Why Agile?

The Economics, Psychology, and Science
of Agile’s Success

@MatthewRenze

#PrairieCode

Purpose

Explain why Agile practices are so successful

Insights from Economics, Psychology, and Science

Top 7 most important ideas

Ideas that are not typically covered

Overview

1. The World after Midnight
2. Inverted Constraints
3. Prioritizing Value
4. Embracing Change
5. Self-Organization
6. Effective Communication
7. Feedback

About Me

Independent software consultant

Education

- B.S. in Computer Science
- B.A. in Philosophy

Community

- Public Speaker
- Pluralsight Author
- Microsoft MVP
- ASPInsider
- Open-Source Software

IOWA STATE UNIVERSITY

PLURALSIGHT

MVP Microsoft® Most Valuable Professional

ASP INSIDERS

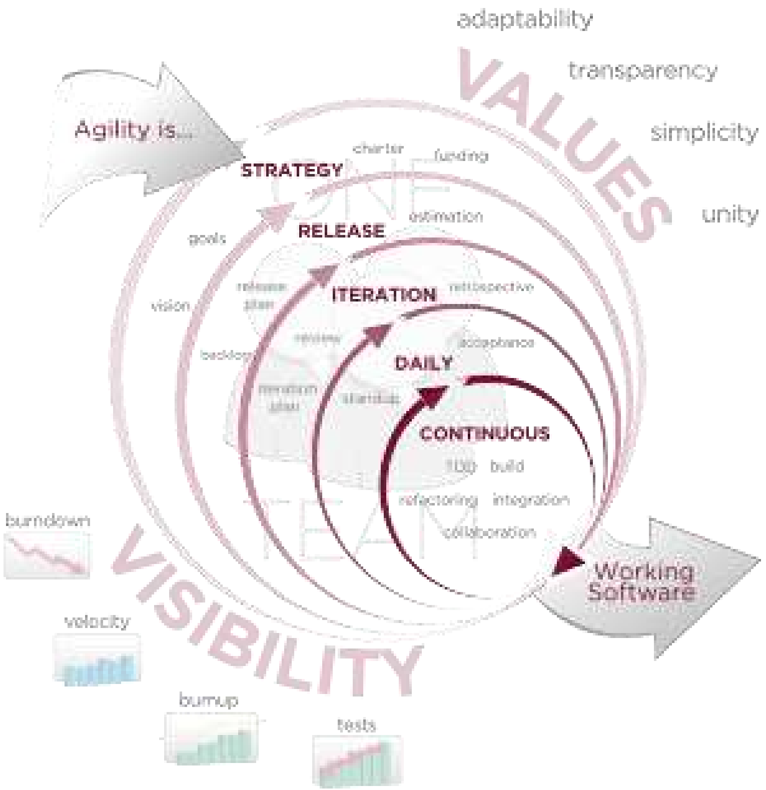
A Brief Review of Agile

What is Agile?

Agile Manifesto

- 4 value propositions
- 12 principles

Common practices



Source: Wikipedia

What is Agile?

Agile is *not*:

- A methodology itself
- A magic silver bullet

Source: http://www.best-story.net/userfiles/silver-bullets.jpg

Agile Values

Individuals and interactions

over processes and tools

Working software

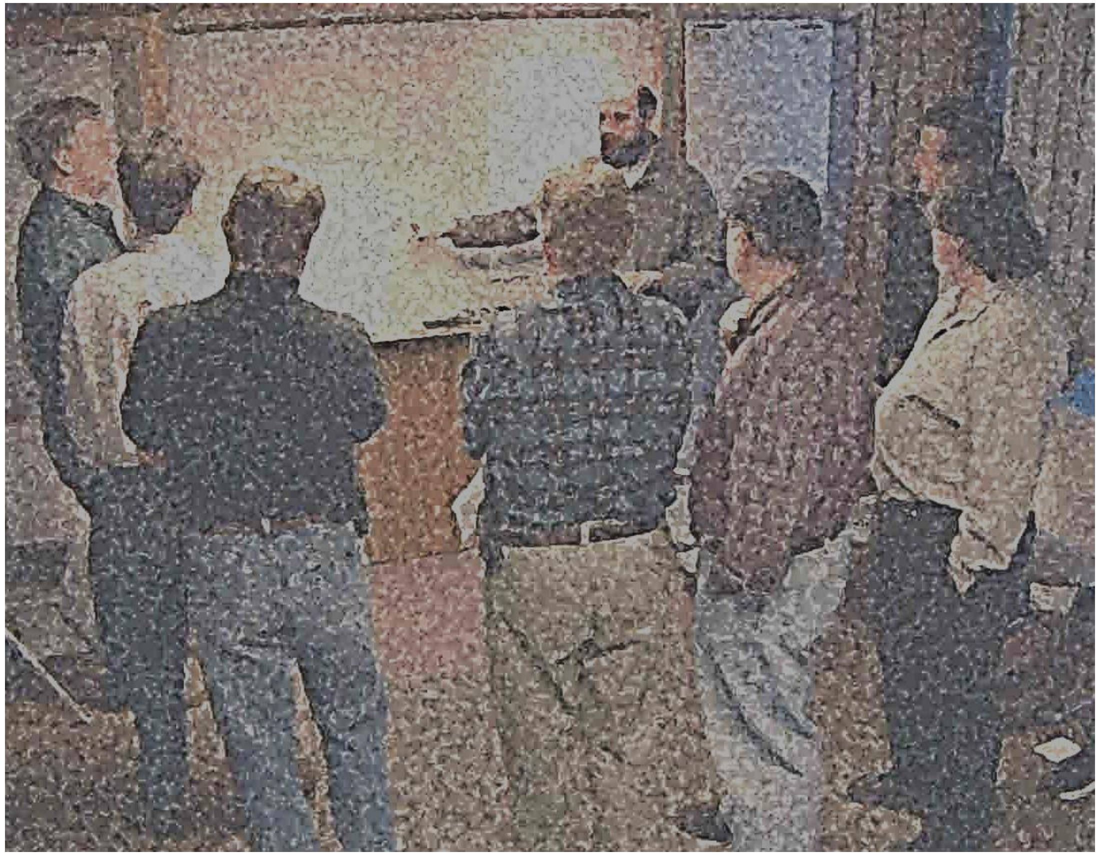
over comprehensive documentation

Customer collaboration

over contract negotiation

Responding to change

over following a plan



Source: http://agilemanifesto.org/

12 Principles of Agile

1. Continuous delivery of value
2. Embrace changing requirements
3. Frequent deployment
4. Customer collaboration
5. Motivated individuals
6. Face-to-face conversation

12 Principles of Agile

7. Working software as measure of progress
8. Sustainable development
9. Technical excellence
10. Simplicity
11. Self-organization
12. Continuous improvement

Agile Methodologies

Scrum

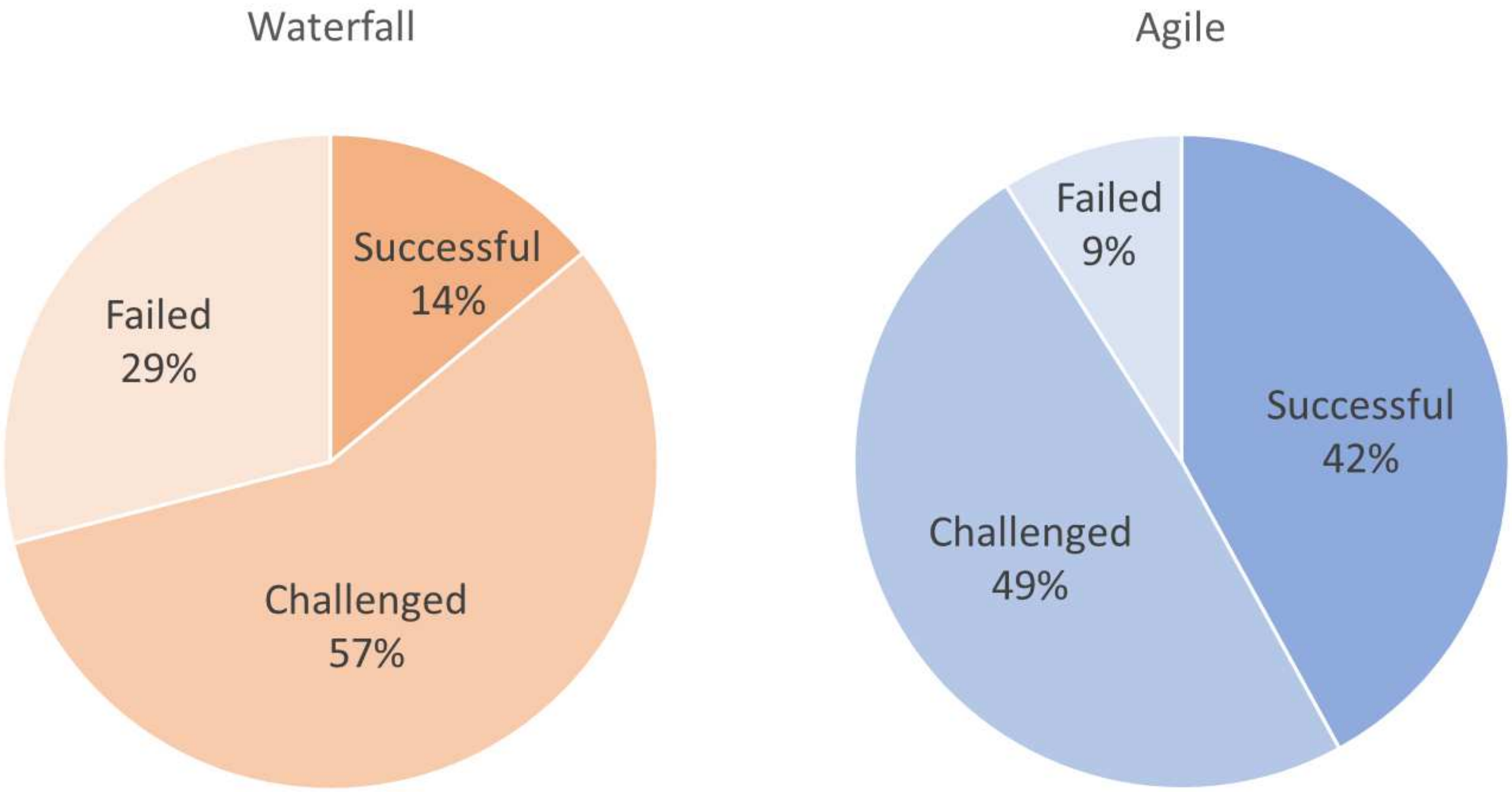
XP

Kanban

Lean

Source: http://parkertoddloesch.files.wordpress.com/2011/09/umbrella.jpg

Is Agile More Successful?



