AGILE PROJECT MANAGEMENT

Course 1. Applied Scrum for Project Management

Module 1 of 4

John Johnson, PMP CSM SPC
A. James Clark School of Engineering, University of Maryland, College Park

Week 1. Agenda

1. Agile Basics
2. Proof Agile Works
3. Evolution of Agile
4. Case Study 1 – Netflix
5. Case Study 2 – 18 F

Valuable Sprints & Dispelled Myths

Agile Basics

Intro to Agile: the Manifesto

Agile was codified in 2001 at the Snowbird Resort by 17 practitioners of Iterative Development. The Agile Manifesto was written by XP, DSDM, and Scrum practitioners stating

http://logicboost.com/agilemanifesto.html

Intro to Agile: the Manifesto

"...while there is value in items on the right, we value items on the left more…"

- Individuals and Interactions over ***processes and tools***
- Working Software over ***comprehensive documentation***
- Customer Collaboration over ***contract negotiation***
- Responding to Change over ***following a plan***

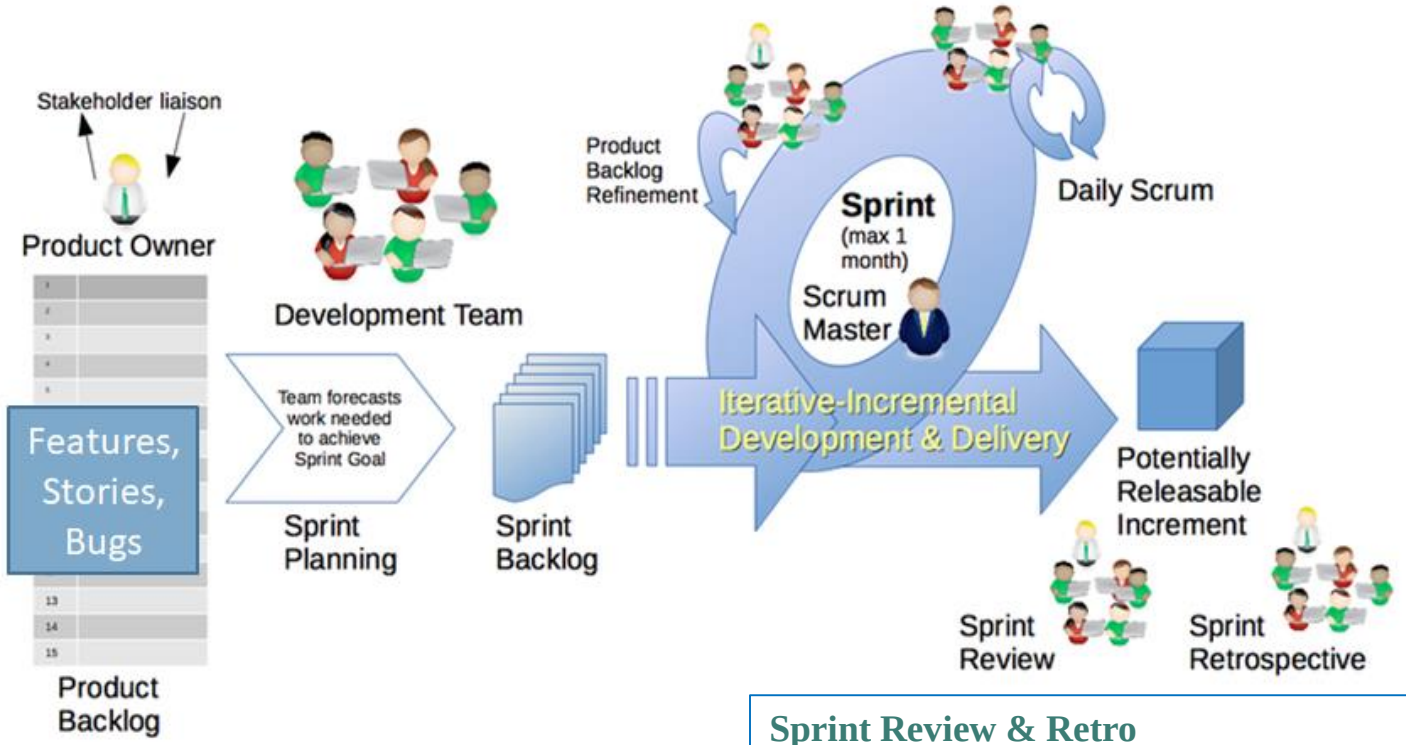
Sprint Basics

Sprint Planning 1

- Team & Product Owner select work
- Team commits to complete work inside the Sprint
- All work is stated as a "User Story" with a clear "who, what, why" and acceptance criteria
- Scrum Master facilitates and guides

