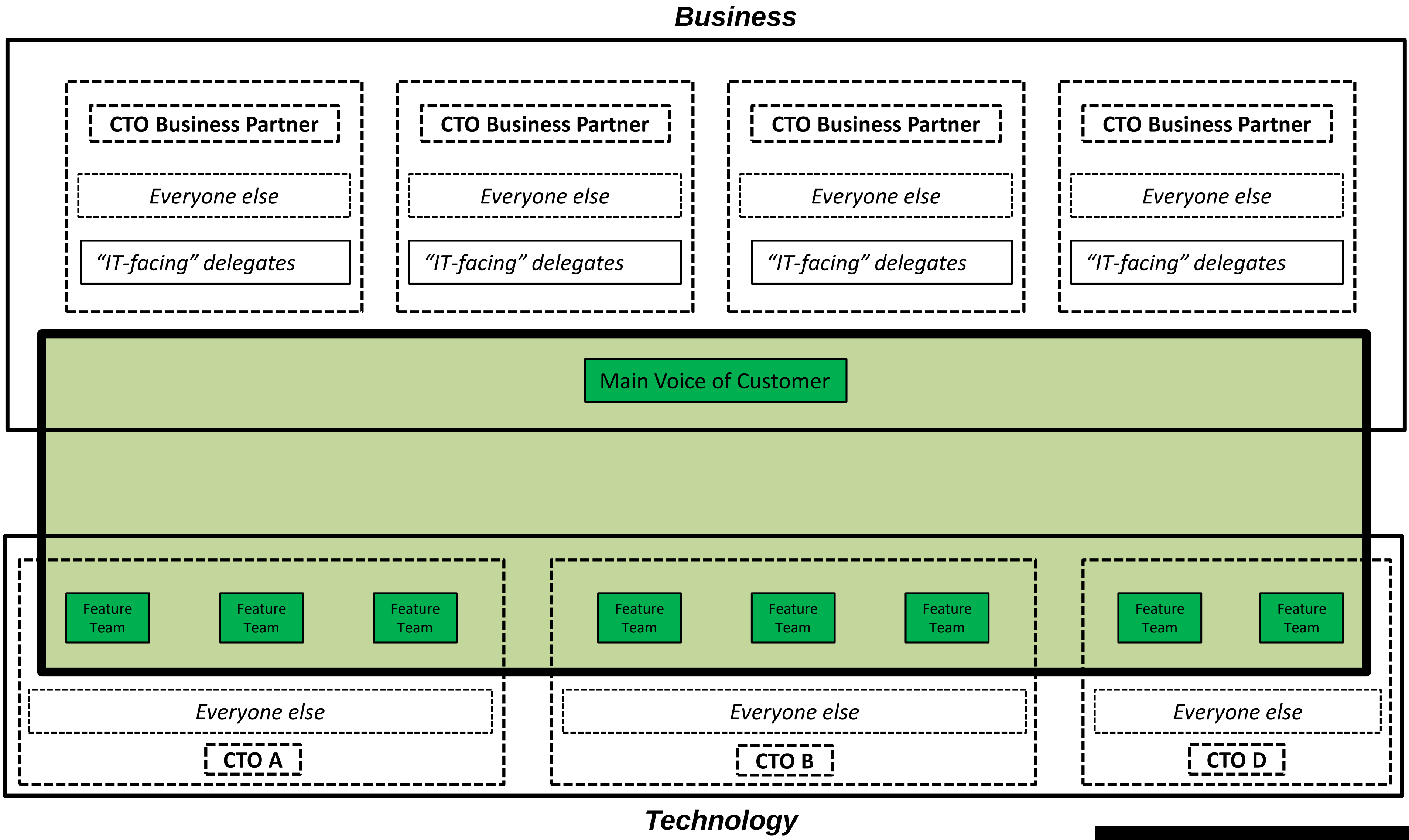
Better Internal Design (Business)



What must be considered?

- Dependencies *within* which Business vertical?
- Dependencies *across* which Business verticals?
- Existing roles/responsibility/workload/compensation/career path?
- Current relationships with IT?

Improved (Overall) Organizational Alignment



Business

CTO Business Partner

Everyone else

"IT-facing" delegates

CTO Business Partner

Everyone else

"IT-facing" delegates

CTO Business Partner

Everyone else

"IT-facing" delegates

CTO Business Partner

Everyone else

"IT-facing" delegates

Main Voice of Customer

Feature Team

Feature Team

Feature Team

Feature Team

Feature Team

Feature Team

Feature Team

Feature Team

Everyone else

CTO A

Everyone else

CTO B

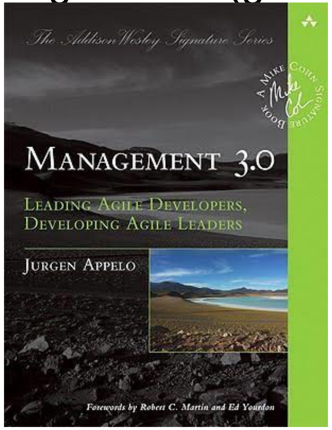
Everyone else

CTO D

Technology

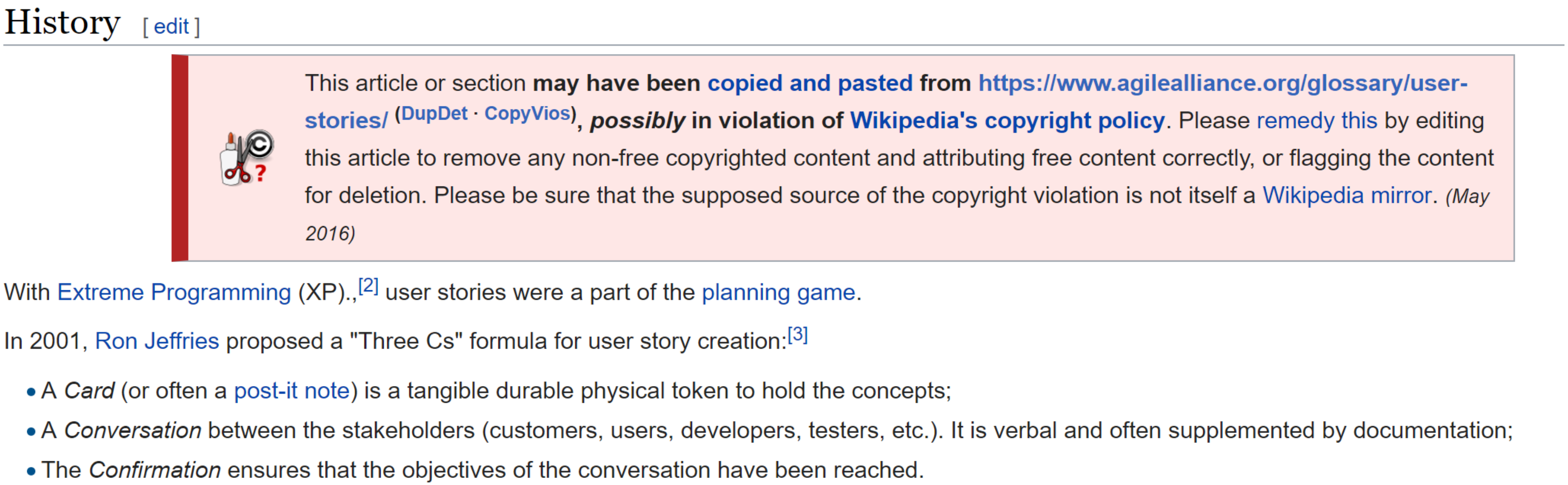
These Topics Should Not be Off Limits

- HR Policies
- Individual Appraisals
- Monetary Rewards
- Internal Competition
- Comp. & Career Path
- Taylorian Mngmnt (bad)
- Mngmnt 3.0 (good)


The Addison Wesley Signature Series
A Mike Cohn Signature Book
Management 3.0
Leading Agile Developers, Developing Agile Leaders
Jurgen Appelo
Forewords by Robert C. Martin and Ed Yourdon

User Stories

History of User Stories



History [edit]

This article or section **may have been copied and pasted from https://www.agilealliance.org/glossary/user-stories/** (DupDet · CopyVios)**, *possibly* in violation of Wikipedia's copyright policy**. Please remedy this by editing this article to remove any non-free copyrighted content and attributing free content correctly, or flagging the content for deletion. Please be sure that the supposed source of the copyright violation is not itself a Wikipedia mirror. *(May 2016)*

With Extreme Programming (XP).,[2] user stories were a part of the planning game.

In 2001, Ron Jeffries proposed a "Three Cs" formula for user story creation:[3]

- A *Card* (or often a post-it note) is a tangible durable physical token to hold the concepts;
- A *Conversation* between the stakeholders (customers, users, developers, testers, etc.). It is verbal and often supplemented by documentation;
- The *Confirmation* ensures that the objectives of the conversation have been reached.

Stories are TOLD!!!

Story Format

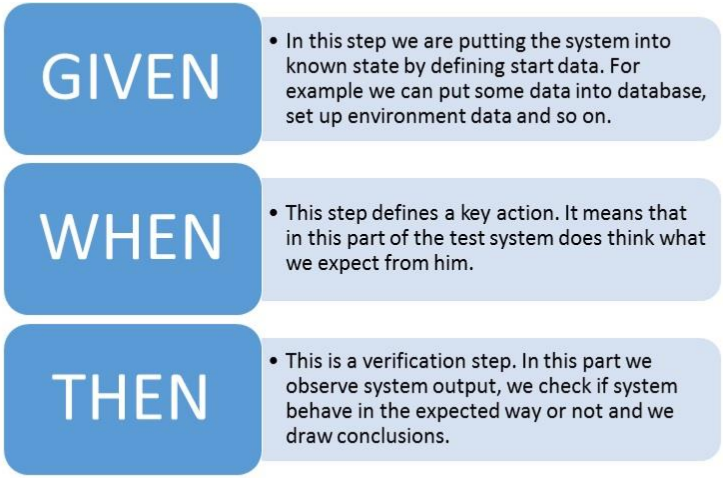
As a <role>
I want <goal>
So that <benefit>

Acceptance criteria:

...

https://www.testingexcellence.com/wp-content/uploads/2016/05/User_Story.jpg

or



GIVEN

- In this step we are putting the system into known state by defining start data. For example we can put some data into database, set up environment data and so on.

WHEN

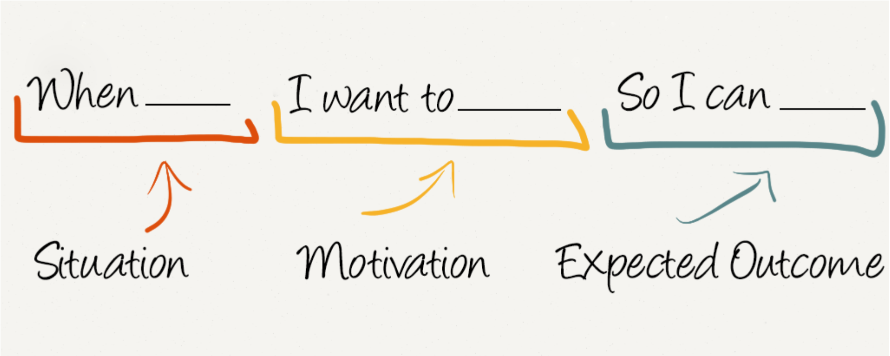
- This step defines a key action. It means that in this part of the test system does think what we expect from him.

THEN

- This is a verification step. In this part we observe system output, we check if system behave in the expected way or not and we draw conclusions.

http://www.blog.j-labs.pl/uploads/Given-When-Then.jpg?1488202157599

or

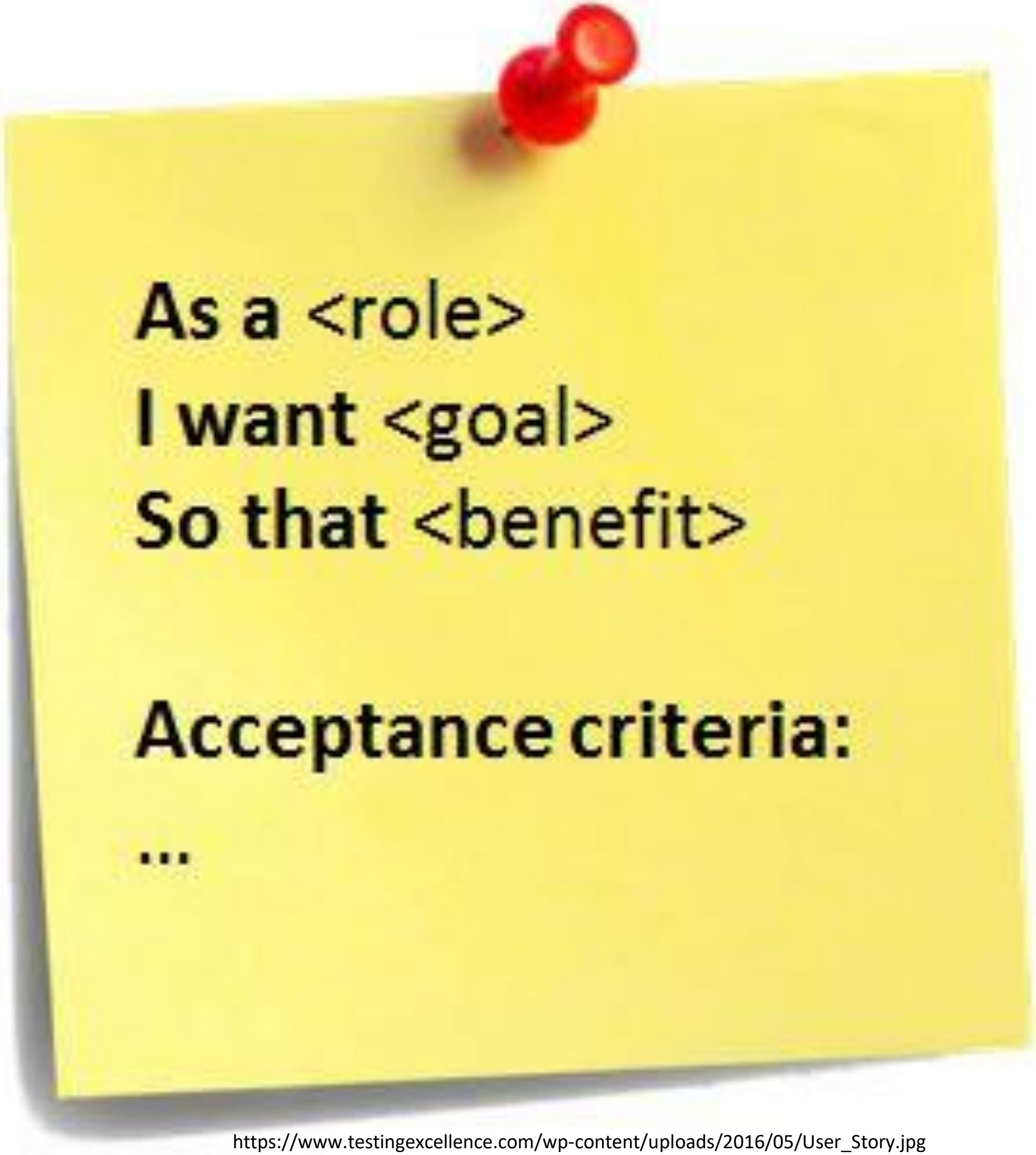


When ____ I want to ____ So I can ____

Situation Motivation Expected Outcome

https://www.productmanagerhq.com/wp-content/uploads/2014/10/User-Stories.png

Most Popular Story Format



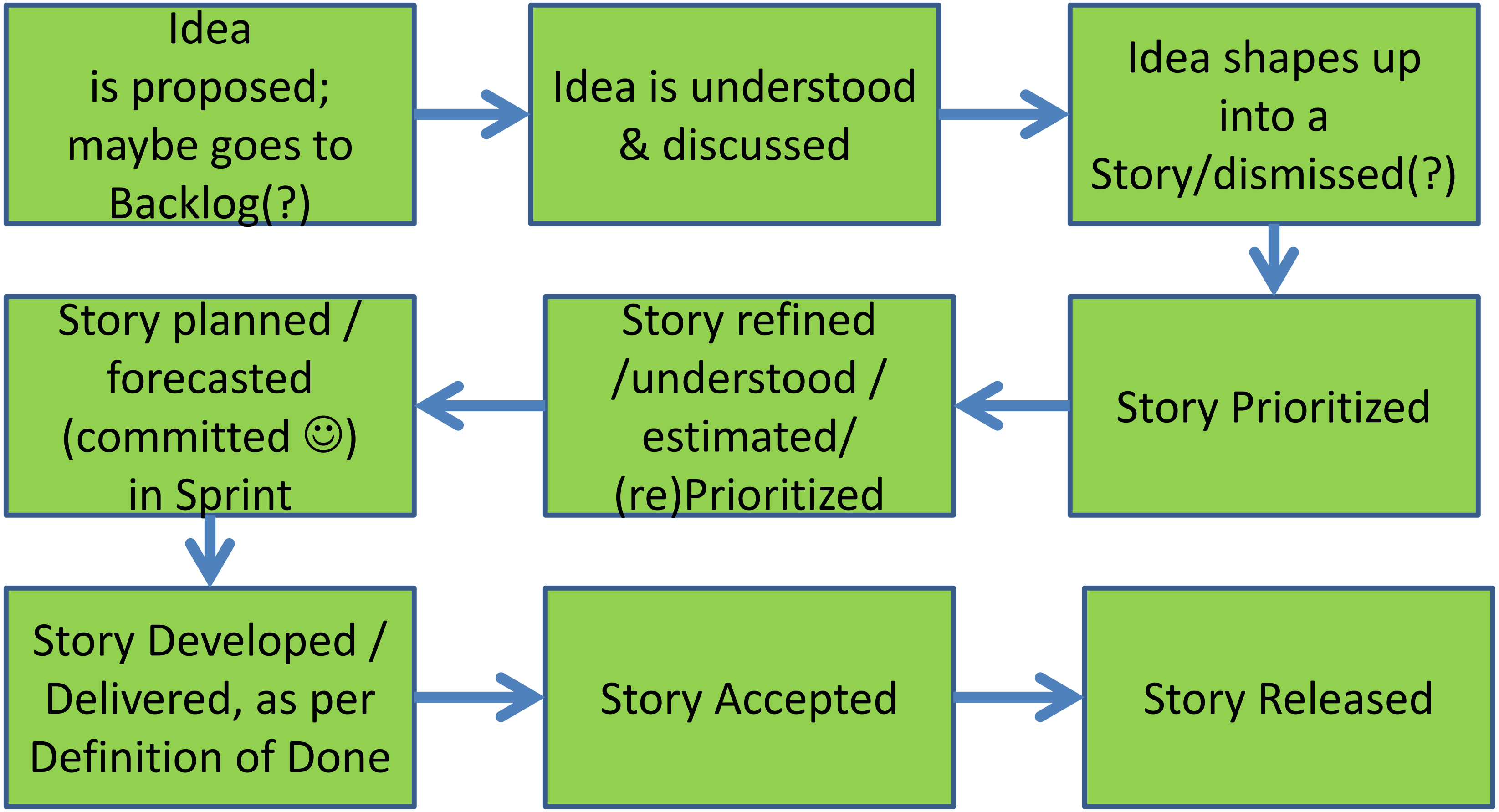
As a <role>
I want <goal>
So that <benefit>

Acceptance criteria:

...

https://www.testingexcellence.com/wp-content/uploads/2016/05/User_Story.jpg

User Story Life Cycle: “Happy” Path



Idea
is proposed;
maybe goes to
Backlog(?)

Idea is understood
& discussed

Idea shapes up
into a
Story/dismissed(?)

Story Prioritized

Story refined
/understood /
estimated/
(re)Prioritized

Story planned /
forecasted
(committed ☺)
in Sprint

Story Developed /
Delivered, as per
Definition of Done

Story Accepted

Story Released

What User Stories Are NOT?

What User Stories Are NOT?

- System-centric Functional Specifications (FRS, FSD)
- Comprehensive, long, voluminous BRDs
- Complex User–centric Use Cases
  - …There are some exceptions, however, such as:
    - A single (happy path) Use Case
    - Essential Use Case, stripped of any hidden assumptions about technology or implementation
- Complex Scenarios (usually ~~ multiple stories)
- Screen Design (UI/UX) ALONE …or DB ALONE…

INVEST - able Stories

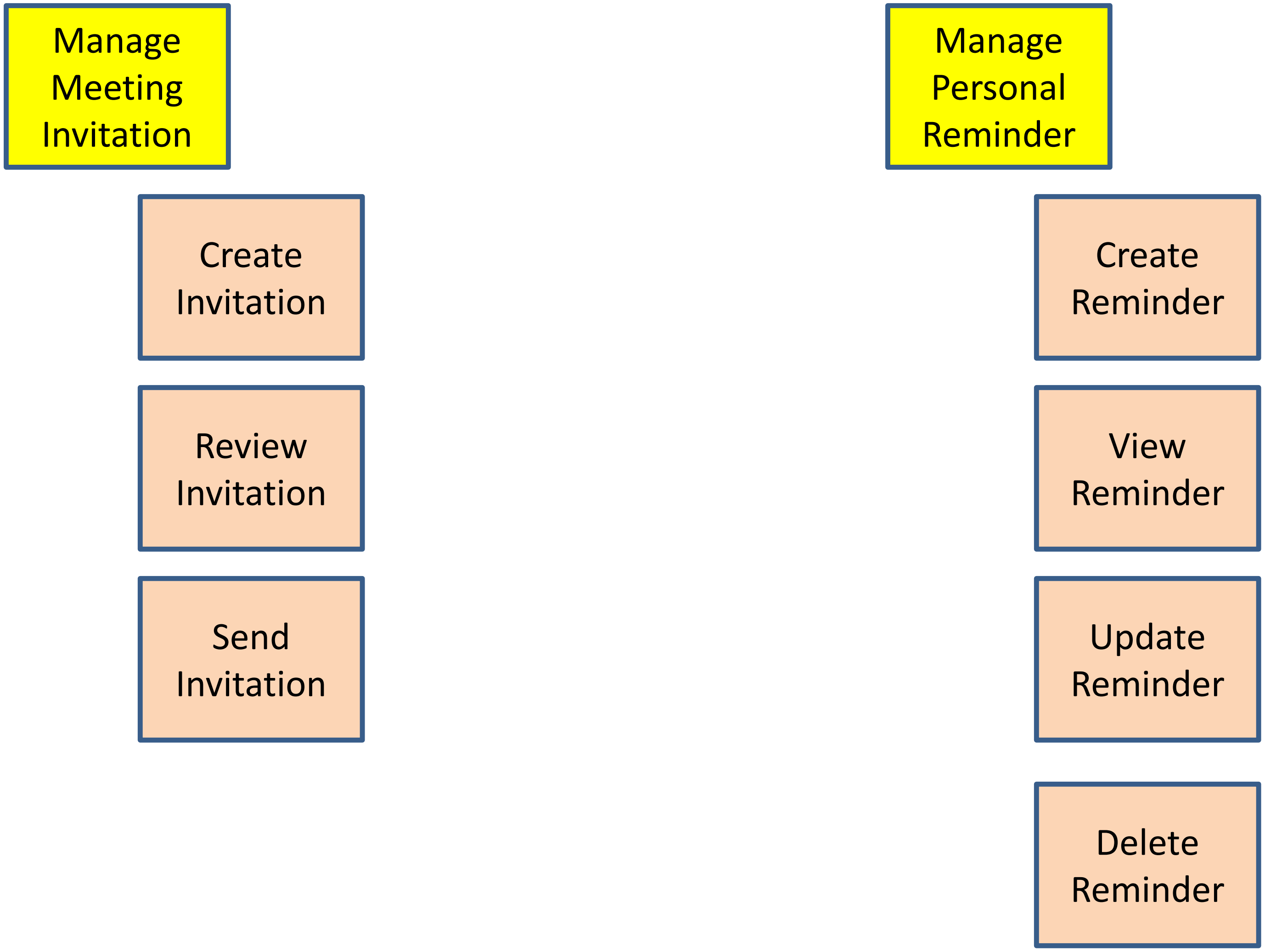
- **Independent** - as small as possible, without making it dependent on other Stories
- **Negotiable** - are not meant to be precise, detailed specifications of a feature. They should not be fixed in stone
- **Valuable** - should be of value to the user
- **Estimatable** - useful way to measure whether a User Story is good or not
- **Small** – small enough to be fully developed & tested in one sprint
- **Testable** - has to do with size and clarity of a story

Compound vs. Compleх Stories

Story Splitting: Can be done any time but often done as a part of Backlog Refinement

- Large stories (Epics) can be of two types:
  - **Compound** Story
    - Consists of multiple shorter stories
    - Easily decomposable
    - Characterized by a large volume of work
  - **Complex** Story
    - Inherently large
    - Cannot be easily disaggregated into smaller stories
    - Can be split into Investigative (spike) and Development stories
    - Characterized by low level of understanding

Splitting Compound Stories



Manage Meeting Invitation

- Create Invitation
- Review Invitation
- Send Invitation

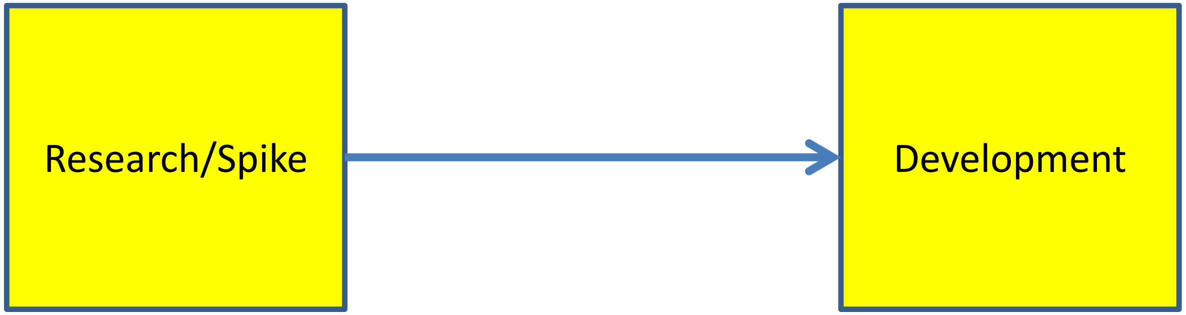
Manage Personal Reminder

- Create Reminder
- View Reminder
- Update Reminder
- Delete Reminder

Splitting Compound Stories

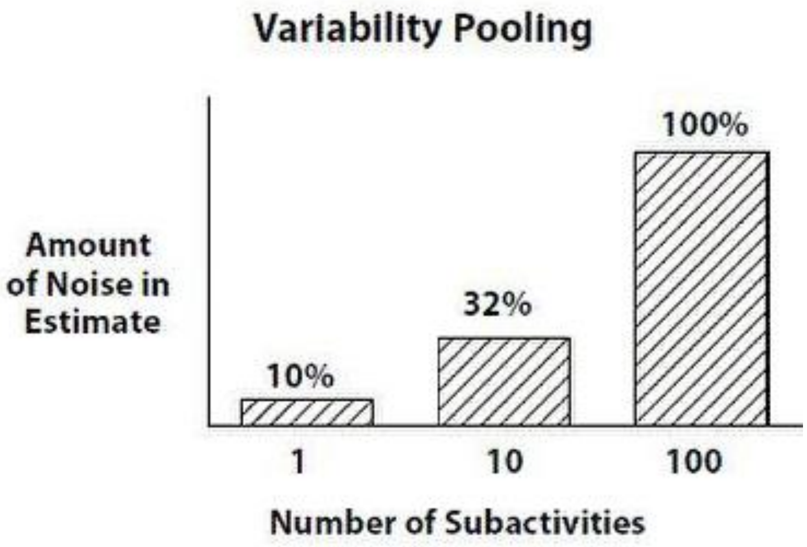
Note: As much as possible, keep research/analysis/non-implementation as a part of a story, not a separate story.

However…



“Stapling” Stories

Reducing ‘Noise’ in Estimation



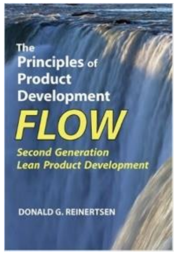
$$\sigma_T = \sqrt{\sigma_1^2 + \sigma_2^2 + \sigma_3^2 ..... + \sigma_n^2}$$

Figure 4-7 When we combine activities with uncorrelated variability, the total standard deviation grows as a square root function. The more we subdivide activities, the higher the coefficient of variation in each subactivity. We can reduce variability by pooling activities together.