Original Source: The Standish Group, The CHAOS Report 2012

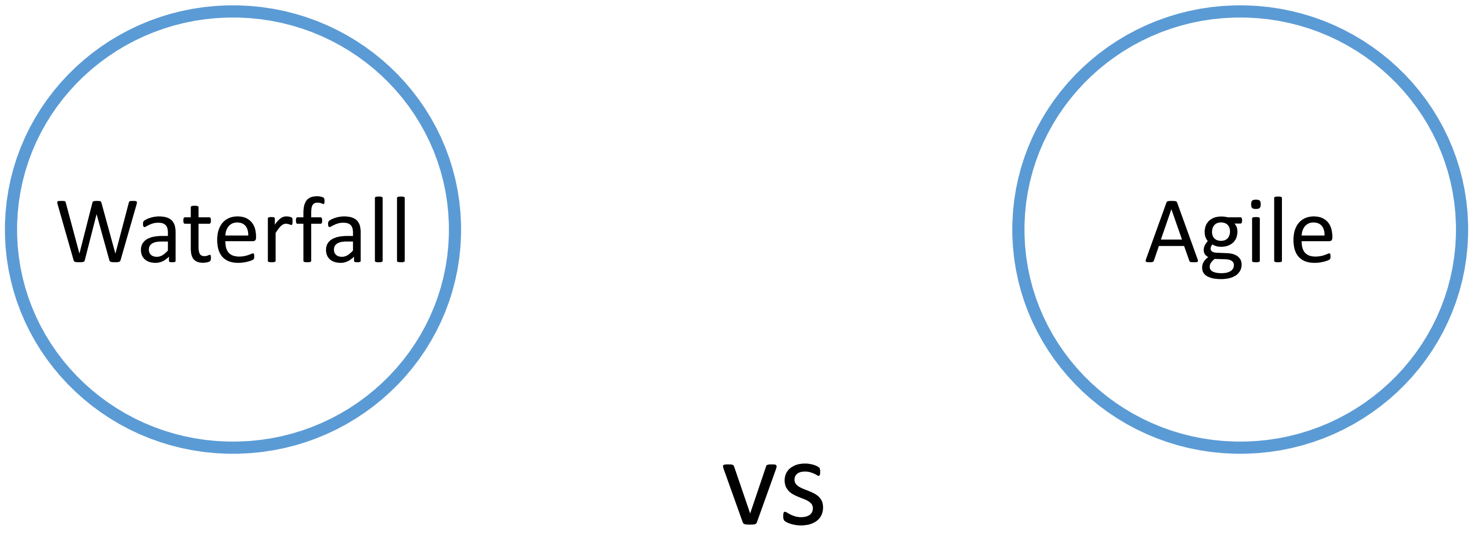
🚫

Agile = Good

Waterfall = Bad

Waterfall

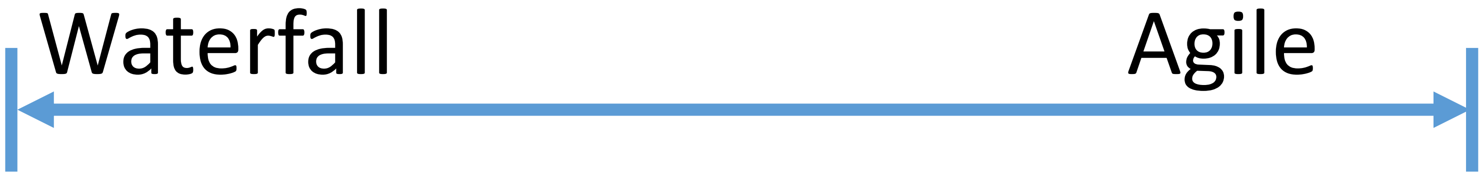
Agile



Waterfall

Agile

vs



Waterfall

Agile

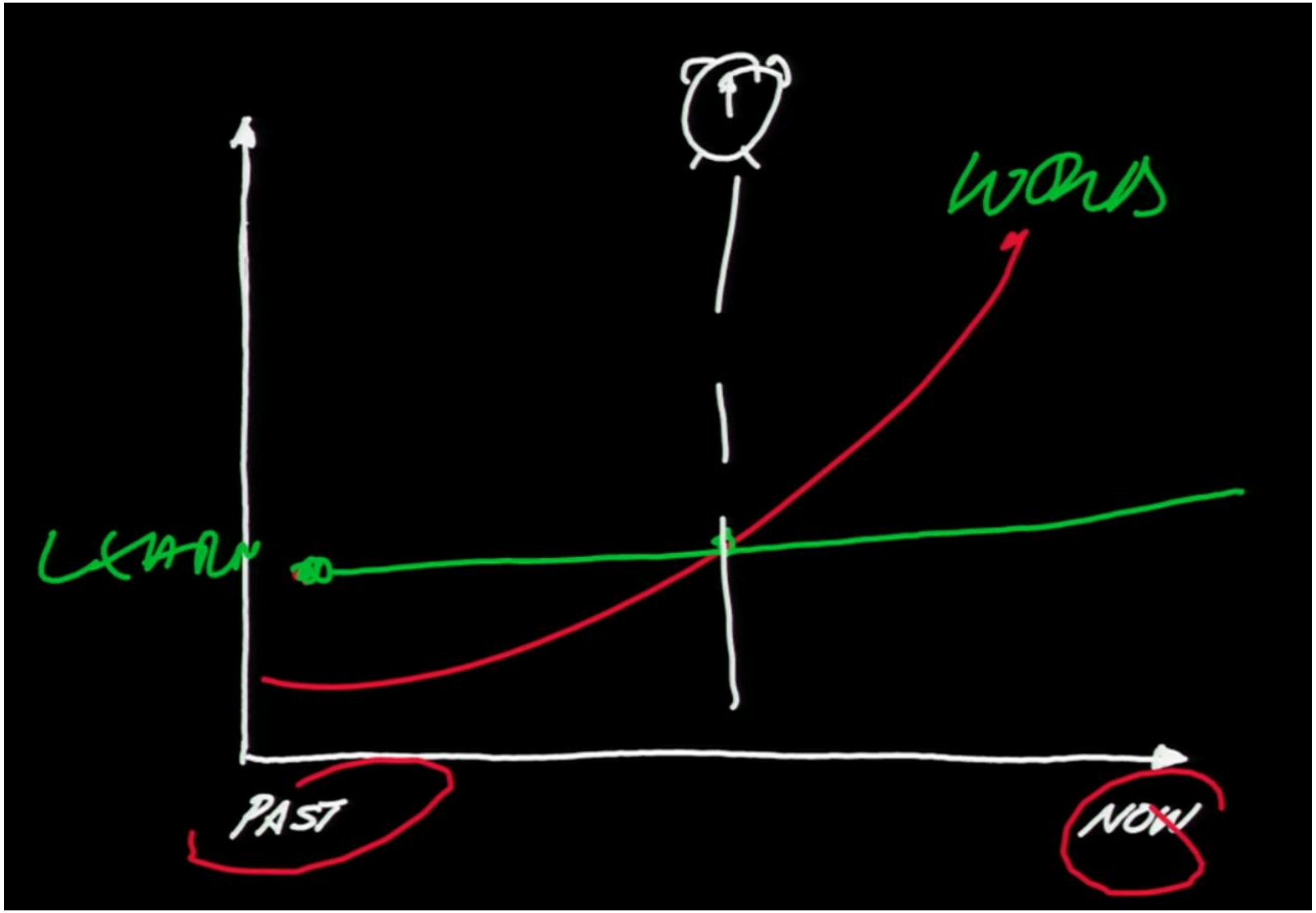
1. The World after Midnight

Source: www.ted.com

The World After Midnight

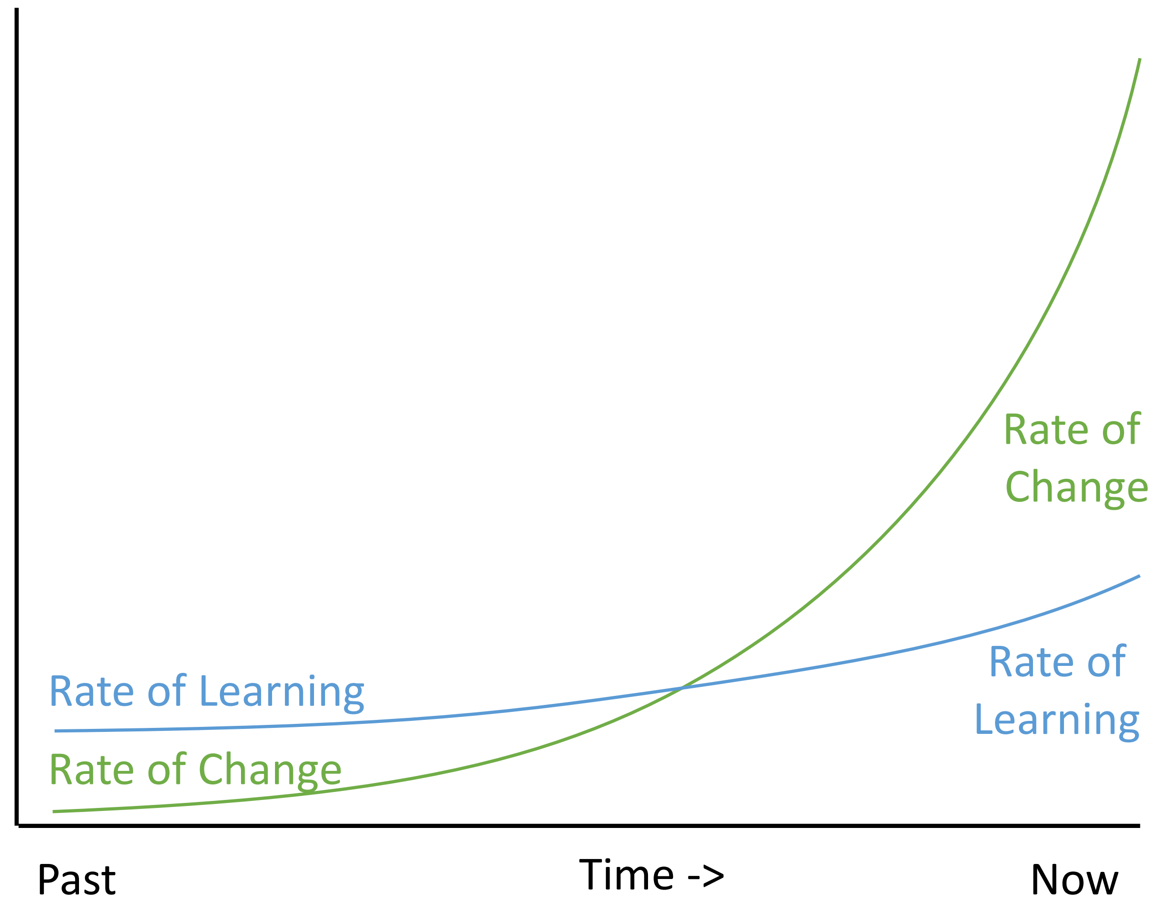
About fifteen years ago all the ‘Rules’ about how to run a business, organization, or government successfully, were changed or deleted and a completely new set of ‘Rules’ has been in operation ever since, which means that ***we keep acting rationally in response to a world we recognize and understand… but which no longer exists!***

- Eddie Obeng



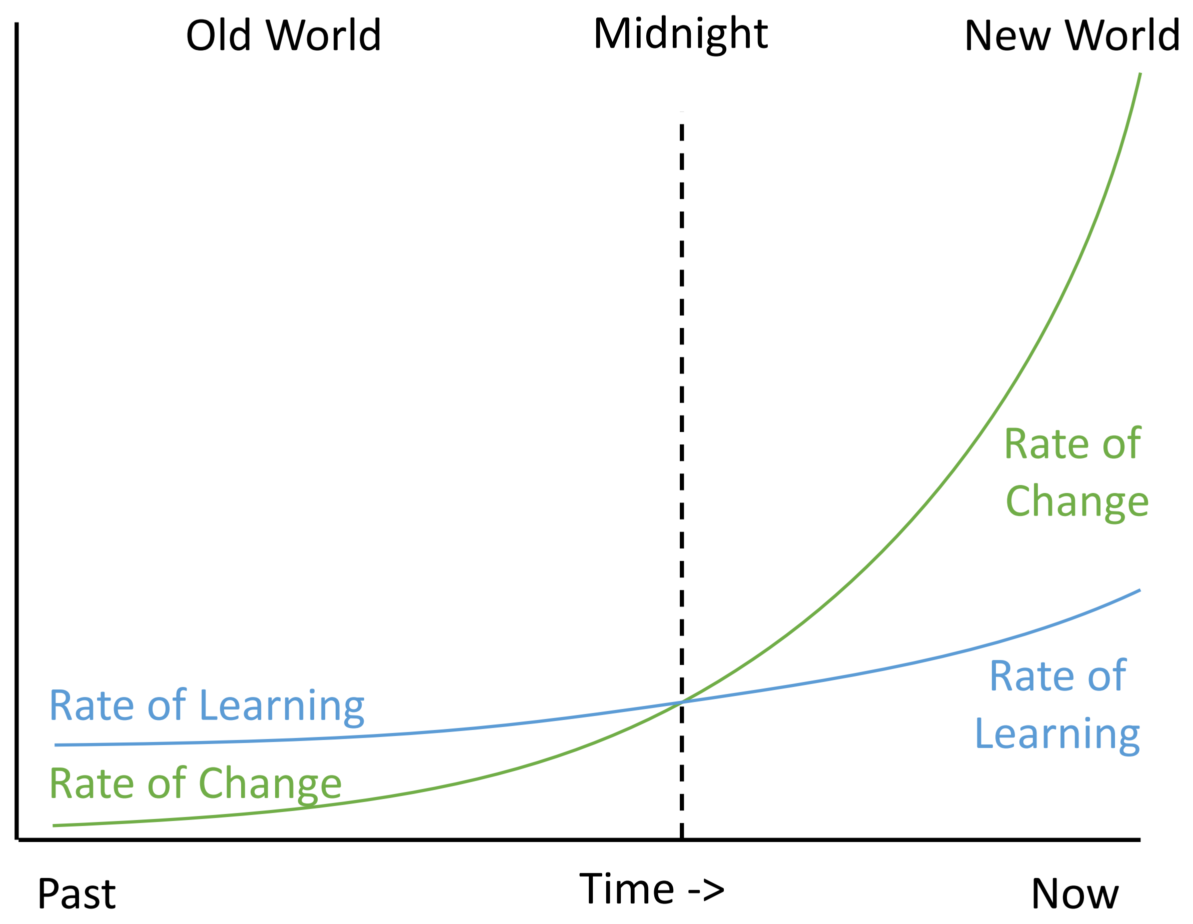
Source: http://www.ted.com/talks/eddie_obeng_smart_failure_for_a_fast_changing_world.html

The World after Midnight



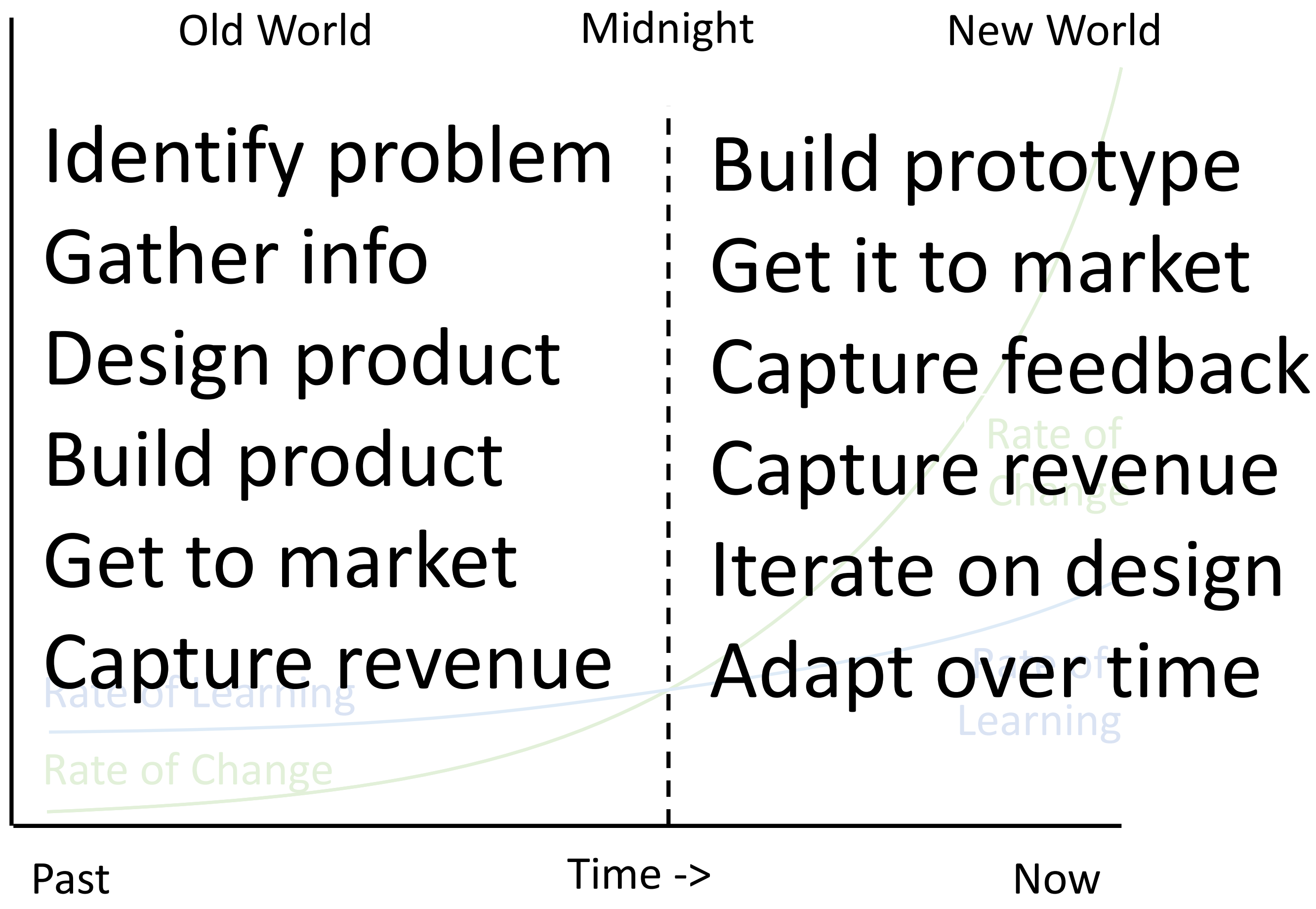
Original Source: http://pentacle.co.uk/Downloads/InnovationResources/Old%20New%20World.jpg

The World after Midnight



Original Source: http://pentacle.co.uk/Downloads/InnovationResources/Old%20New%20World.jpg

The World after Midnight

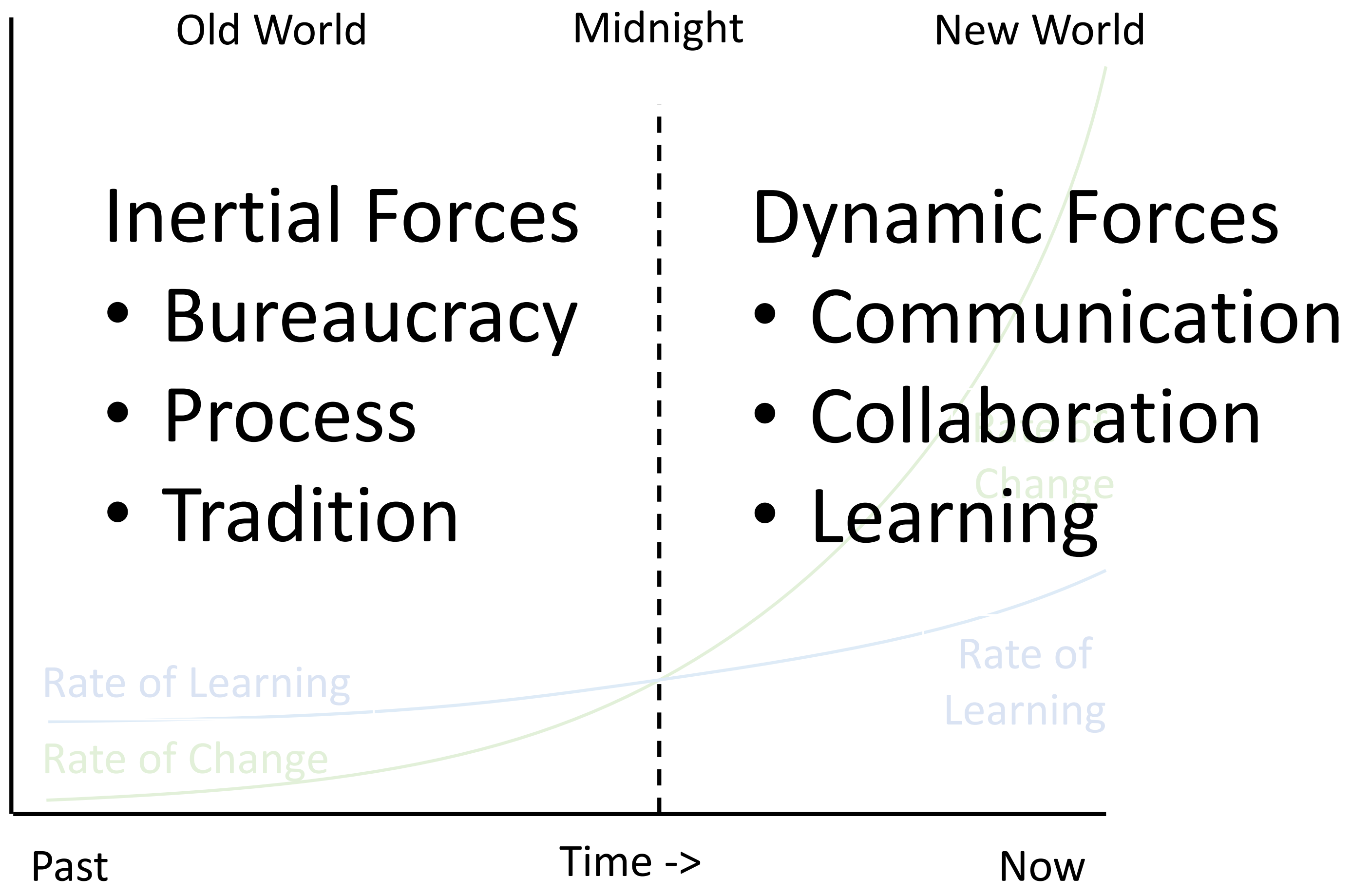


Identify problem
Gather info
Design product
Build product
Get to market
Capture revenue

Build prototype
Get it to market
Capture feedback
Capture revenue
Iterate on design
Adapt over time

Original Source: http://pentacle.co.uk/Downloads/InnovationResources/Old%20New%20World.jpg

The World after Midnight



Inertial Forces

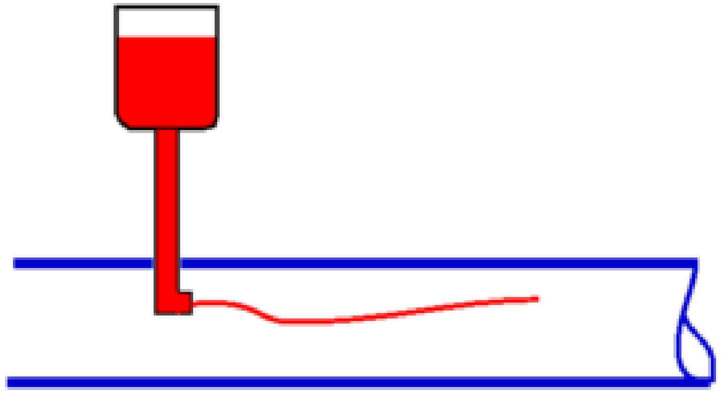
- Bureaucracy
- Process
- Tradition

Dynamic Forces

- Communication
- Collaboration
- Learning

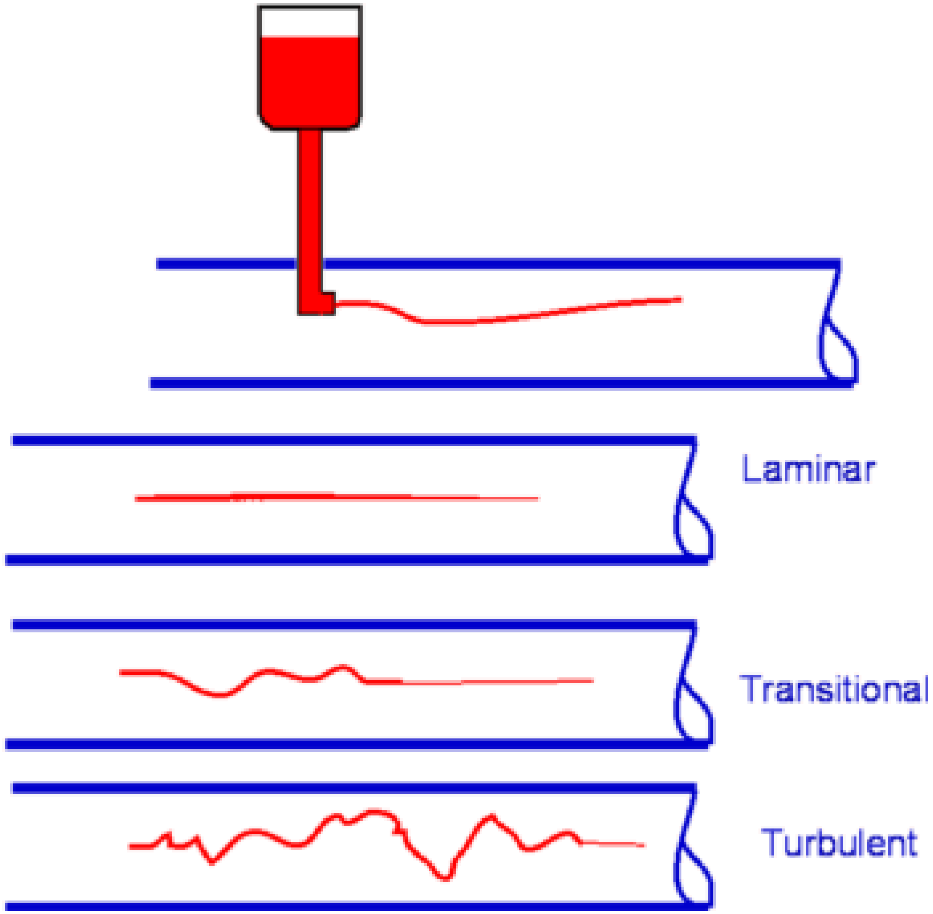
Original Source: http://pentacle.co.uk/Downloads/InnovationResources/Old%20New%20World.jpg