Sprint Development 2

- Team meets daily to decompose & assign work
- Team self-organizes based on skills
- No client can interrupt or change their work
- Product Owner liaisons with end users
- Product Owner builds and prioritizes backlog
- Scrum Master facilitates and tracks

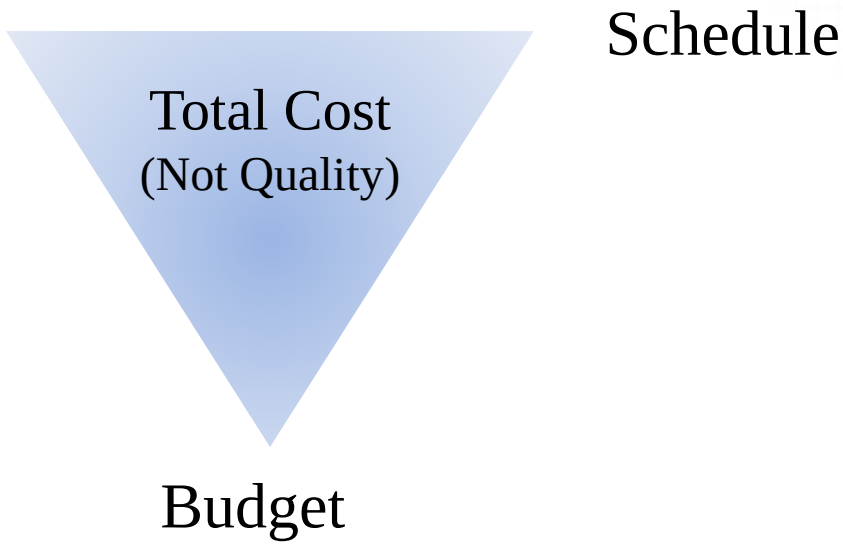


https://en.wikipedia.org/wiki/Scrum_(software_development)#/media/File:Scrum_Framework.png

Sprint Review & Retro 3

- Team presents completed work to customer
- Team reviews work performed
- Team performs retrospective to improve itself
- Scrum Master facilitates and guides

Comparing Agile, Traditional, and Lean



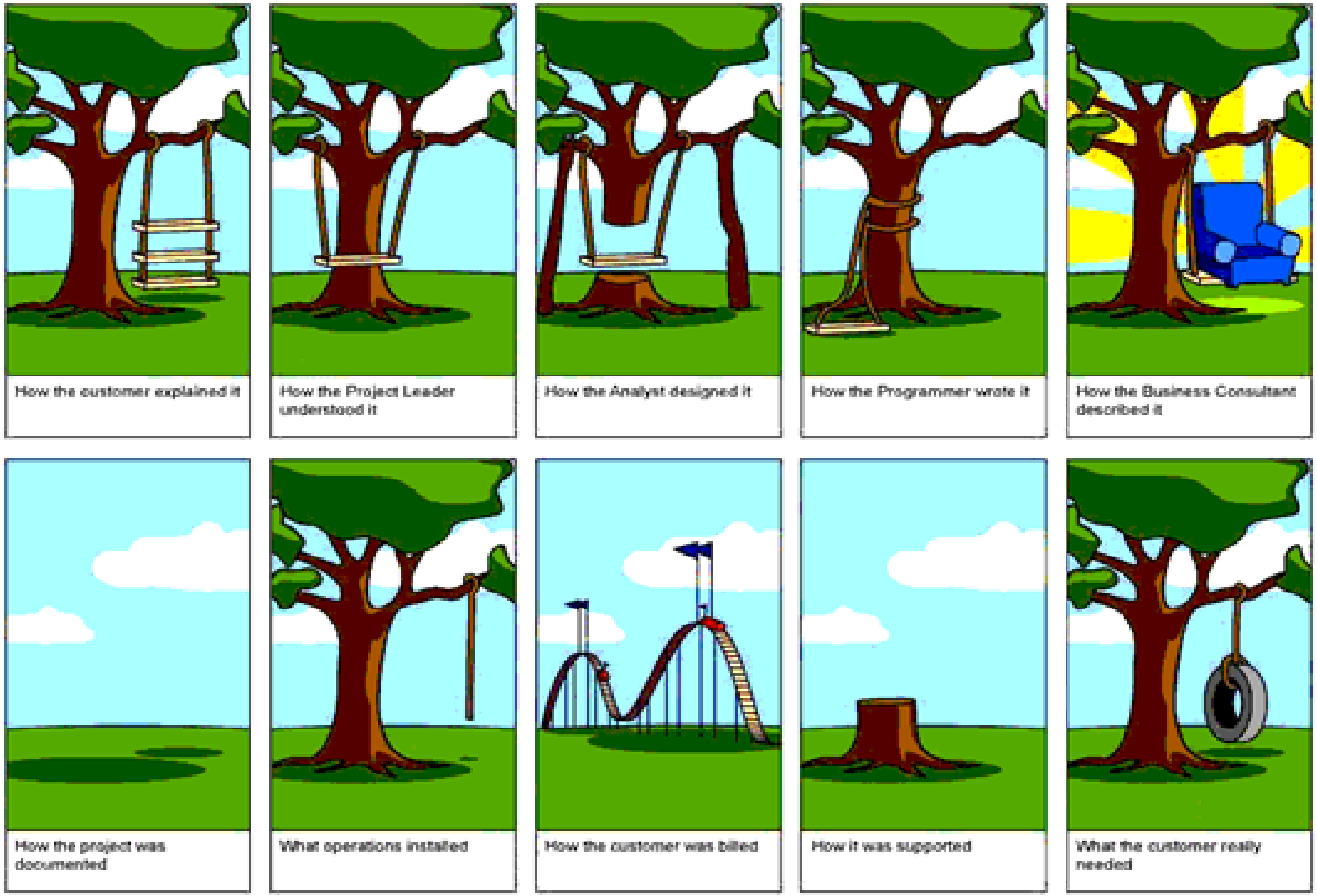
	Agile	Traditional	Lean
Adjust	Scope	Budget	Schedule
Requires	Trust	Efficiency	Expertise
Goal	Speed	Predictability	Innovation