Accuracy

!=

Precision


The Principles of Product Development
FLOW
Second Generation Lean Product Development
DONALD G. REINERTSEN

The Principles of Product Development Flow: Second Generation Lean Product Development, by Donald G. Reinertsen

Story “Stapling”

Combining User Stories is recommended for more accurate estimation and planning:

- Small bug fixes can be combined into user stories
- Small UI/UX changes
- Technical Tasks – erroneously presented as User Stories
- If technology thinks that stories are too granular, combine them:
  - “A user can add and edit personal contact information”
  - “A user can add and edit education information”
  - “A user can add and edit working history information”
  - “A user can add and edit compensation information”
  - “A user can add and edit hobbies information”
  - “A user can add and edit references information”

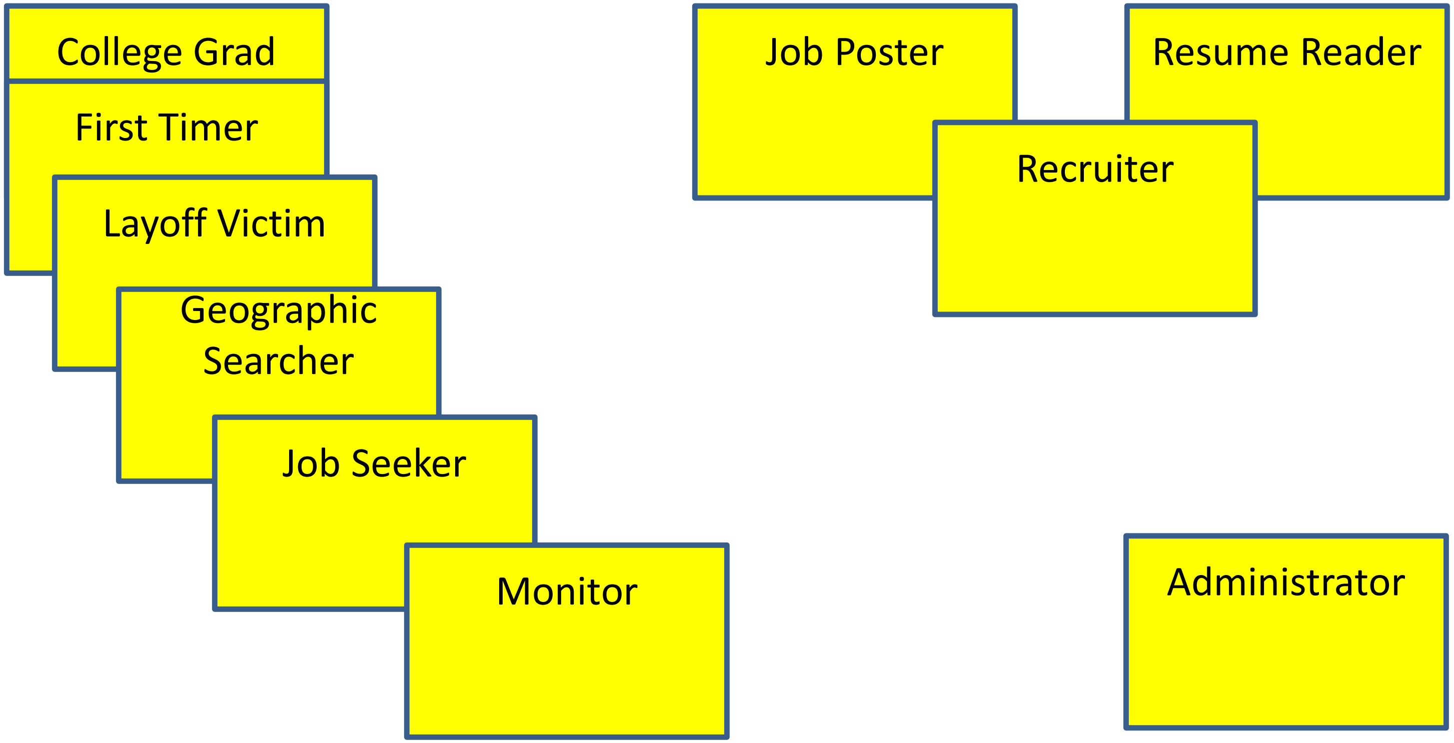
Story “Conversion”

- High Priority Functional Defects ~~ keep as defects
- Low Priority Functional Defects ~~ candidates for user story conversion (low business value)
- Any cosmetic defects ~~ combining into one user story
- Prioritize defects along side with user stories

https://media.licdn.com/mpr/mpr/AAEAAQAAAAAAAAgQAAAAJGNjMTRmMmRhLTc0NmMtNGY4NS1hODVlLTE1YjliYmI5NjI5NA.jpg

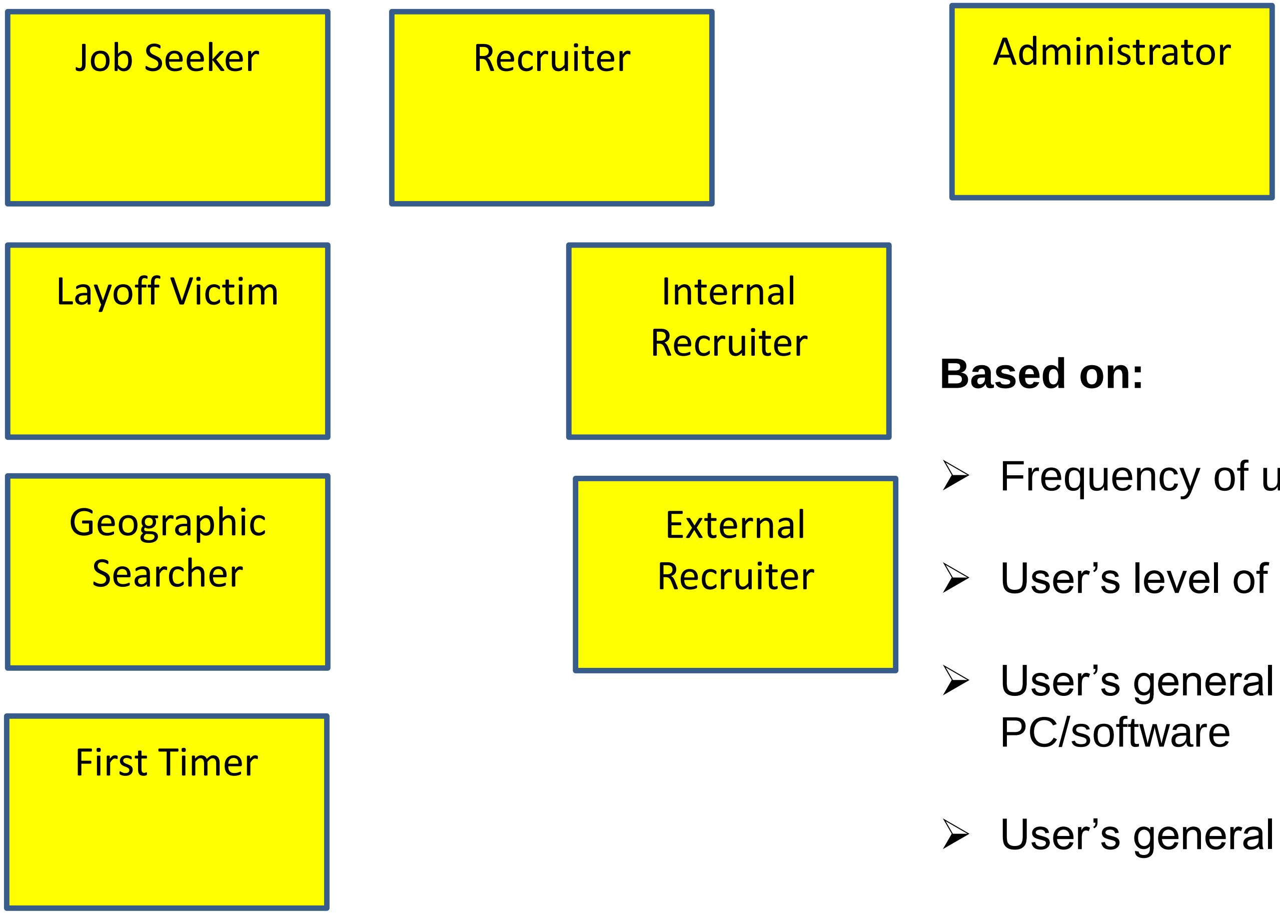
User Role Modeling

User Role Modelling: organizing User Roles for www.MoneyJos.com .   Discover Level of relevance/overlapping



User Role Modeling

User Role Modelling: Group-by, based on relevance/overlapping



Job Seeker

Recruiter

Administrator

Layoff Victim

Internal Recruiter

Geographic Searcher

External Recruiter

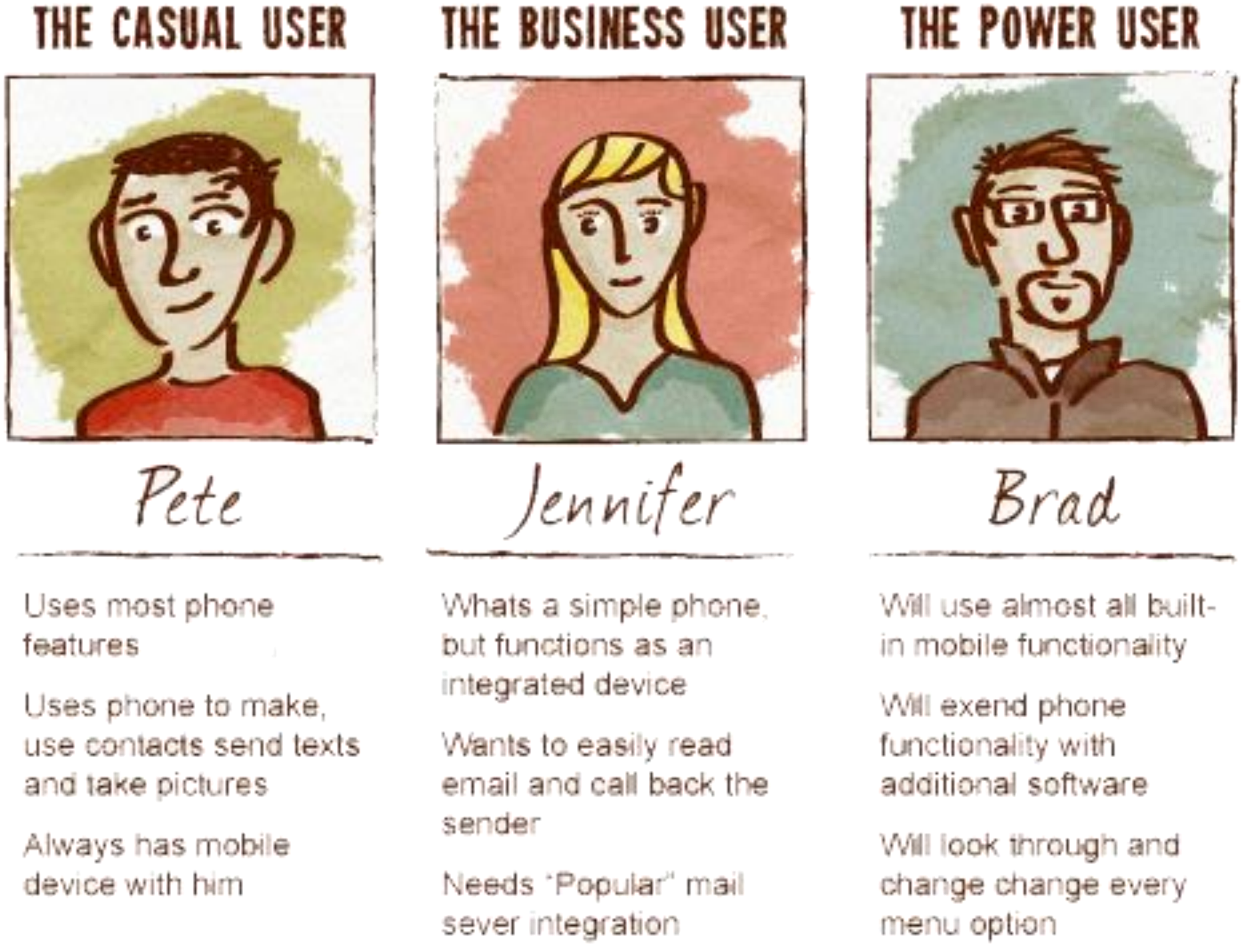
First Timer

Based on:

- Frequency of usage
- User’s level of expertise with domain
- User’s general level of proficiency with PC/software
- User’s general goal for using software

Identifying Personas for User Role Modelling

…thinking of real people, with real job responsibilities, performing real daily tasks….



THE CASUAL USER

Pete

Uses most phone features

Uses phone to make, use contacts send texts and take pictures

Always has mobile device with him

THE BUSINESS USER

Jennifer

Whats a simple phone, but functions as an integrated device

Wants to easily read email and call back the sender

Needs 'Popular' mail sever integration

THE POWER USER

Brad

Will use almost all built-in mobile functionality

Will exend phone functionality with additional software

Will look through and change change every menu option

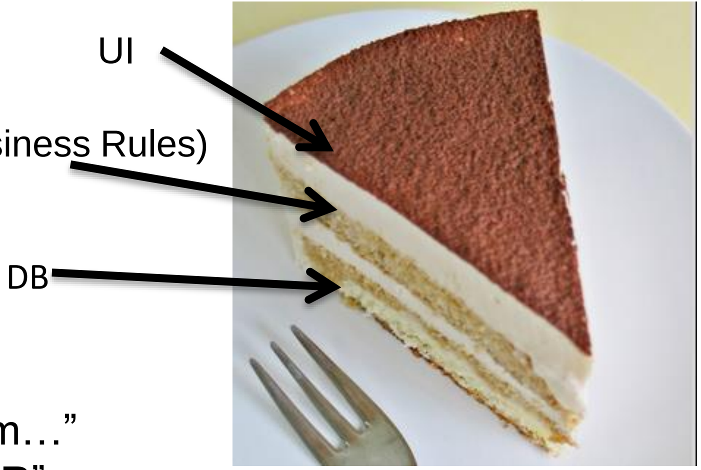
https://www.woodst.com/wp-content/uploads/2014/03/journal-illustration-profiles-lg.png

Story Slicing

Horizontal Slicing

No "Business Value" per slice

Vertical Slicing

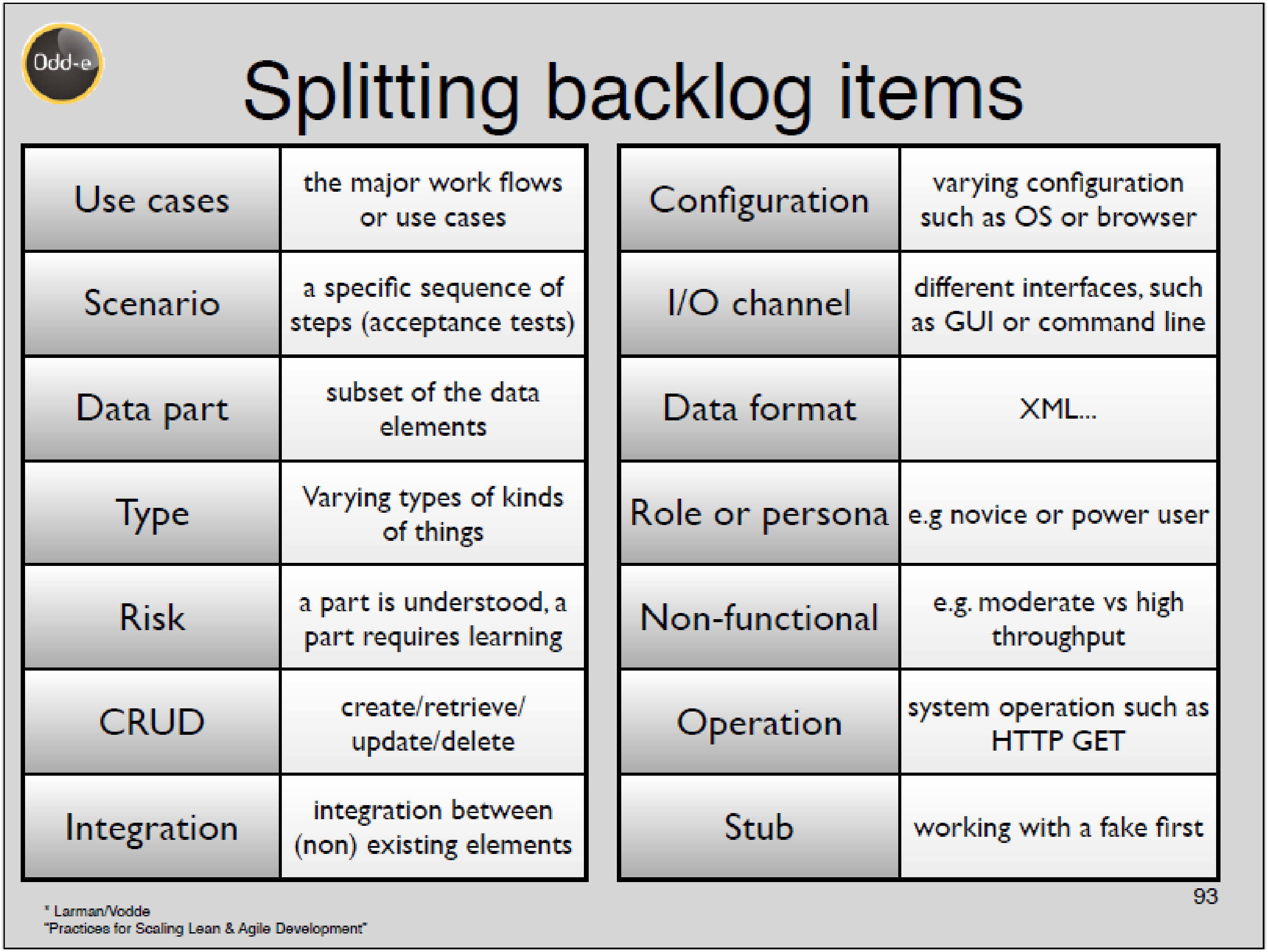


Some "Business Value" per slice

- **Bad** slicing (component-centric/horizontal):
  - "A Job Seeker can fill out **HTML** resume form…"
  - "Information on a resume form is saved to **DB**"

- **Good** slicing (feature-centric/vertical):
  - "A Job Seeker can fill out a resume form that includes ONLY basic information (name, address, education)"
    - OR
  - "A Job Seeker can fill out a resume form that includes ALL of the information that an employer wants to see (employment history)"

Dimensions for User Story Slicing



Odd-e

Splitting backlog items

Use cases	the major work flows or use cases
Scenario	a specific sequence of steps (acceptance tests)
Data part	subset of the data elements
Type	Varying types of kinds of things
Risk	a part is understood, a part requires learning
CRUD	create/retrieve/ update/delete
Integration	integration between (non) existing elements

Configuration	varying configuration such as OS or browser
I/O channel	different interfaces, such as GUI or command line
Data format	XML...
Role or persona	e.g novice or power user
Non-functional	e.g. moderate vs high throughput
Operation	system operation such as HTTP GET
Stub	working with a fake first

93

* Larman/Vodde
"Practices for Scaling Lean & Agile Development"

Sourced from LeSS training materials of C. Larman and B. Vodde

Story is Simple: Single…Single…Single

User Story is always <u>spoken</u>:

- On behalf of One User
- Performing One Key function
- For one Key benefit (could be secondary benefits, but not necessary to include all)

- Examples of 'loose' stories:

- As Market Editor **and** Municipal Bond Trader, I want….blah….blah…

- As Audit Officer, I want to have access to personal files of employees **and** a company's vacation policy **and** CEO's personal bank account….blah…blah….

- As Audit Officer, I want to have access to personal files of employees, so I know where they live, **and** where they went to school **and** history of their prior employment **and** their eye color…blah…blah..

Story “Voicing”

- Write a story in **Active Voice**:

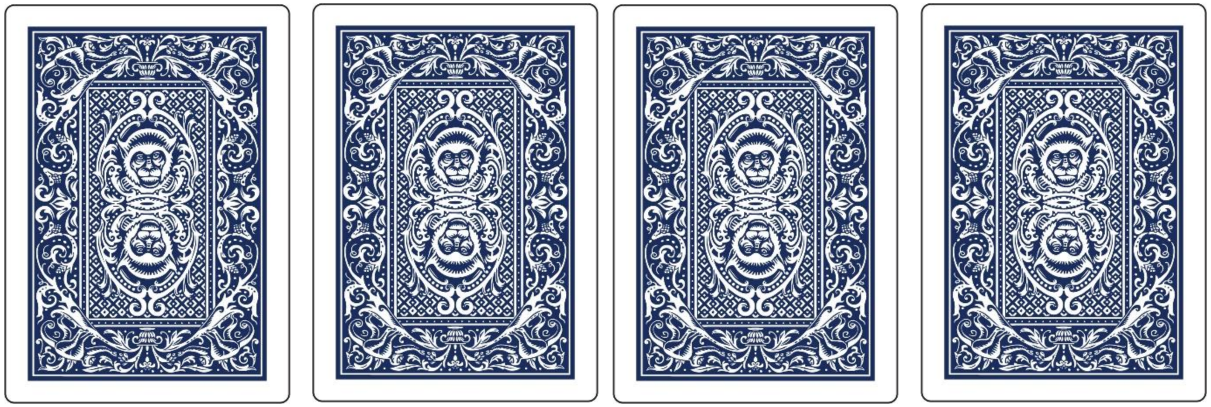
“A Job Seeker can post a resume” <u>is *better* </u>than “A resume can be posted by a Job Seeker”

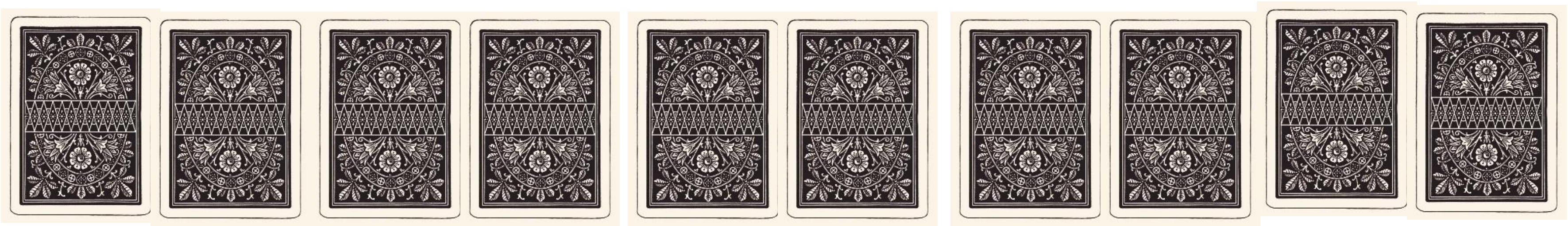
https://harnessconsulting.com/wp-content/uploads/2016/10/tumblr_static_2cl8p3l0edc0gscwkwckgsgw0.png

Non-Functional Requirements on User Stories

- Hardware
- Software
- Performance
- Number of concurrent users
- Page response time
- Usability
- Maintainability
- Extensibility
- Security
- Availability
- Reliability

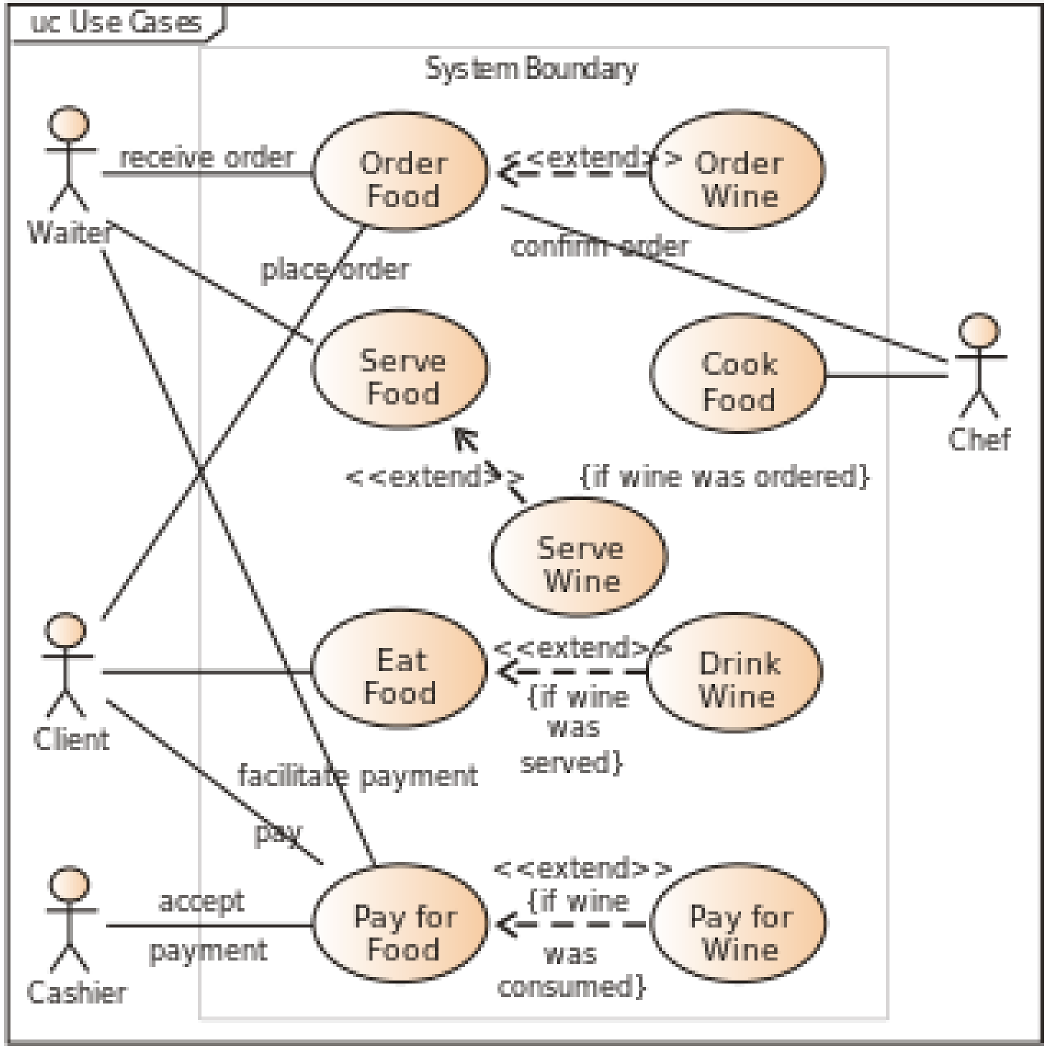
Highest (logically possible) level of abstraction ☺





Use Case – to – User Story Decomposition

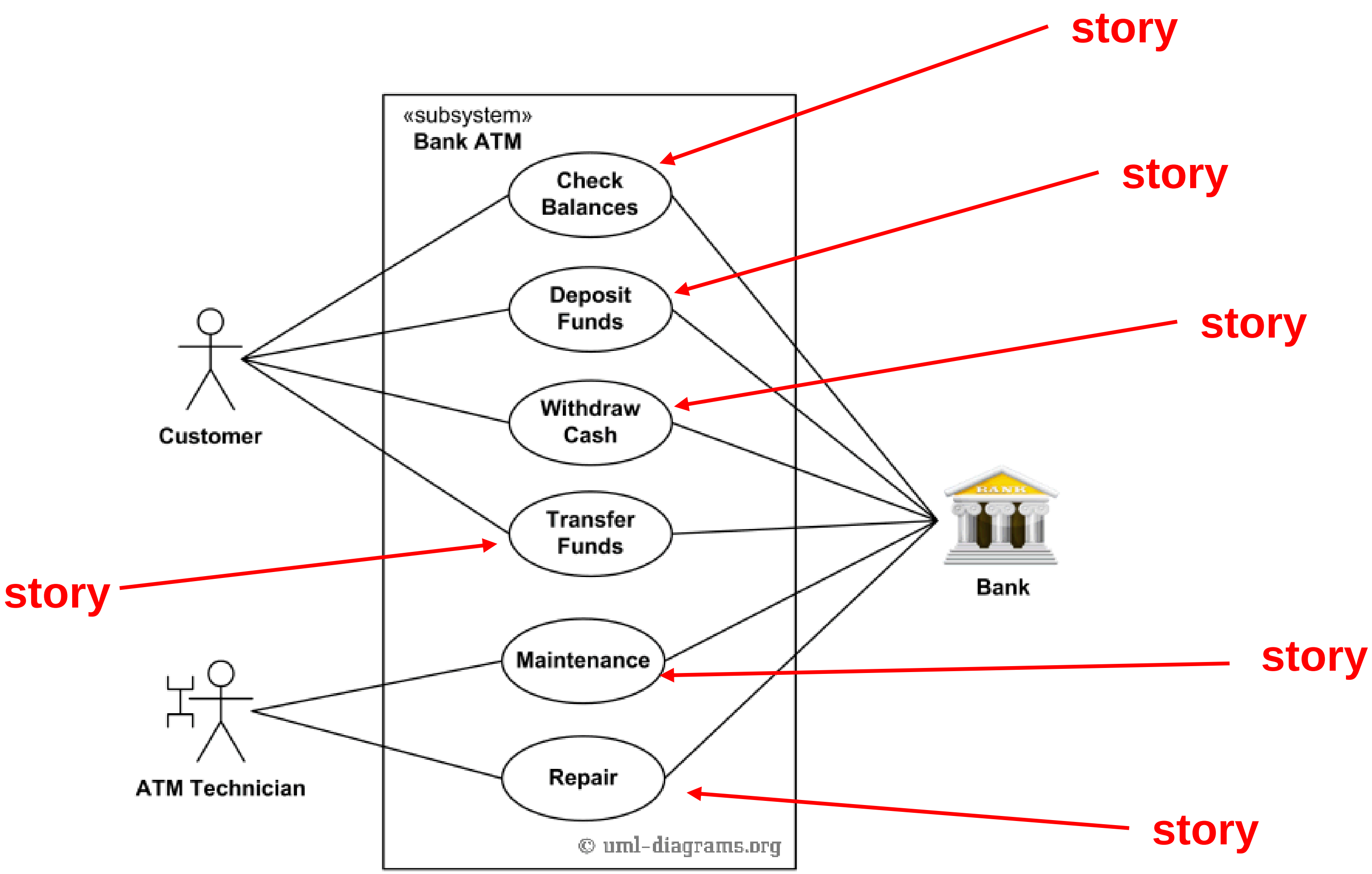
Use Case Refresher 101 ☺ (example)



https://en.wikipedia.org/wiki/Use_case

Use Case – to – User Story Decomposition

Use Case Refresher 101 ☺ (example)