Laminar Flow vs. Turbulent Flow



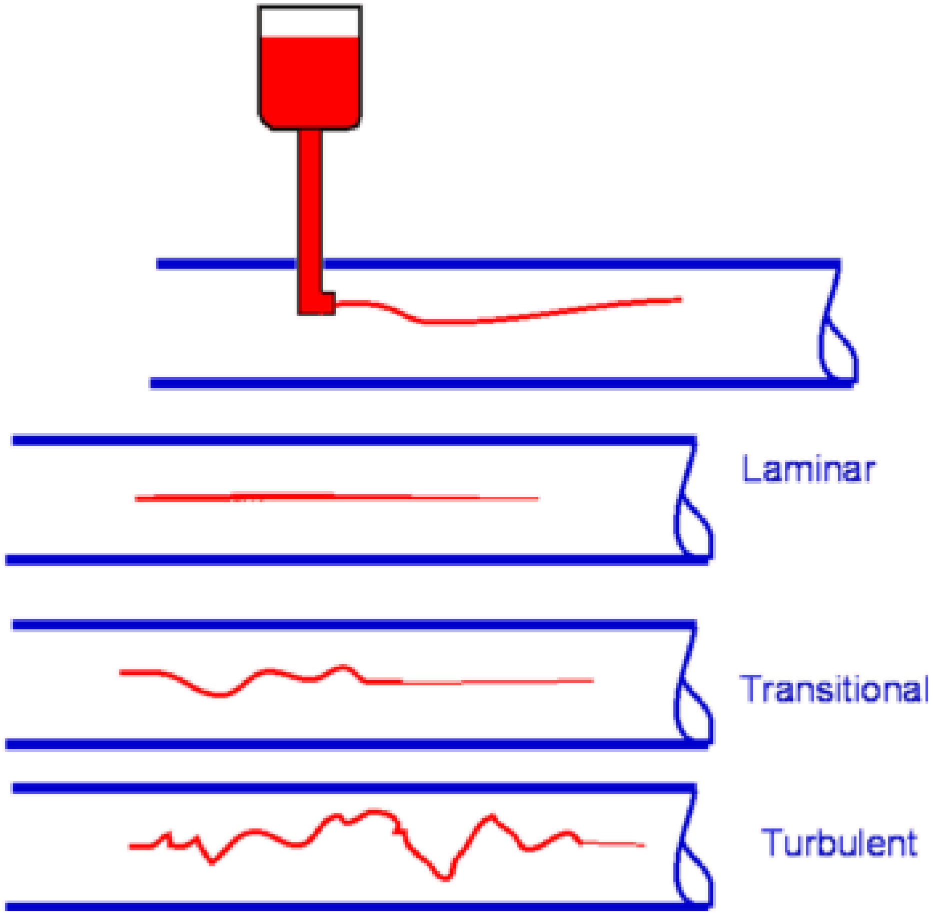
Source: http://www-mdp.eng.cam.ac.uk/web/library/enginfo/
aerothermal_dvd_only/aero/fprops/pipeflow/node8.html

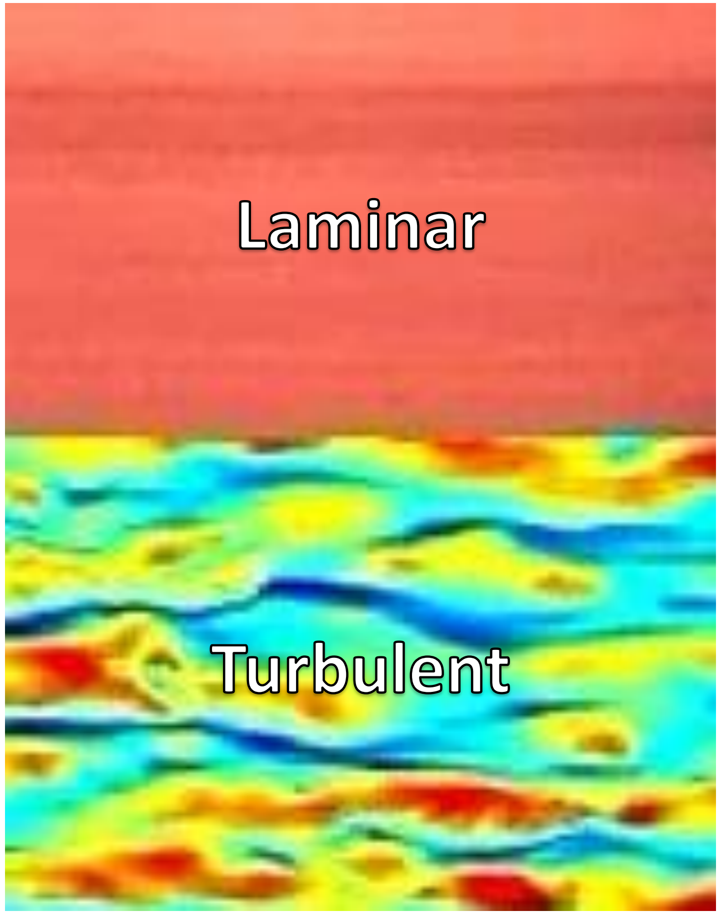
Laminar Flow vs. Turbulent Flow



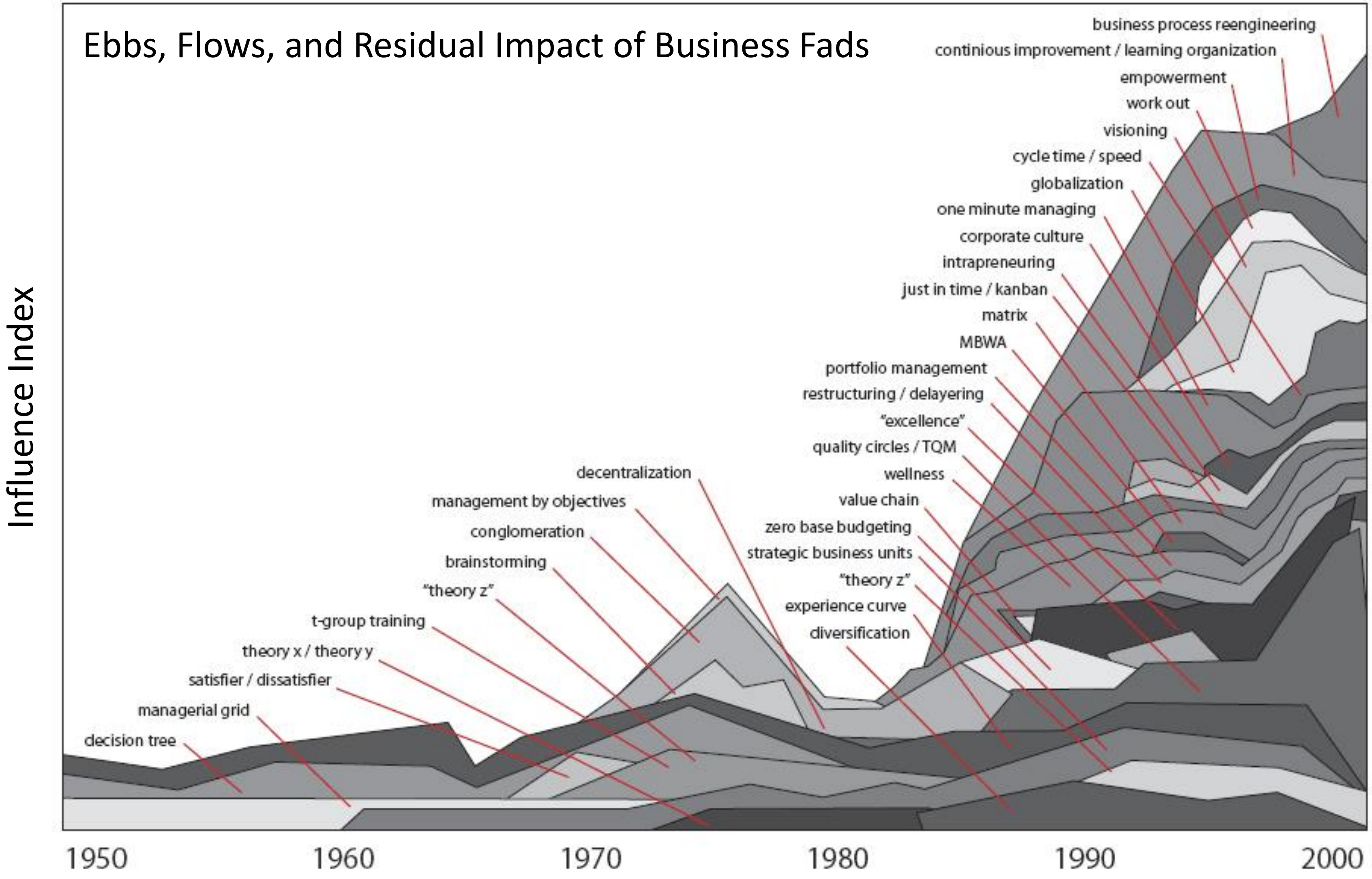
Source: http://www-mdp.eng.cam.ac.uk/web/library/enginfo/aerothermal_dvd_only/aero/fprops/pipeflow/node8.html

Laminar Flow vs. Turbulent Flow





Source: http://www-mdp.eng.cam.ac.uk/web/library/enginfo/aerothermal_dvd_only/aero/fprops/pipeflow/node8.html



Ebbs, Flows, and Residual Impact of Business Fads

Source: 'The Ebbs, Flows and Residual Impact of Business Fads 1950 – 1995' by R. Pascale

Why is this important?

Problem	**Solution**
World has changed	Adapt to new physics
Markets change rapidly	Faster time-to-market
Requirements change rapidly	Better response to change
High degree of uncertainty	Continuous and rapid feedback

Agile is very well suited to operate in the physics of this new world!

2. Inverted Constraints

Four Levers of Software Development

Scope

Resources

Schedule

Quality

Source: http://farm6.staticflickr.com/5300/5521479079_36815225e4_z.jpg

Four Levers of Software Development

Working software

Max value

Min cost

Source: http://farm6.staticflickr.com/5300/5521479079_36815225e4_z.jpg

Constraints

Restriction on freedom

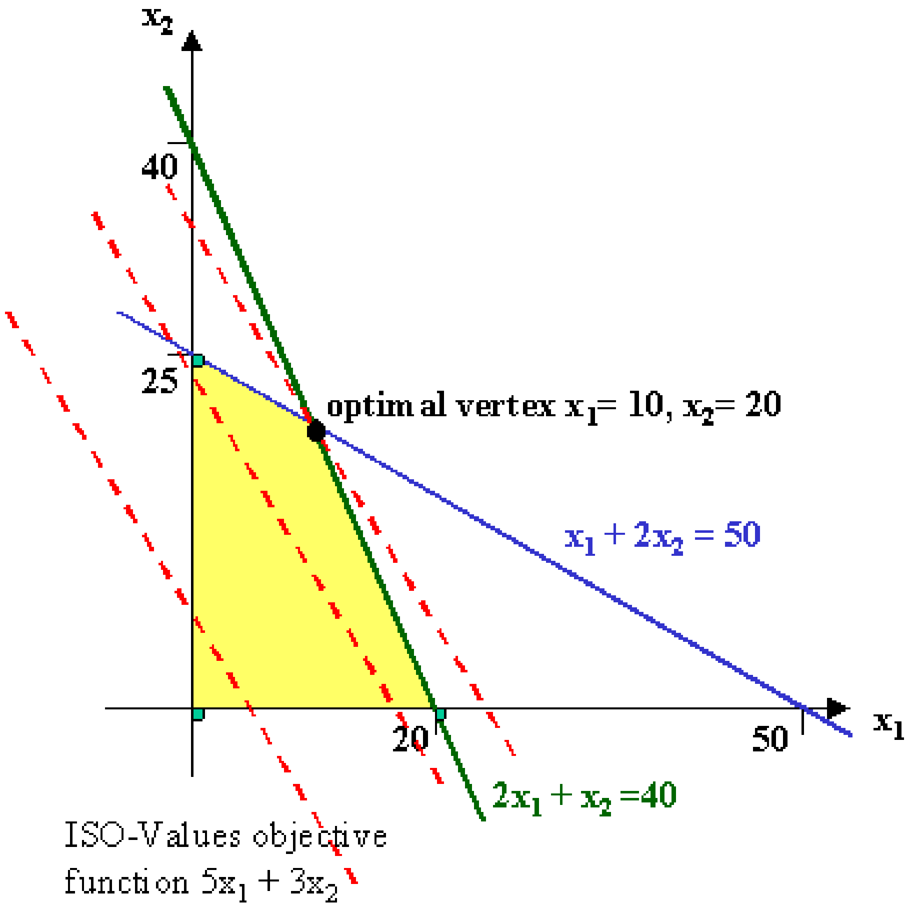
Prevents achieving goal

Examples

- Time
- Money
- Talent

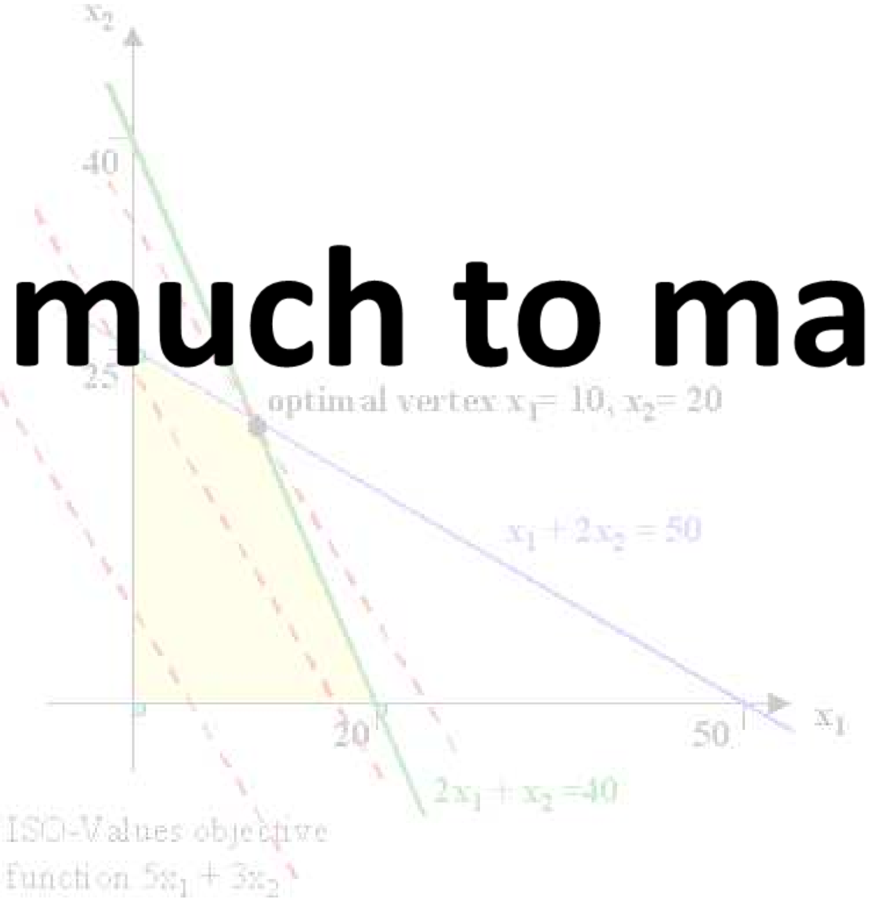
Source: http://www.myspaceantics.com

Constrained Optimization



Source: http://home.ubalt.edu/ntsbarsh/business-stat/opre/partVIII.htm

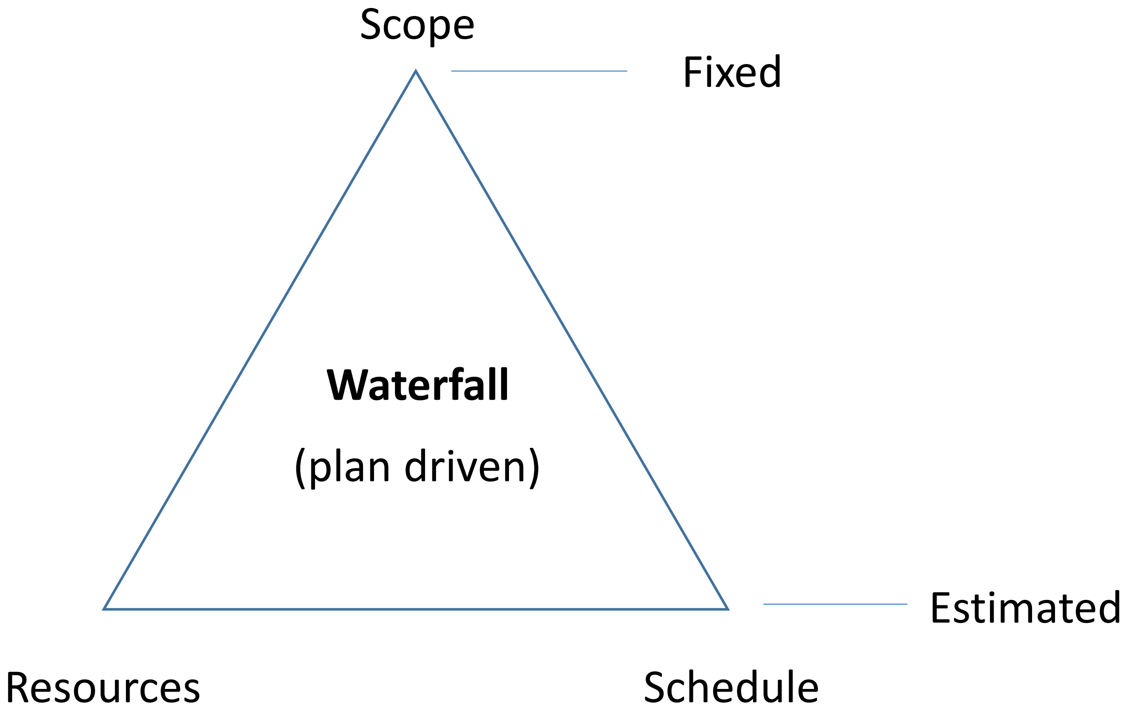
Constrained Optimization



Too much to math!