False Comparisons

The following is true for not only Traditional, but also Lean, and Agile projects. Each methodology has:

- ***A Charter***
- ***A Plan***
- ***Documentation***
- ***Design***
- ***Testing***

The Story


How the customer explained it
How the Project Leader understood it
How the Analyst designed it
How the Programmer wrote it
How the Business Consultant described it
How the project was documented
What operations installed
How the customer was billed
How it was supported
What the customer really needed

Wait, so What Is Agile Again?

Agile is a Project Management ***Methodology*** we can defined as having:

- Shared Vision Robust to Change (can vary tech scope)
- Whole Teams (customer + a cross-functional team)
- Incremental Delivery (learn by doing using small “Sprints”)
- Continuous Integration & Testing (teams test increments early often)

Scrum, *SAFe*, *Disciplined*, *Kanban* - these are ***Frameworks*** which offer a structure for conducting Agile projects.

From Spacecraft to Supercomputers

Proof That Agile Works

Proof Agile Works: Skunkworks

- Clarence Leonard "Kelly" Johnson, Lockheed Martin Engineer in WWII.
- In 1943, tasked with extending range of fighter jets
- He and his team colocated in a tent because they needed the space…
- Program was called "Skunk Works"

Designed and built the first jet-fighter, "P-80 Shooting Star," in just 143 days

Keys to Skunkworks Success

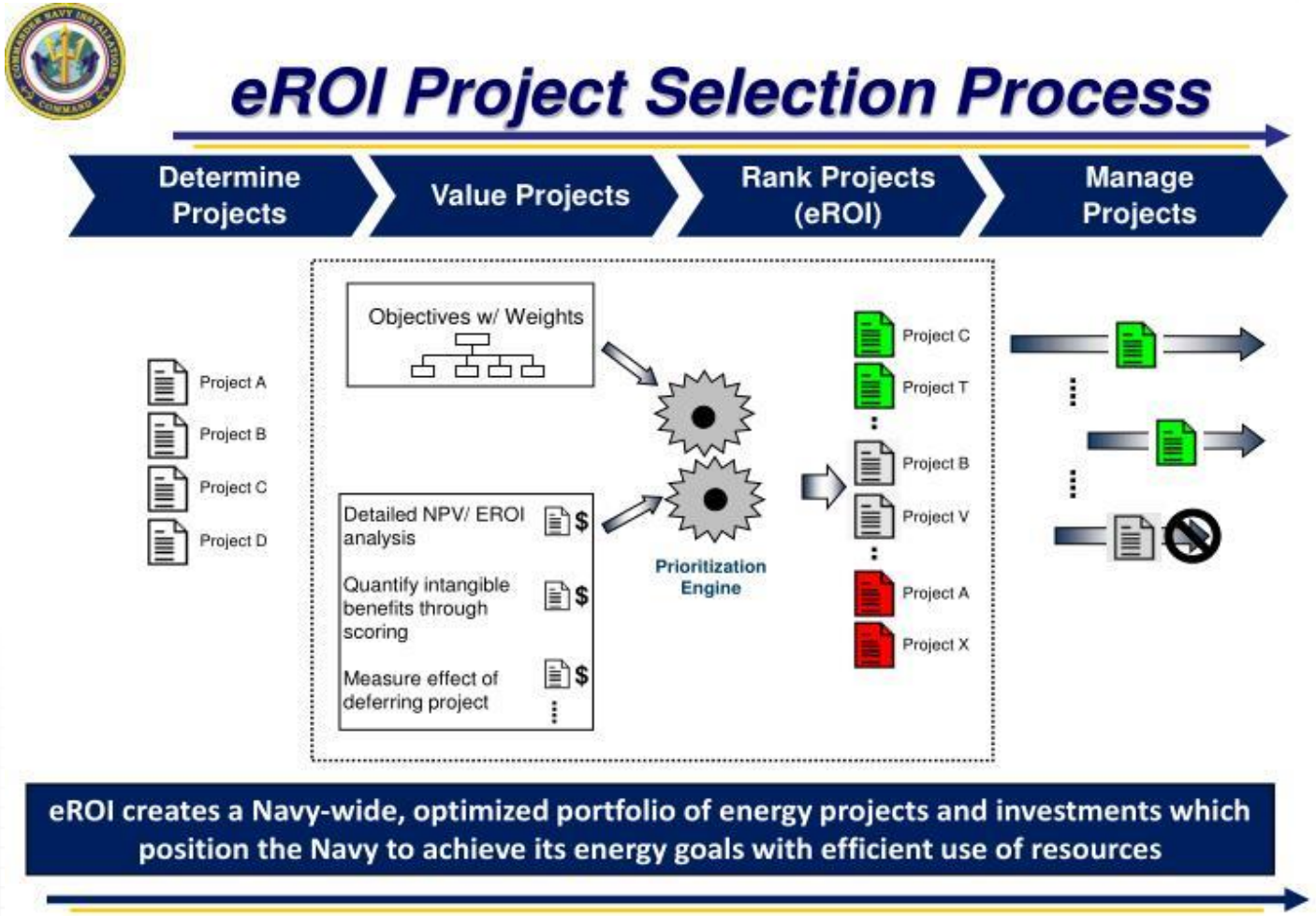
Kelly Johnson's Skunkworks Program had 14 Rules of Management, which roughly translate to:

- Small, Strong, Self-Directed Cross-functional Teams
- Owners and Vendors must Collaborate and Trust
- Manage and Respond to Change
- Minimize Reports, But Record Important Work
- Incremental Development with Self-Testing Teams

Remember! Agile is a Project Management ***Methodology*** we can defined as:

- Shared Vision Robust to Change (can vary tech scope)
- Whole Teams (customer + a cross-functional team)
- Incremental Delivery (learn by doing using small "Sprints")
- Continuous Integration & Testing (teams test increments early often)

Proof Agile Works: Navy Energy Program



Source: https://image1.slideserve.com/1599299/slide3-n.jpg

Proof Agile Works: Navy Energy Program

Navy Shore Energy Program, Energy Return on Investment (eROI) Support

Booz Allen Hamilton (BAH)

Scope: Build decision support systems to identify, evaluate, and select $500M/yr. in shore energy projects

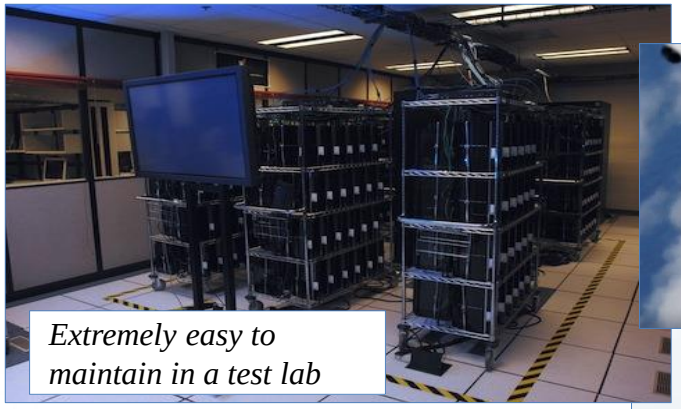
Total Cost: $5M over 4 years (T&M)

- 2 Fully Cross-Functional Teams
- BAH Personnel: 8 (1 PM, 3 Devs, 4 BA/Testers)
- Navy Personnel: 5 (1 PgM, 3 Officers, 1 Analyst)