story

story

story

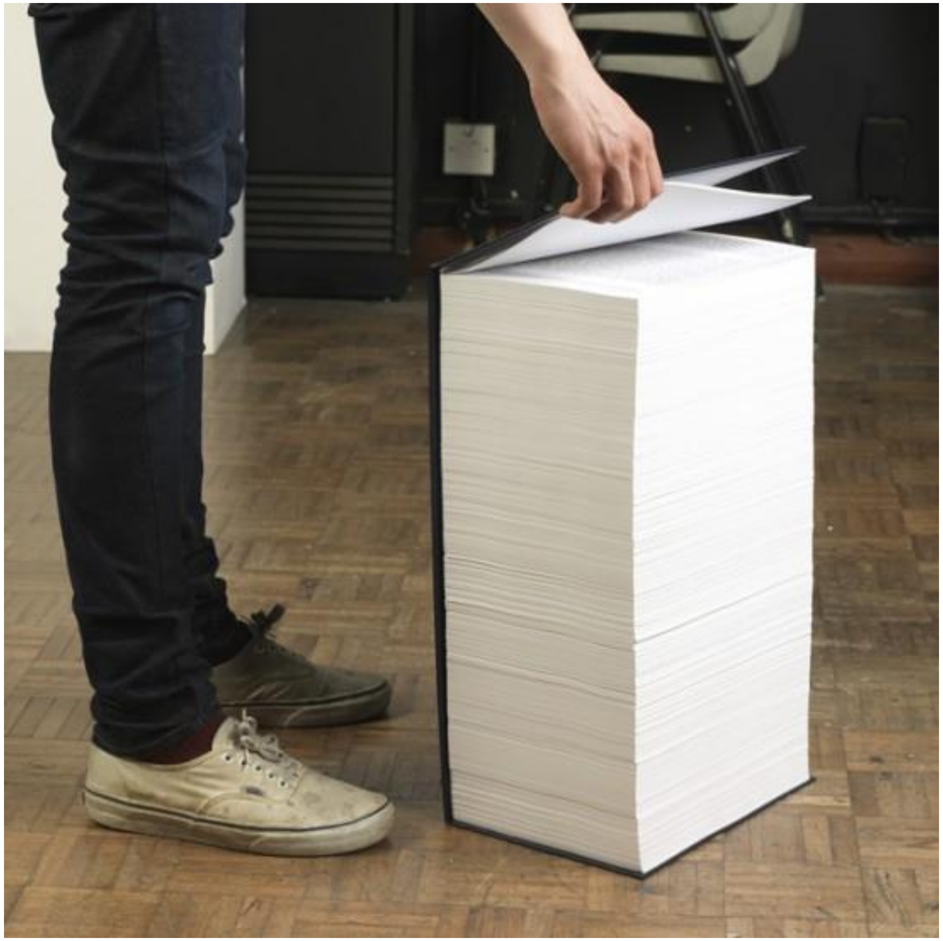
story

story

story

http://www.uml-diagrams.org/bank-atm-uml-use-case-diagram-example.html

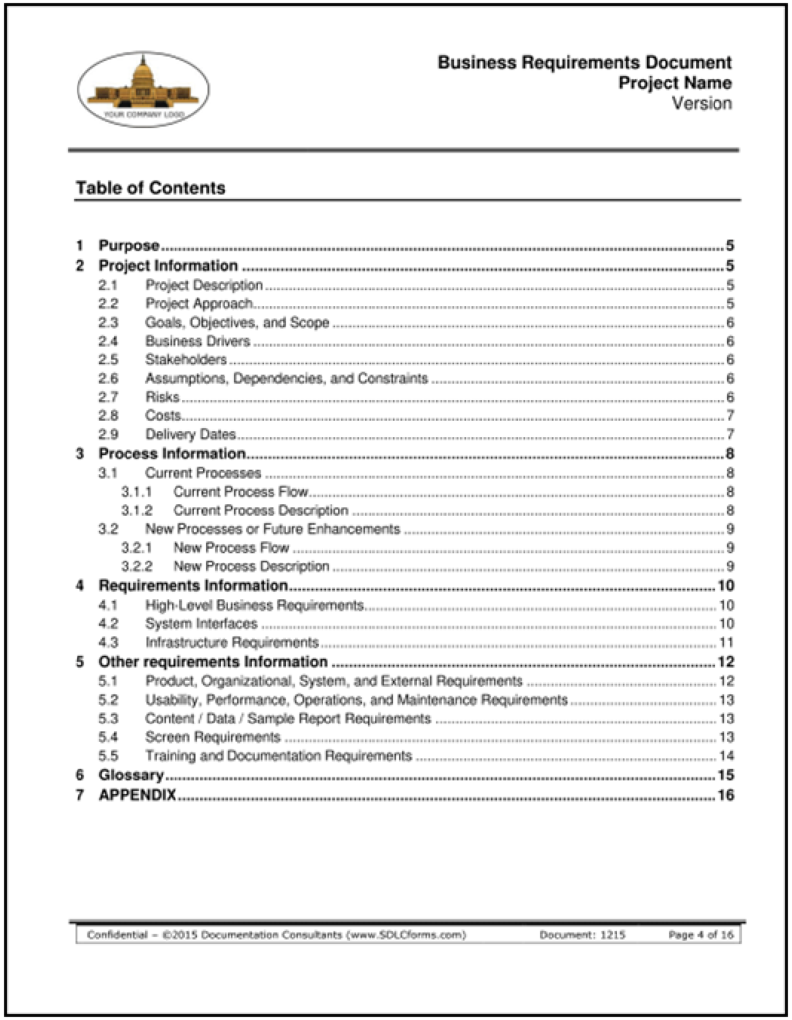
BRD – to – User Story Decomposition



http://gnoted.com/wp-content/uploads/2009/06/5_wikipedia-2-550x550.jpg

Deep breath….

BRD – to – User Story Decomposition



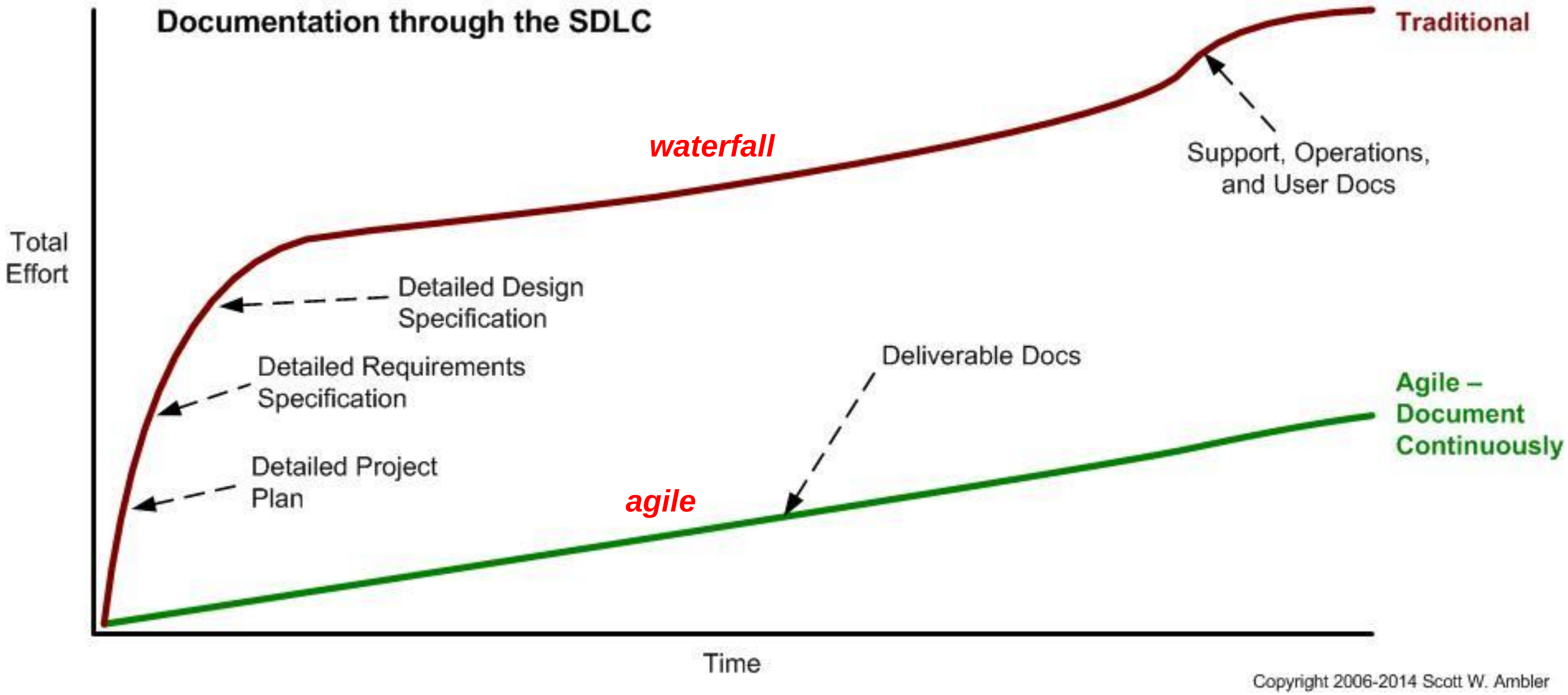
Business Requirements Document
Project Name
Version

Table of Contents

Confidential – ©2015 Documentation Consultants (www.SDLCforms.com) Document: 1215 Page 4 of 16

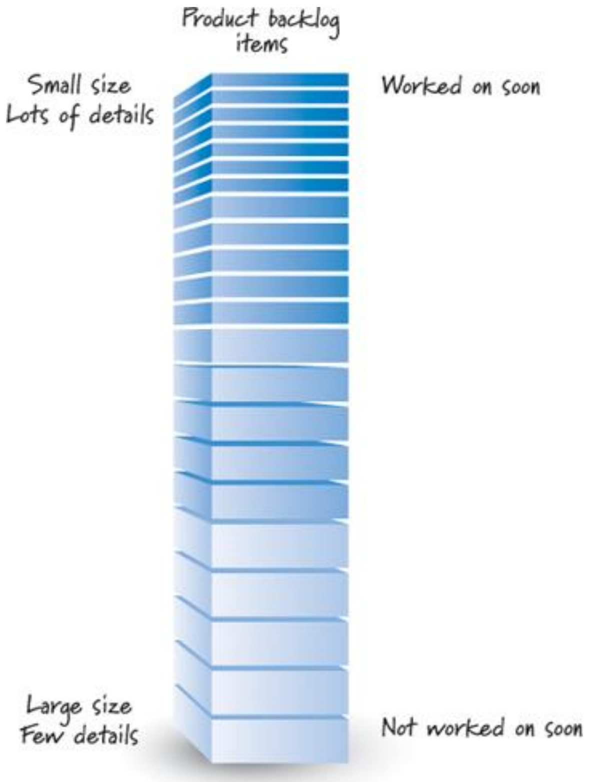
http://www.sdlcforms.com/WOWslider/WOWslider-BusinessRequirementsDocument/data1/images/business_requirements_documentp04500.png

Documentation: Agile vs. Waterfall

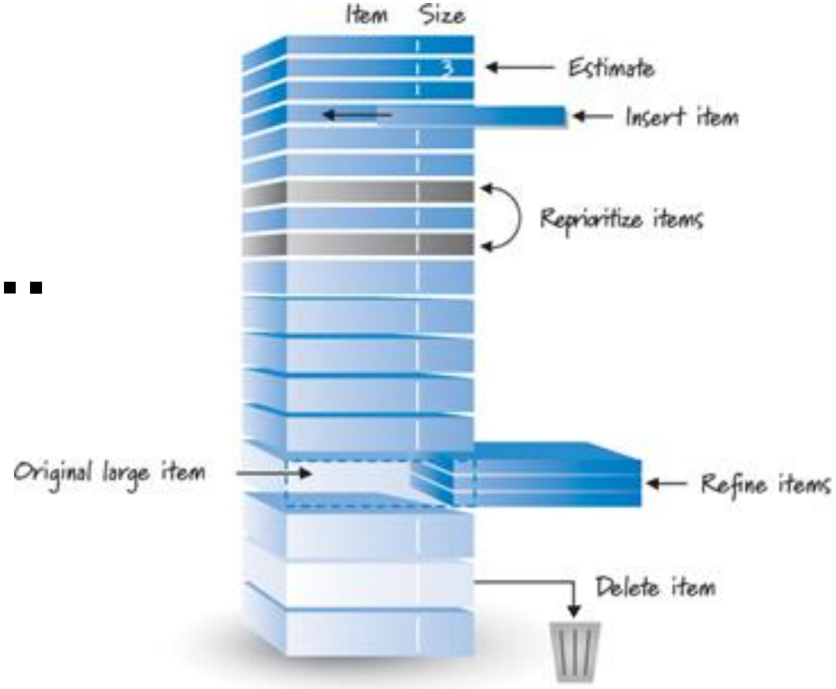


Source: http://www.agilemodeling.com/essays/agileDocumentation.htm

Where Do Stories Belong?



http://www.informit.com/content/images/chap6_9780137043293/elementLinks/06fig03.jpg

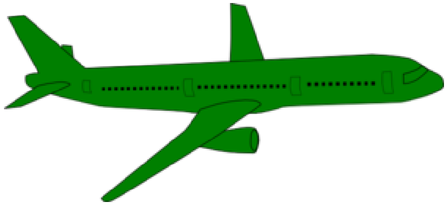


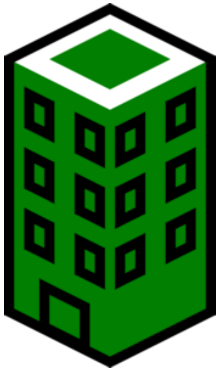
http://www.informit.com/content/images/chap6_9780137043293/elementLinks/06fig06.jpg

Story Writing Example

Reverse Engineering

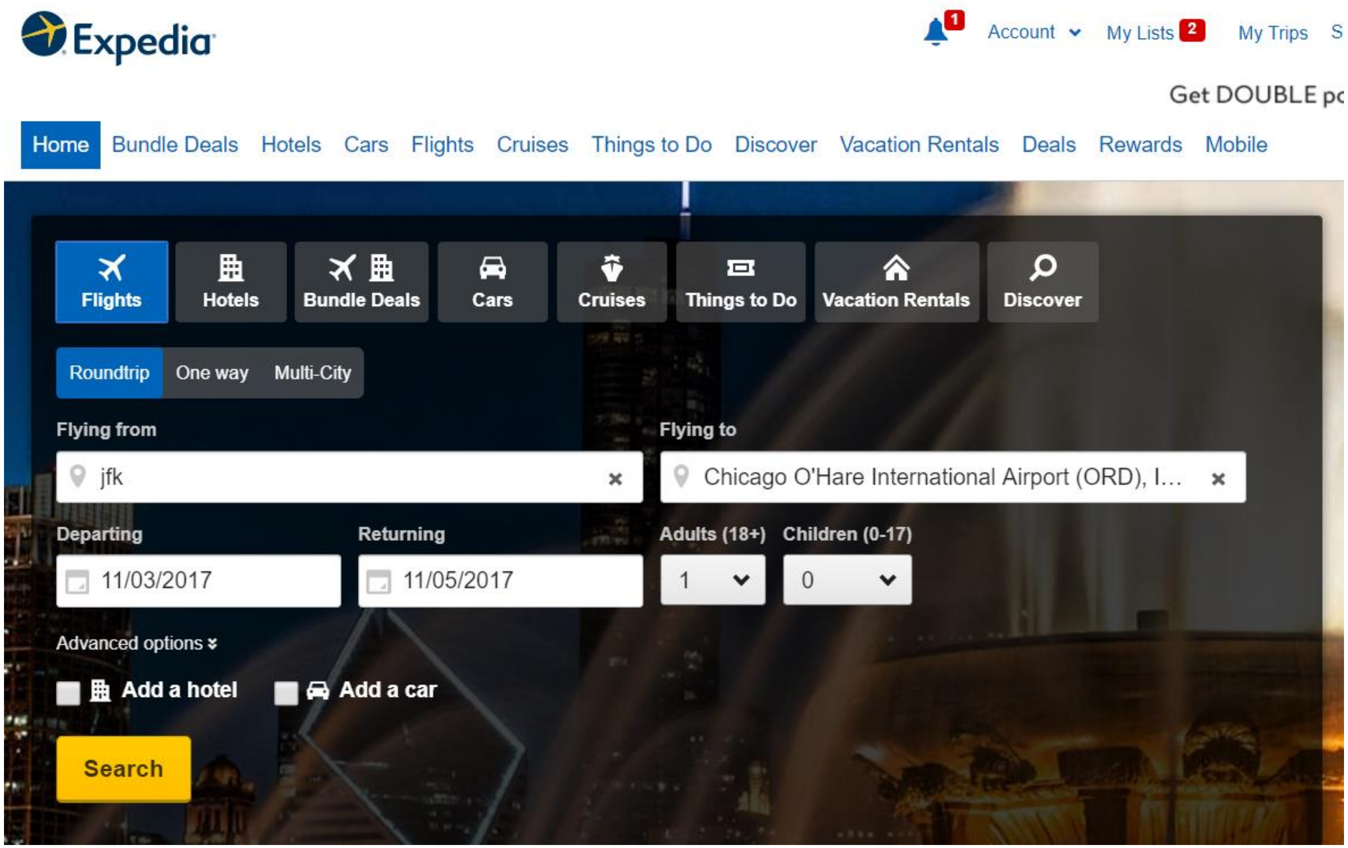
As <Frequent Traveler>, I WANT <to have a system that helps me with my travel accommodations>, SO THAT <I can always find the cheapest and most convenient deals.>





Story Writing Example

As <Frequent Traveler>, I WANT < to be able to search for flights>, SO THAT <I can make the most convenient and cost-effective flight booking>



Flying from: jfk

Flying to: Chicago O'Hare International Airport (ORD), I…

Departing: 11/03/2017

Returning: 11/05/2017

Adults (18+): 1

Children (0-17): 0

□ Add a hotel □ Add a car

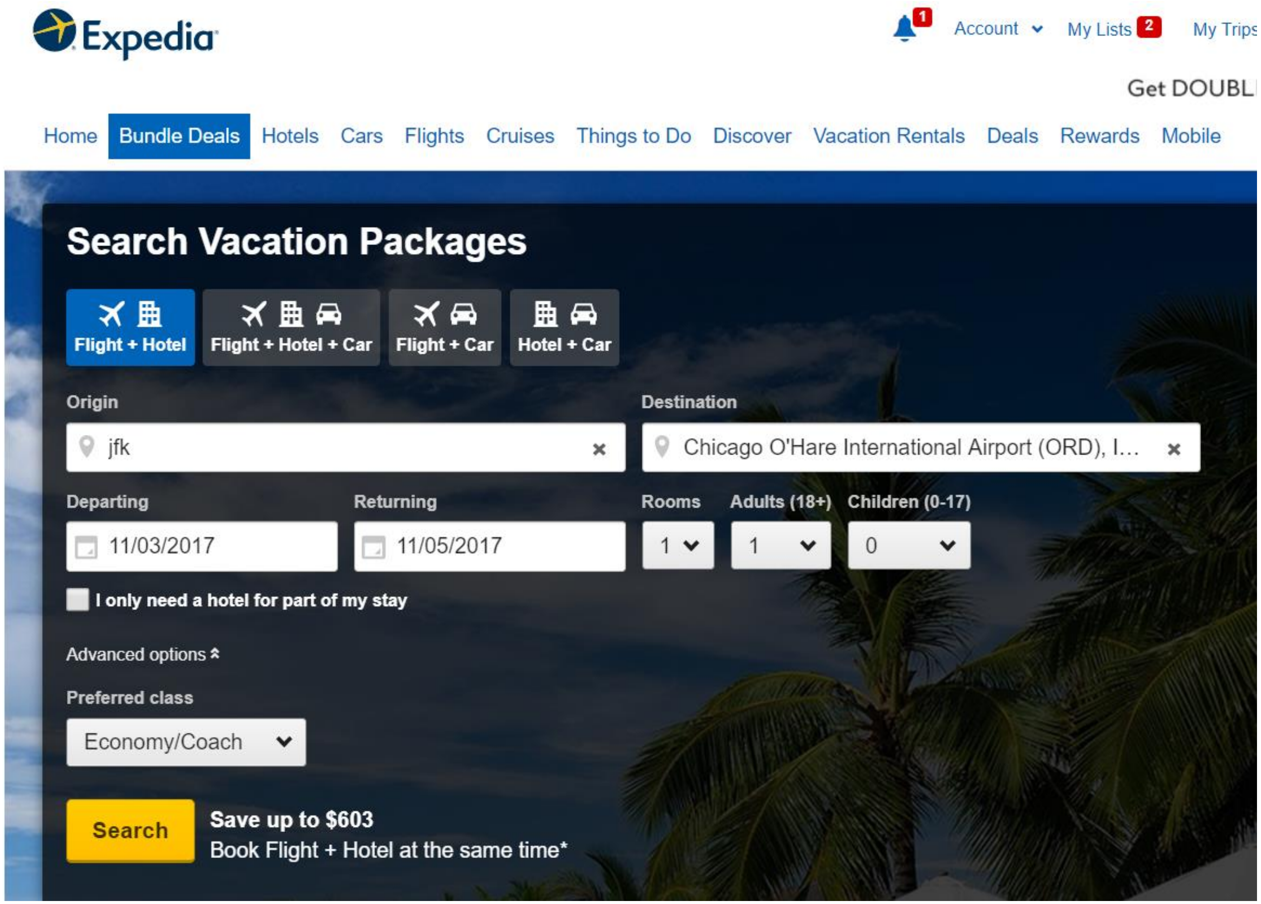
Always ask yourself a question:

↓

What are potential INVEST-able stories that can be "spun off" of this story?

Story Writing Example

As <Frequent Traveler>, I WANT < to be able to search for flights and hotels>, SO THAT <I can make the most convenient and cost-effective flight booking and lodging reservation.>



Search Vacation Packages

Origin: jfk

Destination: Chicago O'Hare International Airport (ORD), I…

Departing: 11/03/2017

Returning: 11/05/2017

Rooms: 1

Adults (18+): 1

Children (0-17): 0

□ I only need a hotel for part of my stay

Preferred class: Economy/Coach

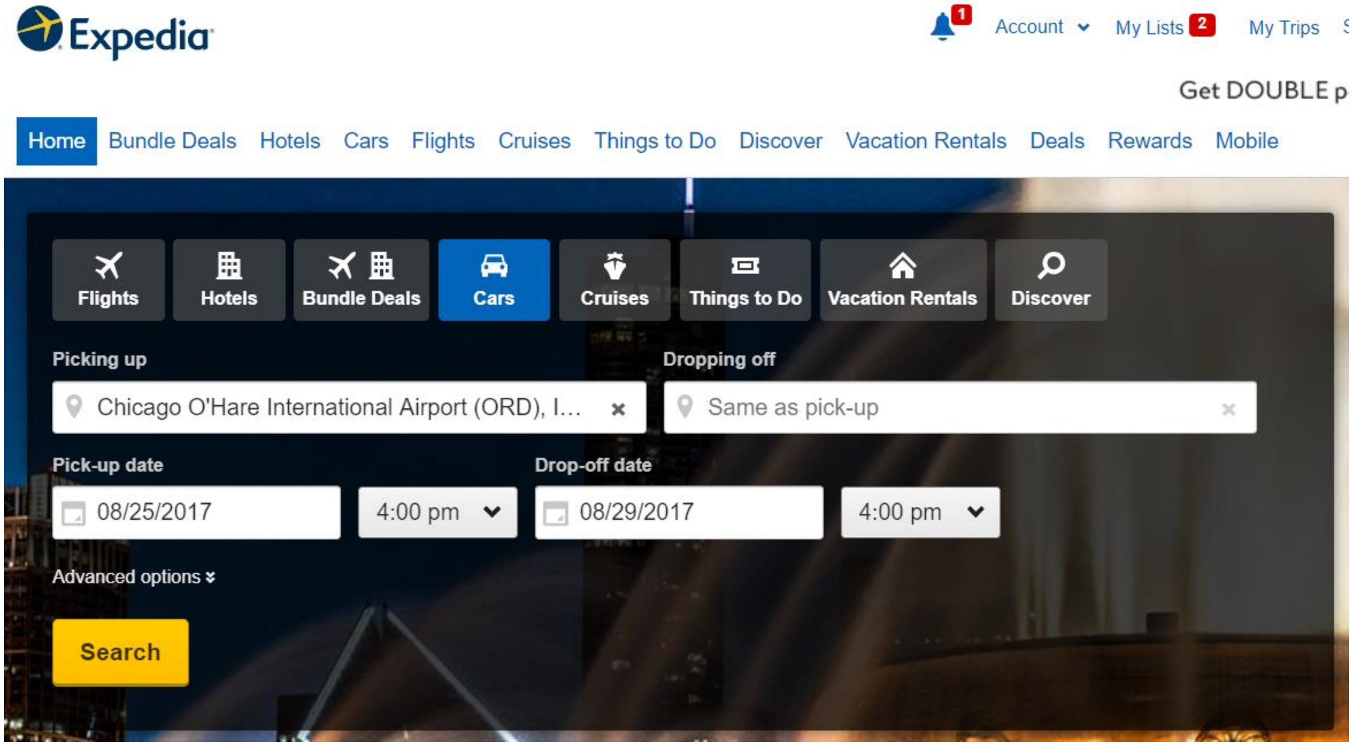
Save up to $603
Book Flight + Hotel at the same time*

Always ask yourself a question:

What are potential INVEST-able stories that can be "spun off" of this story?

Story Writing Example

As <Frequent Traveler>, I WANT < to be able to search for car rental services>, SO THAT <I can rent a car at a destination of my travels.>



Picking up: Chicago O'Hare International Airport (ORD), I…

Dropping off: Same as pick-up

Pick-up date: 08/25/2017 4:00 pm

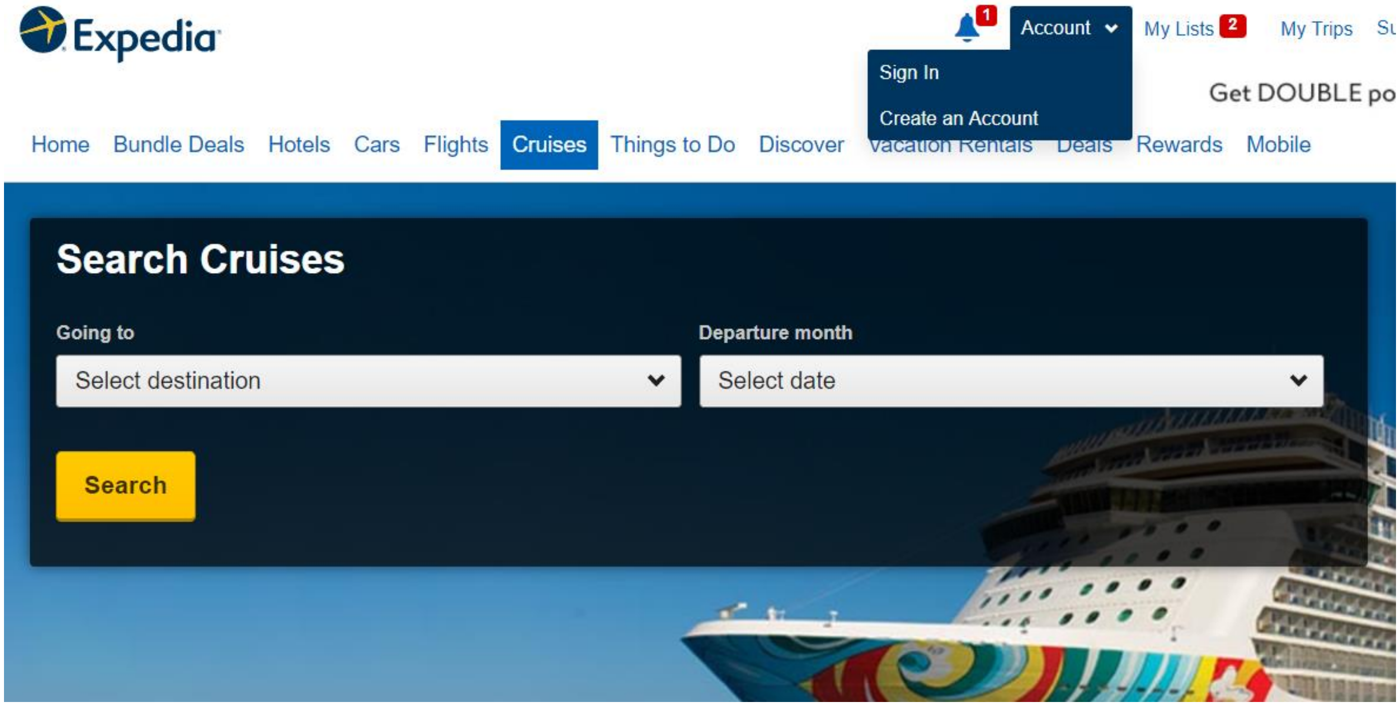
Drop-off date: 08/29/2017 4:00 pm

Always ask yourself a question:

What are potential INVEST-able stories that can be "spun off" of this story?