Source: http://home.ubalt.edu/ntsbarsh/business-stat/opre/partVIII.htm

Waterfall Constraints



Scope

Fixed

Waterfall

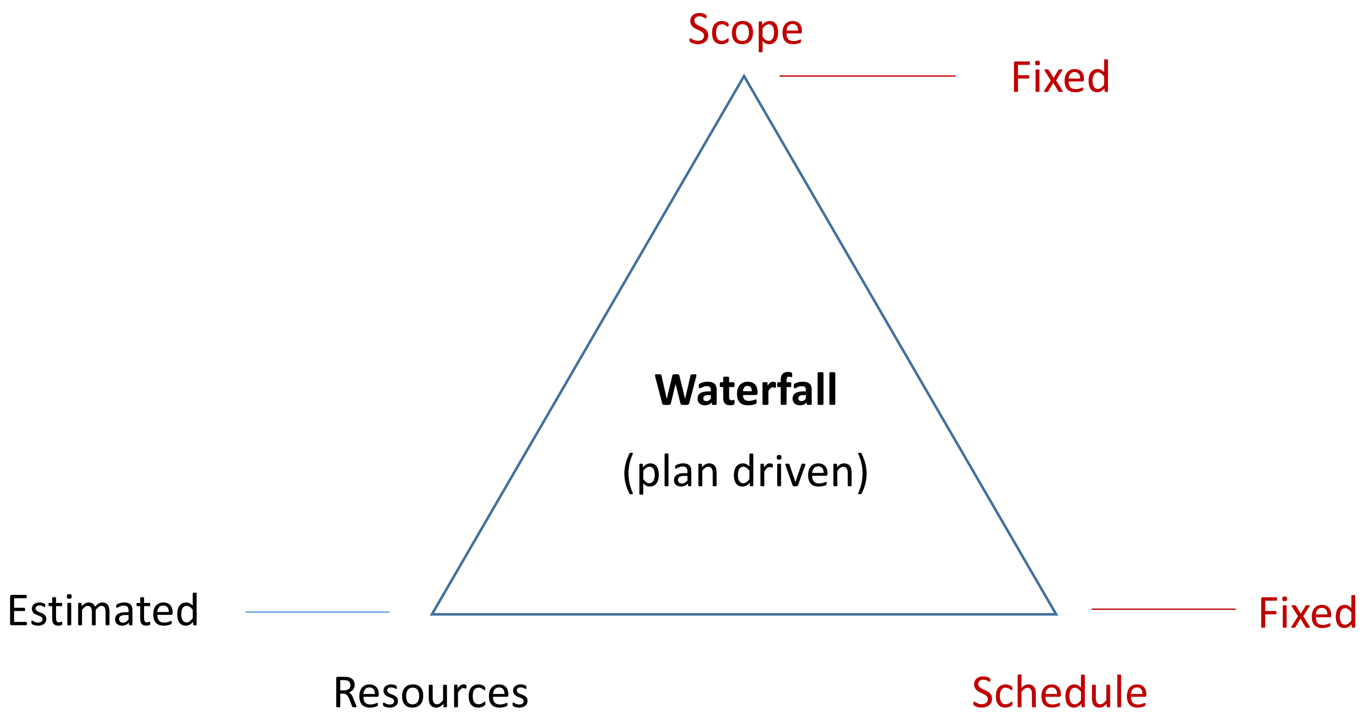
(plan driven)

Estimated

Resources

Schedule

Waterfall Constraints



Scope

Fixed

Waterfall

(plan driven)

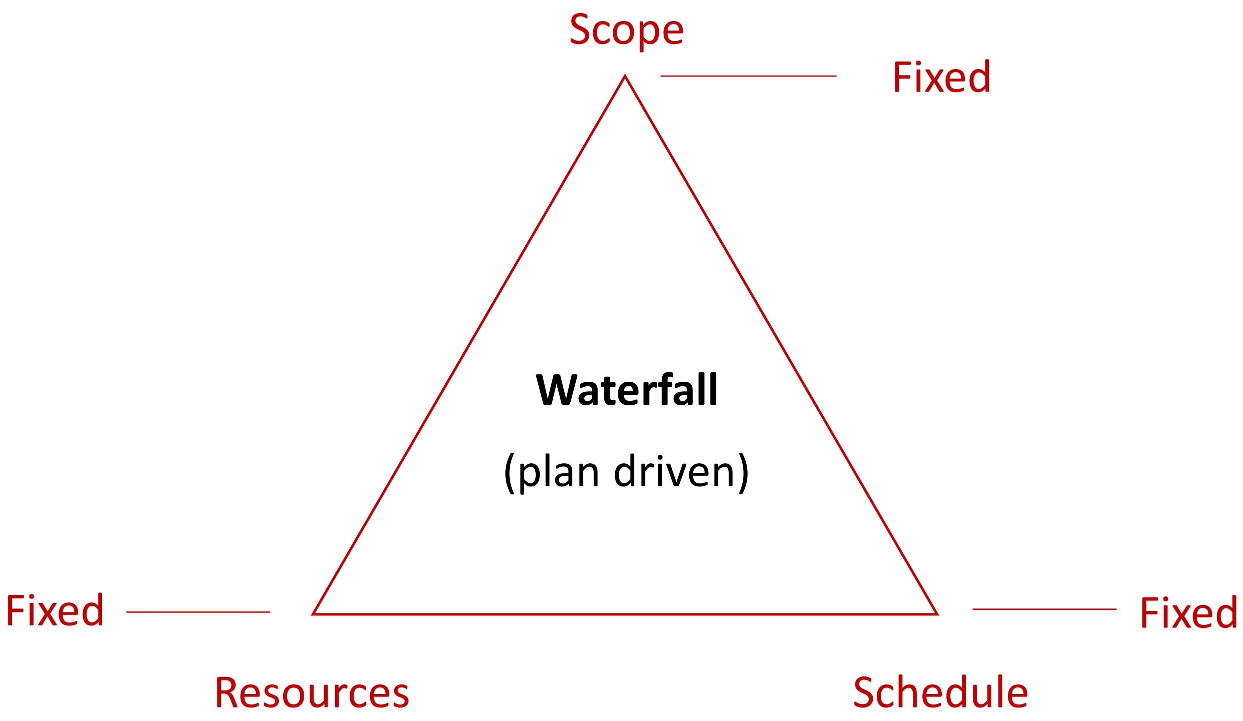
Estimated

Fixed

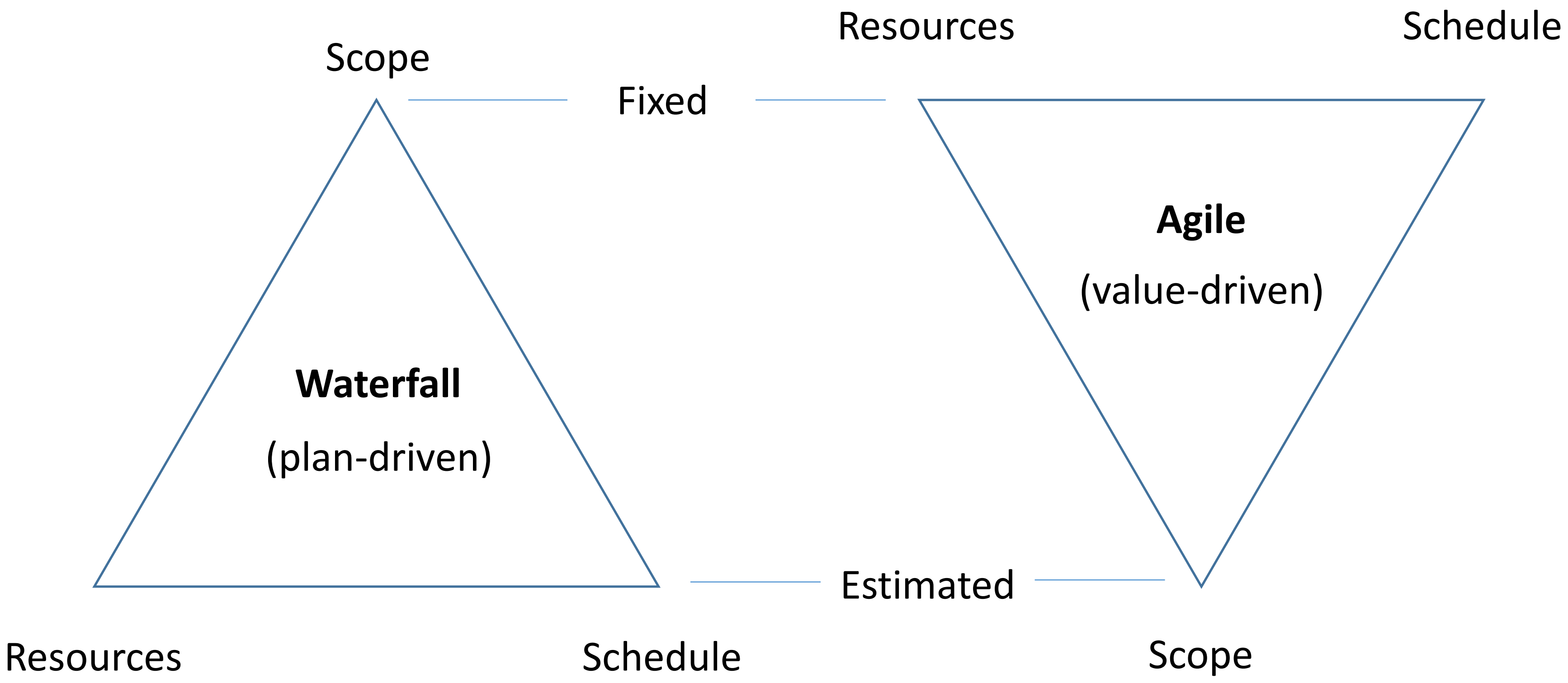
Resources

Schedule

Waterfall Constraints



Agile Constraints



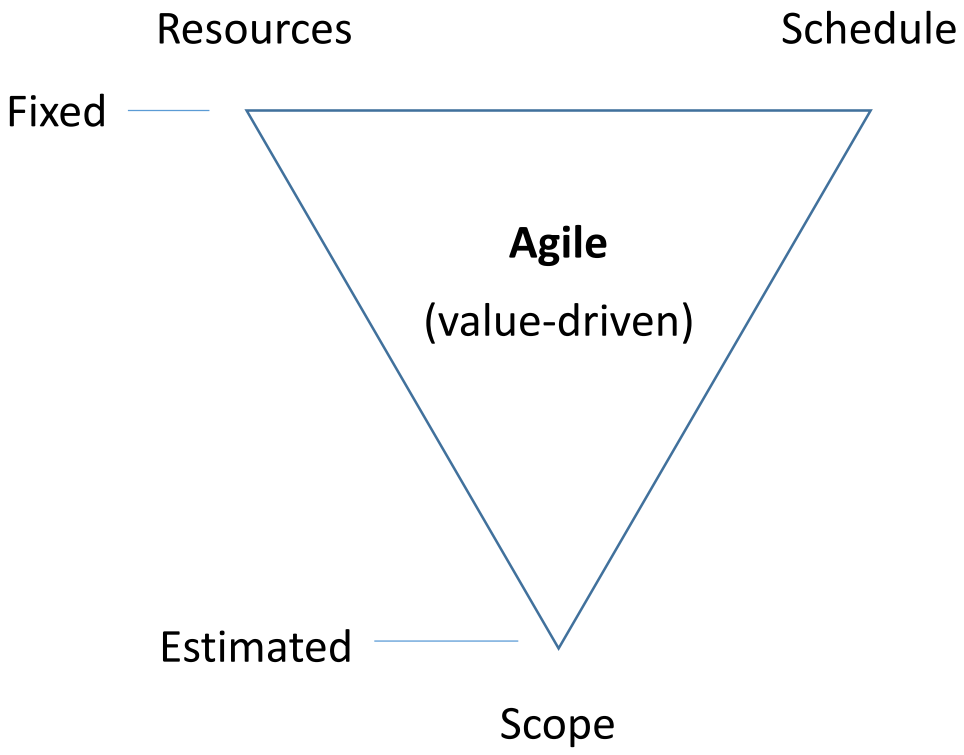
Agile Constraints

Fixed team size

Fixed releases

Estimated features

Team controls quality



Why is This Important?

Problem	**Solution**
Mythical man-month	Limit team size
Slipping release dates	Fix schedule
Scope creep	Estimate scope
Technical debt	Protect quality