Output: ***Project ROI: 50***

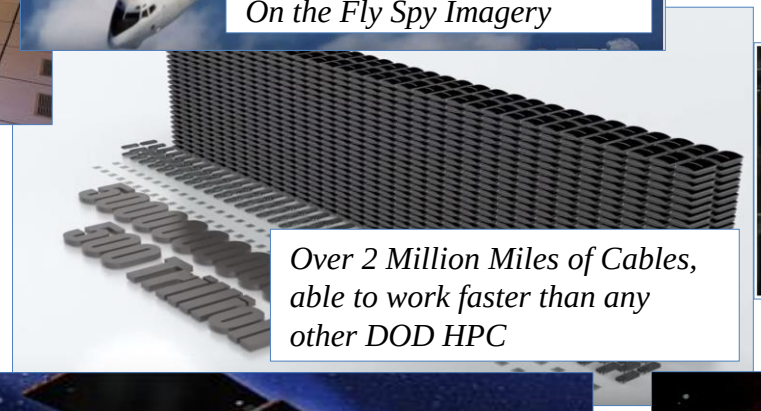
- QA/QC avoided $20M/yr. in net-loss projects
- Improved selection by $30M/yr. annualized returns
- Modeled investments with 95% accuracy by year 3
- BAH sole sourced the $10M/yr Renewables Program

Proof Agile Works: Spacecraft to Supercomputers



Extremely easy to maintain in a test lab

Can be loaded on a plane On the Fly Spy Imagery



Over 2 Million Miles of Cables, able to work faster than any other DOD HPC

Image Source: http://www.zdnet.com/article/what-the-dods-playstation-powered-condor-cluster-means-for-the-future-of-supercomputing/

We went from days to seconds!

Condor Cluster Supercomputer, Air Force 2010

NASA led the Faster, Better, Cheaper (FBC) initiative in the 1990s

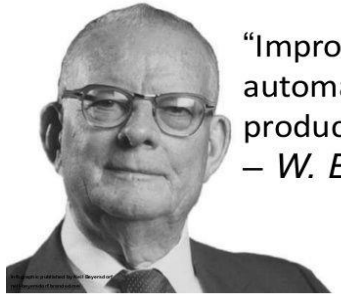
Stardust

Shoemaker

Spiraling Away from Waterfall: a Total Quality Revolution

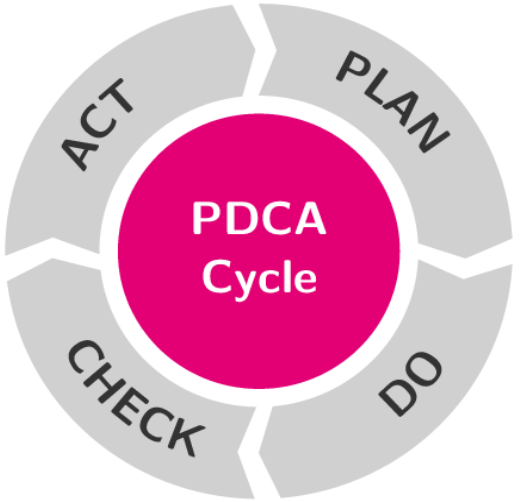
Evolution of Agile

Total Quality Management (TQM)



"Improve quality, you automatically improve productivity."
– *W. Edwards Deming*

https://i.pinimg.com/736x/98/bc/1f/98bc1f7c7ce266dd7e2fe796be001285--teacher-w-edwards-deming.jpg



- Improving quality decreases costs
- Must continuously improve (systems and people)
- Key is pride of workmanship, cross-functional teams, and trust
- Plan – Do – Check – Act (PDCA)

Proof it works: turned around Ford Motors in 1986 from $B losses to first profits in years

Toyota Production System (TPS)

https://c1.staticflickr.com/9/8110/8472007819_485415e875_b.jpg

https://lifehacker.com/productivity-101-how-to-use-personal-kanban-to-visuali-1687948640

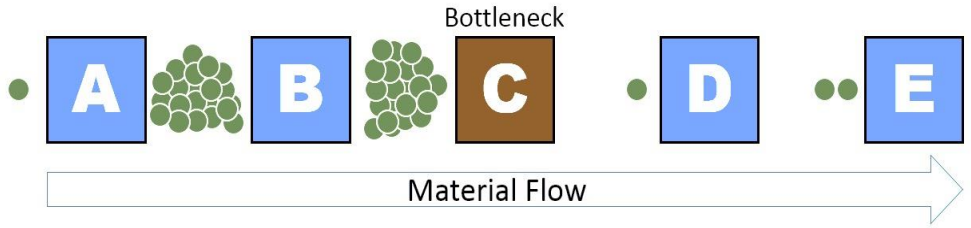
Toyota Production System (TPS) – Taichii Ohno and Lean (1980s - Present)

- Eliminate 7 Wastes - Movement, Inventory, Motion, Waiting, Overproduction, Over-processing, Defects
- Small Batches – addresses most of the waste – *Kanban!*
- Continuous Improvement w/ Fixed Reporting Schedules & Metrics (KPIs)