Story Writing Example

As <Frequent Traveler>, I WANT < to be able to search for ship cruises>, SO THAT <I can find the most exciting trips by sea.>

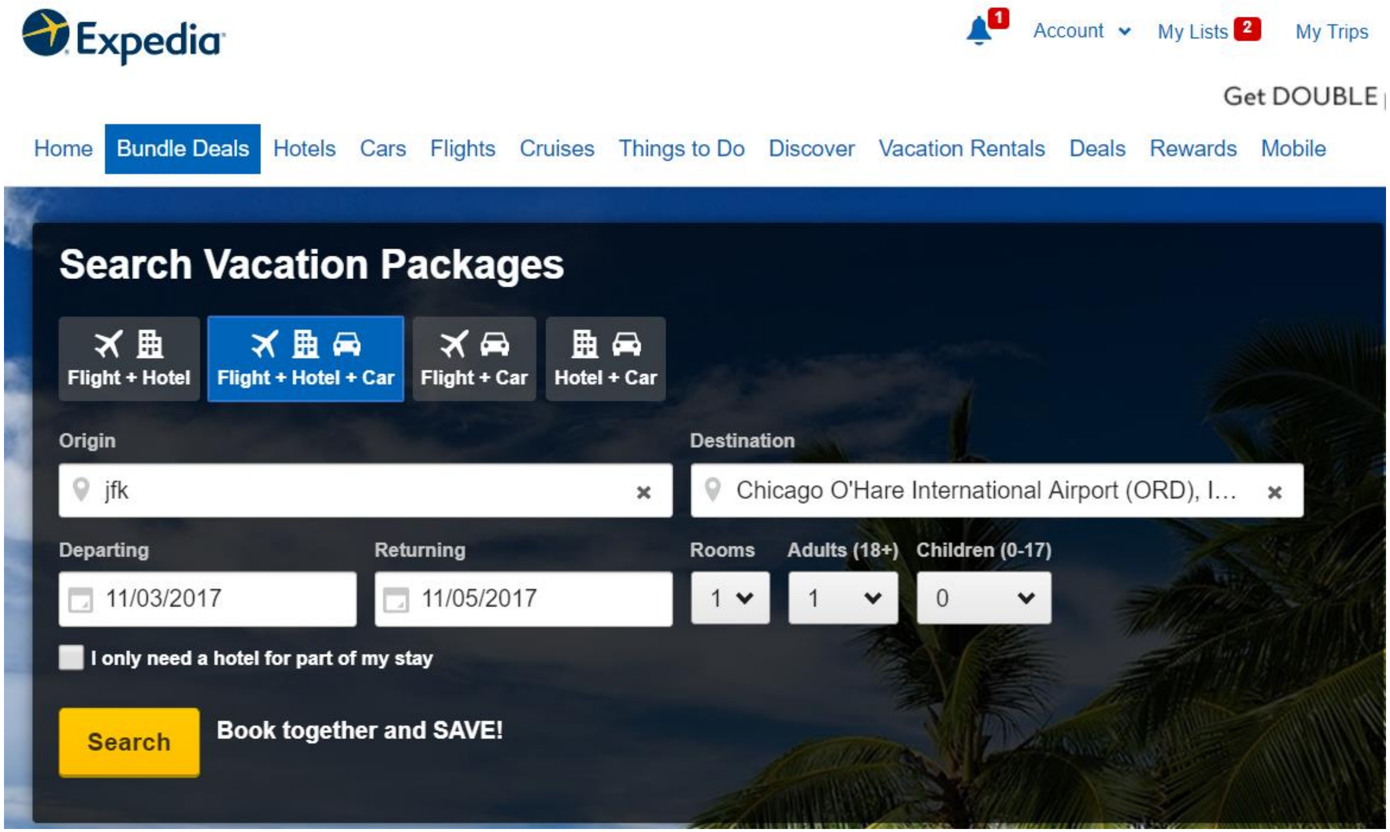


Always ask yourself a question:

What are potential INVEST-able stories that can be "spun off" of this story?

Story Writing Example

As <Frequent Traveler>, I WANT < to be able to search for bundle deals>, SO THAT <I can find the most optimal flight, lodging and ground transportation options, *at once*.>



Search Vacation Packages

Origin: jfk

Destination: Chicago O'Hare International Airport (ORD), I...

Departing: 11/03/2017

Returning: 11/05/2017

Rooms: 1

Adults (18+): 1

Children (0-17): 0

□ I only need a hotel for part of my stay

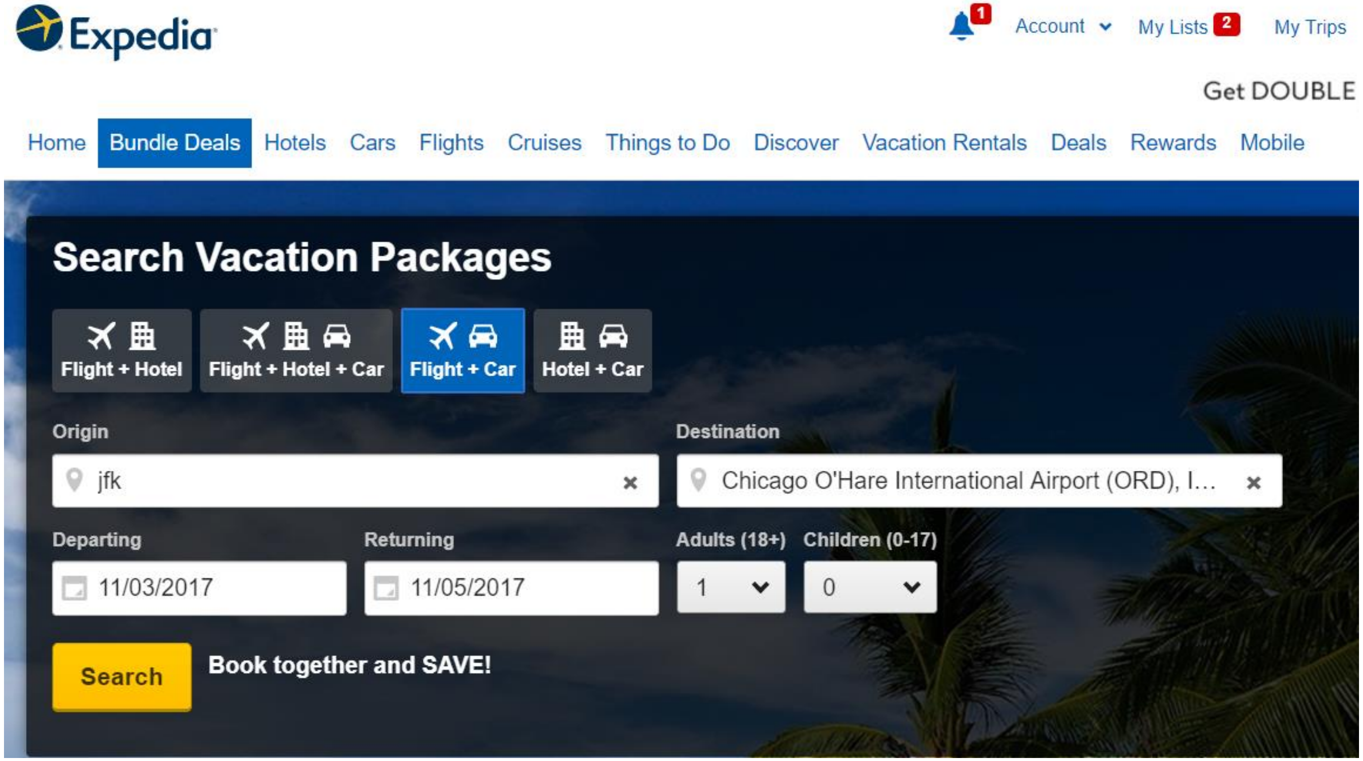
Book together and SAVE!

Always ask yourself a question:

What are potential INVEST-able stories that can be "spun off" of this story?

Story Writing Example

As <Frequent Traveler>, I WANT < to be able to search for bundle deals>, SO THAT <I can find the most optimal flight and ground transportation options, *without* lodging.>



Search Vacation Packages

Origin: jfk

Destination: Chicago O'Hare International Airport (ORD), I...

Departing: 11/03/2017

Returning: 11/05/2017

Adults (18+): 1

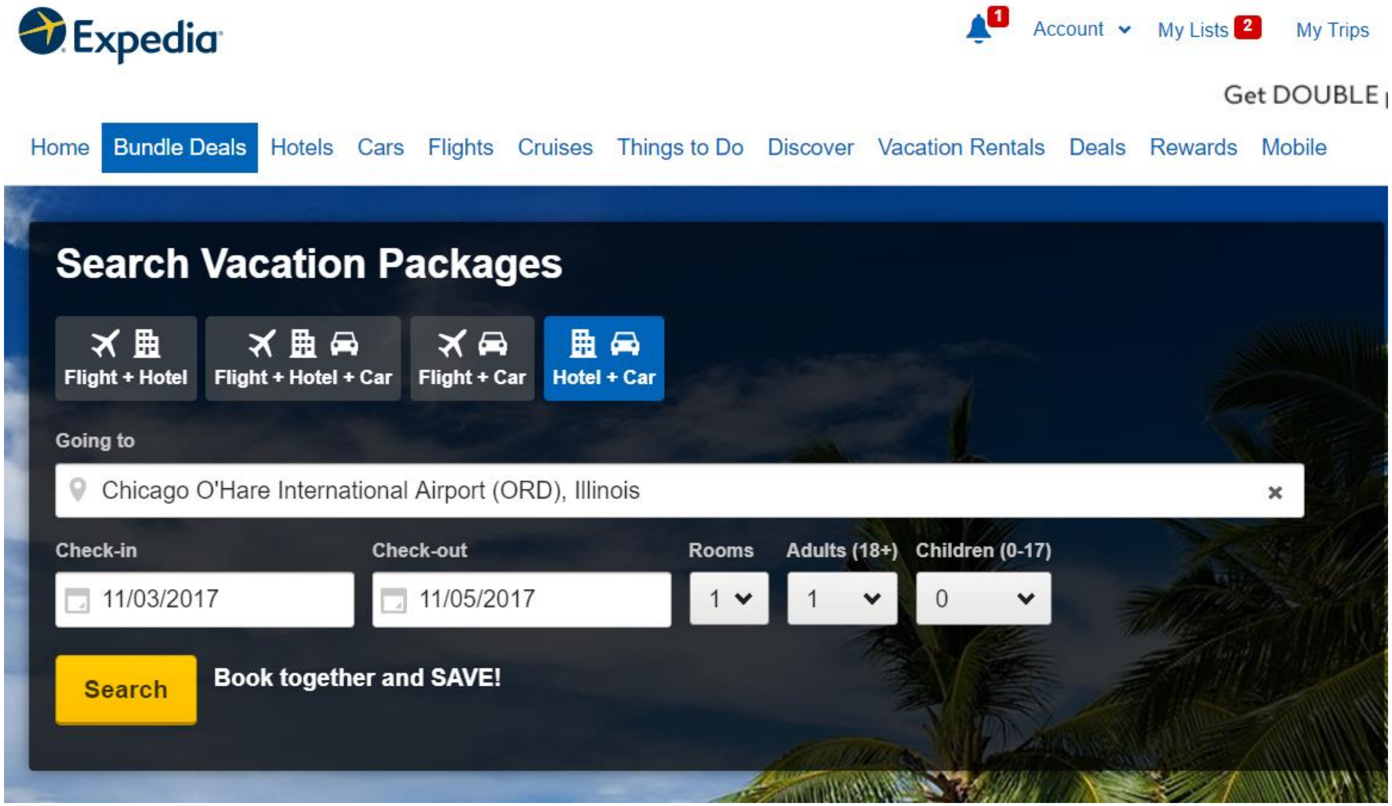
Children (0-17): 0

Always ask yourself a question:

What are potential INVEST-able stories that can be "spun off" of this story?

Story Writing Example

As <Frequent Traveler>, I WANT < to be able to search for bundle deals>, SO THAT <I can find the most optimal lodging and ground transportation options, *without* flying options.>



Search Vacation Packages

Going to: Chicago O'Hare International Airport (ORD), Illinois

Check-in: 11/03/2017

Check-out: 11/05/2017

Rooms: 1

Adults (18+): 1

Children (0-17): 0

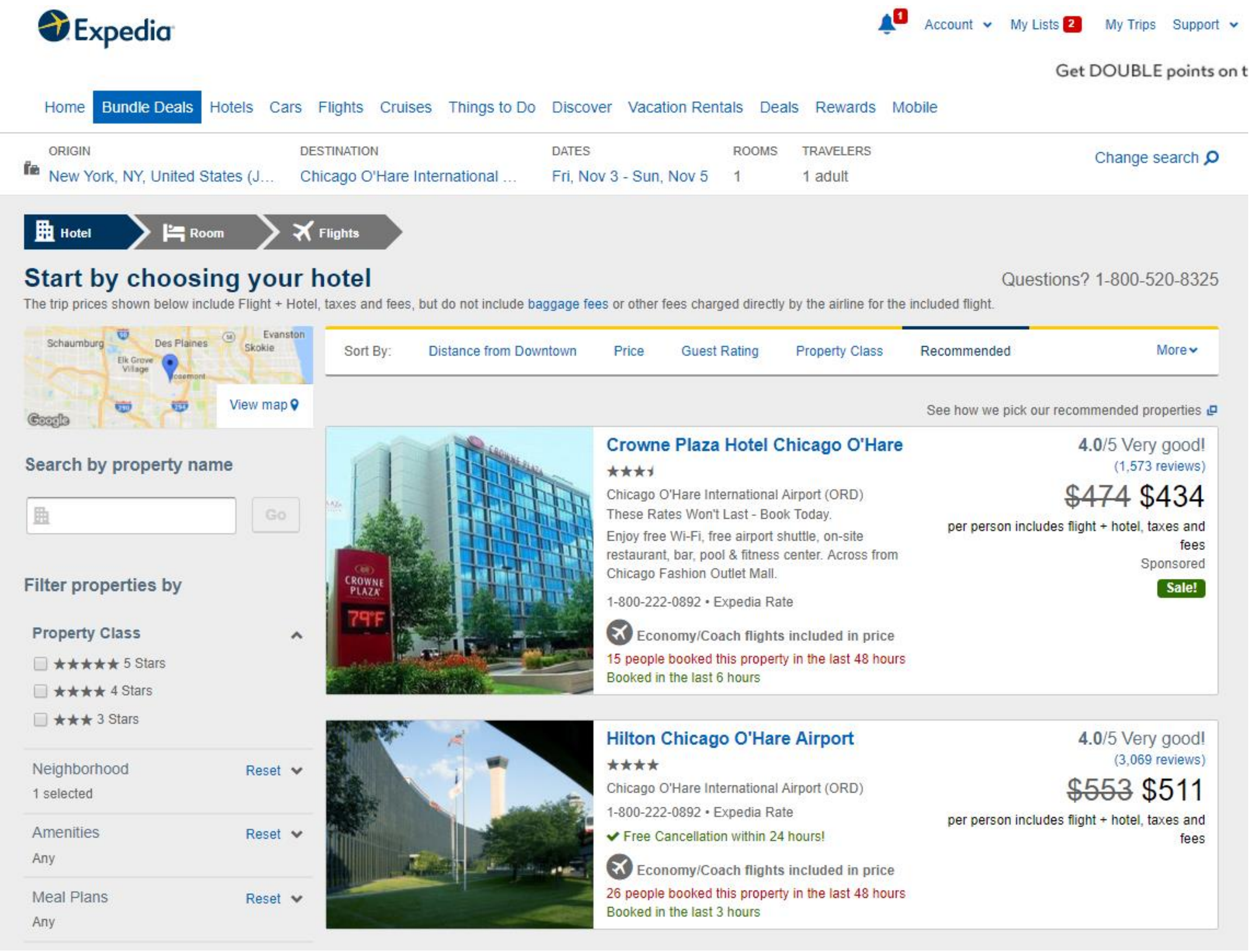
Book together and SAVE!

Always ask yourself a question:

What are potential INVEST-able stories that can be "spun off" of this story?

Story Writing Example

As <Frequent Traveler>, I WANT <to be able to sort all flight results>, SO THAT <they are all presented in the order of my selection preferences>



Expedia

Account My Lists 2 My Trips Support

Get DOUBLE points on t

Home | Bundle Deals | Hotels | Cars | Flights | Cruises | Things to Do | Discover | Vacation Rentals | Deals | Rewards | Mobile

ORIGIN	DESTINATION	DATES	ROOMS	TRAVELERS
New York, NY, United States (J...	Chicago O'Hare International ...	Fri, Nov 3 - Sun, Nov 5	1	1 adult

Change search

Hotel › Room › Flights

Start by choosing your hotel

Questions? 1-800-520-8325

The trip prices shown below include Flight + Hotel, taxes and fees, but do not include baggage fees or other fees charged directly by the airline for the included flight.

View map

Sort By: Distance from Downtown | Price | Guest Rating | Property Class | Recommended | More

See how we pick our recommended properties

Search by property name

Go

Filter properties by

Property Class

□ ★★★★★ 5 Stars

□ ★★★★ 4 Stars

□ ★★★ 3 Stars

Neighborhood Reset
1 selected

Amenities Reset
Any

Meal Plans Reset
Any

Crowne Plaza Hotel Chicago O'Hare
★★★½
Chicago O'Hare International Airport (ORD)
These Rates Won't Last - Book Today.
Enjoy free Wi-Fi, free airport shuttle, on-site restaurant, bar, pool & fitness center. Across from Chicago Fashion Outlet Mall.
1-800-222-0892 • Expedia Rate
Economy/Coach flights included in price
15 people booked this property in the last 48 hours
Booked in the last 6 hours

4.0/5 Very good!
(1,573 reviews)
~~$474~~ $434
per person includes flight + hotel, taxes and fees
Sponsored
Sale!

Hilton Chicago O'Hare Airport
★★★★
Chicago O'Hare International Airport (ORD)
1-800-222-0892 • Expedia Rate
✓ Free Cancellation within 24 hours!
Economy/Coach flights included in price
26 people booked this property in the last 48 hours
Booked in the last 3 hours

4.0/5 Very good!
(3,069 reviews)
~~$553~~ $511
per person includes flight + hotel, taxes and fees

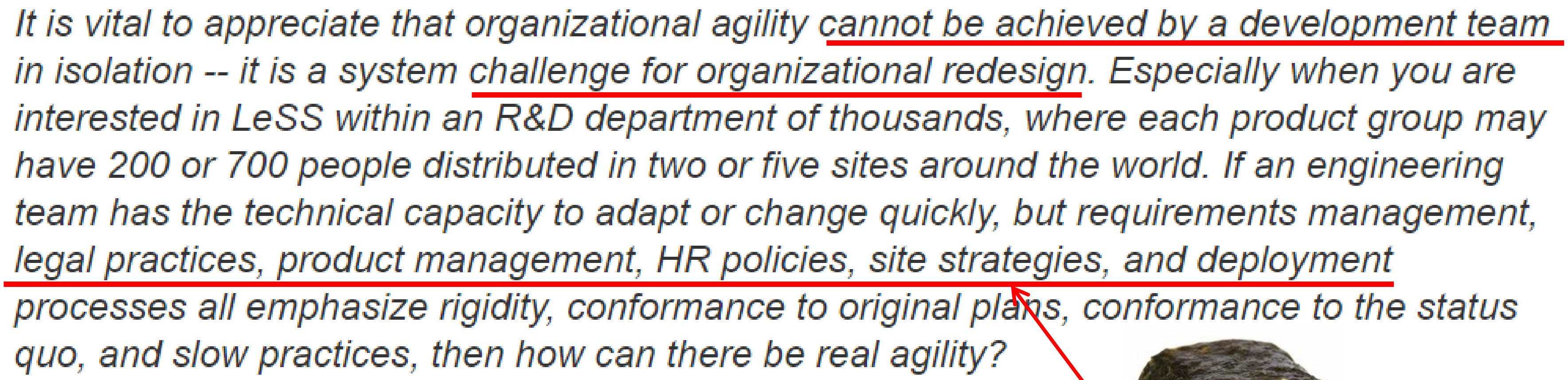
Always ask yourself a question:

↓

What are potential INVEST-able stories that can be "spun off" of this story?

Example of Organizational Story

As <Organizational Leader>, I WANT <to make my organization more agile>, SO THAT <it becomes more completive in a market place>



It is vital to appreciate that organizational agility cannot be achieved by a development team in isolation -- it is a system challenge for organizational redesign. Especially when you are interested in LeSS within an R&D department of thousands, where each product group may have 200 or 700 people distributed in two or five sites around the world. If an engineering team has the technical capacity to adapt or change quickly, but requirements management, legal practices, product management, HR policies, site strategies, and deployment processes all emphasize rigidity, conformance to original plans, conformance to the status quo, and slow practices, then how can there be real agility?

Source: https://www.scrumalliance.org/community/spotlight/craig-larman/june-2015/less-agile-or-less-agile

Example of Organizational Story

As <Organizational Leader>, I WANT <to make my organization more agile>, SO THAT <it becomes more completive in a market place>

- As <CTO>, I WANT <to make my Technology improve engineering practices>, SO THAT <our organization becomes more flexible and less resistant to frequently changing business priorities>
- As <Head of Head of Fixed Income Trading>, I WANT <to make sure that our technology-facing financial analysts are strongly engaged and motivated> , SO THAT <our frequently changing business needs are seamlessly communicated to IT, including priorities and details>
- As <Head of HR>, I WANT <to discontinue stack ranking-driven monetary incentives>, SO THAT <employees act towards each other more like teammates, and less like competitors >
- As <CFO>, I WANT <to make my organization start doing flexible budgeting>, SO THAT <system gaming of "*I need to spend all of my budget this year*" comes to end>
- As <Senior Property Manager>, I WANT <to collocate individuals that closely work together>, SO THAT <time-zone/geography difference is minimized and quality of collaboration increases>

Example

As <CTO>, I WANT <to make my technology improve engineering practices>, SO THAT <we can become more flexible and less resistant to frequently changing business priorities>

- …I WANT….<introduce TDD>…SO THAT….
- …I WANT….<ensure CI/CD>…SO THAT….
- …I WANT….<ensure full test automation>…SO THAT….
- …I WANT….<ensure unit test coverage> …SO THAT….
- …I WANT….<enforce “zero bug deployment strategy”>…SO THAT….
- …I WANT….<remand pair programming>…SO THAT….

Example

- **As <CFO>, I WANT <to make my organization start doing flexible budgeting>, SO THAT <"*I need to spend all of my budget before Dec 31*" mindset is changed>**
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….

Example

As <Head of HR>, I WANT <to discontinue stack-ranking-based monetary incentives allocation>, SO THAT <employees start acting towards each other more like teammates, and less as competitors >

- …I WANT….<to make feedback incremental>…SO THAT….
- …I WANT….<to make feedback voluntary>…SO THAT….
- …I WANT….<to discontinue using bell-shaped curve>…SO THAT….
- …I WANT….<to incentivize teams, not than individuals> …SO THAT….
- …I WANT….<to make subjective incentives more like profit-sharing>…SO THAT….

Example

As <Location Strategy Expert>, I WANT <to collocate individuals that closely work together>, SO THAT <time-zone/geography difference is minimized and quality of collaboration increases>

- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….

Example

- **As <CFO>, I WANT <to make my organization start doing flexible budgeting>, SO THAT <system gaming of "*I need to spend all of my budget this year*" comes to end>**
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….
- …I WANT….<xxxxxxxxxxxxxx>…SO THAT….

User Story Mapping

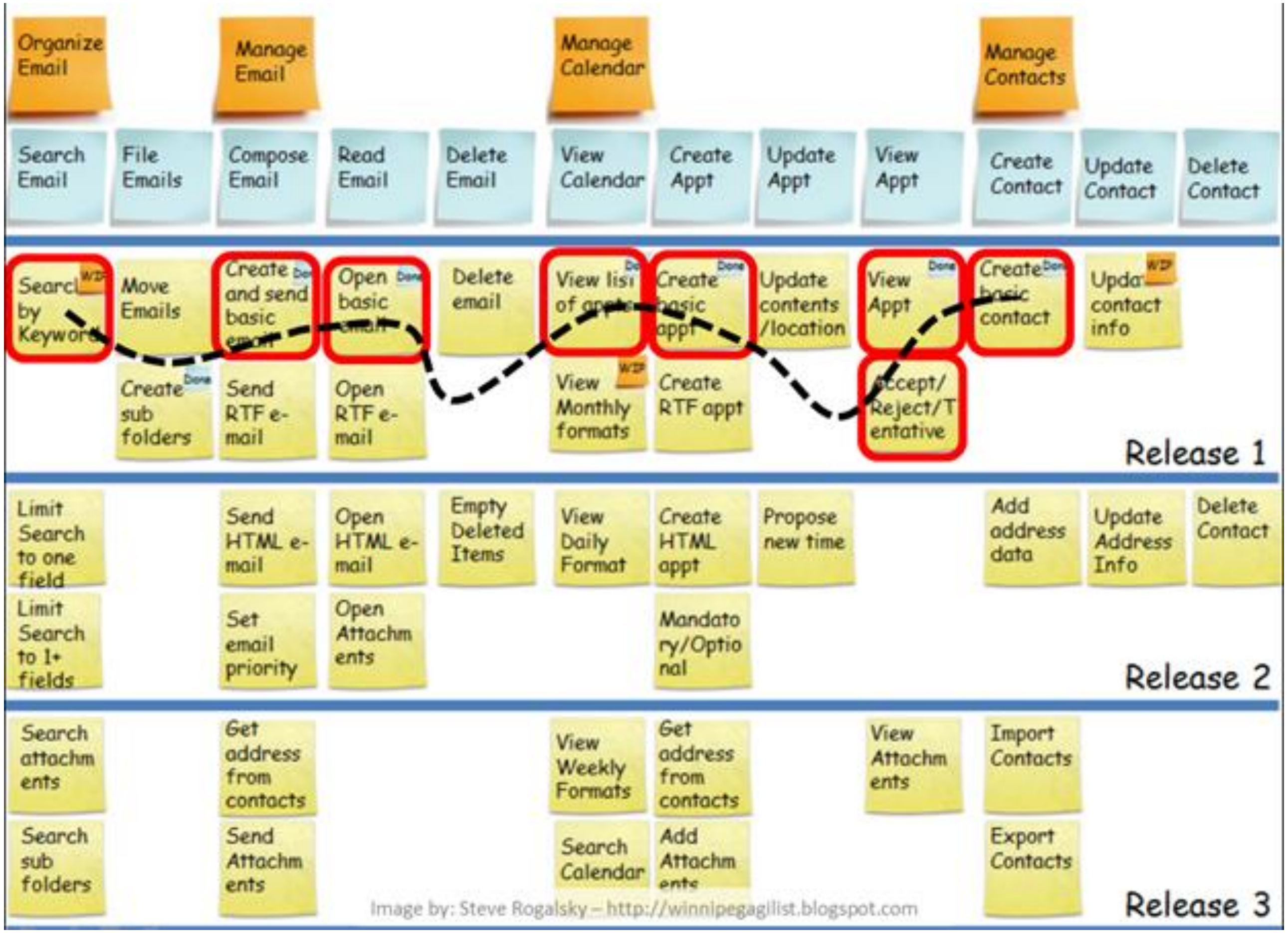
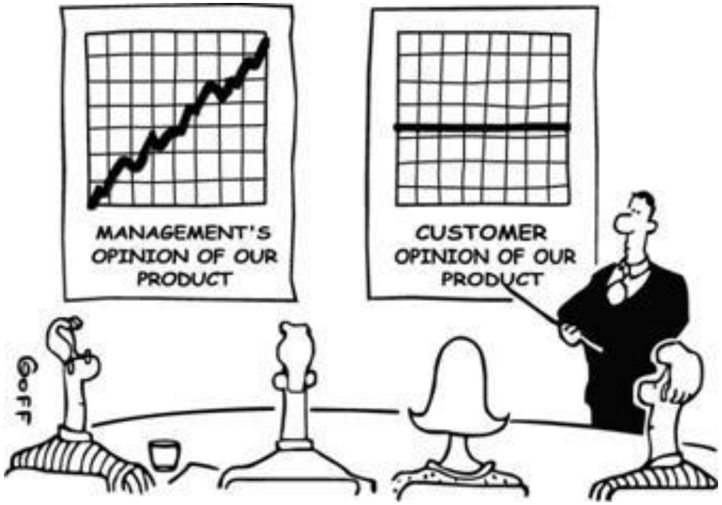


Image by: Steve Rogalsky – http://winnipegagilist.blogspot.com

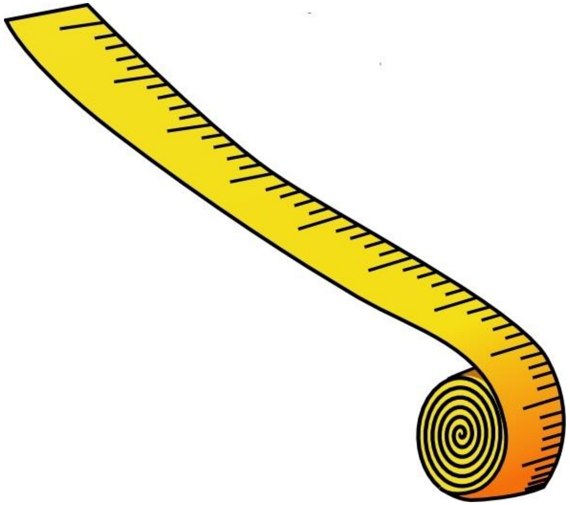
Source: http://winnipegagilist.blogspot.com/2014/11/story-maps-testing-tool-after-all.html

Estimation, Planning & Forecasting

(without System Gaming ☺)


MANAGEMENT'S OPINION OF OUR PRODUCT
CUSTOMER OPINION OF OUR PRODUCT
GOFF

Source: google images



Work Volume vs. Work Complexity

- Both are taken into account when estimating in Story Points:
  - Volume – it defines size of work
  - Complexity (as in Potential “Complication”) - it may influence Team’s ability to complete the same volume of work

Types of Planning

- **Velocity Driven Planning (@ Release level)**
  - Historical Velocity is used to determine Team's Capacity
    - "Yesterday's Weather"
    - Looking at Historical Average
    - **Can be done by using Story Points**

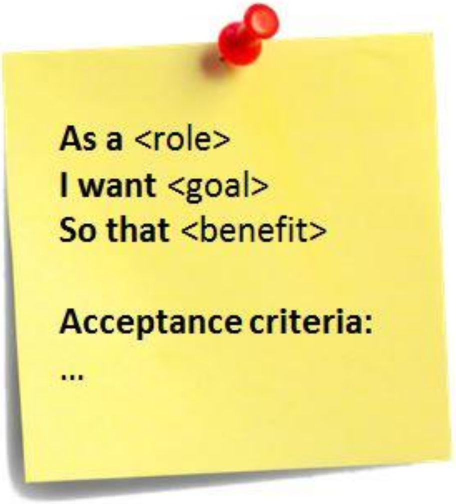
- **Commitment/Capacity Driven Planning (@ Sprint level)**
  - Determining Capacity in hours for individuals and their Skillset
  - Taking one Story at a time from top of Backlog and asking yourself: "**can we do to it?"**
  - Overly ambitions commitment leads to dropping hours mid-sprint
  - The goal is to understand how many hours of work Team can commit to, in one Sprint
  - **Can be done by using Days/Hours**

Based on Agile Estimating and Planning, by Mike Cohn

The Main Goal of Estimation

3C's

Card **Conversation** **Confirmation**



As a <role>
I want <goal>
So that <benefit>

Acceptance criteria:
...