Agile is more flexible

3. Prioritizing Value

Quick Lesson in Economics

1. Return on Investment
2. Pareto Principle
3. Opportunity Cost

Source: http://myhomeworkhelp.com/economics-homework-help/

Return on Investment

$$ROI = \frac{Value\ -\ Cost}{Cost}$$

High ROI => lots of value

Low ROI => some value

Neg. ROI => lost value

Return on Investment

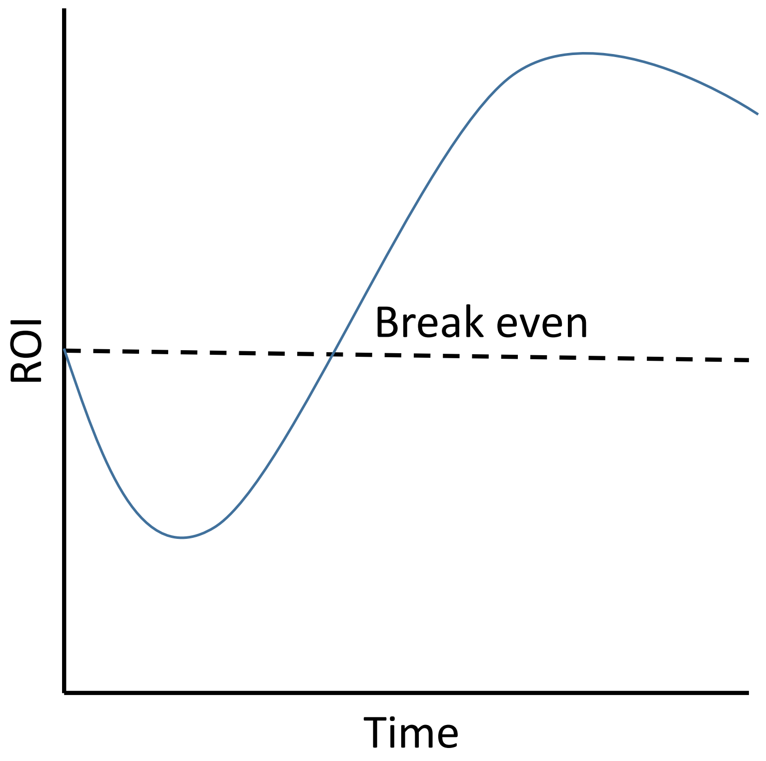
$$ROI = \frac{Value\ -\ Cost}{Cost}$$

High ROI => lots of value

Low ROI => some value

Neg. ROI => lost value

ROI Curve for an Investment



Return on Investment

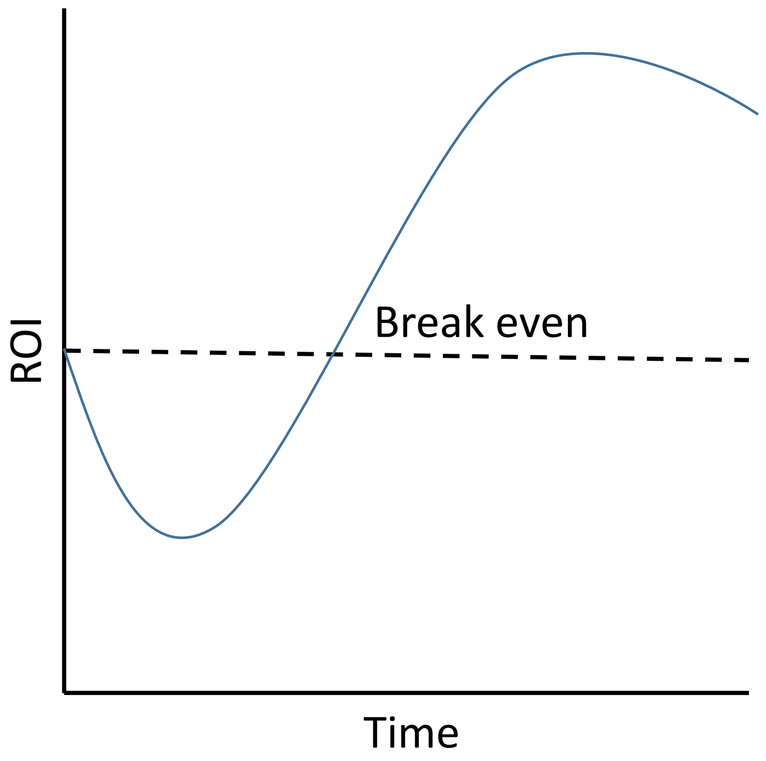
Each feature has ROI

- Cost to develop
- Value to business

Project ROI is sum of feature ROIs

Goal is to maximize ROI

ROI Curve for an Investment

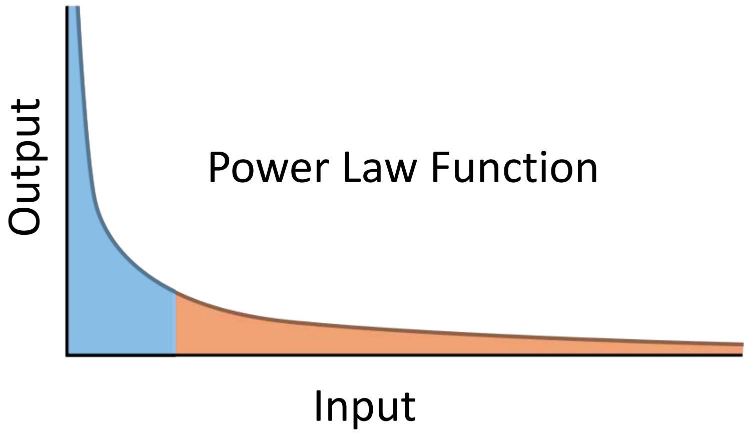


Pareto Principle

80/20 rule

Power law function

Diminishing marginal returns

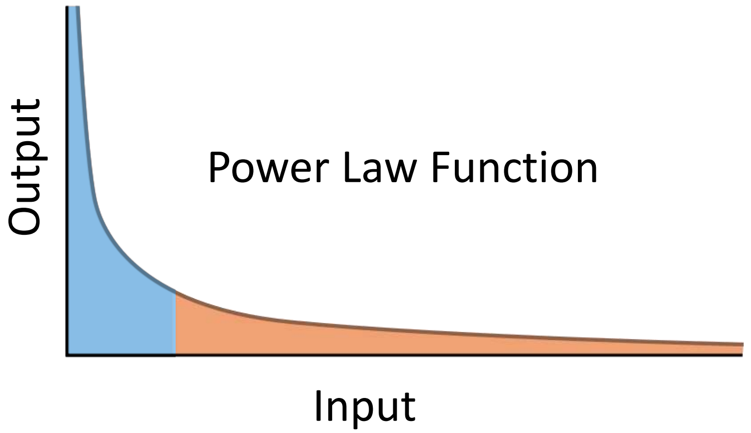


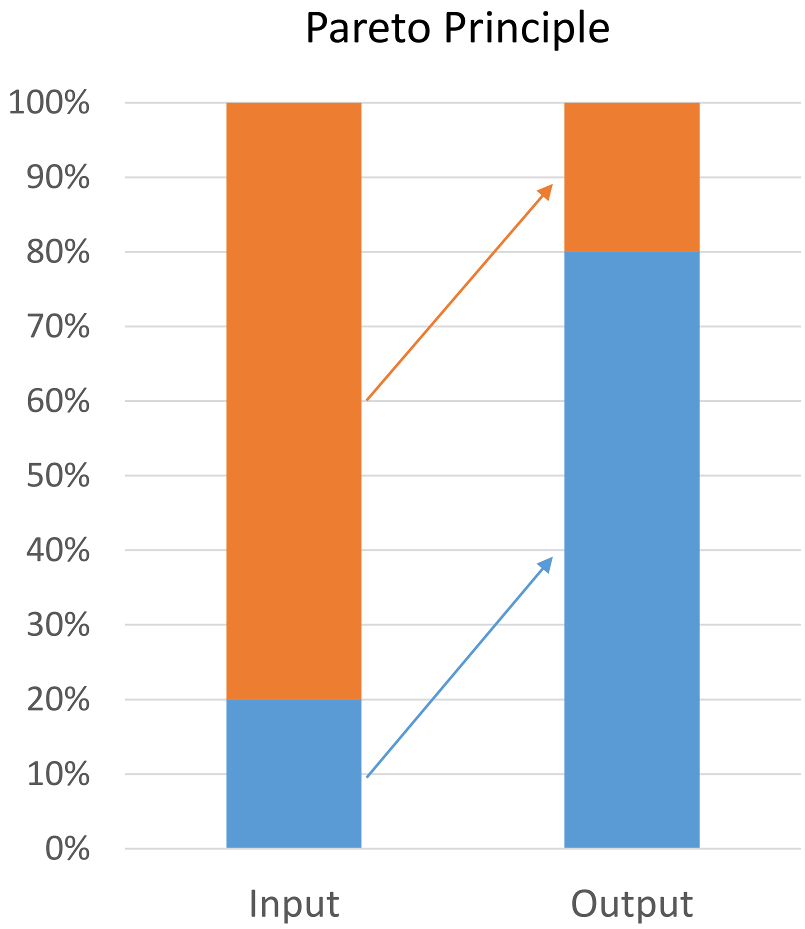
Pareto Principle

80/20 rule

Power law function

Diminishing marginal returns





Pareto Principle of Software Feature Usage

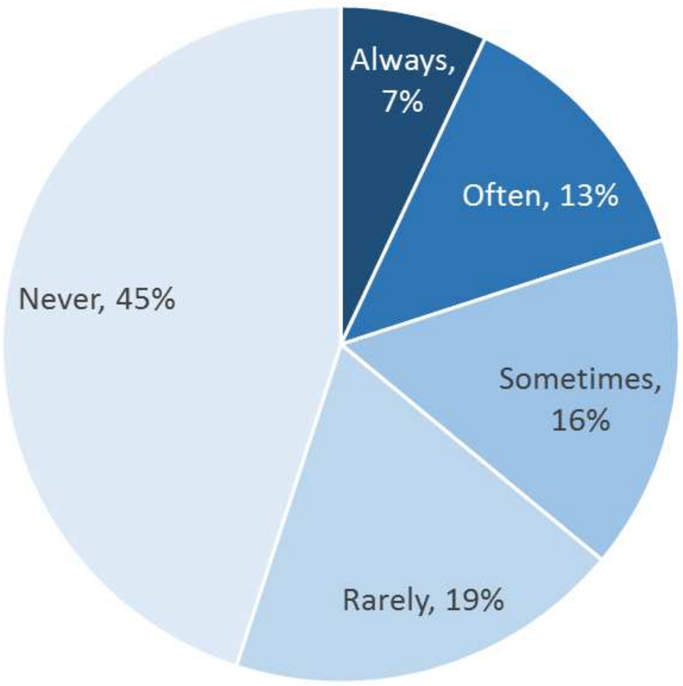
Features

- 20% of features
- 80% of value

Traditional software is

- 20% high-value features
- 80% low-value features

Software Features Used



Source: Standish Group

Opportunity Cost

Source: http://www.ethicurean.com/2009/03/03/free-lunch-program-in-new-england/

Opportunity Cost

Cost of foregone alternative options

True cost = explicit cost + implicit cost

Must be included in cost-benefit analysis

Stu's Views
© 2004 Stu All Rights Reserved www.STUS.com
S U
I hope you appreciate that each "walk" costs $175 of my billable time.
I hope you appreciate that I'm your only friend.

Source: http://www.stus.com/

Prioritizing Features by Business Value

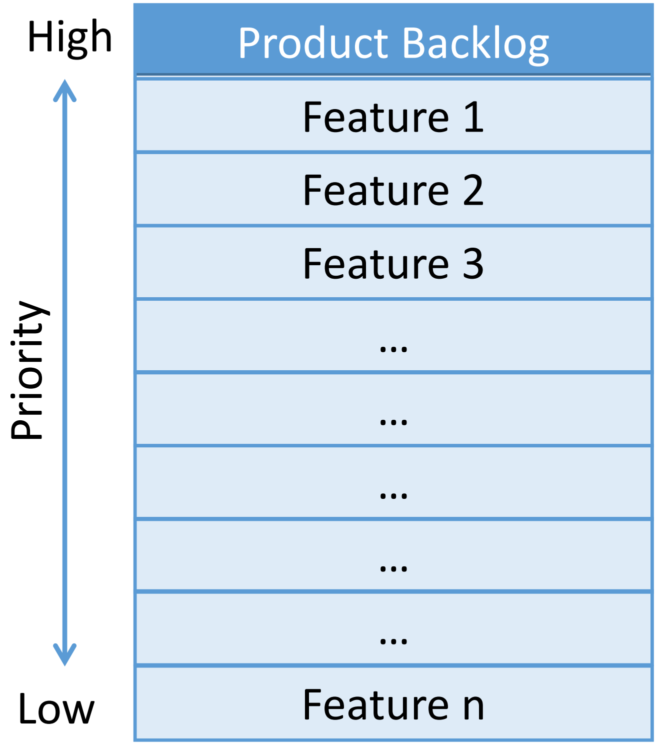
Product backlog

List of features

Ordered by business value

Highest priority on top

Create and deliver in order



High

Priority

Low

Product Backlog
Feature 1
Feature 2
Feature 3
…
…
…
…
…
Feature n

Why is This Important?

Problem

Need to maximize ROI

Low-value features

Opportunity cost

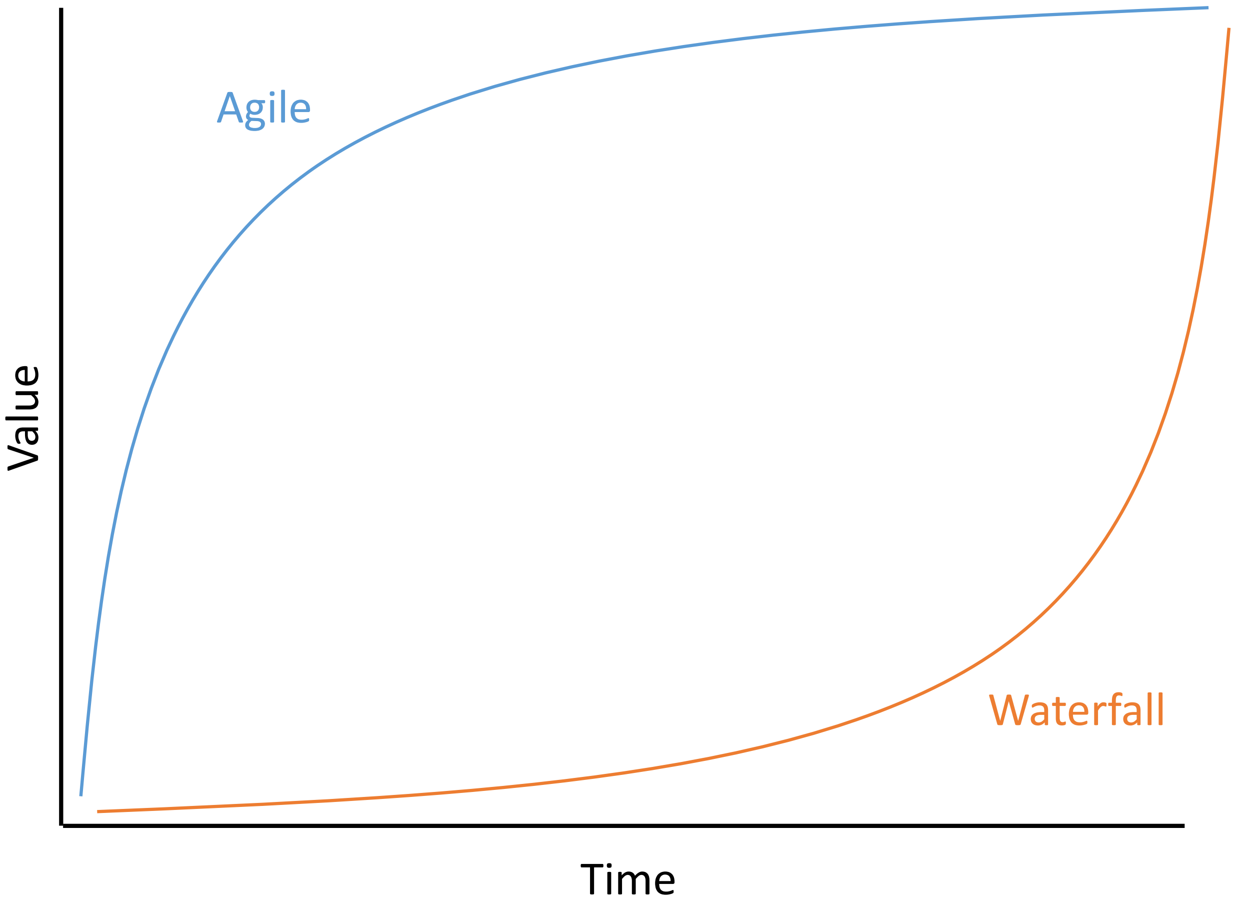
Solution

Prioritize features by ROI

Deliver highest-value first

Prioritize features relative

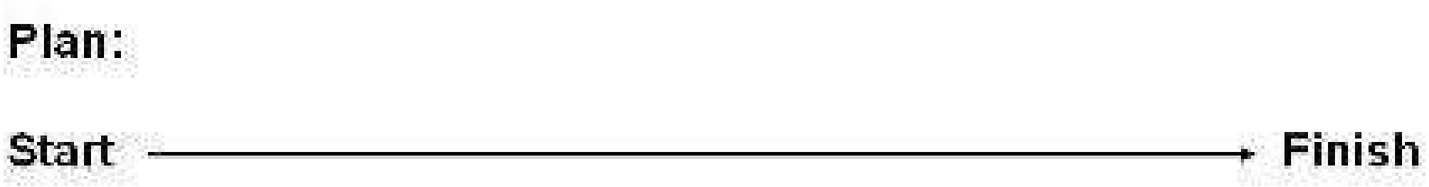
Agile Produces More Value



Original Source: http://www.versionone.com/Agile101/Agile-Software-Development-Benefits/