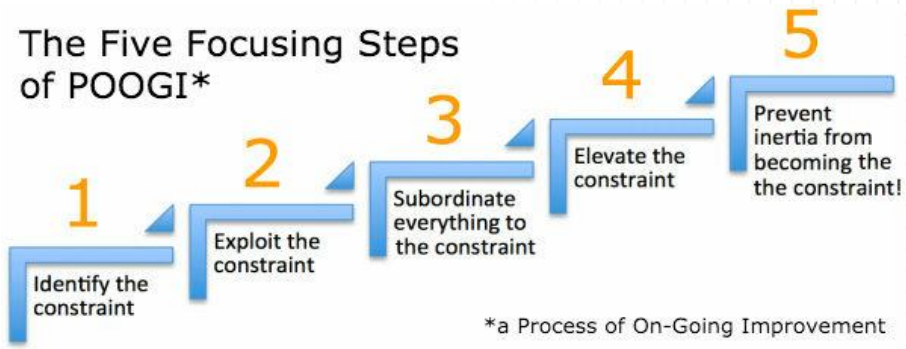
Proof it works: Toyota's a Top 3 Car Manufacturer with ***70% employee satisfaction***
....Note that employee satisfaction is only 30% avg. Nationally

Theory of Constraints (TOC)

https://en.wikipedia.org/wiki/Eliyahu_M._Goldratt



https://en.wikipedia.org/wiki/Bottleneck_(production)



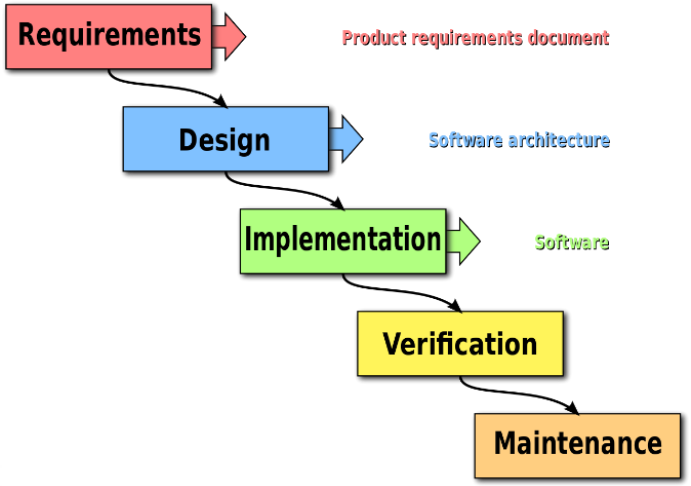
https://www.tocinstitute.org/five-focusing-steps.html

Theory of Constraints (TOC) – Eli Goldratt (1980s to Present)

- Optimize "System Throughput" not "Cost Centers" towards a ***Goal***
- Five Focusing Steps to Exploit System Constraints (Physical, Paradigm, Policy, Market)

Proof it works: BP used TOC to save $200M and rapidly clean 10,000 boats after Gulf Oil Spill