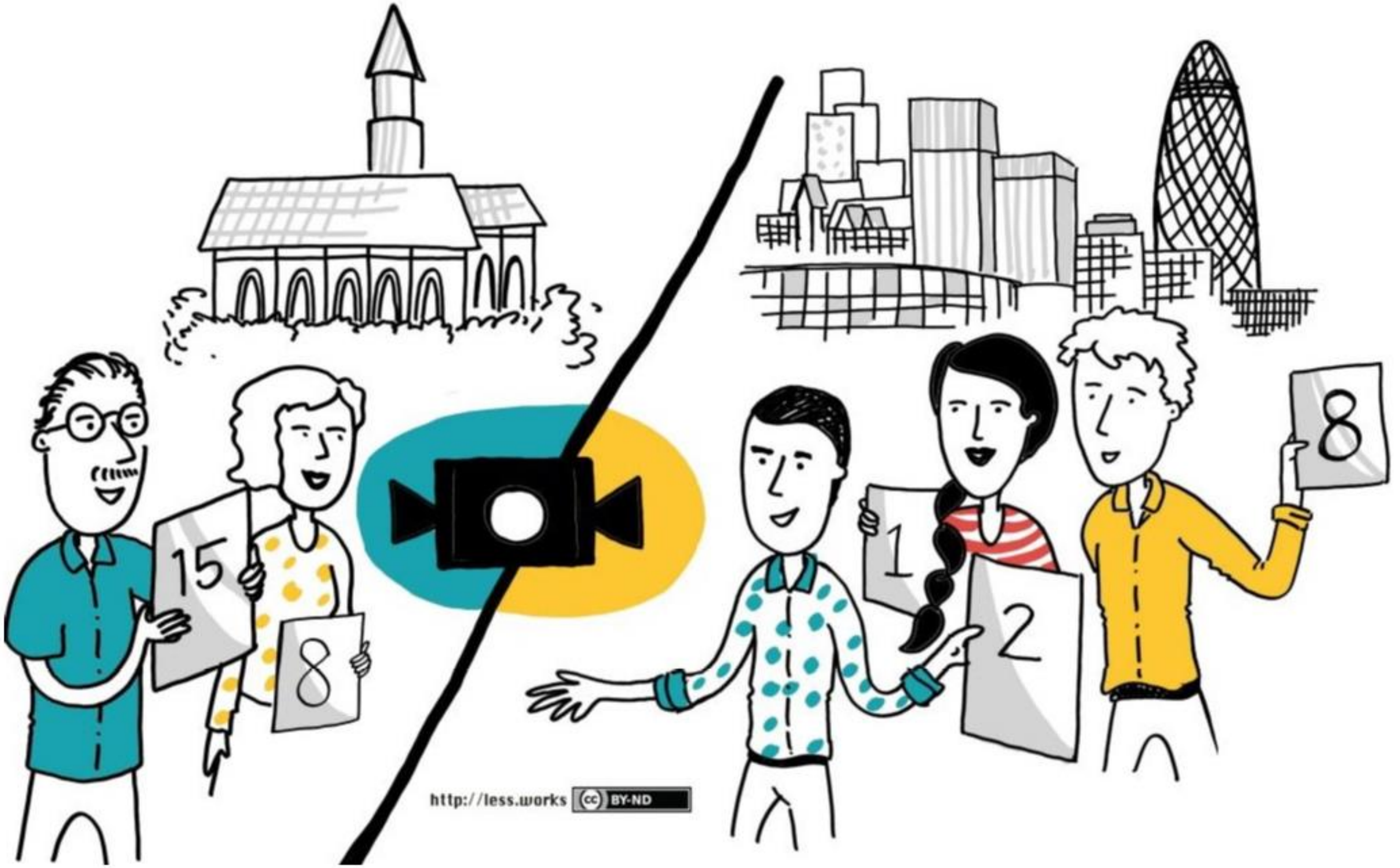
SO THIS COVERS ALL THE WORK RIGHT?
THIS IS REGULATORY, PAOLO. IT'S NEVER FINISHED OR CLEAR.

NOT to produce a number/metric

Free Byproduct of Estimation is…

NOT the other way around. Please!!!

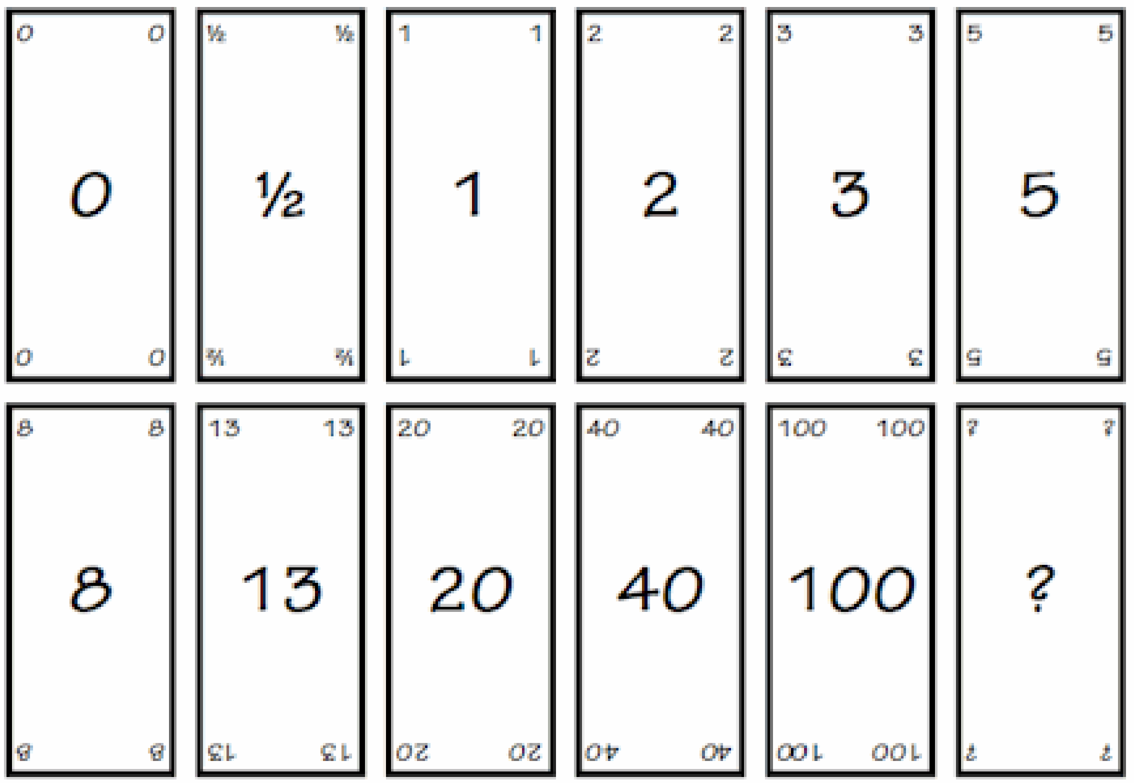
Estimation Dynamics



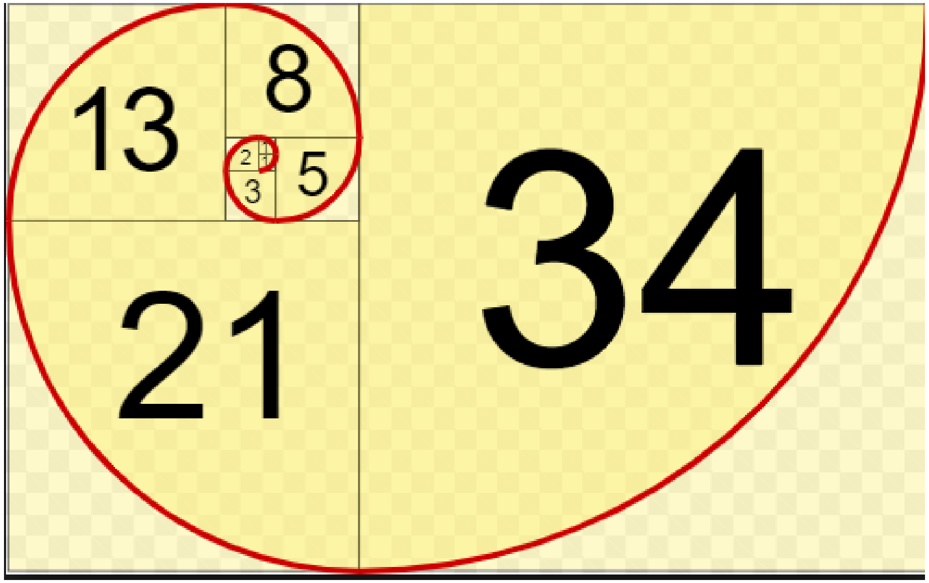
Estimation Techniques

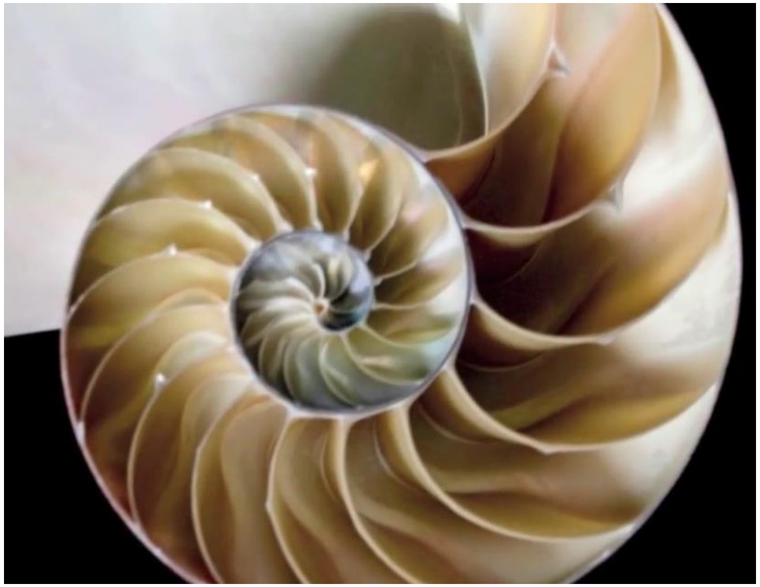
Planning/Agile Poker

- Objective, unbiased; removes anchoring
- Encourages discussions
- Uses various scales:
  - **Fibonacci Sequence**
  - Arithmetic Sequence
  - T-shirt sizes
  - Dog breeds

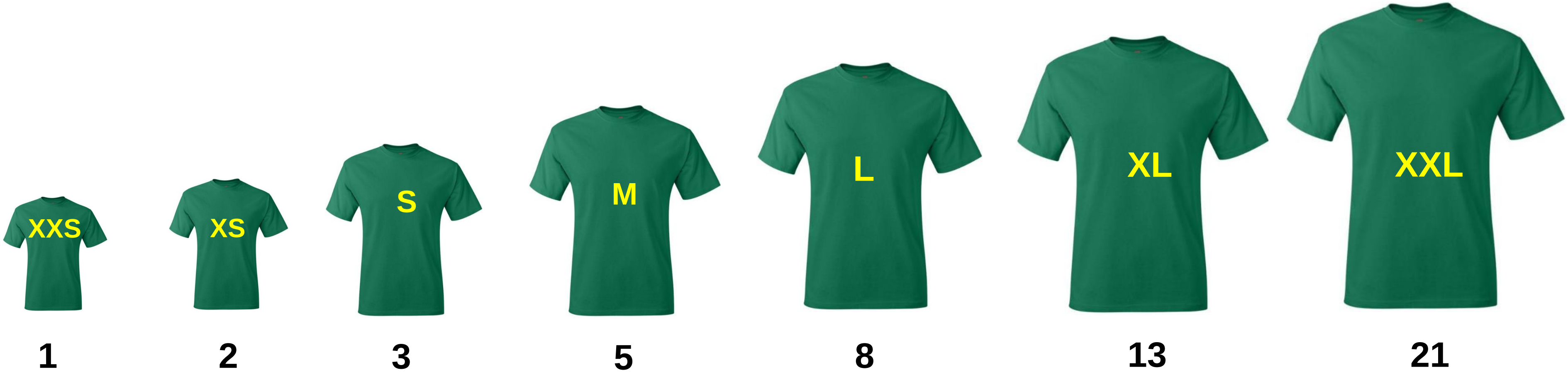


Estimation Techniques





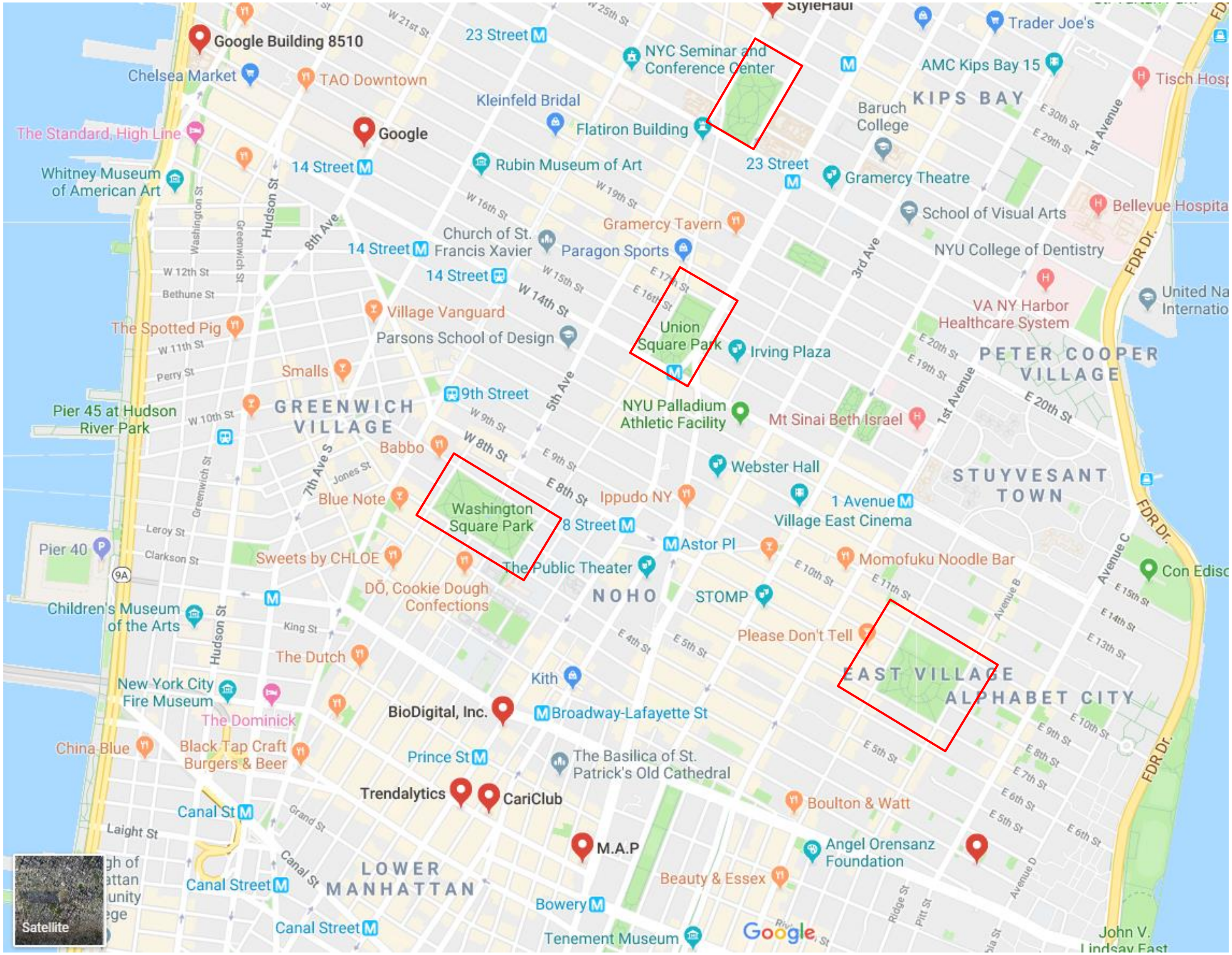
Estimation Techniques



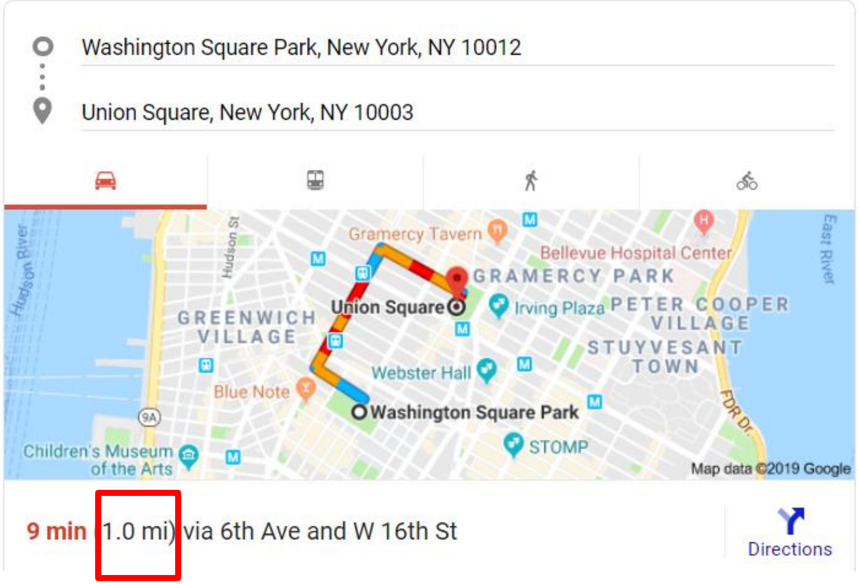
Question: Why in Planning Poker, teams use “20”, not “21”?

Accuracy != Precision

Estimation Techniques



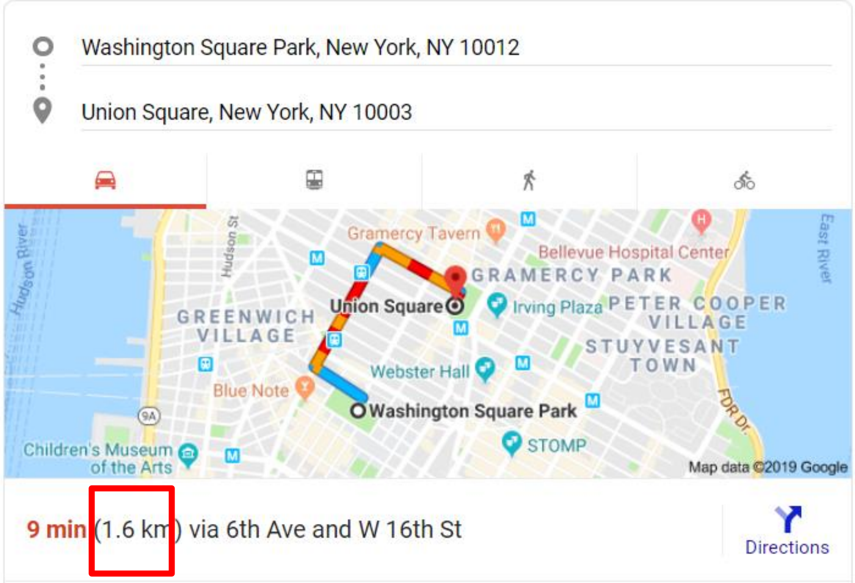
Simple “Tooling”



Washington Square Park, New York, NY 10012

Union Square, New York, NY 10003

9 min (1.0 mi) via 6th Ave and W 16th St



Washington Square Park, New York, NY 10012

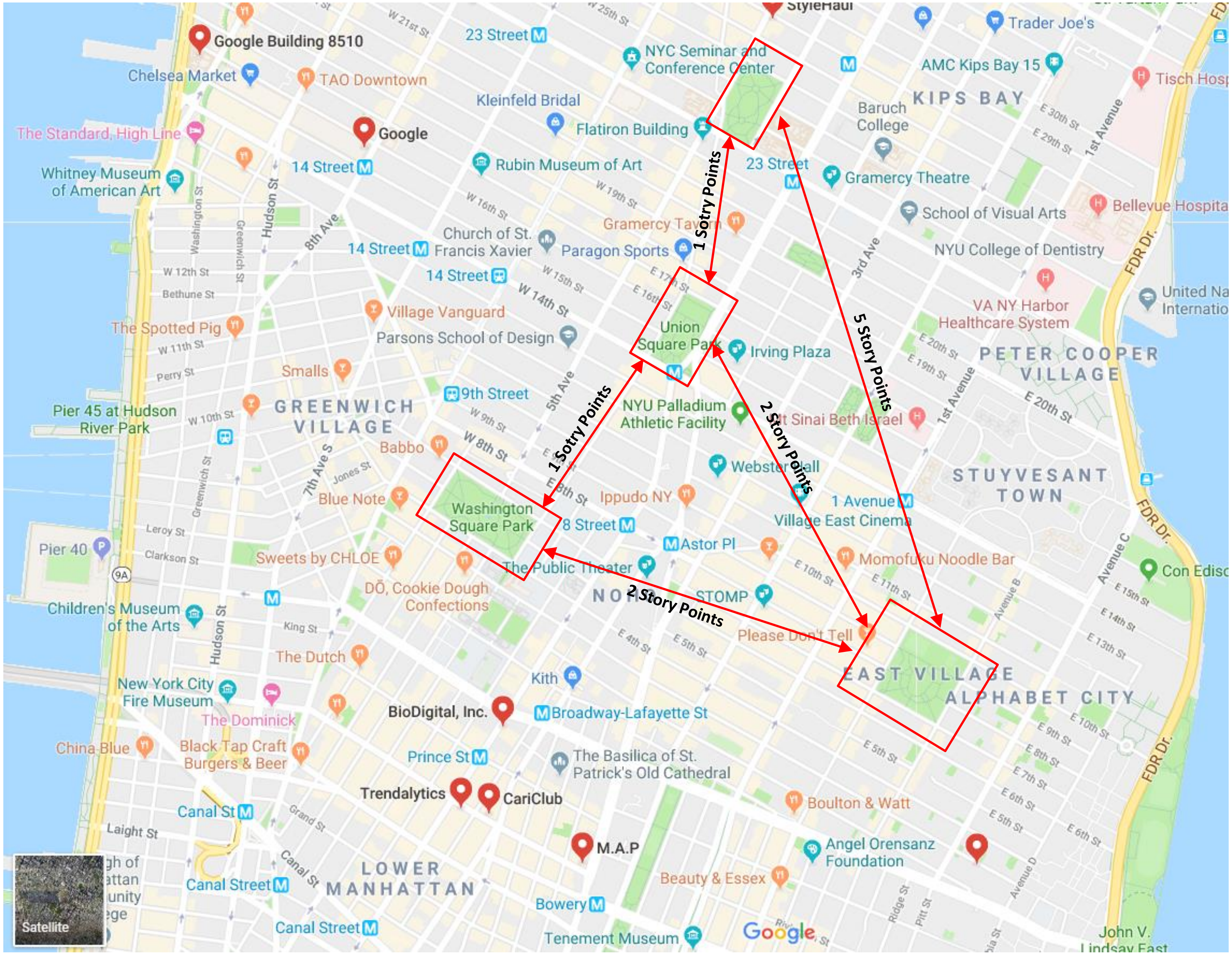
Union Square, New York, NY 10003

9 min (1.6 km) via 6th Ave and W 16th St

“Normalizing” Miles and Kilometers into…8 City Blocks???

…or arbitrarily calling it a 1 story point…

Estimation Techniques



...all relative...

Conditions Required for Relative Estimation

- **Mandatorily**, theoretical/conceptual understanding of Relative Estimation

- **At minimum**, two people+ that need to '*normalize*' their understanding about volume and complexity of work. (A single-performer can simply estimate in days/hours/minutes. There is no need for 'normalize' with anyone)

- **Ideally**, cross-functional, T-shaped workers, capable of performing multiple types of work on the same team. **At minimum**, people - understanding each other's work

- **Mandatorily**, people that officially belong to the same team and share ownership/responsibilities with others

- **Unequivocally**, people that trust each other

Anchoring

- **Inability** to make one’s own independent, unbiased, uninfluenced, objective decisions

- Caused by:
  - Lack of strong knowledge
  - Pressure: peer, customer, management
  - Status Quo

- Can be avoided by:
  - “Military Democracy” (“privates speak before generals”)
  - Using discrete estimation techniques (e.g. Planning Poker)

Estimation, Planning & Forecasting: Example

Imagine a team….☺

Individual Capacity Management – In one Sprint

Team Member	**Primary Skills**	**Secondary Skills**	**Sprint Capacity**
John	Java, Flex	SQL, BA, Ruby	60 hours
Jeff	Ruby on Rails, Flex	SQL, VBScript, Java	55 hours
Jill	Java, .Net, Pascal	C++, C#, Java, Flex	65 hours
Jim	Flex, Java, .Net	DHTML, Cold Fusion	55 hours
Julie	Cobol, Fortran	Pascal, SQL	65 hours
Total			**295 hours**

Estimation, Planning & Forecasting: Example

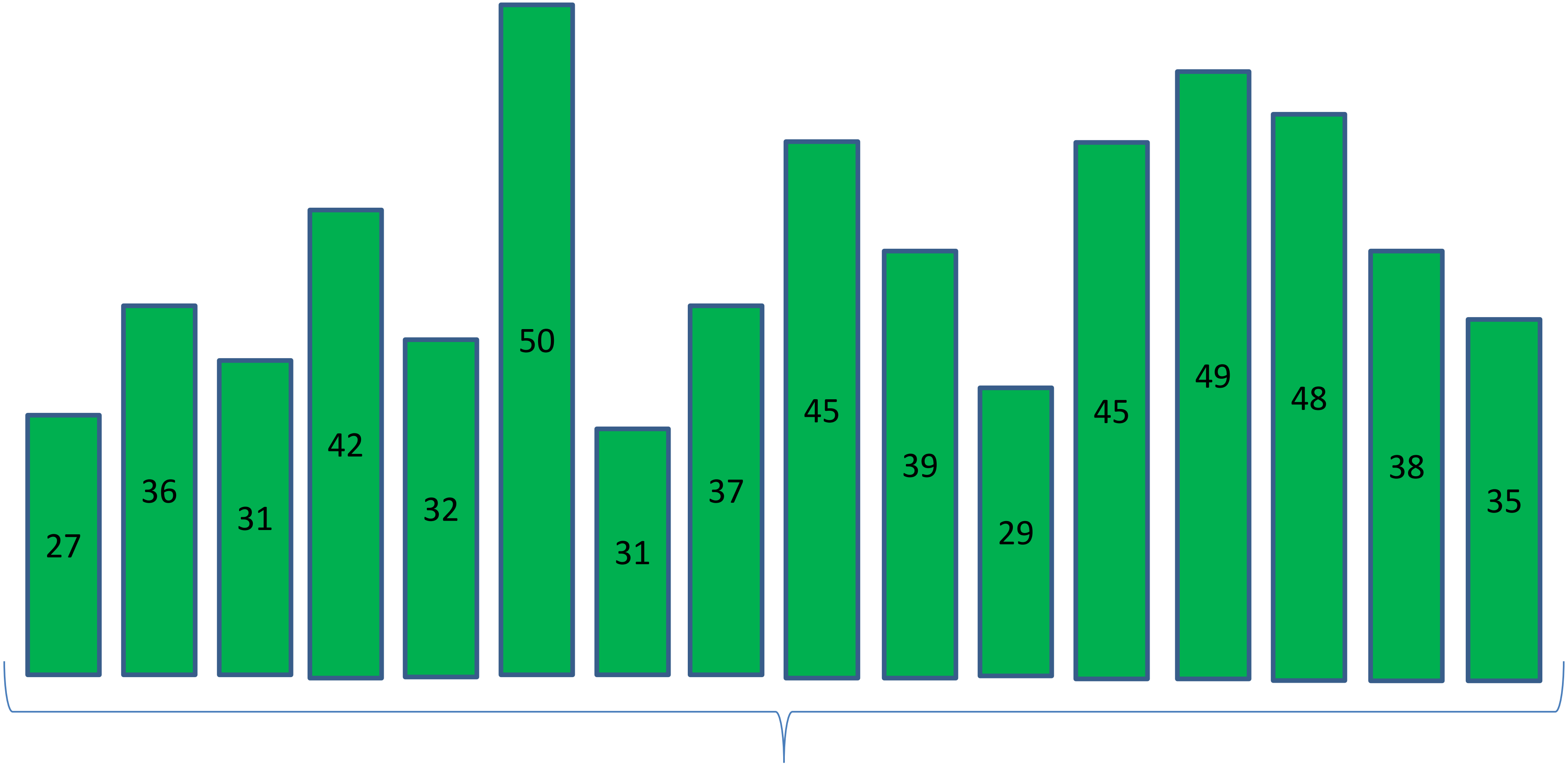
Imagine a team….☺

Skillset Capacity Management in one Sprint

Skillset	**Capacity**
Java	150 hours
Flex	50 hours
Ruby	30 hours
.Net	30 hours
Cobol, Fortran, Pascal	35 hours
Total	**295 hours**