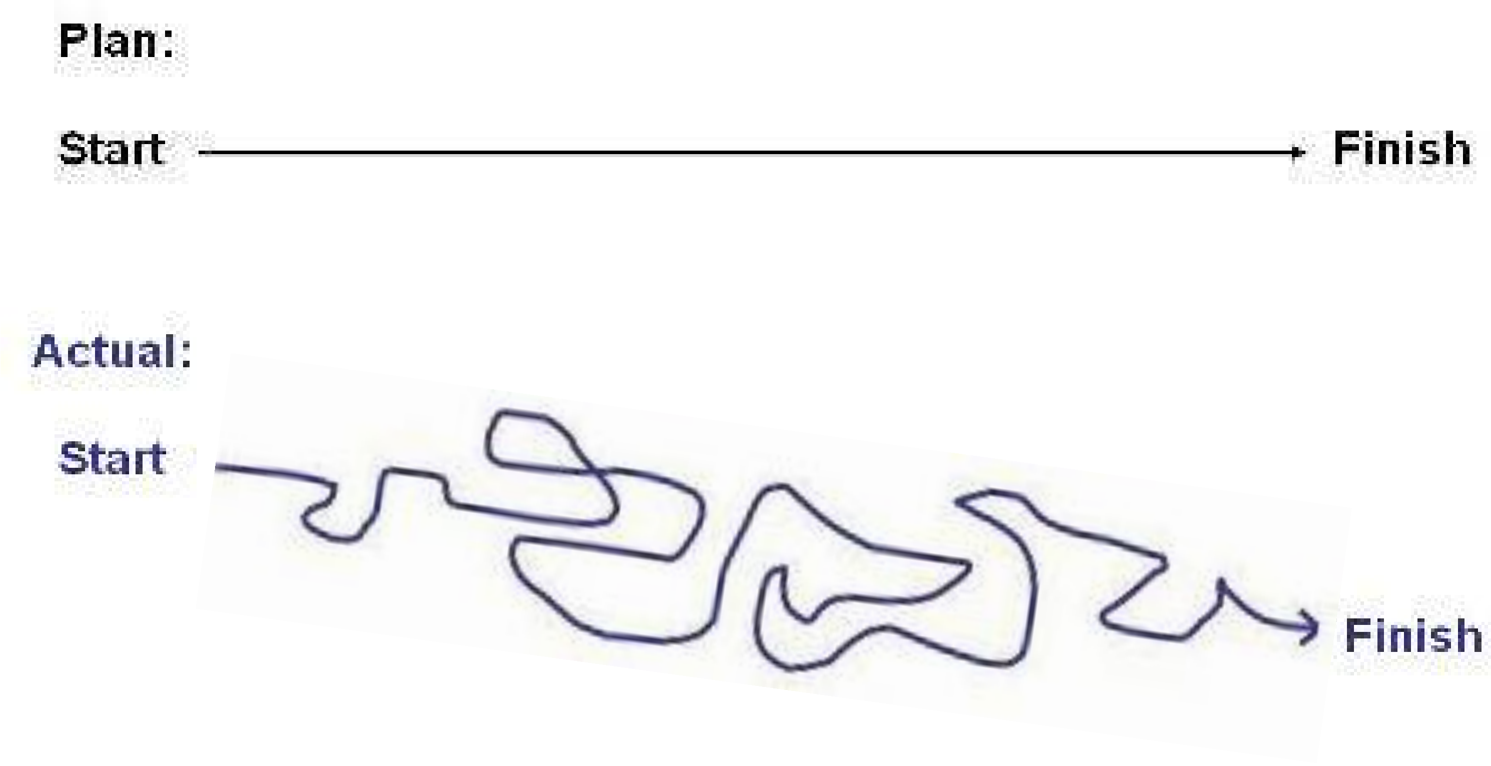
4. Embracing Change

Waterfall’s Key Assumption



Source: Doug DeCarlo – eXtreme Project Management

Waterfall’s Key Assumption



Source: Doug DeCarlo – eXtreme Project Management

Waterfall Assumptions

Users actually know what they want

Markets will not change during development

There is nothing new or unknown

Technology is stable and mature

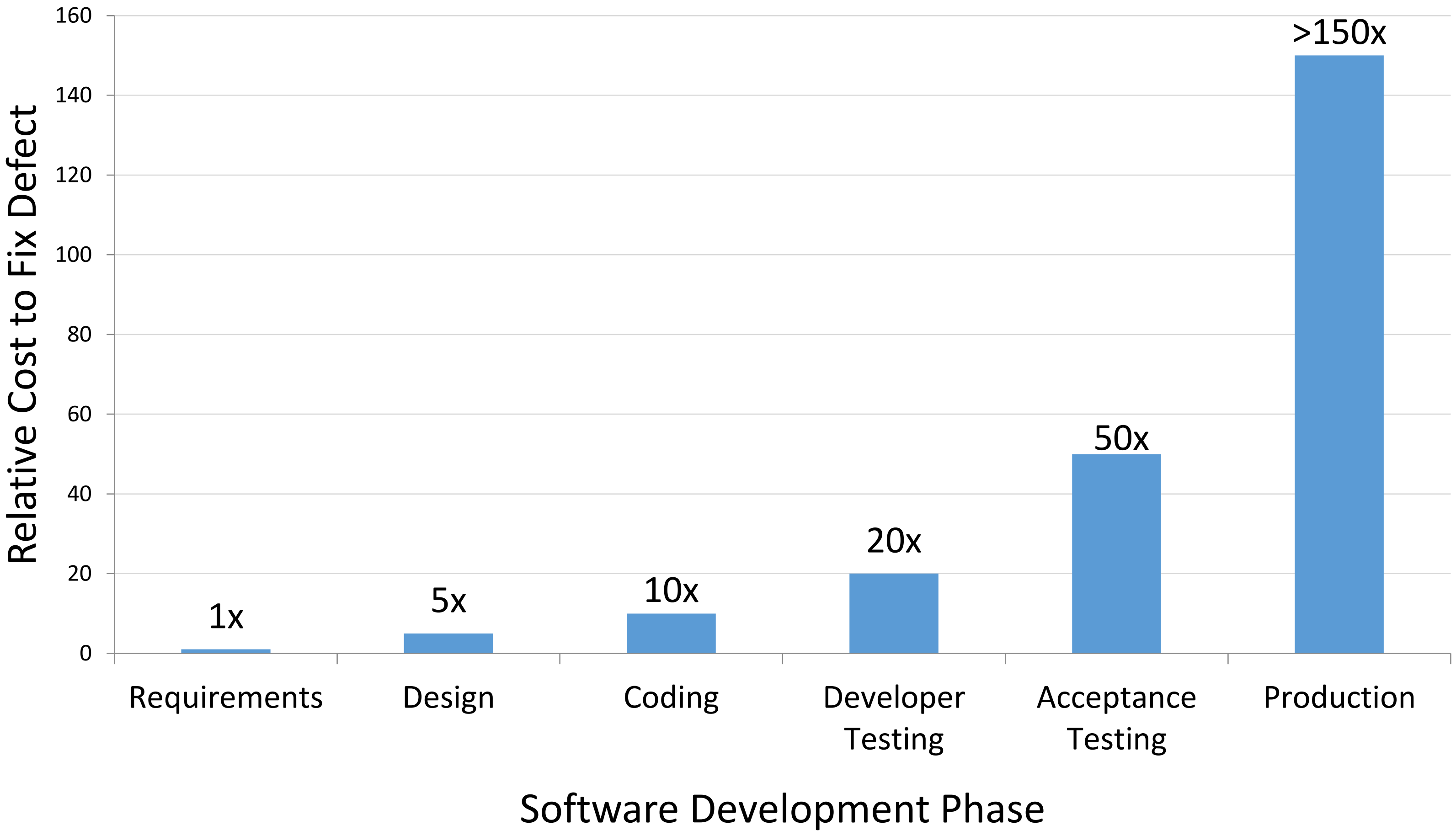
All of the pieces will fit together in the end

Waterfall Reality

Requirements are not stable

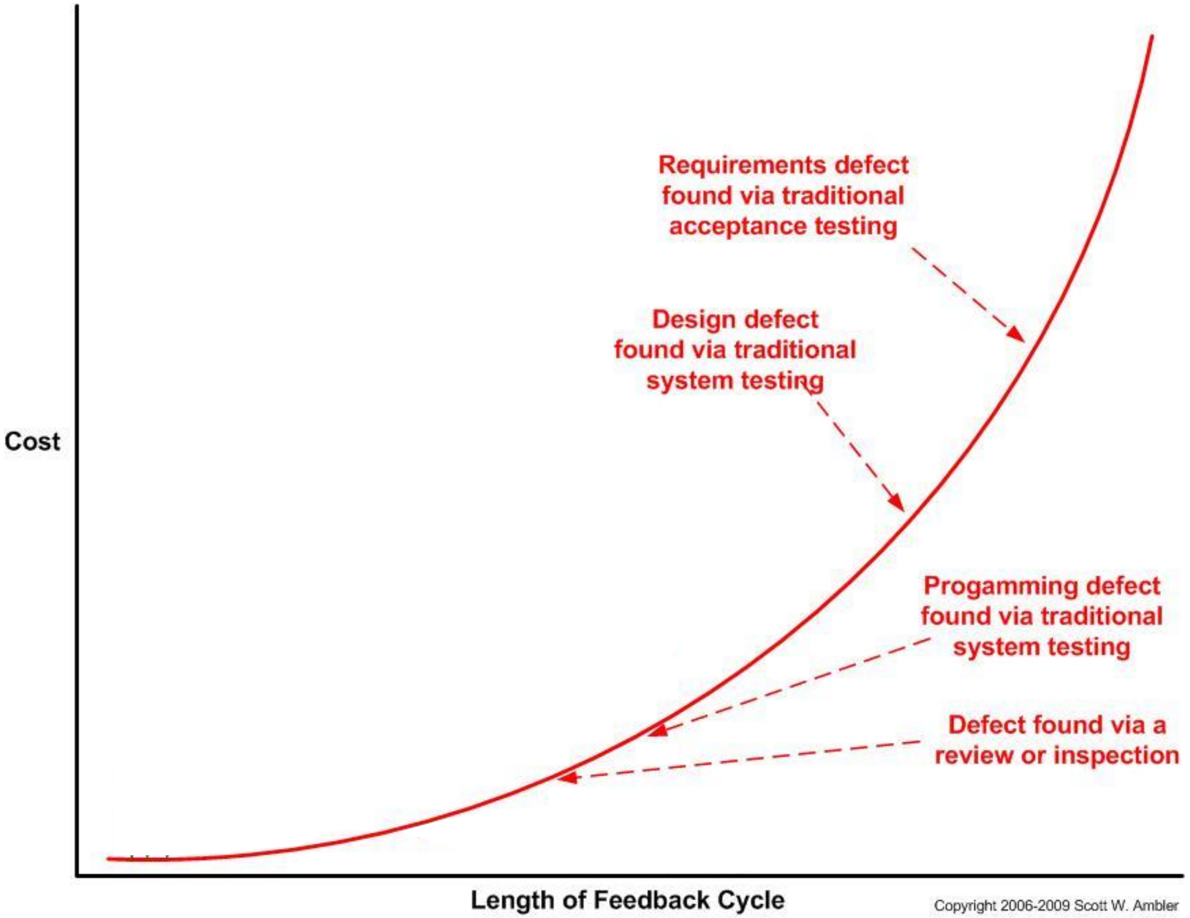
Requirements are just assumptions

Cost of Fixing Defects in Waterfall



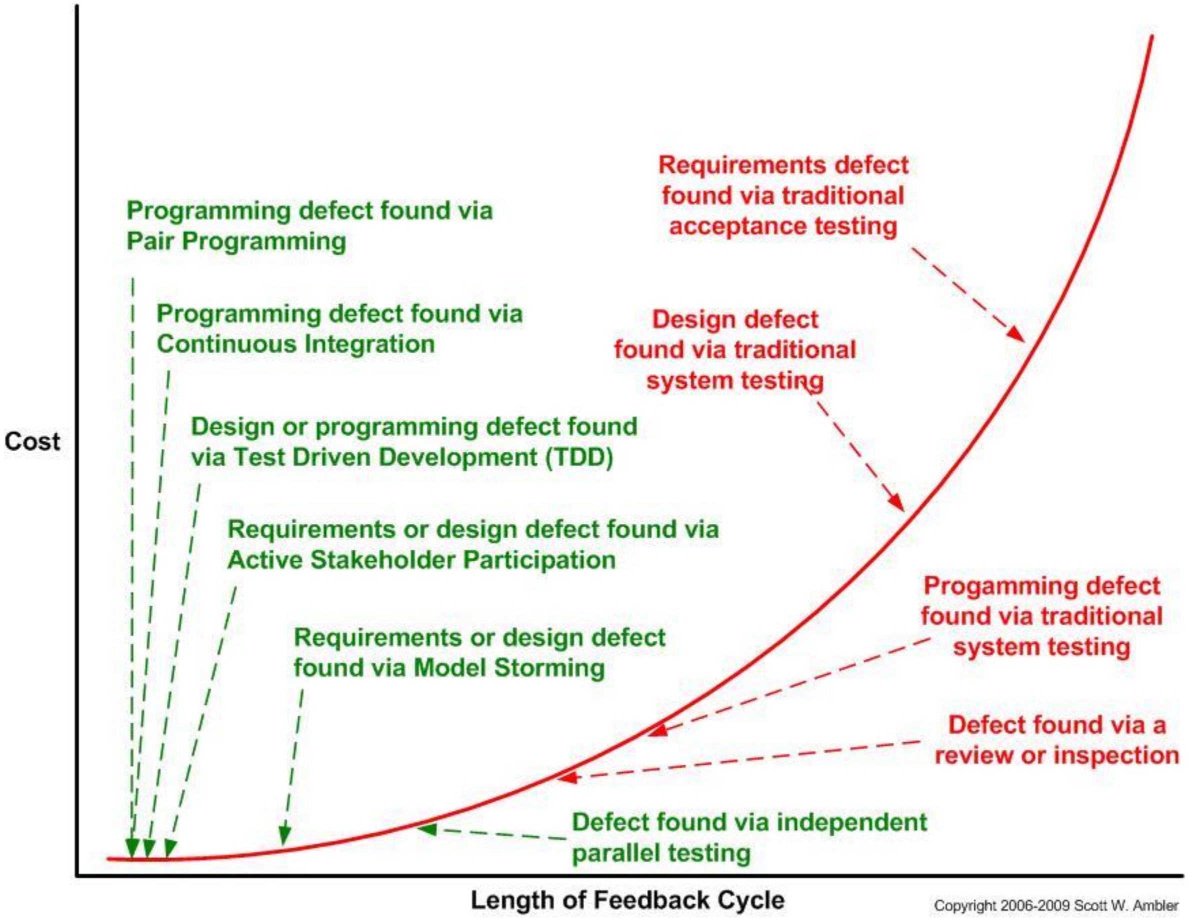
Original Source: Barry Boehm, “Equity Keynote Address” March 19, 2007

Finding Defects in Waterfall



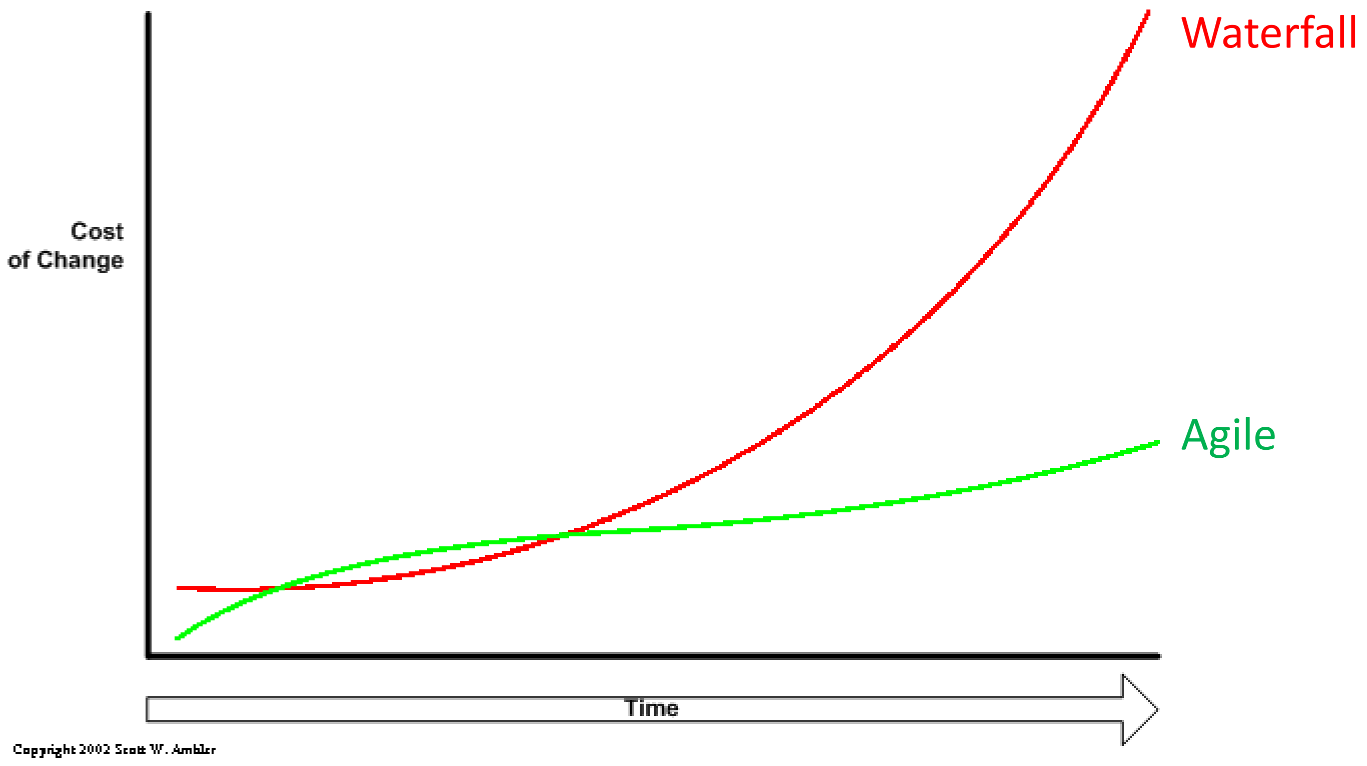
http://www.agilemodeling.com/essays/costofchange.htm

Finding Defects in Agile



http://www.agilemodeling.com/essays/costofchange.htm

Cost of Change in Agile



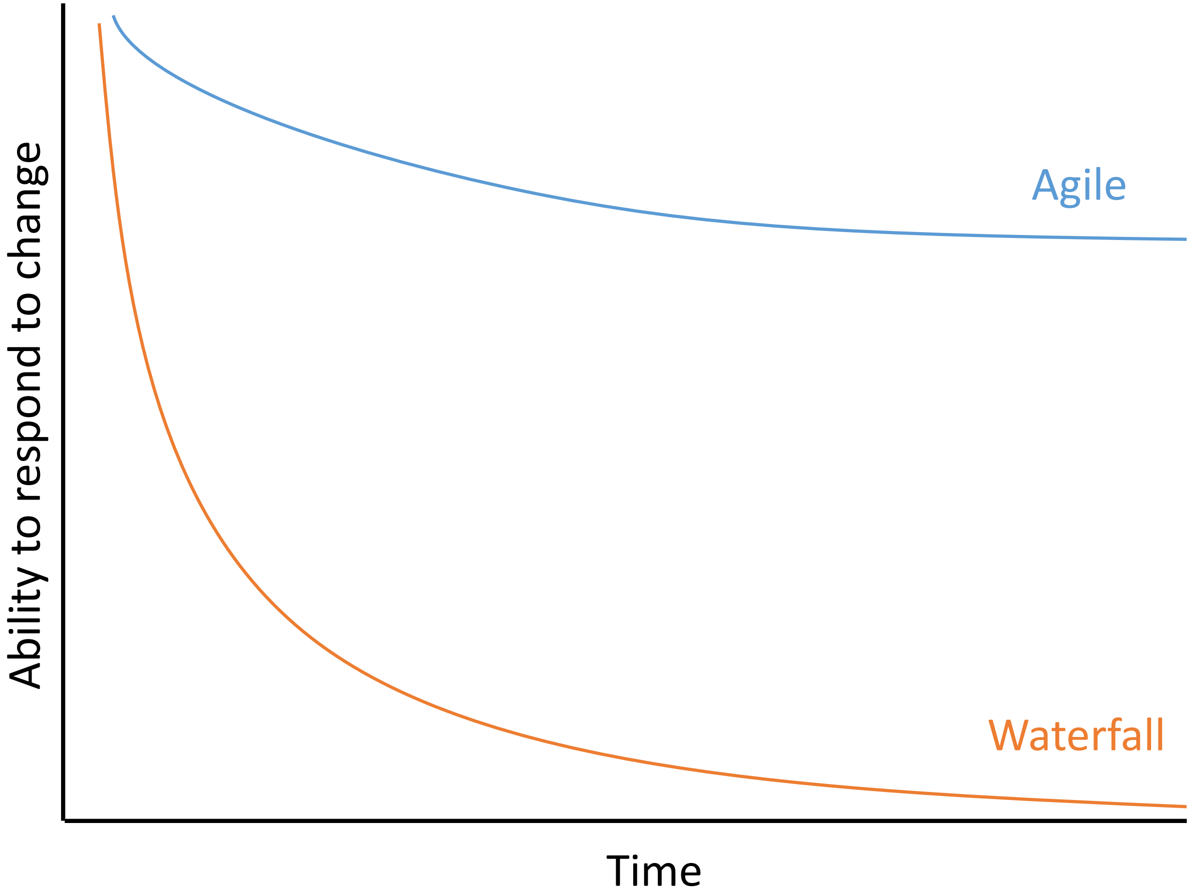
Copyright 2002 Scott W. Ambler

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

Why is This Important?