The Waterfall Mistake



The average Software Project had a 10% success rate in the 1970s

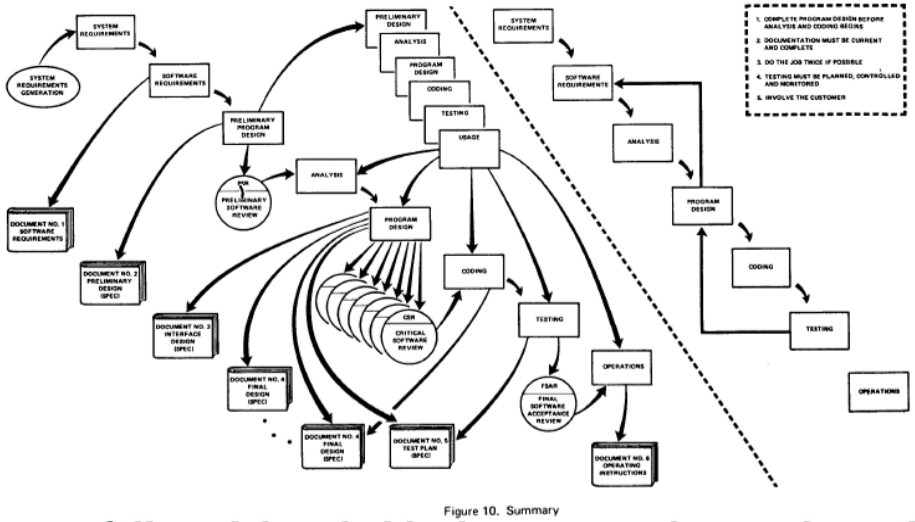


Figure 10. Summary

Waterfall model probably the most costly mistake in the world

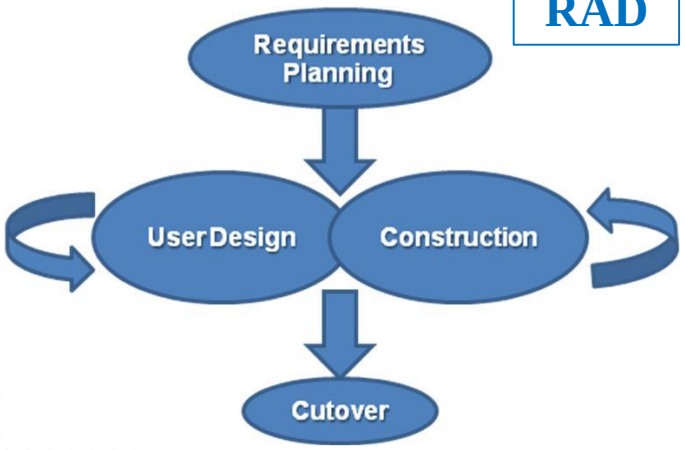
http://valueatwork.se/waterfall-model-probably-the-most-costly-mistake-in-the-world/?lang=en

By the 1980s "Waterfall" was the predominant methodology, but it was a poor fit for the immaturity of the software development world (although embraced by DoD until 1996)
.....Tom Cargill of Bell Labs said it all with his "Ninety-Ninety" Rule said it all:

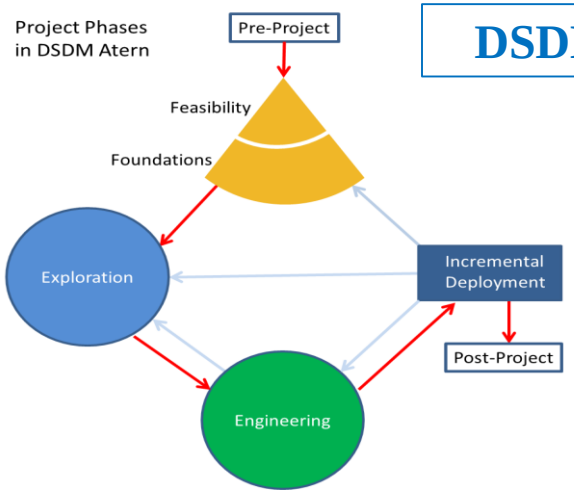
The **first 90 percent** of the code accounts for the **first 90 percent** of the **development time**.
The **remaining 10 percent** of the code accounts for **the other 90 percent** of the **development time**.

Spiral to Scrum

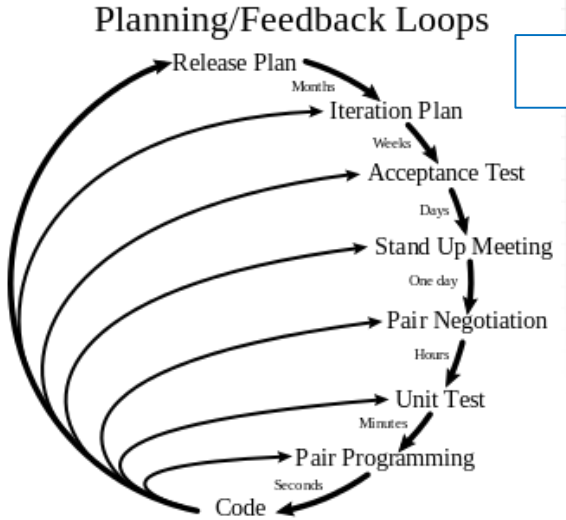
RAD



DSDM



XP



In response to failure rates as high as 90%, "iterative development" was born:

- Rapid Application Development (RAD) 1970s - 1980s
- Dynamic System Development Methodology (DSDM) 1980s - 1990s
- Extreme Programming (XP) 1990s - 2000s

Spiral to Scrum

Key Tenants of Iterative Development:

- ***Consolidated Up-Front Planning*** - single "Systems Design" phase with Stakeholders
- ***Iterative Development*** – Users Propose and Test Product Throughout Development
- ***Timeboxes*** - Emphasizes On-Time Delivery
- ***User Stories*** - Emphasizes Business Needs, Not Tech Specs
- ***Test-Driven Development*** - Incorporation of "best practices"

2013 Cross-Industry Study
173 Industry Respondents

http://www.ambysoft.com/surveys/success2013.html

Agile	Traditional
64% Successful	49% Successful
28% Challenged	33% Challenged
8% Failed	18% Failed

https://clearcode.cc/blog/agile-vs-waterfall-method/

How Netflix Wins!