Estimation, Planning & Forecasting: Example

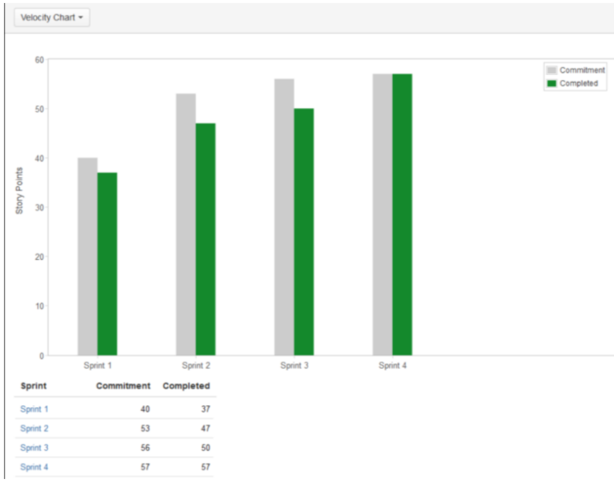
Historical Velocity Analysis



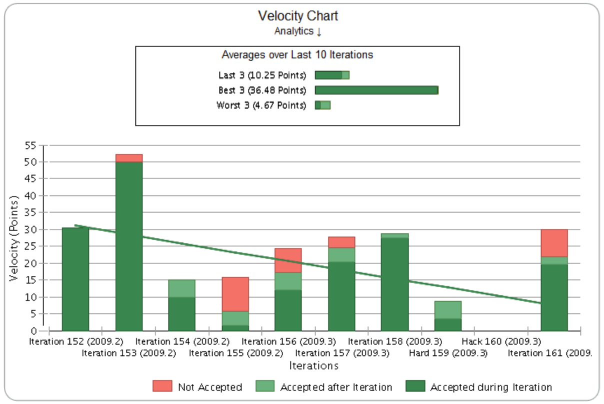
16 sprints

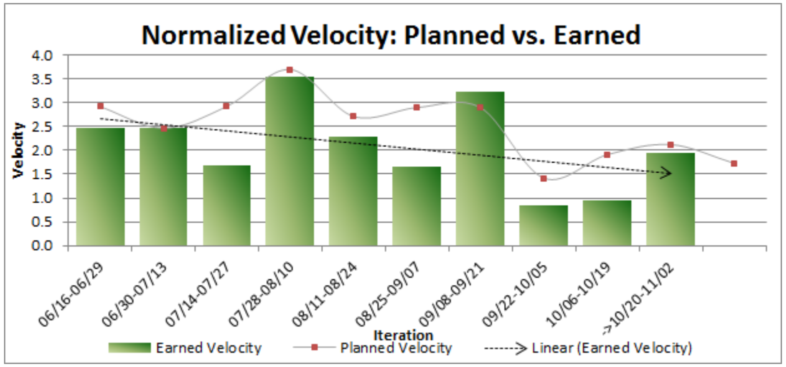
Estimation, Planning & Forecasting: Example

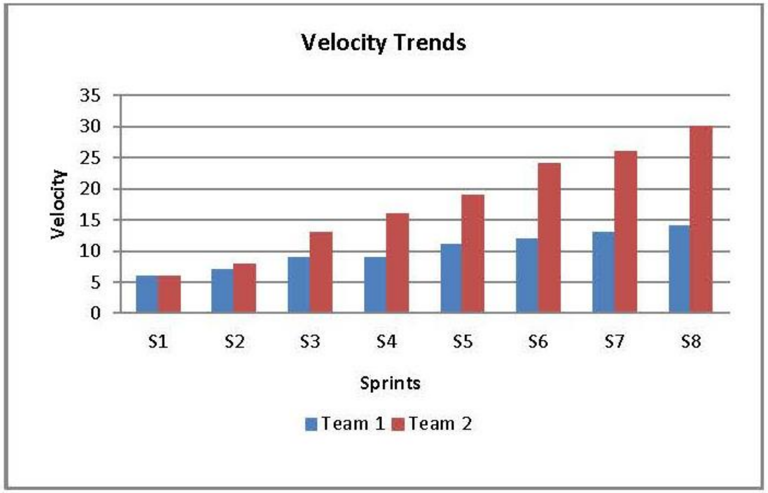
Historical Velocity (Real Life Examples)



Sprint	Commitment	Completed
Sprint 1	40	37
Sprint 2	53	47
Sprint 3	56	50
Sprint 4	57	57



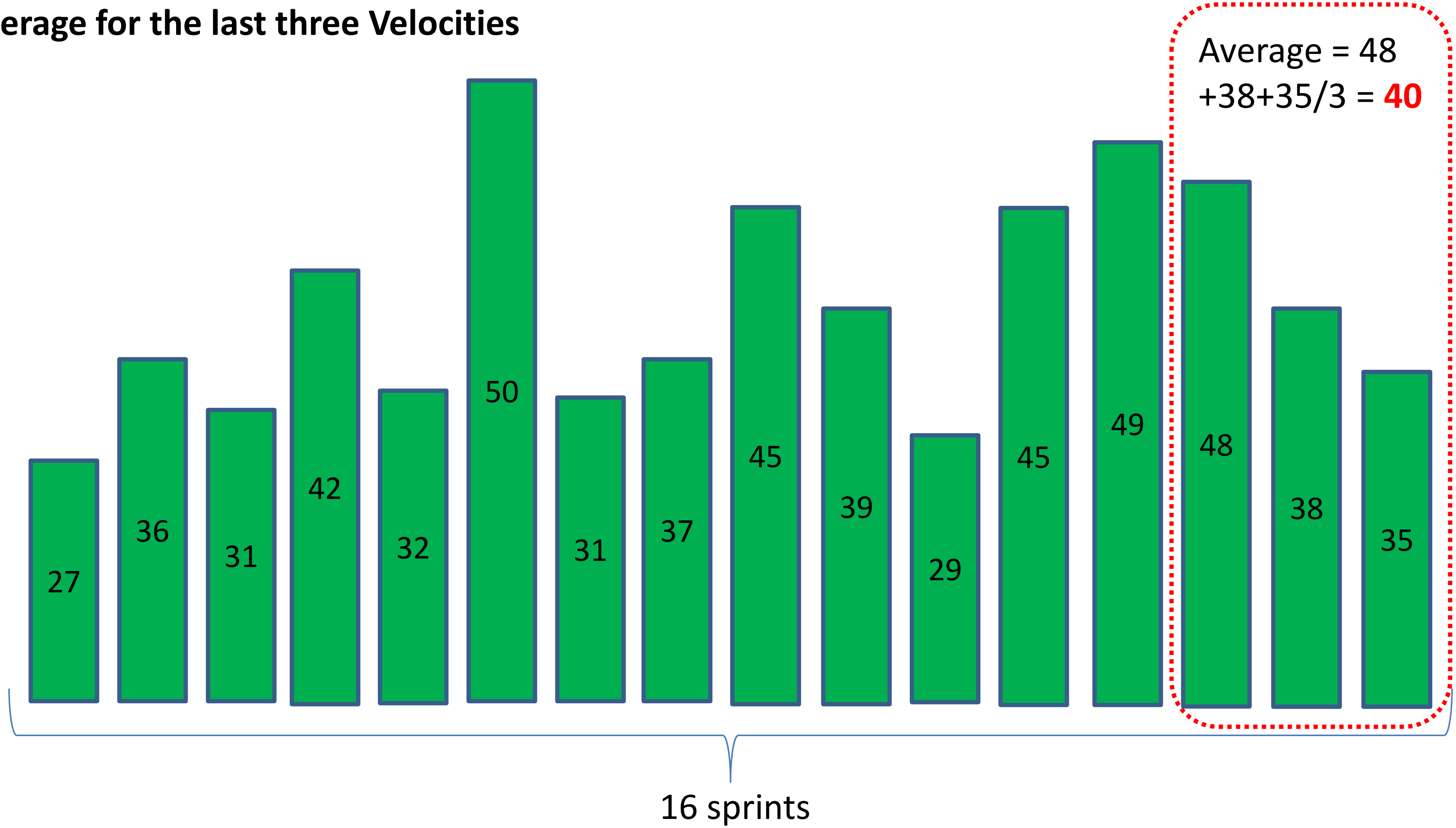




Estimation, Planning & Forecasting: Example

Work Estimation & Planning

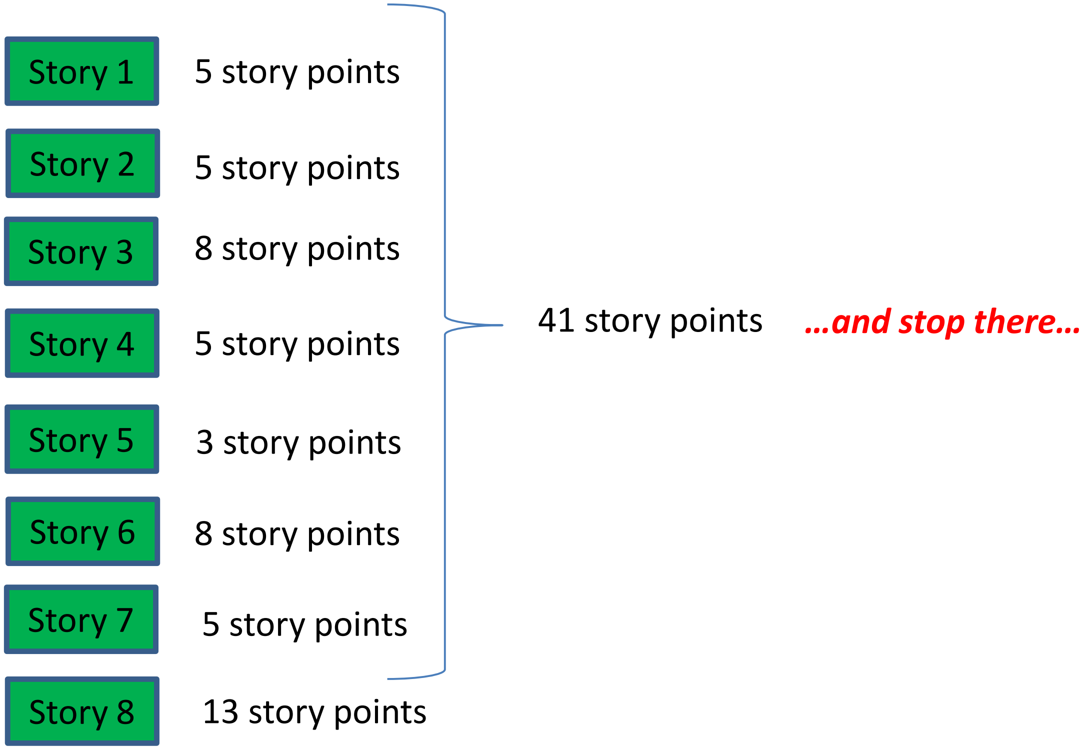
Average for the last three Velocities



Average = 48 +38+35/3 = **40**

Estimation, Planning & Forecasting: Example

Identifying Stories-Candidates

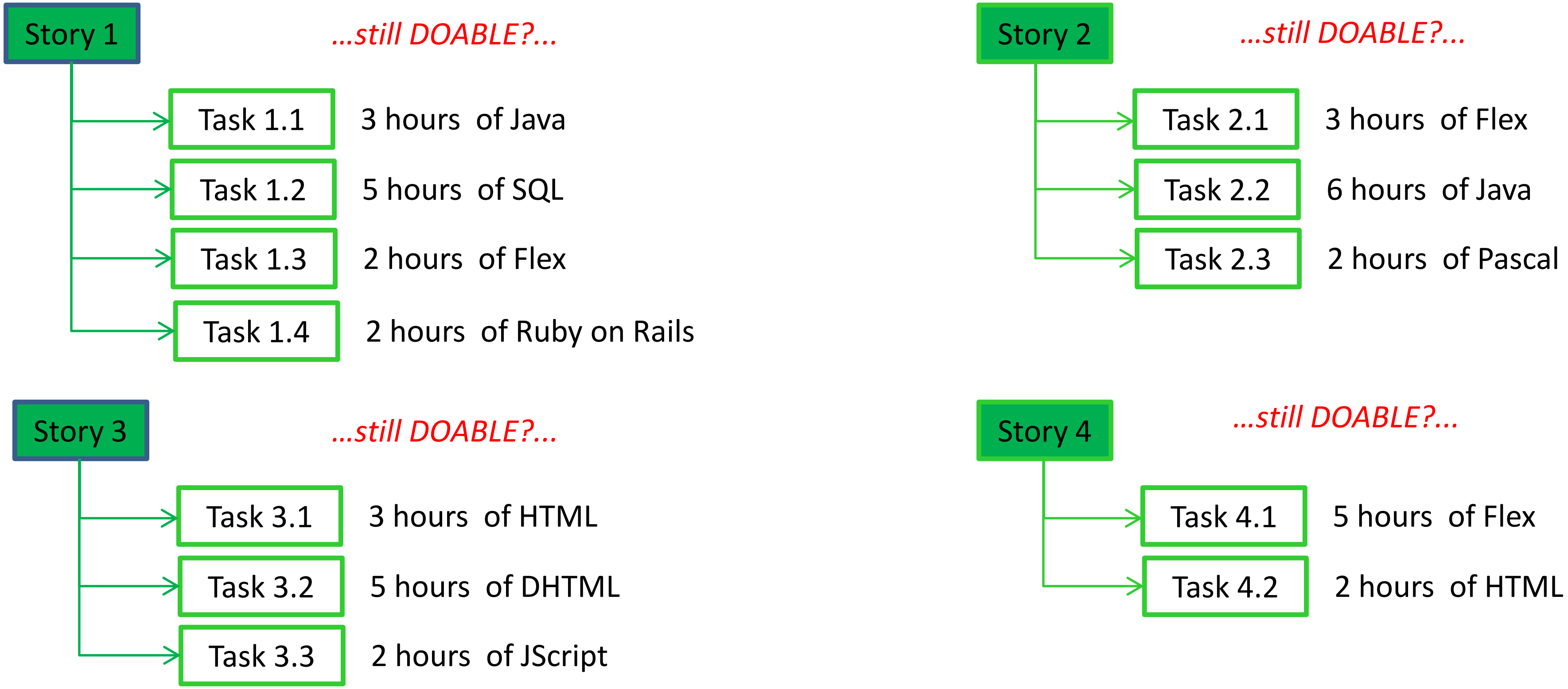


Story 1	5 story points
Story 2	5 story points
Story 3	8 story points
Story 4	5 story points
Story 5	3 story points
Story 6	8 story points
Story 7	5 story points
Story 8	13 story points

41 story points ***…and stop there…***

Estimation, Planning & Forecasting: Example

Capacity-Driven Sprint Planning: Breaking Stories into Tasks



Story 1 …still DOABLE?...

- Task 1.1 3 hours of Java
- Task 1.2 5 hours of SQL
- Task 1.3 2 hours of Flex
- Task 1.4 2 hours of Ruby on Rails

Story 2 …still DOABLE?...

- Task 2.1 3 hours of Flex
- Task 2.2 6 hours of Java
- Task 2.3 2 hours of Pascal

Story 3 …still DOABLE?...

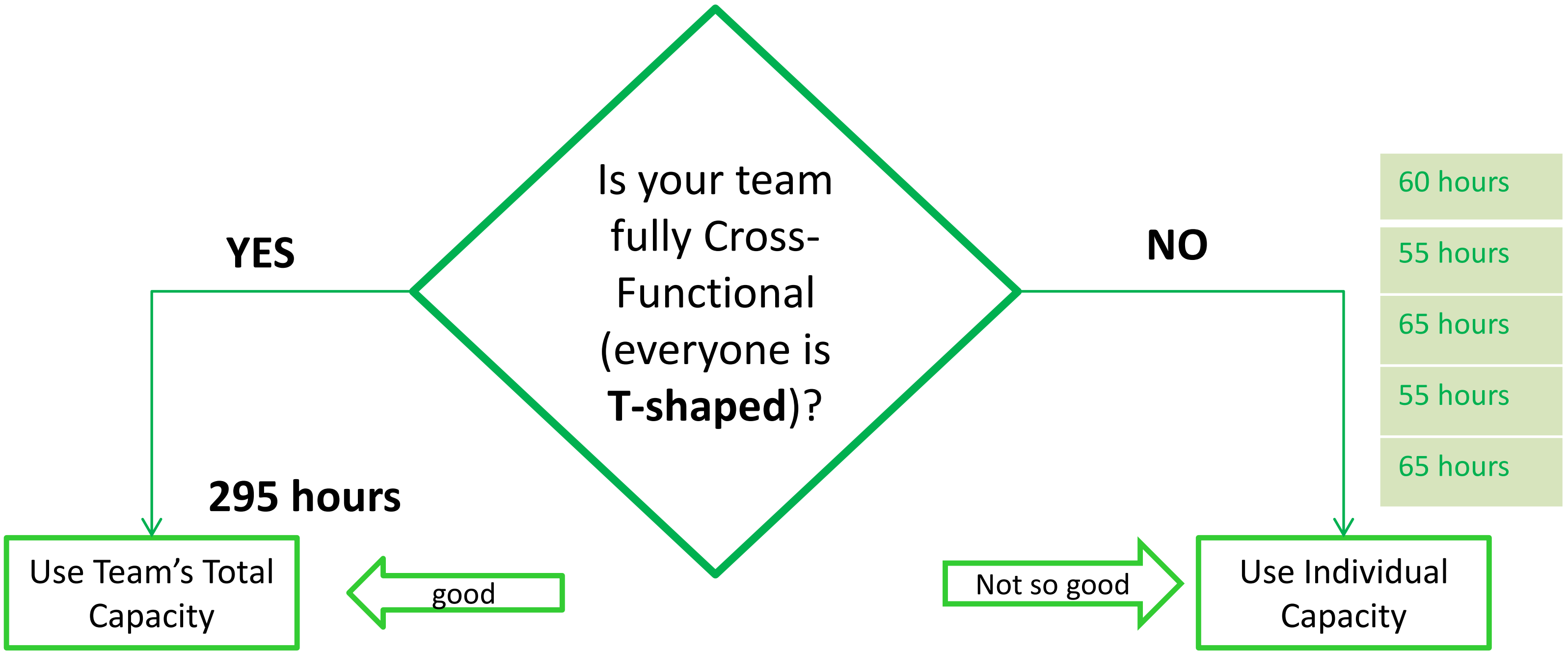
- Task 3.1 3 hours of HTML
- Task 3.2 5 hours of DHTML
- Task 3.3 2 hours of JScript

Story 4 …still DOABLE?...

- Task 4.1 5 hours of Flex
- Task 4.2 2 hours of HTML

Estimation, Planning & Forecasting: Example

Capacity-Driven Planning: two options



Is your team fully Cross-Functional (everyone is **T-shaped**)?

YES

295 hours

Use Team's Total Capacity

good

NO

60 hours
55 hours
65 hours
55 hours
65 hours

Use Individual Capacity

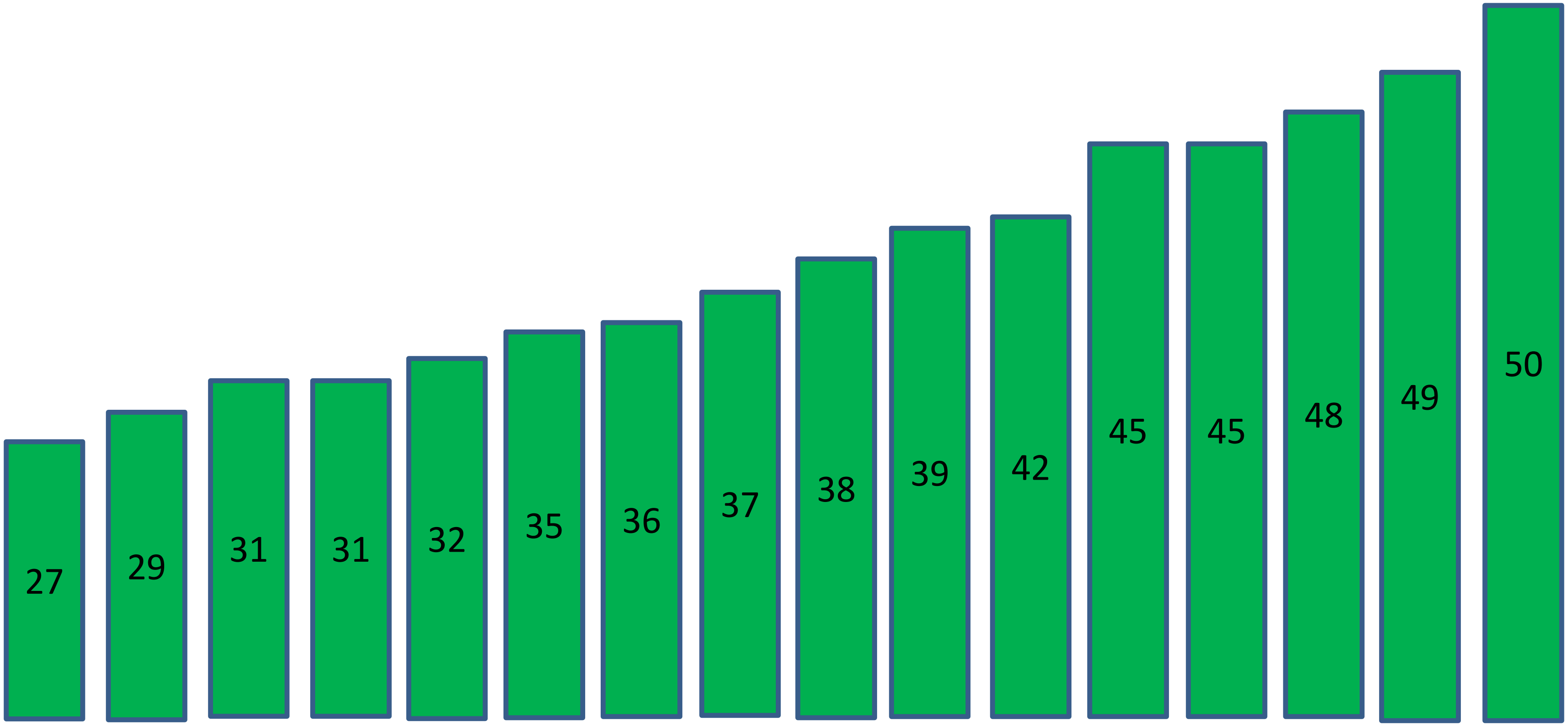
Not so good

…swarming and collective ownership is expected…

…unfortunately, need to consider who is doing what…

Estimation, Planning & Forecasting: Example

Historical Velocity Analysis



Based on Agile Estimating and Planning, by Mike Cohn

Estimation, Planning & Forecasting: Example

Discarding Outliers when Data Pool is Large

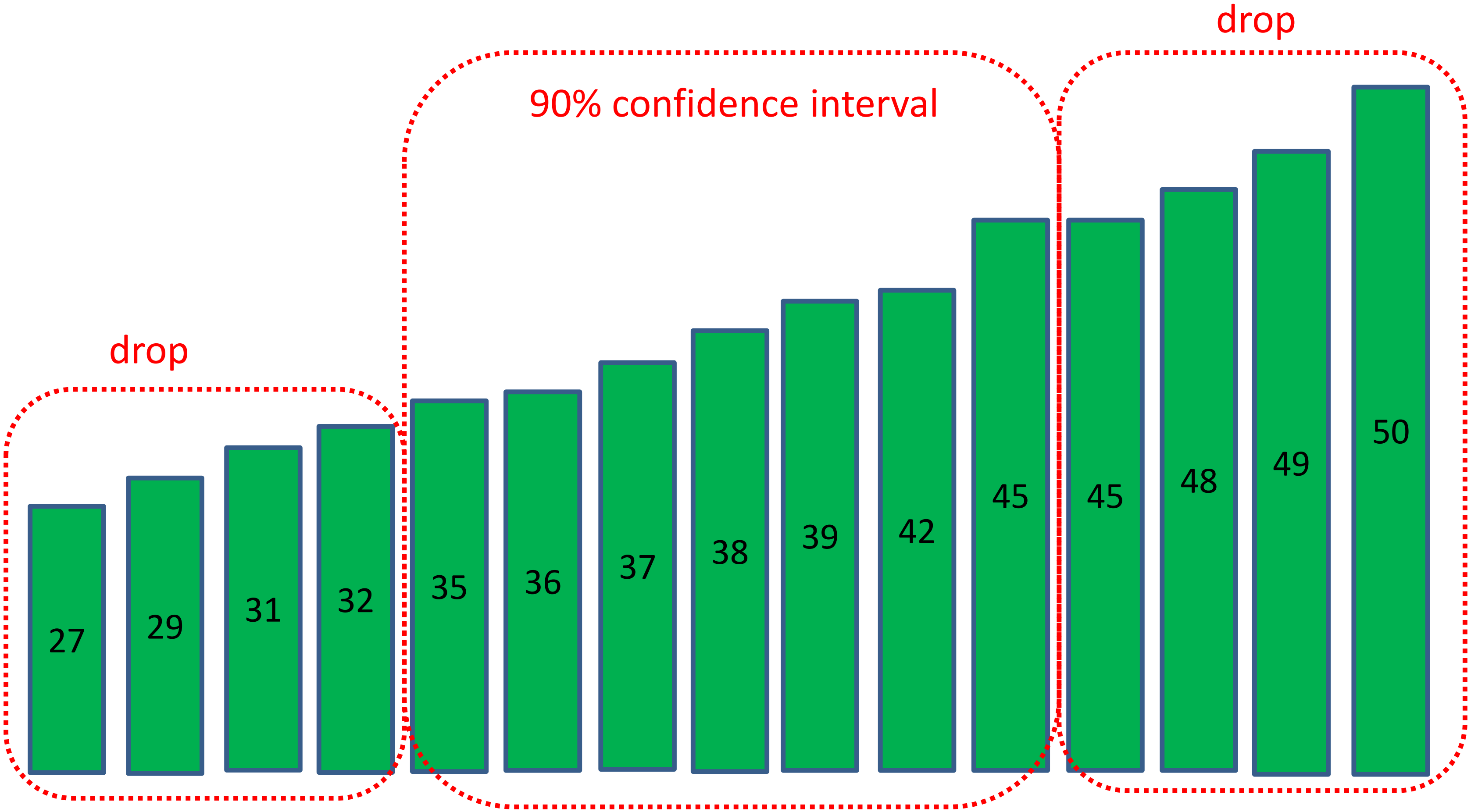
Number of Data Sets	Number Outliers to Drop on Each End
0-7	0
8-10	1
11-12	2
13-15	3
16-17	4
18-20	5
21-22	6
23-25	7
26+	8

Based on Confidence Interval around the Median using Binomial Distribution

Based on Agile Estimating and Planning, by Mike Cohn

Estimation, Planning & Forecasting: Example

Historical Velocity Analysis – getting rid of Outliers

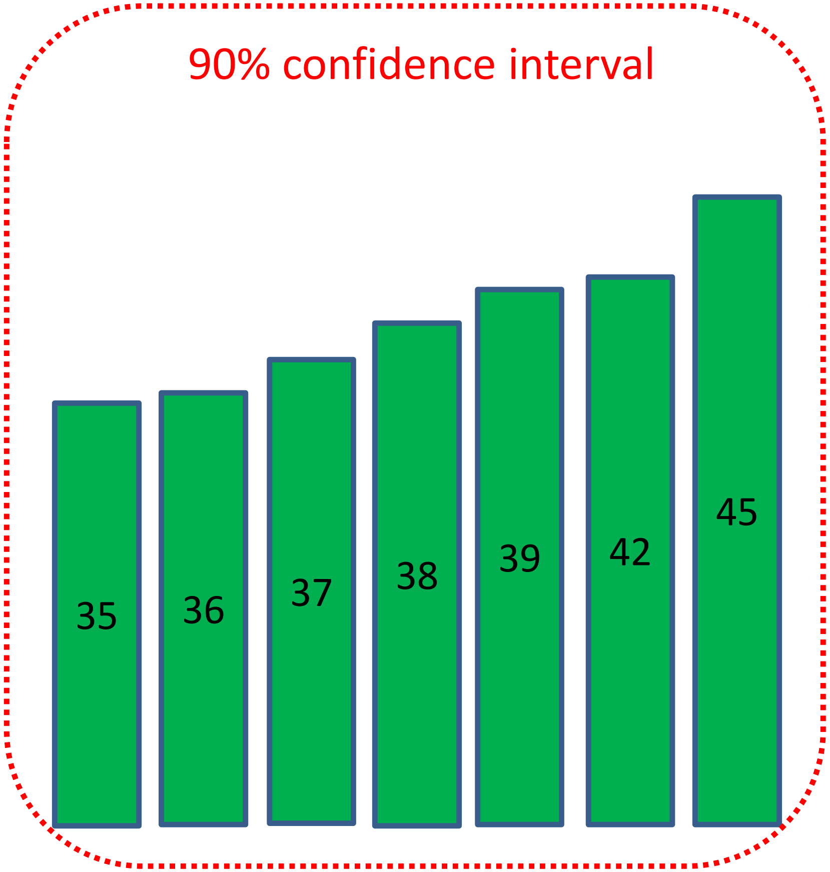


Based on Agile Estimating and Planning, by Mike Cohn

Estimation, Planning & Forecasting: Example

Historical Velocity Analysis – getting rid of Outliers

“Official” Low Velocity = 35



“Official” High Velocity = 45

Based on Agile Estimating and Planning, by Mike Cohn

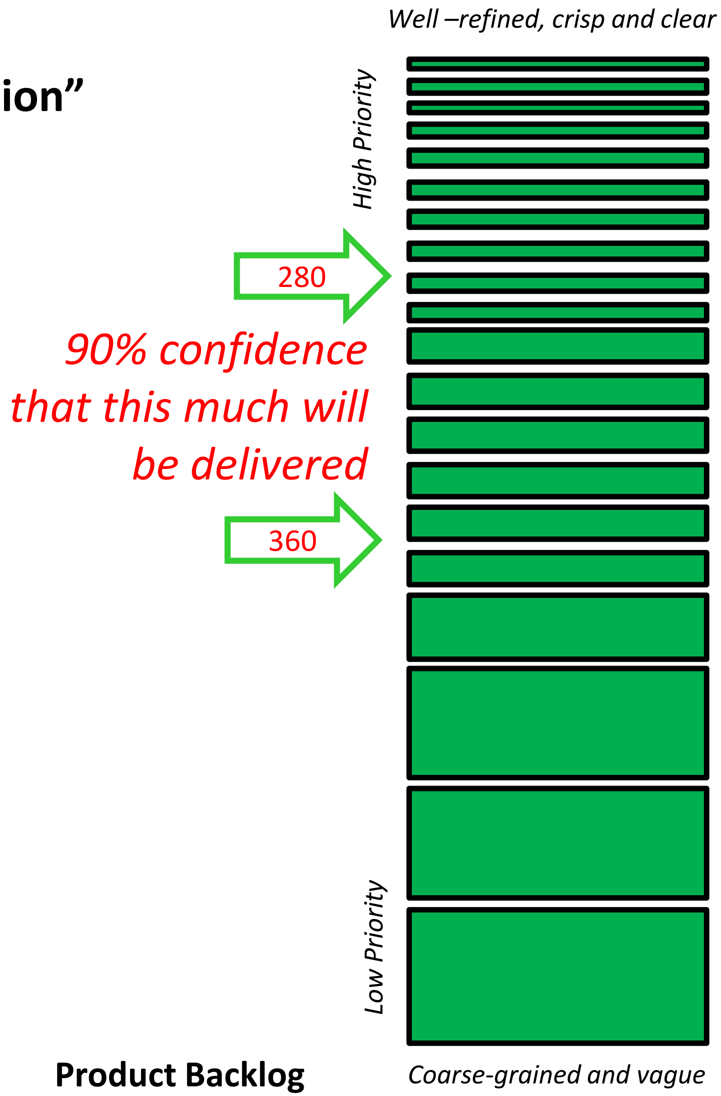
Estimation, Planning & Forecasting: Example