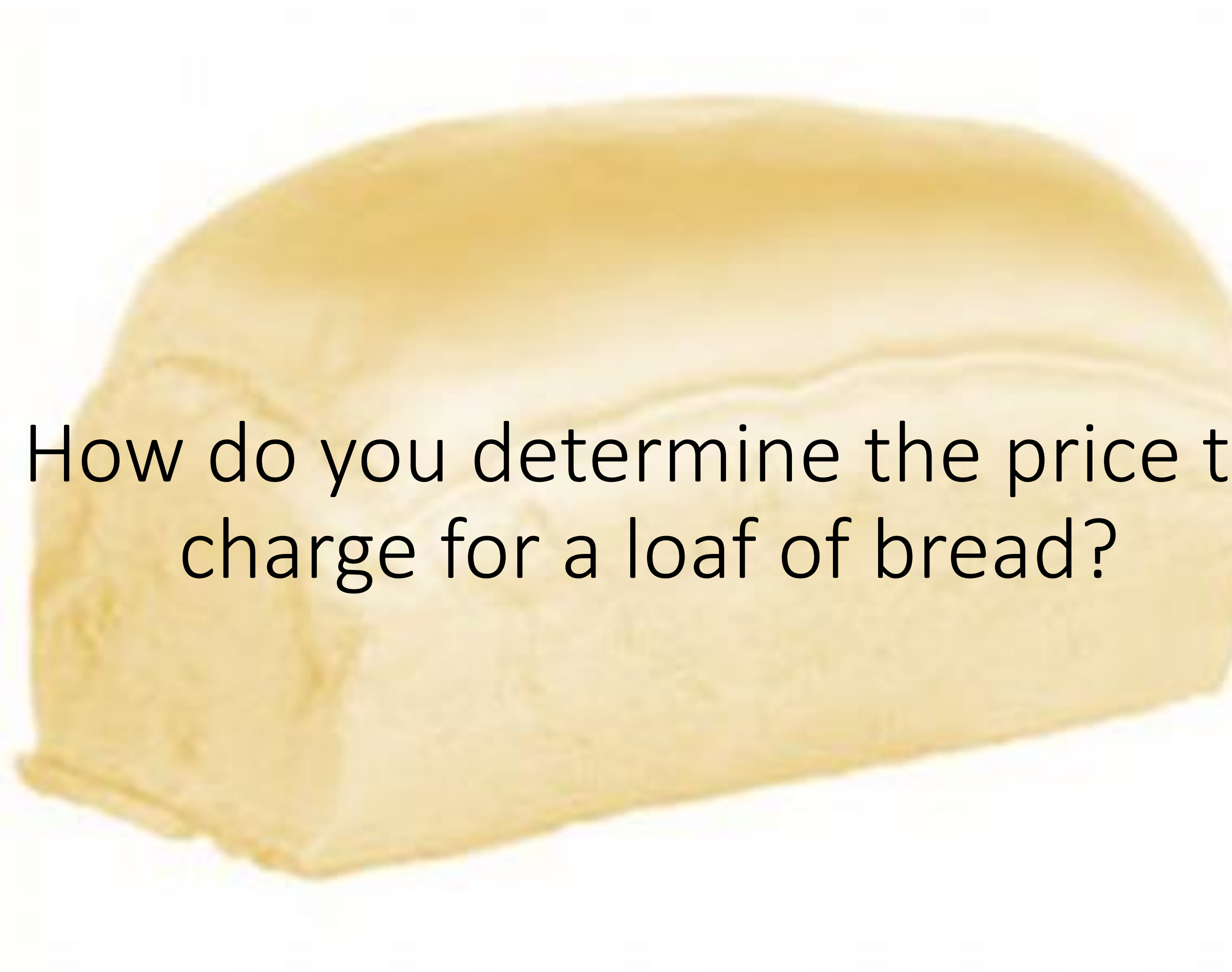
Problem	**Solution**
Requirements change	Embrace change
Fixing defects late is costly	Fix defects early
Late changes are costly	Build in flexibility

Agile is More Adaptable



Original Source: http://www.versionone.com/Agile101/Agile-Software-Development-Benefits/

5. Self-Organization



How do you determine the price to charge for a loaf of bread?

Market Economy

Market makes decisions

Produces and consumers

Supply and demand

Millions of decisions

Source: Britannica

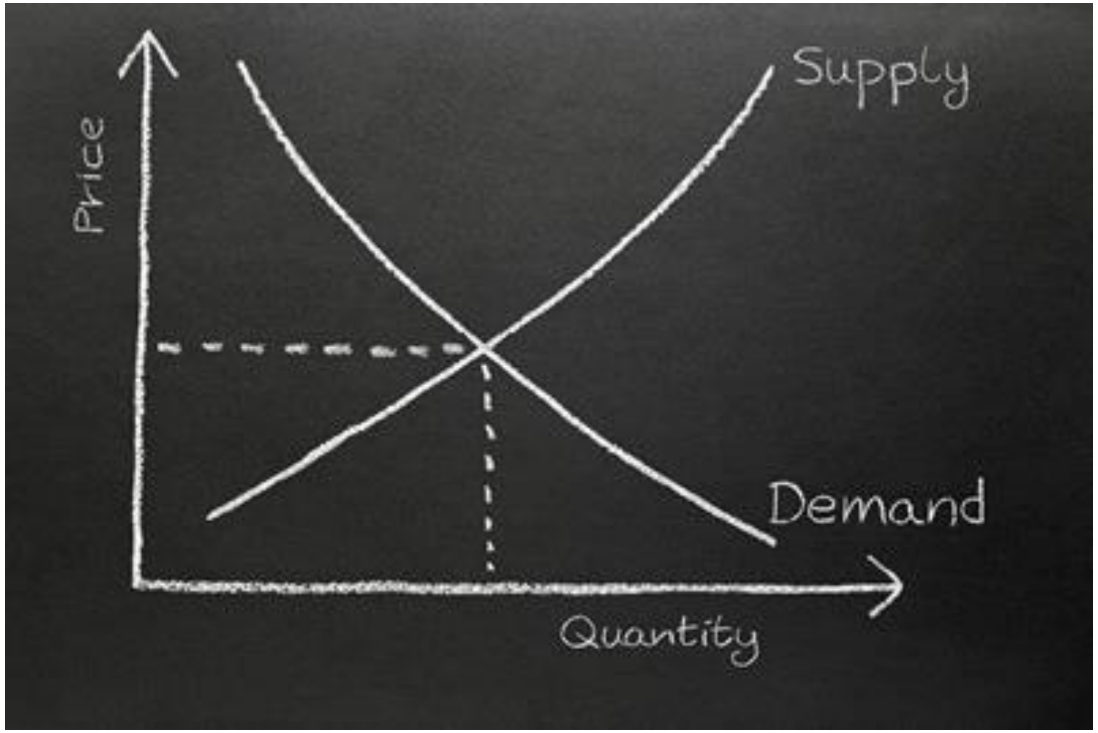
Market Economy

Goal: Maximize social welfare

Competitive market equilibrium

Extremely efficient

“Chaotic success”



Source: https://content.dodea.edu/
VS/HS/DVHS_Courses/Economics/syllabus.html

Complex Adaptive Systems

System

collection of interconnected things

Complex

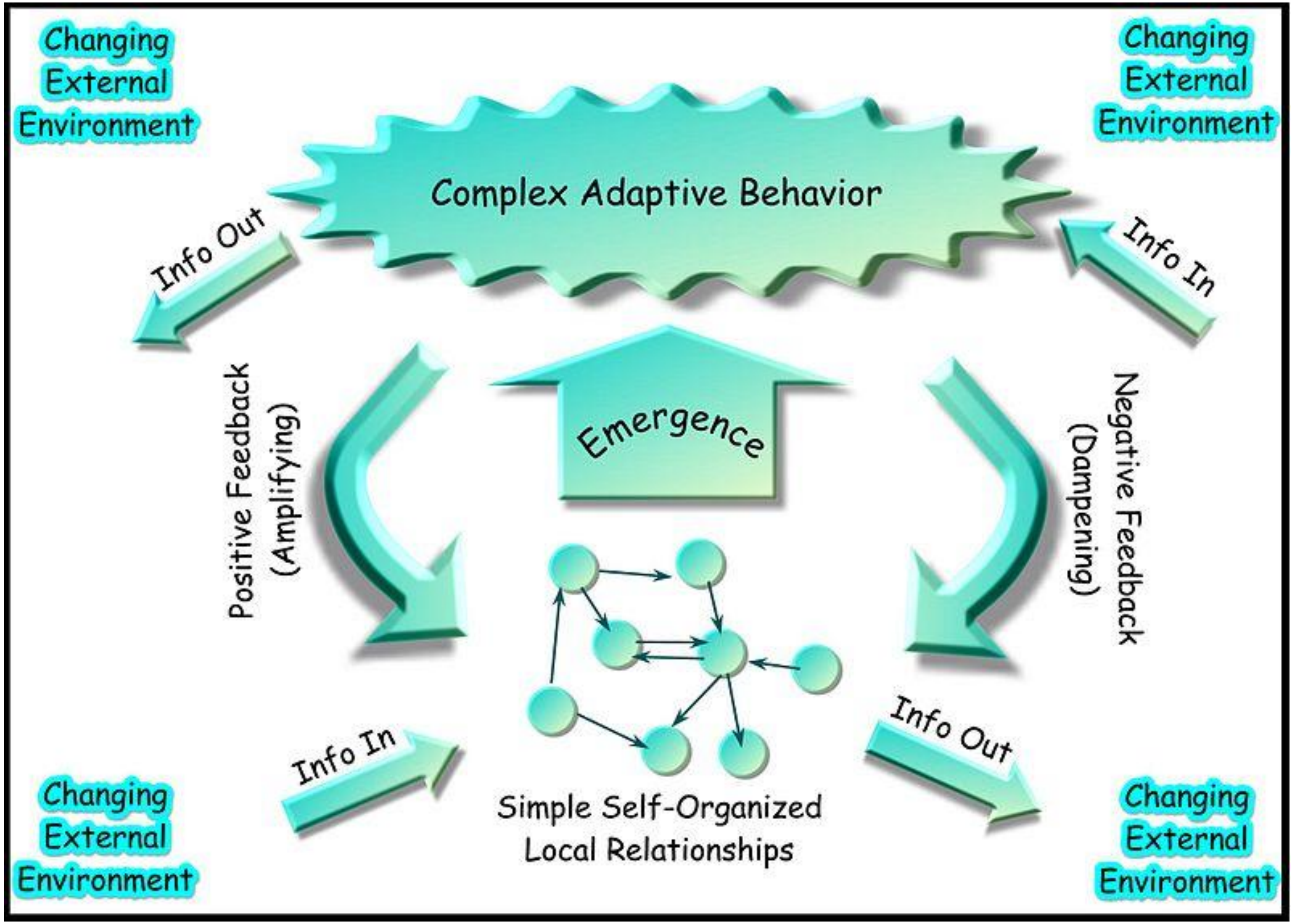
dynamic network of interactions

Adaptive

changes in response to environment

to increase survivability

Source: http://integral-options.blogspot.com/2013/03/peter-fryer-brief-description-of.html


Changing External Environment
Changing External Environment
Complex Adaptive Behavior
Info Out
Info In
Positive Feedback (Amplifying)
Emergence
Negative Feedback (Dampening)
Info In
Info Out
Simple Self-Organized Local Relationships
Changing External Environment
Changing External Environment

Source: Wikipedia

Inversion of Control

Top-down

Command and Control

Bureaucracy

Source: Wikipedia

Inversion of Control

Top-down

Command and Control

Bureaucracy

vs.

Bottom-up

Self-organization

Adhocracy

Source: http://funnyasduck.net/post/10458

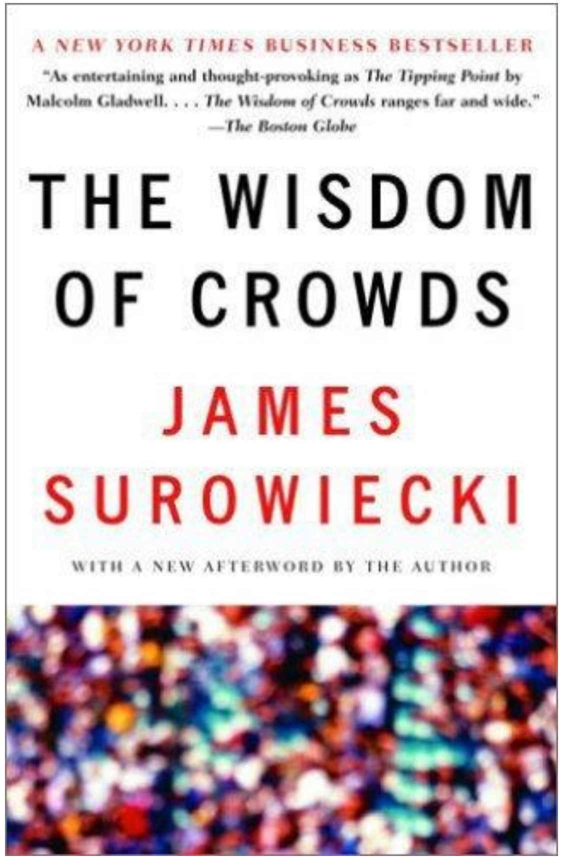
Wisdom of the Crowd

Collective guesses of crowd

Aggregate better than expert

Only some types of knowledge

Not all crowds are wise!



A *NEW YORK TIMES* BUSINESS BESTSELLER

"As entertaining and thought-provoking as *The Tipping Point* by Malcolm Gladwell. . . . *The Wisdom of Crowds* ranges far and wide."
—*The Boston Globe*

THE WISDOM OF CROWDS

JAMES SUROWIECKI

WITH A NEW AFTERWORD BY THE AUTHOR

Why is This Important?

Problem	**Solution**
Top-down is inefficient	Self-organizing teams
Poor information flow	Invert control to bottom-up
Ineffective decisions	Wisdom of the Crowds

Self-organizing Agile teams are more efficient

6. Effective Communication

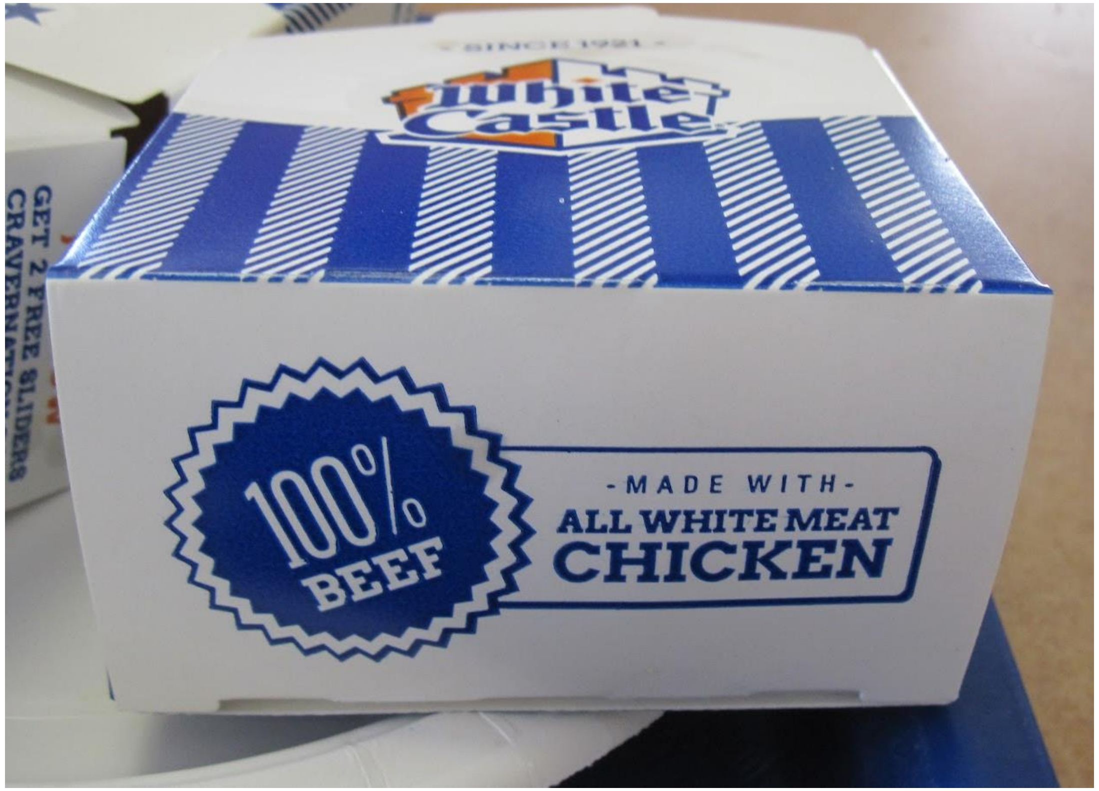
Cost of Poor Communication

Cost is enormous

Hard to quantify

Hidden cost

Expense is real



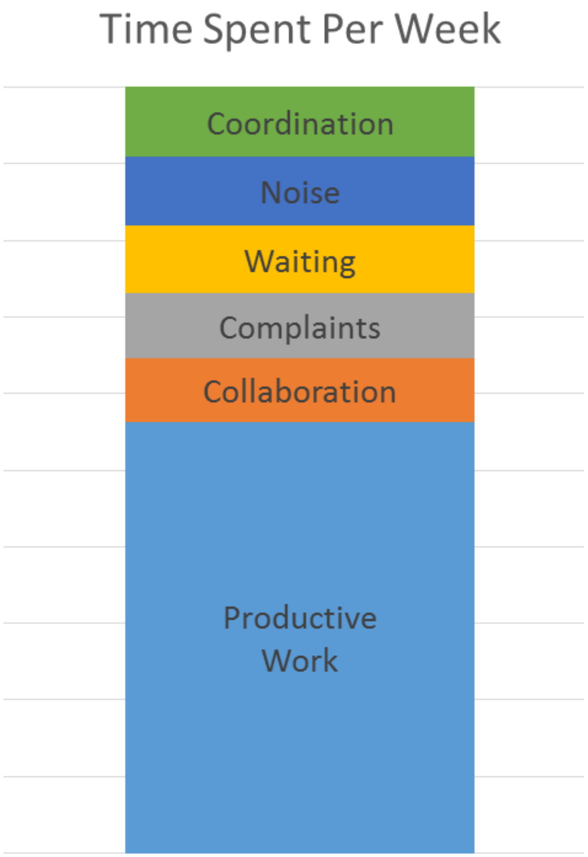
Source: http://www.cathy.willman.com/2012/06/what-boys-need.html