Commercial Case Study

Netflix Case Study

Watch the speech here:
https://www.youtube.com/watch?v=wyWI3gLpB8o

SPEED WINS!

Keynote: Velocity and Volume
(or Speed Wins) by Adrian Cockcroft

Adrian Cockroft names four (4) things required to turn Netflix from a manufacturing company into an web-centric large-scale business:

1. Culture of Innovation - *respond to opportunities*
2. Data and Analytics - *compare alternatives*
3. Decentralized Decisions - *assign resources quickly*
4. Agile and Self-Service Deploy - *freedom & responsibility culture*

18F to the Rescue!

Government Case Study

18F Case Study

General Services Administration (GSA) supports CA Social Services...

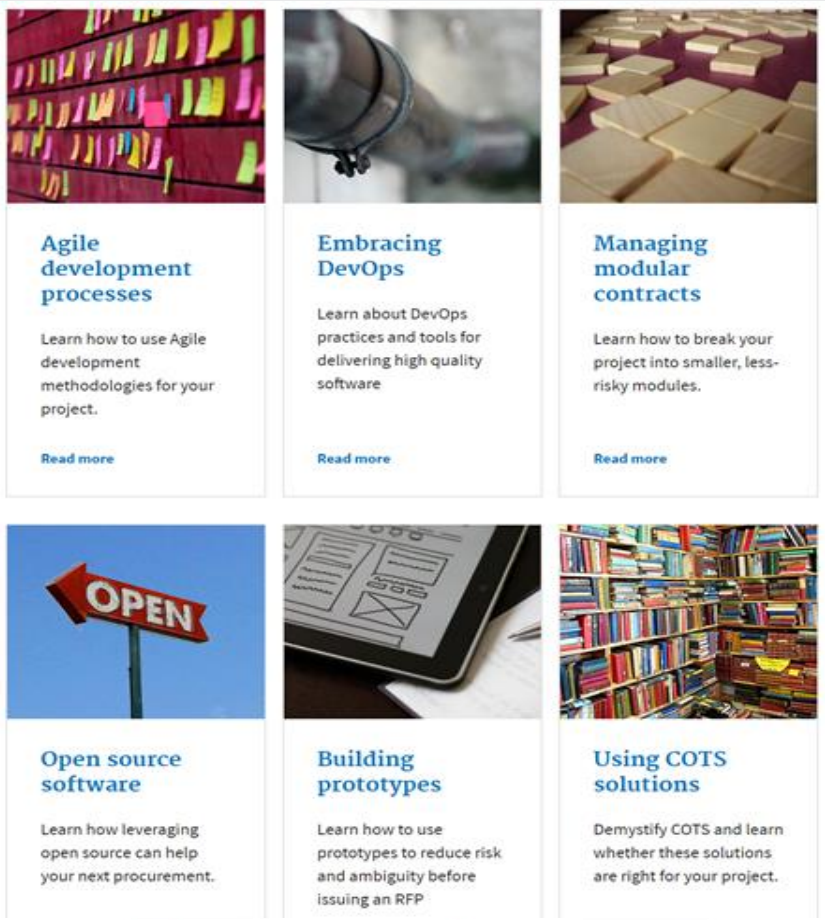
In 2015, the State of California began a process to modernize their child welfare services case management system.

- Used by more than 20,000 social workers
- Track and manage the more than 500,000 cases of child abuse and neglect annually

18F worked with California's Department of Social Services and Office of Systems Integration to add:

- modular contracting,
- agile development,
- user-centered design, and
- open source practices

This project is still in the early stages, but this change in strategy has started to produce greater vendor competition, cost savings, a vastly improved end product, and a better contracting experience.



https://www.youtube.com/watch?v=lNSmF7-xisU

You've just competed Module 1 of *Applied Scrum for Project Management*

Thank You!

[*CLICK here to go to the EdX Course Page*]

WE HAVE THE

MASTERS LEVEL EDUCATION OPTIONS

YOU NEED.

- [x] Technical undergraduate degree
- [ ] Project Management skills

- Master of Science
- Master of Engineering in Project Management
- Graduate Certificate

COURSES OFFERED
100% ONLINE,
100% ON-CAMPUS,
OR A
COMBINATION
OF BOTH.

MORE INFORMATION AT PM.UMD.EDU

UNIVERSITY OF MARYLAND 1856
PROJECT MANAGEMENT
CENTER FOR EXCELLENCE
A.J. CLARK SCHOOL OF ENGINEERING
Civil & Environmental Engineering Department