Historical Velocity Analysis – forecasting Backlog "penetration"

If you have a release in 4 months and your sprint is 2 weeks in duration, it means that you have 8 sprints in total

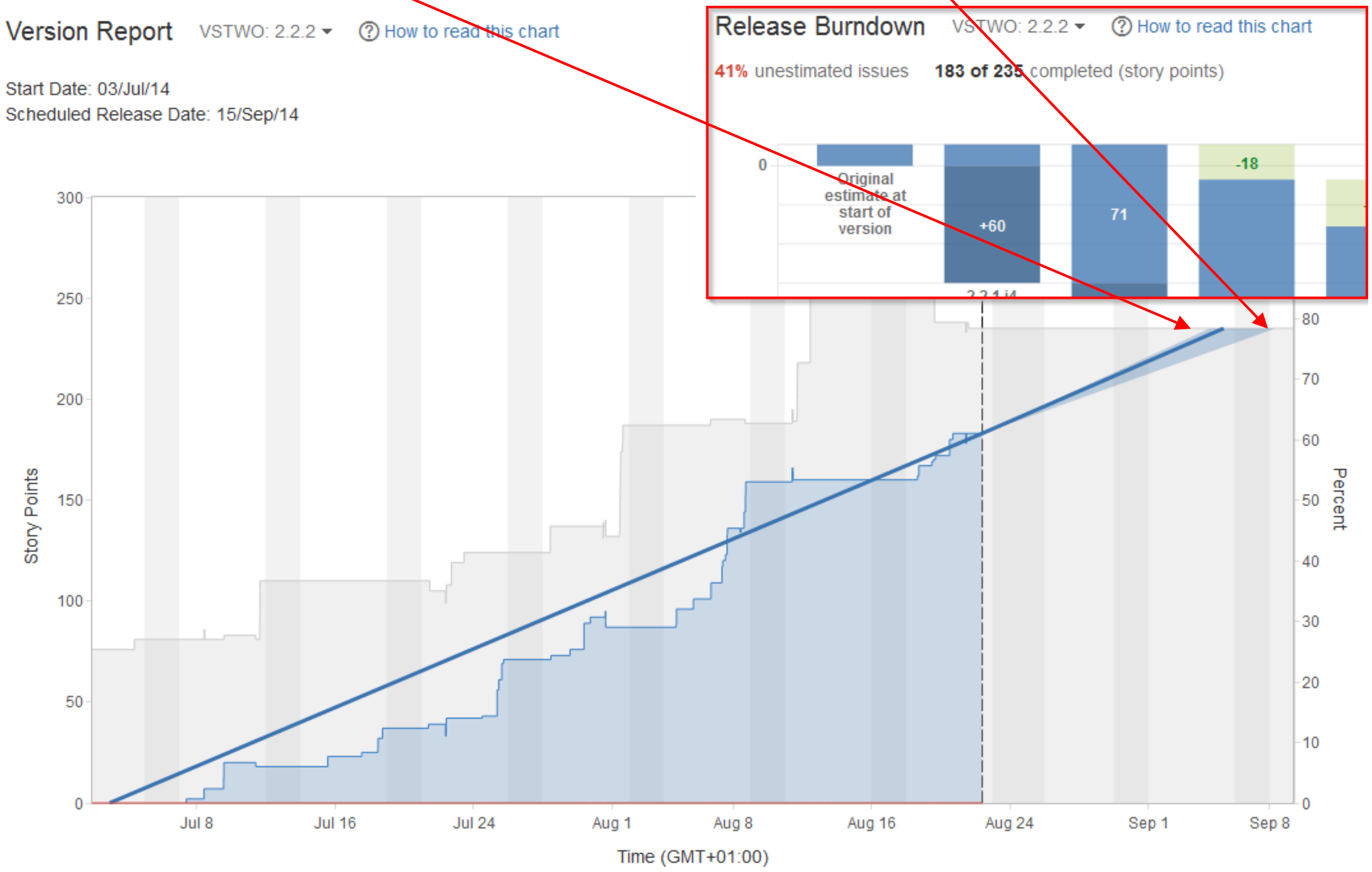
8 Sprints x "Official" Low Velocity of 35 = **280** story points (pessimistic projection)

8 Sprints x "Official" High Velocity of 45 = **360** story points (optimistic projection)


Well –refined, crisp and clear
High Priority
280
90% confidence that this much will be delivered
360
Low Priority
Product Backlog
Coarse-grained and vague

Estimation, Planning & Forecasting: Example

Cone of Uncertainly: Optimistic vs. Pessimistic forecasting



Version Report VSTWO: 2.2.2 ▾ ⓘ How to read this chart

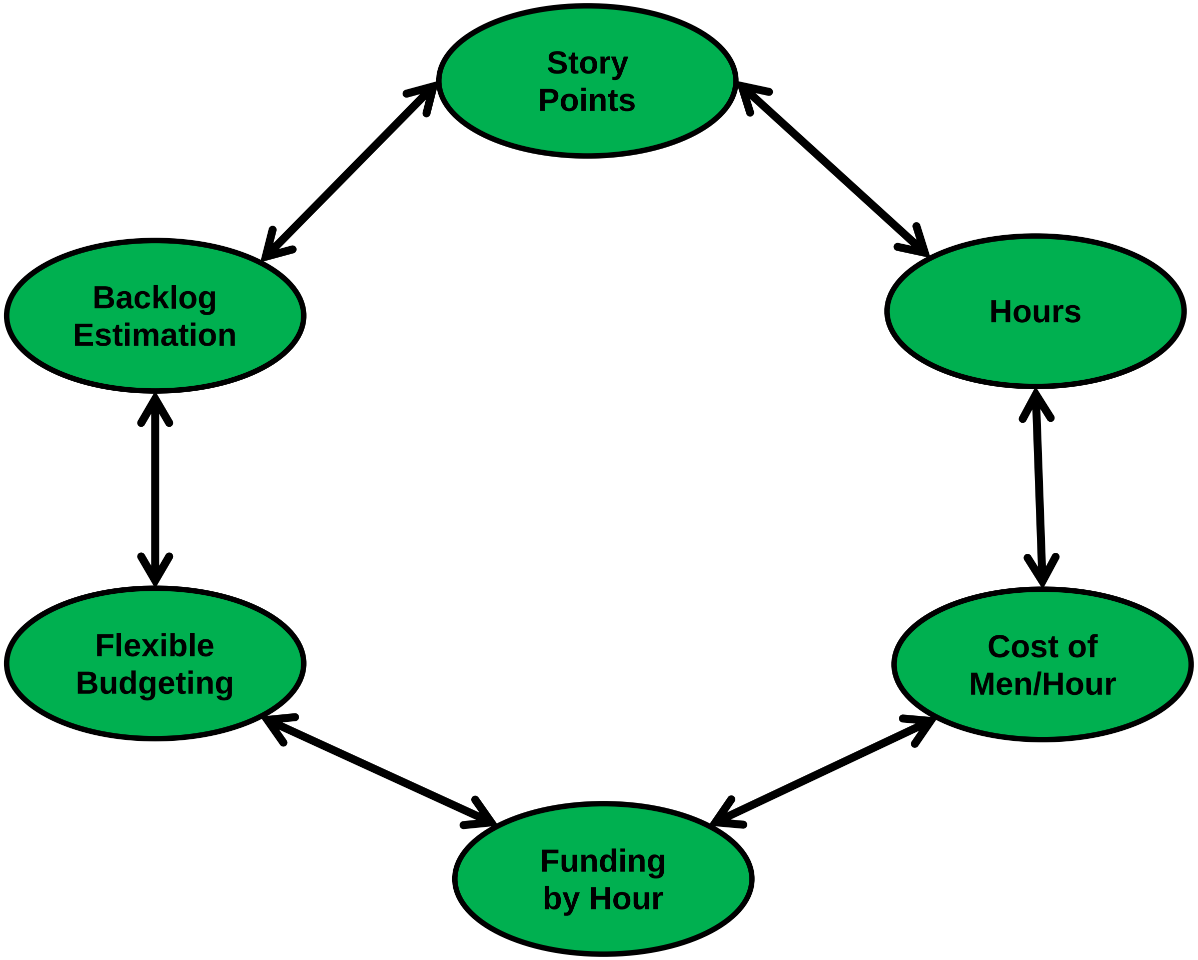
Start Date: 03/Jul/14
Scheduled Release Date: 15/Sep/14

Release Burndown VSTWO: 2.2.2 ▾ ⓘ How to read this chart

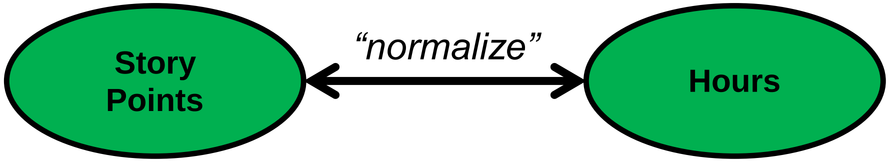
41% unestimated issues **183 of 235** completed (story points)

Estimation, Planning & Forecasting Example

(Everything is related)



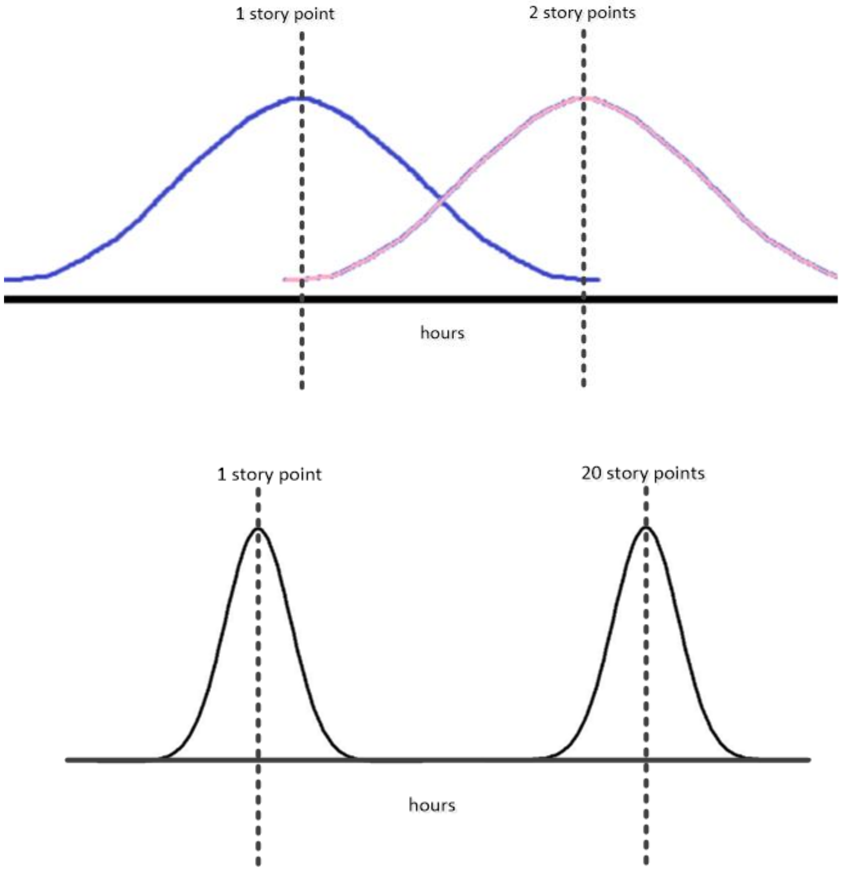
Estimation, Planning & Forecasting: Example



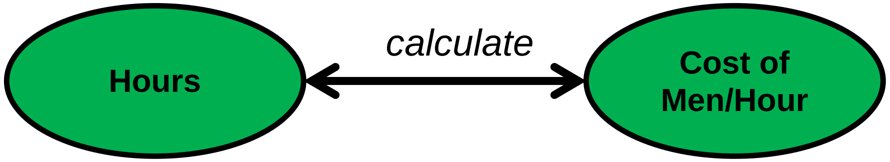
Goal: to derive Average Time Spent per Story Point

- □ Team Stability
- □ Team Focus
- □ Team Composition (Skill set)
- □ Team's Estimation Techniques
- □ Team's Estimation (not by Managers)

Probability of Overlapping



Estimation, Planning & Forecasting: Example



Advice

Goal: to derive Average Cost of Story Point

Worker's Pay Rate **x** Team Size

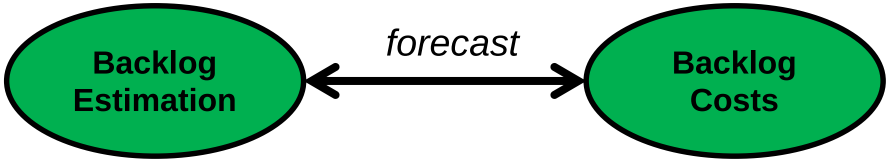
=

Team's Average Pay Rate

Estimation, Planning & Forecasting: Example

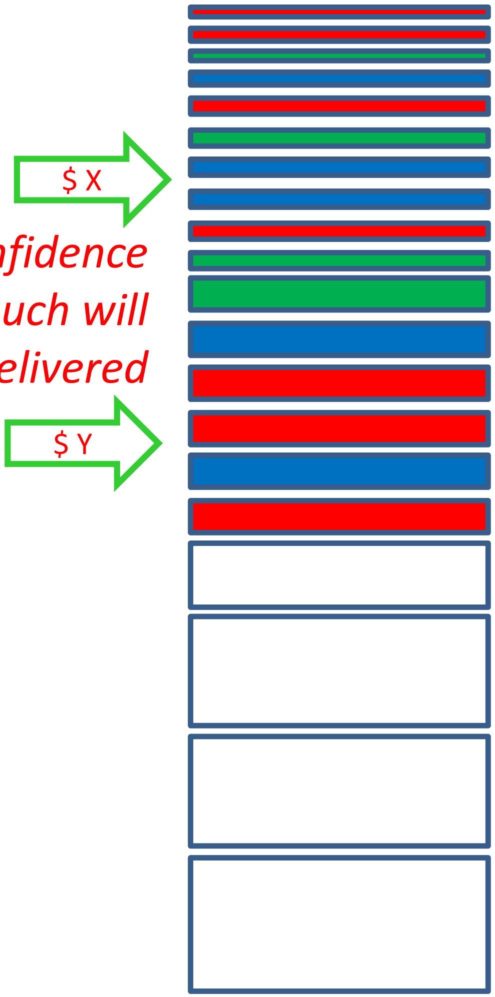
[If Timeline is fixed]

Advice



Goal: to have Flexible Spending

- ❑ Optimistic vs. Pessimistic **Scope**
- ❑ Forecasting Spending, based on Cost of Story Point
- ❑ Asking for more $, if needed
- ❑ Giving back $, if no longer needed (trim tail)

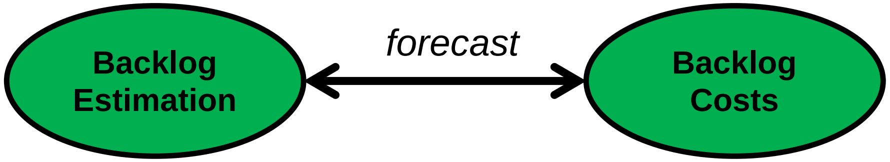


90% confidence that this much will be delivered

Estimation, Planning & Forecasting: Example

[If Scope is fixed]

Advice



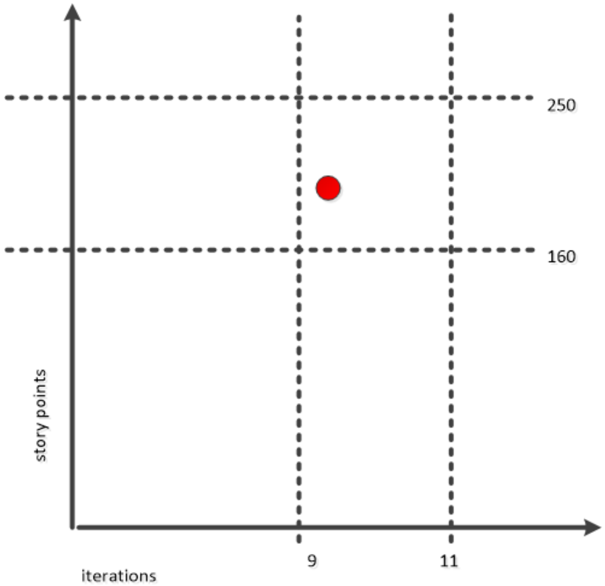
Backlog Estimation

forecast

Backlog Costs

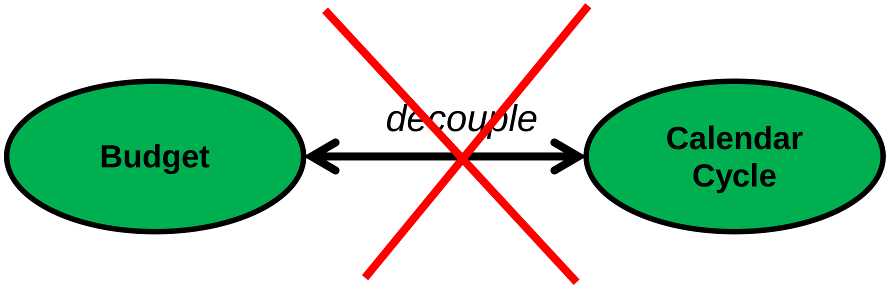
Goal: to have Flexible Spending

- ❑ Optimistic vs. Pessimistic **Completion Dates**
- ❑ Forecasted Spending, based on Cost of Story Point
    - ❑ Asking for more $, if needed
    - ❑ Giving back $, if no longer needed



Based on Agile Estimating and Planning, by Mike Cohn

Estimation, Planning & Forecasting: Example

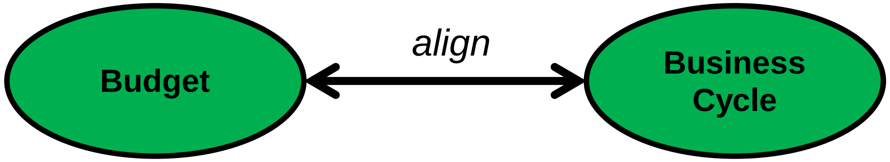


Advice

Goal: to reduce System Gaming

- ❑ Budget Floor = Budget Ceiling
  - ❑ When trying not to exceed Ceiling…end up Trying not to lower Floor ☹

Estimation, Planning & Forecasting: Example



Advice

Goal: to reduce System Gaming

- □ Opportunistic Development
- □ Investigative Spikes
- □ Research and Discovery
- □ Safety to Experiment

Estimation, Planning & Forecasting: Example

Things That Will make Agile Budgeting Very Difficult:

❑ Team instability

❑ Frequent on/off boarding

❑ Single-specialty workers

❑ Competing priorities

❑ ………….

Build a chair exercise

The Product Owner would like to furnish a room:
- Three chairs are required
- Let’s assume the detailed chair stories are written and prioritized in the backlog

Story #1 – Basic Chair
- Needs four legs, a back

Story #2 – Enhanced Chair
- Needs wheels, ability to adjust height

Story #3 - High- concept Future Chair
- Needs to recline, hover in the air

Build a chair exercise

Easy Chair

Build effort:

Test effort:

Score:

Build a chair exercise

“Enhanced” Chair

Build effort:

Test effort:

Score:

Build a chair exercise

Advanced Chair

Build effort:

Test effort:

Score:

Build a chair exercise

Estimation is relative. Both: volume of work and complexity – counts.

Jokes, Games, Videos

Room Information Radiation

- Parking Lot
- “Who Stole My Cheese” (SM, PO, Team, PM, Other)
- DoR & DoD
- CLD Canvas – 1 per team/table

Agile Jokes

Train Ride of Software Engineer & PM:

A young programmer and his project manager board a train headed through the mountains on its way to Great Canyon. They can find no place to sit except for two seats right across the aisle from a young woman and her grandmother. After a while, it is obvious that the young woman and the young programmer are interested in each other, because they are giving each other looks. Soon the train passes into a tunnel and it is pitch black. There is a sound of a kiss followed by the sound of a slap. When the train emerges from the tunnel, the four sit there without saying a word.

- The **grandmother** is thinking to herself, “It was very brash for that young man to kiss my granddaughter, and I’m glad she slapped him.”

  The **project manager** is sitting there thinking, “I didn’t know the young tech was brave enough to kiss the girl, but I sure wish she hadn’t missed him when she slapped me!”

  The young **woman** was sitting and thinking, “I’m glad the guy kissed me, but I wish my grandmother had not slapped him!”

  The **young programmer** sat there with a satisfied smile on his face. He thought to himself, “Life is good. How often does a guy have the chance to kiss a beautiful girl and slap his project manager all at the same time!”

Agile Jokes

Project Management: The Cannibals

Three cannibals get hired at a factory.

They are welcomed to the company and told to go to the cafeteria as long as they don't bother the other workers.

The cannibals promise.
A month later the boss comes to see them and tells them all they have been doing a great job, but the janitor is missing. He asks if anyone knows anything about this.
The cannibals shake their heads no.

After the boss leaves the lead cannibal screams at them. "Who is the idiot that ate the janitor?"
One hand slowly rises.

"You idiot!" says the lead cannibal, "we have been eating the project managers so no one would notice and you go and eat the janitor!"

Agile Jokes

Project Management: Body Parts

The body parts argue over who should be in charge.
The **brain** says he should be in charge because he keeps everything running.
The **blood** says he should be in charge because he delivers oxygen to everything else.
The **stomach** says he should be in charge because he provides energy.

Suddenly, the **rectum** speaks up and says he should be in charge because he is in charge of getting rid of waste.

They all laugh at the rectum and call him names. Frustrated, the rectum shuts down and stops working. Soon the brain is hurting, the stomach is all bloated, and the blood is full of toxins. So, they give in and let the rectum be in charge.

Morale: You do not always have to be smart to be in charge, just an *****hole**.

Agile Jokes

Project Management: 3 Cats

A young girl enters a pet shop to buy a cat. The pet shop worker shows her three cats that look identical.

"This cat here costs $1,000," he explains.
"Why does that cat cost so much?" the girl asks.
"This cat knows how to complete **legal research**," the pet shop worker explains.

The girl asks about the cat in the middle, and the pet shop owner explains that the middle cat costs $2,000 because it knows **legal research and can win any case**.

The girl is curious and asks about the third cat.
"That one is $5,000."
"Well, what can this cat do?" asks the girl.

"Honestly, I have no idea. I have never even seen it do anything at all, but it says it's a **Project Manager**."

Agile Jokes

“Professor – Idiot”

One day a college professor after getting irritated in his college class stands up in front of the class and asks if anyone in the class is an idiot, and if there is one then he/she should stand up. After a minute a young man stands up. The professor then asks that guy if he actually thinks he is an idiot. The boy replied, "No, I just didn't want to see you standing there all by yourself."

Agile Jokes

Software Engineer & Frog-Princess:

A young man was walking along in the forest, when he heard a muffled voice crying for help from behind a log. He leaned over to see a frog sitting in the mud.

The frog looked up at him and said, "I'm actually a beautiful princess, and if you kiss me, I'll transform back into my true self, and be yours for eternity."

Silently, the man scooped up the frog and continued on his walk.

A minute or two later, the frog piped up again, "Hey, buddy, maybe you didn't hear me -- I said, if you kiss me, I'll turn into a princess. What are you waiting for?"

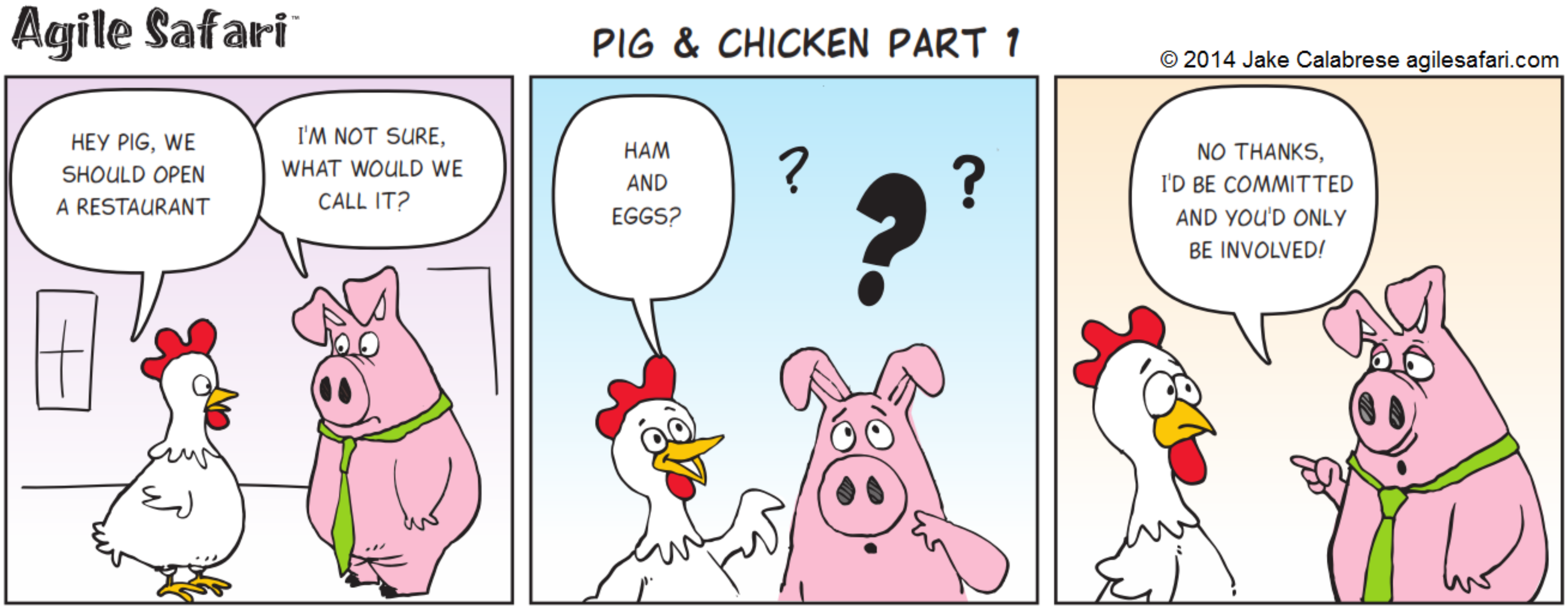
Annoyed, the man stuffed the frog into his coat pocket.

Shocked, the frog yelled from inside the man's pocket, "What the hell? I'm a princess! All you have to do is kiss me!"

Opening his pocket and peering in, the man said, "Listen -- I'm a software engineer. I don't have time for a girlfriend. But a talking frog is kind of cool."

Agile Jokes

Agile: Pig & Chicken


Agile Safari™
PIG & CHICKEN PART 1
© 2014 Jake Calabrese agilesafari.com
HEY PIG, WE SHOULD OPEN A RESTAURANT
I'M NOT SURE, WHAT WOULD WE CALL IT?
HAM AND EGGS?
? ? ?
NO THANKS, I'D BE COMMITTED AND YOU'D ONLY BE INVOLVED!

List of Games

- Build → Plan → Inspect → Adapt Exercise
- Velocity & Workflow Management Exercise
- Multi-Tasking Exercise
- “Who Stole My Cheese” Exercise
- “RAG” Status Escalation Exercise

Build → Plan → Inspect → Adapt Exercise

The Paper Snowflake Game

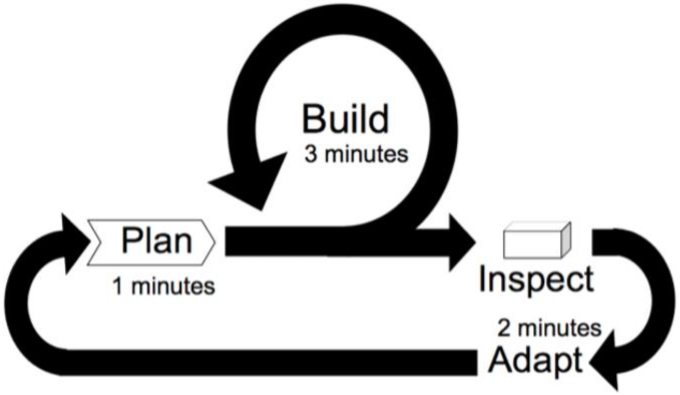
scruminc.

How to Play the Game

Goal: Run a profitable business creating and selling paper snowflakes.

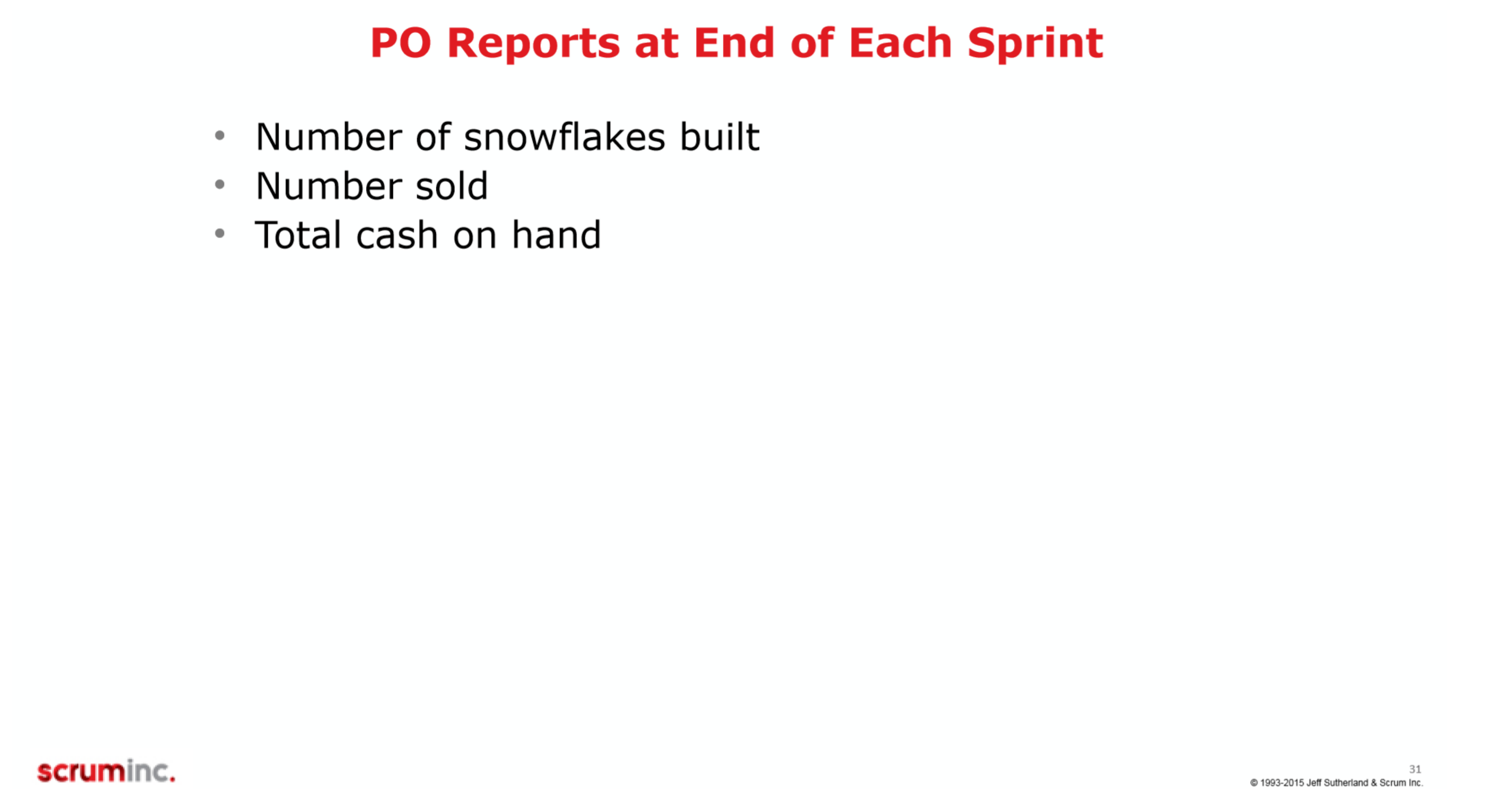
- The teams will construct snowflakes. They start with 5 Scrumbucks, a pair of scissors, and 3 sheets of paper.
- The team's Product Owner transacts with customers (us) at the front of the room…
  - **We buy** snowflakes for 1 - 5 "Scrumbucks" each, depending on how much we like them. We will build piles for each category
  - **We sell** additional supplies:
  - 2 additional paper sheets - S$1
  - An extra pair of scissors - S$3
- The Product Owner shares customer preference feedback with the team

- We will work in 6 minute sprints…



- All building and transactions must take place within the 3min "build" cycle

- At the end of each Sprint, record:
  - The number of snowflakes built
  - Cash on Hand

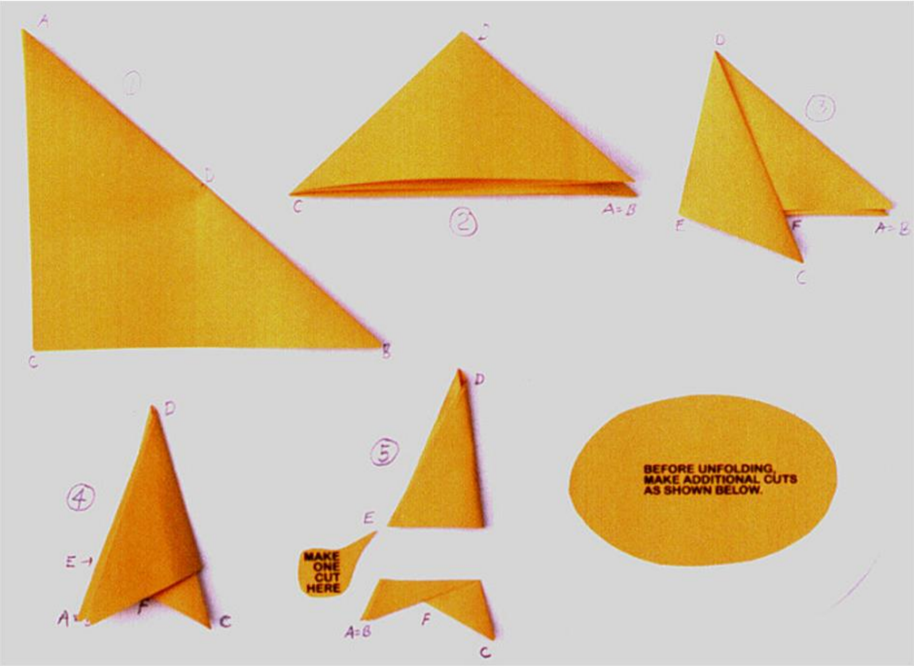


PO Reports at End of Each Sprint

- Number of snowflakes built
- Number sold
- Total cash on hand

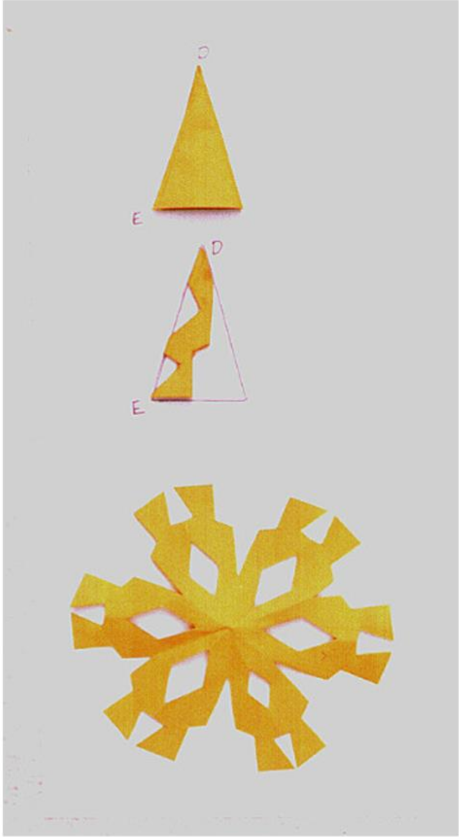
scruminc.

Creating a Paper Snowflake



MAKE ONE CUT HERE

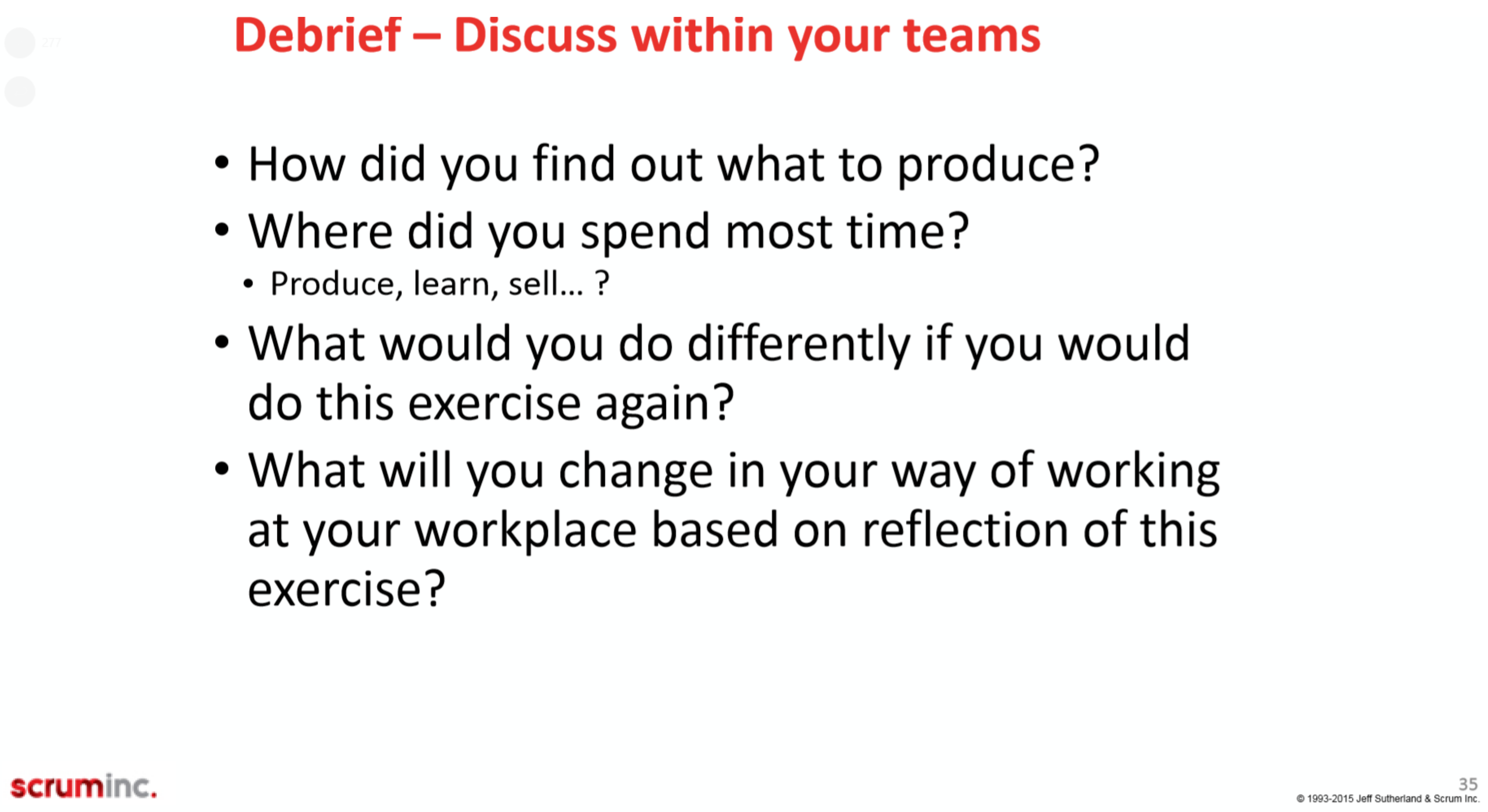
BEFORE UNFOLDING, MAKE ADDITIONAL CUTS AS SHOWN BELOW.



We Buy: Snowflakes 1-5 ScrumBucks

We Sell: 2 papers for 1 buck
1 pair of scissors for 3 bucks

scruminc.



Debrief – Discuss within your teams

- How did you find out what to produce?
- Where did you spend most time?
  - Produce, learn, sell… ?
- What would you do differently if you would do this exercise again?
- What will you change in your way of working at your workplace based on reflection of this exercise?

35

Velocity & Workflow Management Exercise

Velocity and Flow Exercise:
The Dice Game

Velocity and Flow Exercise: The Dice Game

We will need:

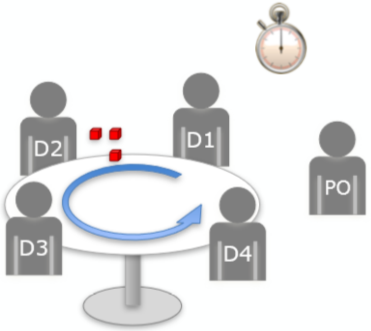
- A Team of 4 workers (dice turners)
- A stack of polyhedral dice

The process:

- Each worker represents one step of the process
- To complete the step, the worker must turn the die so that their step number (i.e. "1" for step one) is facing up
- Once they have completed processing their batch of dice, they pass them to the next worker

The objective:

- **See how long it takes to process dice in batches**
  - **Iteration 1: all dice**
  - **Iteration 2: half the dice (two batches)**
  - **Iteration 3: 1 die at a time**



Velocity and Flow Exercise:
The Dice Game

- **Each Team selects a time-keeper/data recorder**

- **We are not comparing teams (each team gets die of different "complexity" - number of dots)**

- **Can you think of any other way to "optimize" workflow? (different from single-piece workflow, full-batch and half-batch)?**

Multi-Tasking Exercise

Exercise

Original Source: https://blog.crisp.se/2009/02/18/davidbarnholdt/1234986060000
Additional Recap: http://www.keystepstosuccess.com/2017/09/how-detailed-should-business-requirements-be-discovery-through-agile-gaming/

General Instructions

- For the next few minutes, everyone, think of yourself as Leonardo Da Vinci ☺

- Follow instructions that are specific to your Team

- Spend 30 seconds to review instructions in silence

- Send 60 seconds to follow instructions in silence. You are allowed to communicate in sign language

- STOP when the game Facilitator tells you to stop

Instructions: Team 1

Draw a beautiful summer meadow with:

- 10 blue flowers with 5 petals each
- 5 blue flowers with 6 petals each
- 13 red flowers with 6 petals each

- 2 cows with 3 black spots
- 1 cow with 5 black spots
- 2 cows with 4 black spots

- 2 birds to reside in the upper left corner
- 3 birds in the middle
- one sun to the right with 5 sun beams

Instructions: Team 2

Draw a beautiful summer meadow with blue and red flowers in green grass, some cows and birds under a shining sun.

Reflection / Conclusion

- How did dynamics of Team 1 and Team 2 differ in the first 30 seconds?
- Did/how things changed in the next 60 seconds?
- How do people manage/share supplies (colored markers)?
- What can you say about individuals’ anxiety and eagerness to contribute individually, in silos?
- What can you say about the final **output** by each team: details, positioning and proportional size of objects, color choice?
- What can you say about the final **outcome** by each team: overall picture harmony and eye-appeal?
- Overall, what does each Team feel about the level of details in requirements they have received? Too little? Too much? Helpful? Distracting?

Exercise

General Rules

- Each Team Member writes Arabic numbers, Roman numbers and Letters, as per respective **Working Policy (A or B)**.

- Upon completion, each Team Member announces loudly: “DONE”.

- One person (Timer) keeps track of everyone’s time and reads back to each Team Member how many seconds it took to complete work (splits could be short).

- Each Team Member writes down his/her own time on a worksheet.

- When everyone is finished, times of individual Team Members are added together by Timer and provided to Game Facilitator

- Game Facilitator records provided data on a board

Policy-Specific Rules

Working Policy A

Write Arabic numbers from 1 to 10, followed by Roman numbers 1 to 10, followed by letters A through J.

Arabic	Roman	Letters
1	i	A
2	ii	B
3	iii	C
4	iv	D
5	v	E
6	vi	F
7	vii	G
8	viii	H
9	ix	I
10	x	J

Total Time (all team members) =______

Working Policy B

Write Arabic numbers, Roman numbers and letters, row by row (top to bottom): 1→i→A, 2→ii→B, 3→iii→C….etc…

Arabic	Roman	Letters
1	i	A
2	ii	B
3	iii	C
4	iv	D
5	v	E
6	vi	F
7	vii	G
8	viii	H
9	ix	I
10	x	J

Total Time (all team members) =______

Team Member’s Worksheet

Working Policy A

Write Arabic numbers from 1 to 10, followed by Roman numbers 1 to 10, followed by letters A through J.

Arabic	Roman	Letters

Individual (Team Member) Time ______

Working Policy B

Write Arabic numbers, Roman numbers and letters, in rows: 1→i→A, 2→ii→B, 3→iii→C….etc…

Arabic	Roman	Letters

Individual (Team Member) Time ______

Timer’s Calculation Worksheet

Working Policy A

Team Member	Time Lapsed

Total ______

Working Policy B

Team Member	Time Lapsed

Total ______

Game Facilitator’s Worksheet

Team	Time Lapsed (Policy A / Policy B)	% Difference
	/	
	/	
	/	
	/	
	/	
	/	
	/	
	/	
	/	
	/	

Total __/__

Conclusions

- What impact does task-switching/multi-tasking have on individual Cycle Time?

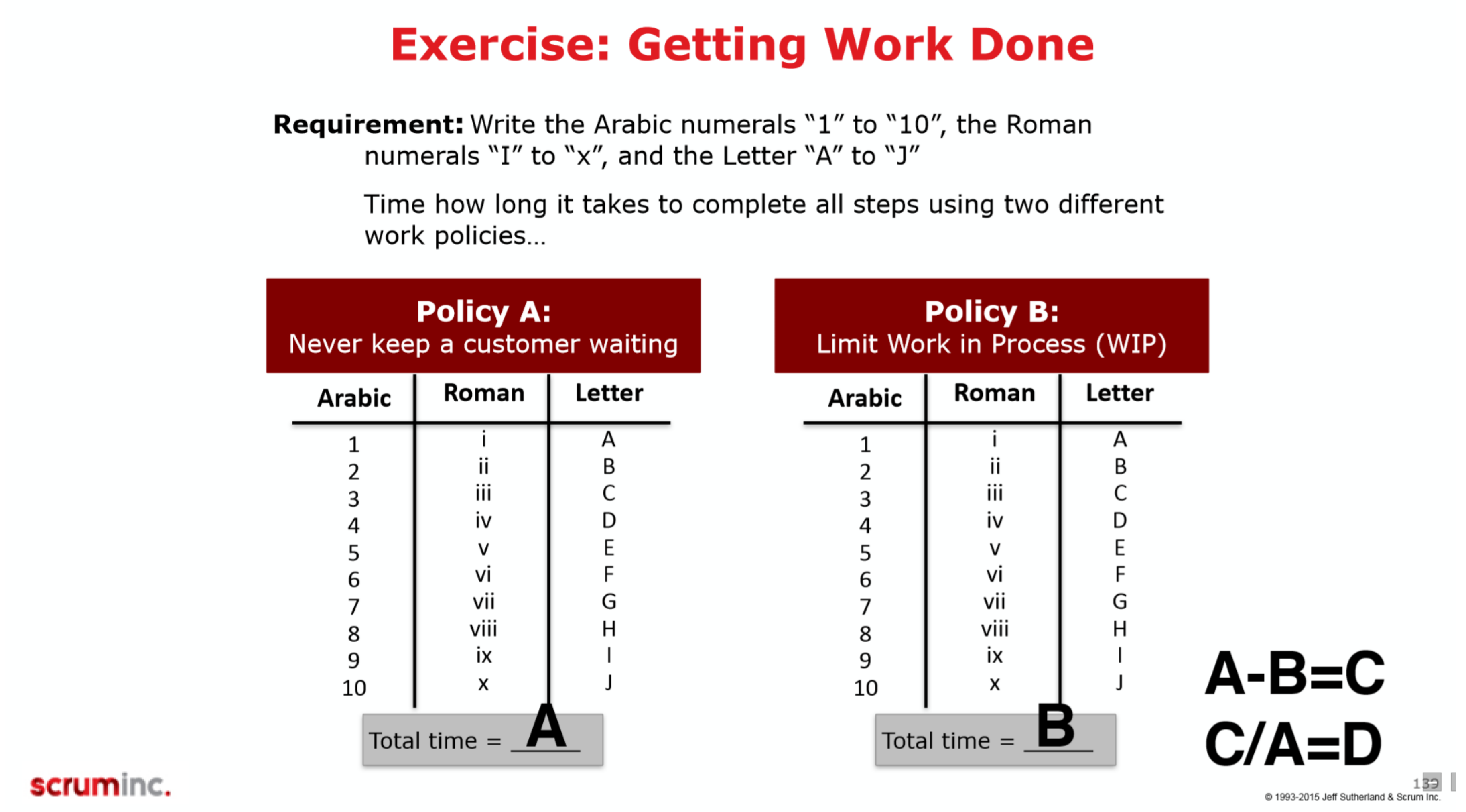
- Is there a compound impact on a team’s Cycle Time with possible intra-team dependencies?

- Will there be a compound impact on Cycle Time of multiple teams working together, with possible cross-team dependencies?

- How does understanding the above, change your perception, when you see too much WIP (In-Progress) work on a team’s Sprint Board? When a team aggressively plans its sprint?

- Does task-switching/multi-tasking help establishing healthy team dynamics? Is it **ECONOMICALLY WISE**?

Multi-Tasking Exercise



Exercise: Getting Work Done

Requirement: Write the Arabic numerals "1" to "10", the Roman numerals "I" to "x", and the Letter "A" to "J"

Time how long it takes to complete all steps using two different work policies…

Policy A:
Never keep a customer waiting

Arabic	Roman	Letter
1	i	A
2	ii	B
3	iii	C
4	iv	D
5	v	E
6	vi	F
7	vii	G
8	viii	H
9	ix	I
10	x	J

Total time = A

Policy B:
Limit Work in Process (WIP)

Arabic	Roman	Letter
1	i	A
2	ii	B
3	iii	C
4	iv	D
5	v	E
6	vi	F
7	vii	G
8	viii	H
9	ix	I
10	x	J

Total time = B

A-B=C

C/A=D

scruminc.

Weinberg Table of Project Switching Waste

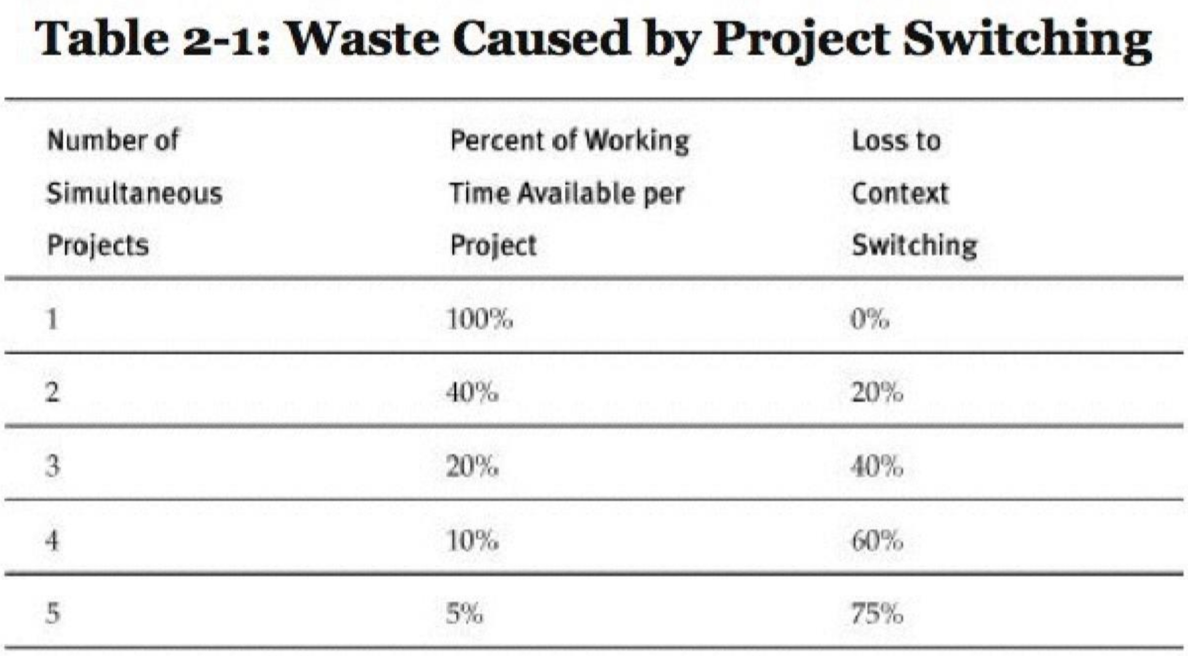


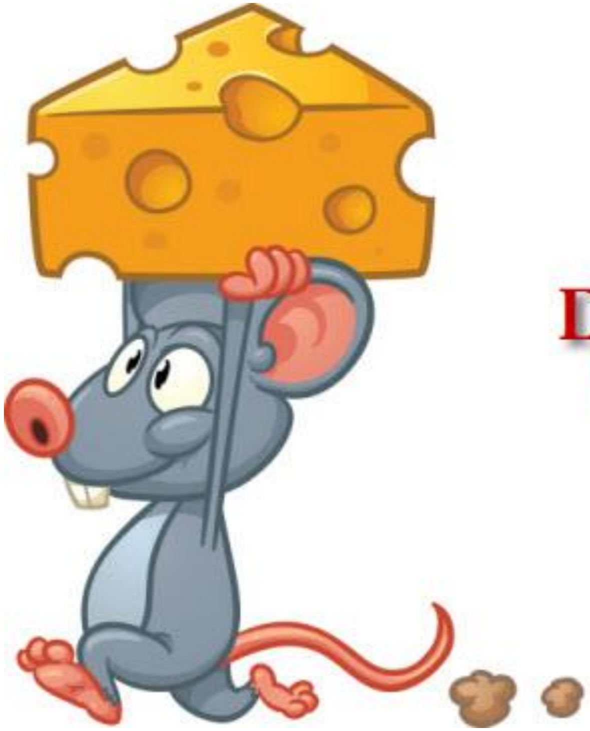
Table 2-1: Waste Caused by Project Switching

Number of Simultaneous Projects	Percent of Working Time Available per Project	Loss to Context Switching
1	100%	0%
2	40%	20%
3	20%	40%
4	10%	60%
5	5%	75%

Weinberg, Gerald M. (1992) Quality Software Management: Systems Thinking. Dorset House, p. 284.

Scrum Role Modeling Exercise

“Who Stole My Cheese”



Did Someone Move Your Cheese?

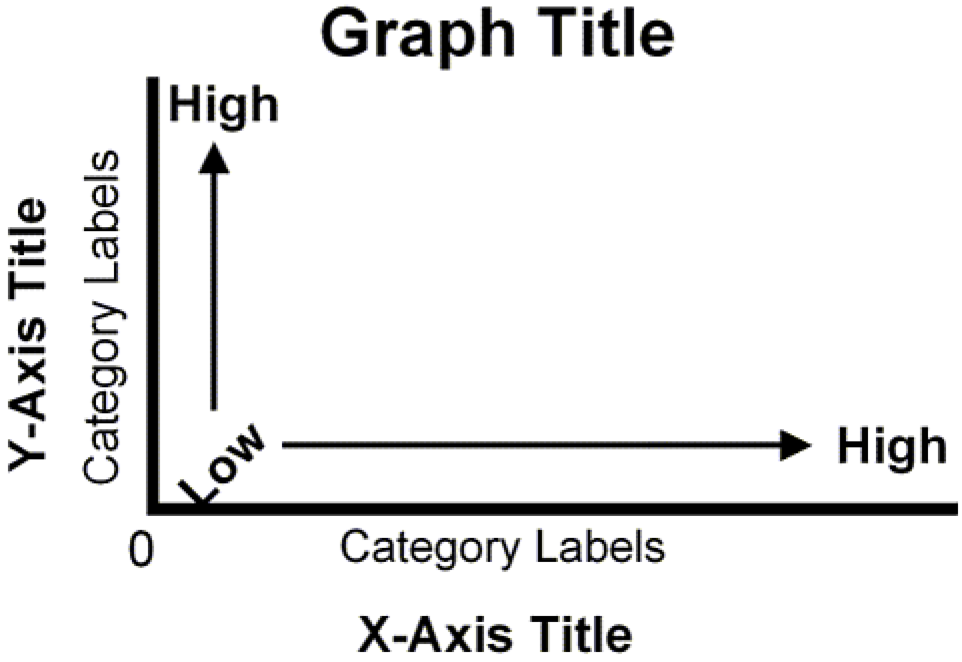
Broken “RAGs”

“RAG” Status Escalation Exercise

The 'Telephone' game shows how easily information can be corrupted via indirect communication.

Take-Home Tools

ScrumMaster “Tool”: Impediment Mapping Exercise

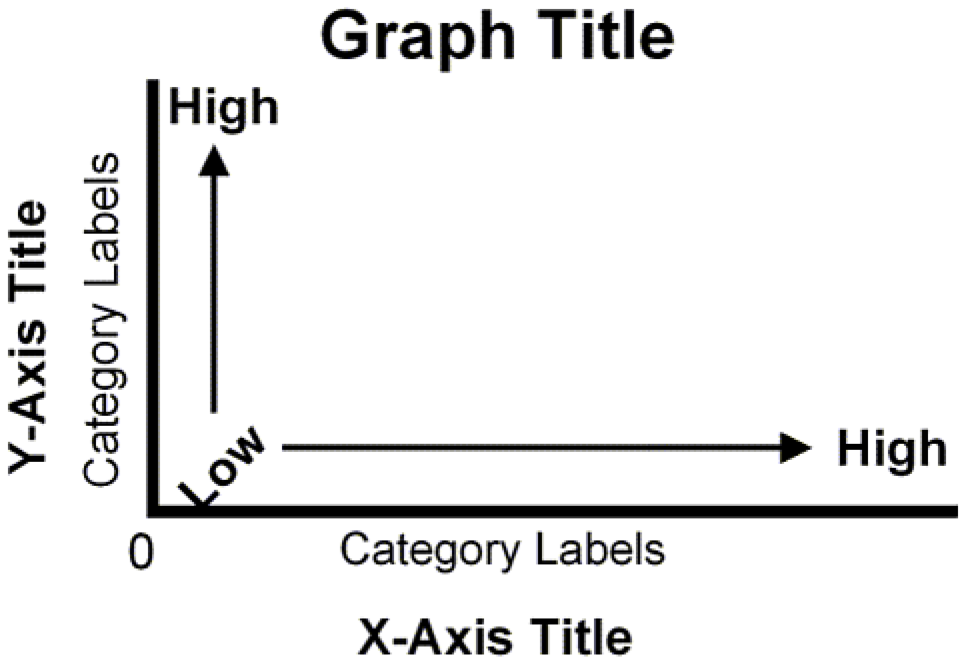


Organizational Level

Seriousness

Product Owner “Tool”: Work Prioritization Exercise

Priority



Size/Complexity