Cost of Poor Communication

17.5 hrs / person / week

Top 5 issues identified:

1. Waiting for information
2. Unwanted communication
3. Inefficient coordination
4. Barriers to collaboration
5. Customer complaints



Source: http://thoughtleadership.sismarketresearch.com/industrial-b2b-journal /2009/3/10/smb-communications-pain-study-white-paper-uncovering-the-hid.html

Total estimated annual cost of poor communication per enterprise knowledge worker: $50,562

Source: http://thoughtleadership.sismarketresearch.com/industrial-b2b-journal
/2009/3/10/smb-communications-pain-study-white-paper-uncovering-the-hid.html

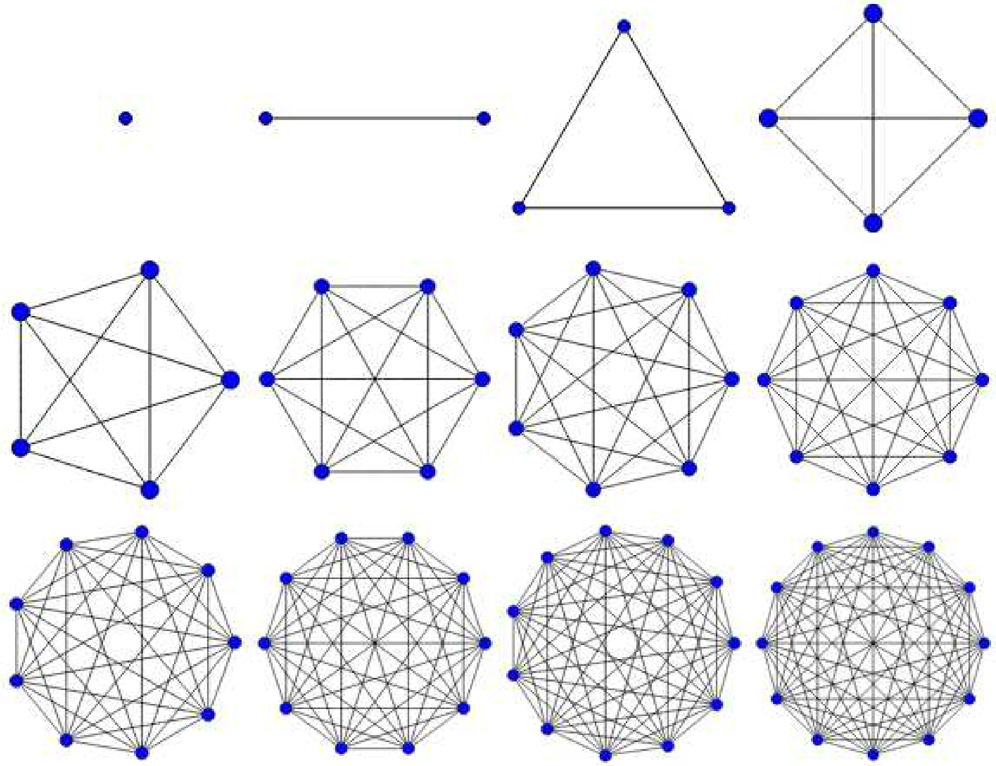
Communication Structures

Fully-connected graph

- Nodes = people
- Edges = channels

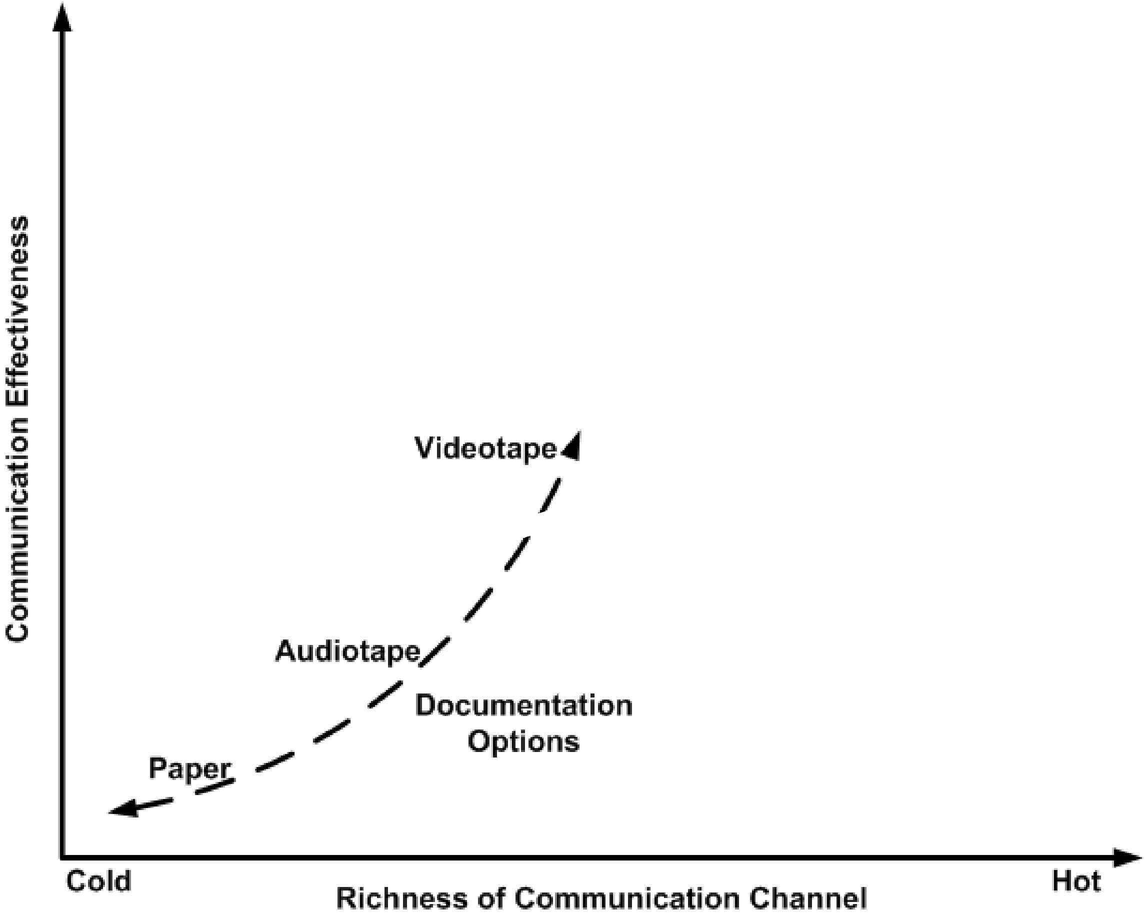
Edges increase by $O(n^2)$

Becomes inefficient very fast



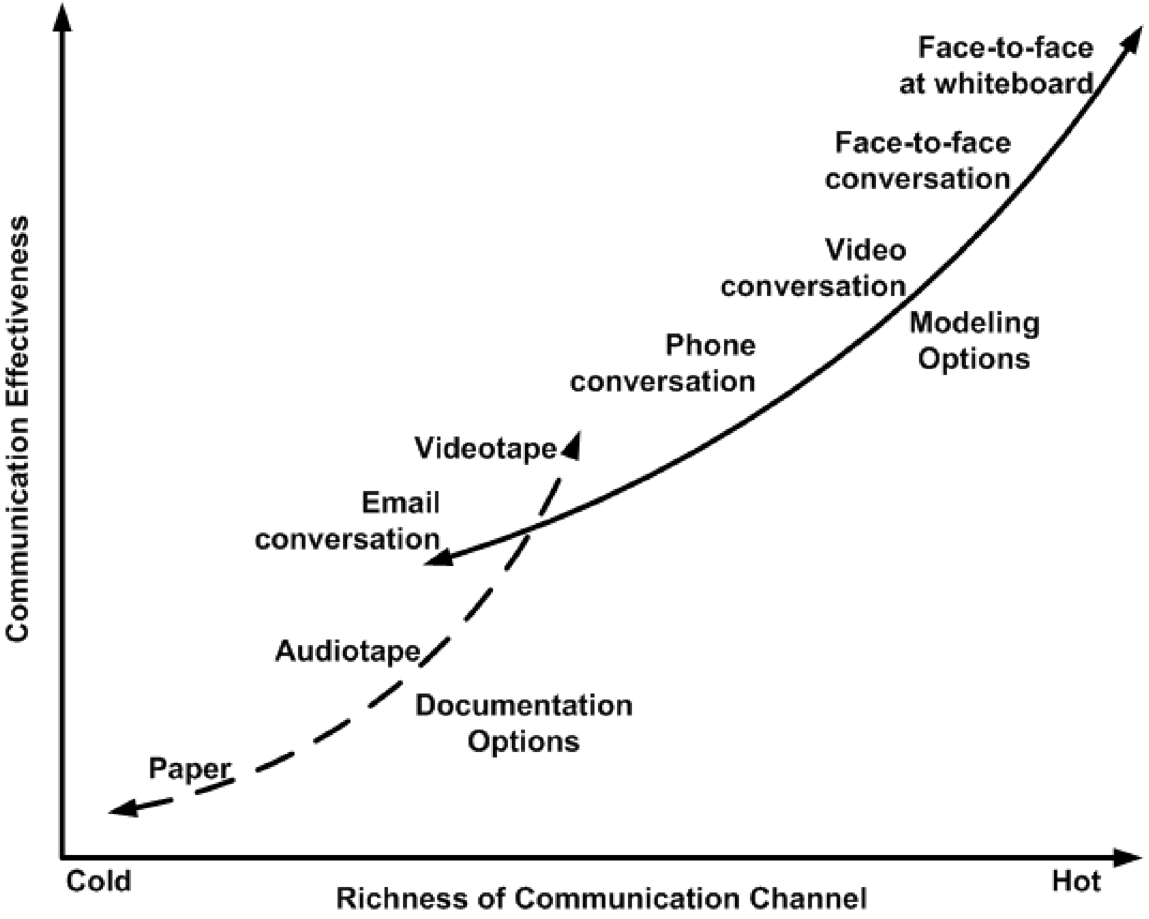
Source: Wikipedia

Effectiveness of Communication



Copyright 2002-2005 Scott W. Ambler
Original Diagram Copyright 2002 Alistair Cockburn

Effectiveness of Communication



Copyright 2002-2005 Scott W. Ambler
Original Diagram Copyright 2002 Alistair Cockburn

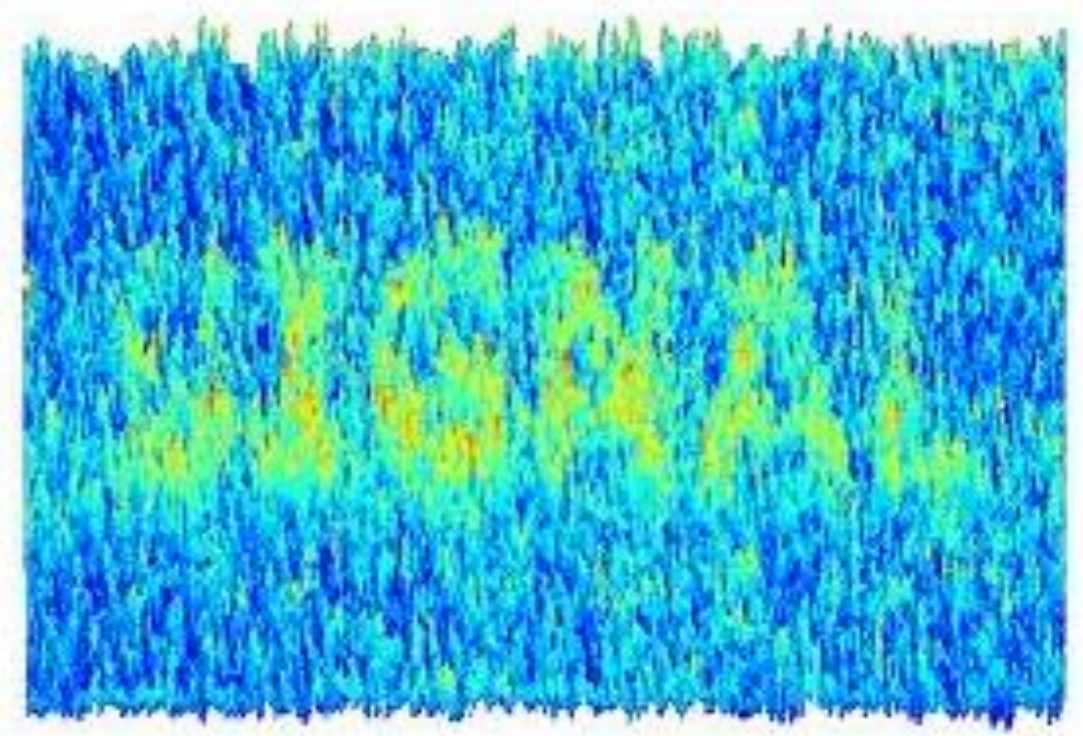
Signal-to-Noise Ratio

SNR = P(signal) / P(noise)

Signal = message

Noise = everything else

Goal is to maximize SNR



Source: http://uber.la/2012/05/signal-to-noise/

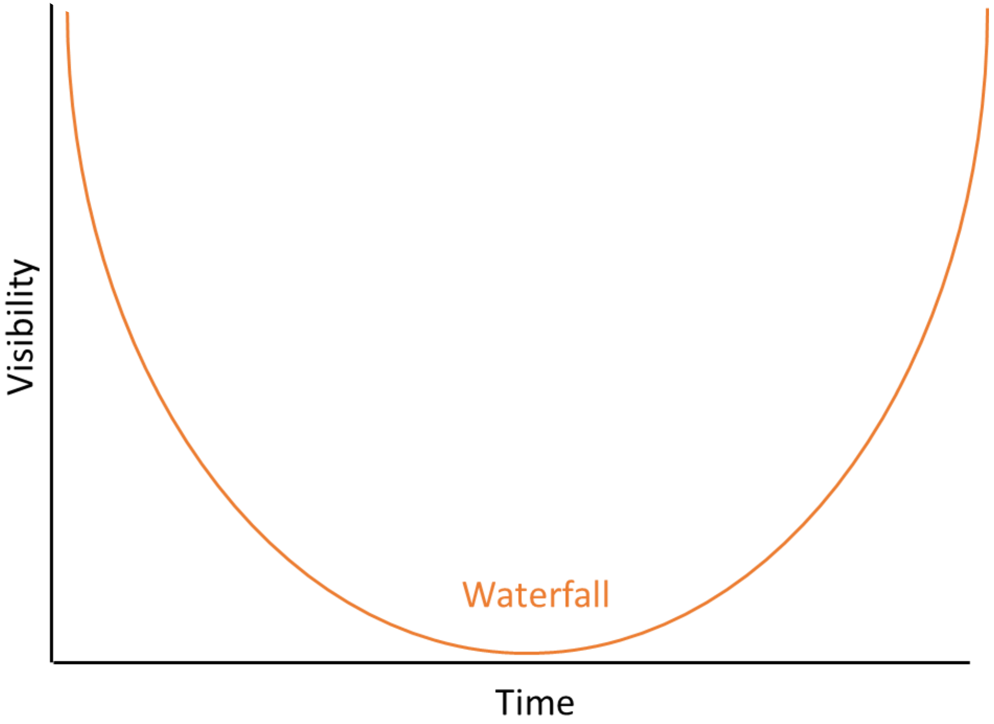
Visibility

Waterfall hides problems

High visibility at start

Low visibility at middle

High visibility at end



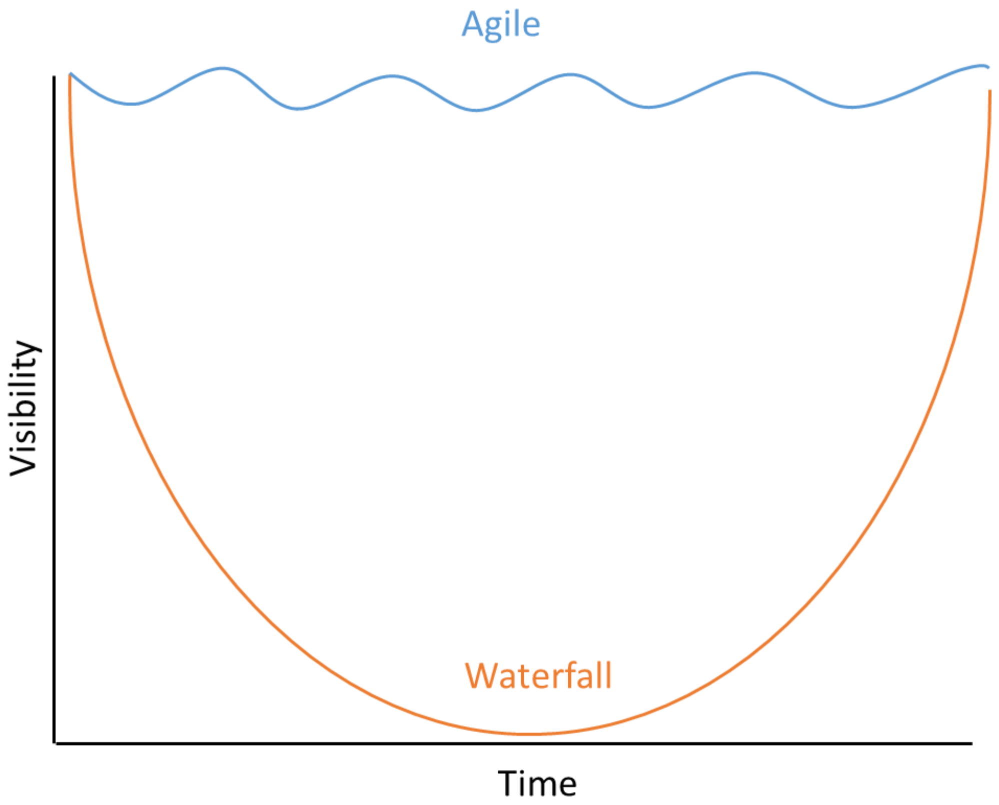
Original source: http://www.versionone.com/Agile101/Agile-Software-Development-Benefits/

Visibility

Agile provides visibility

On the surface with visibility

Problems have no where to hide



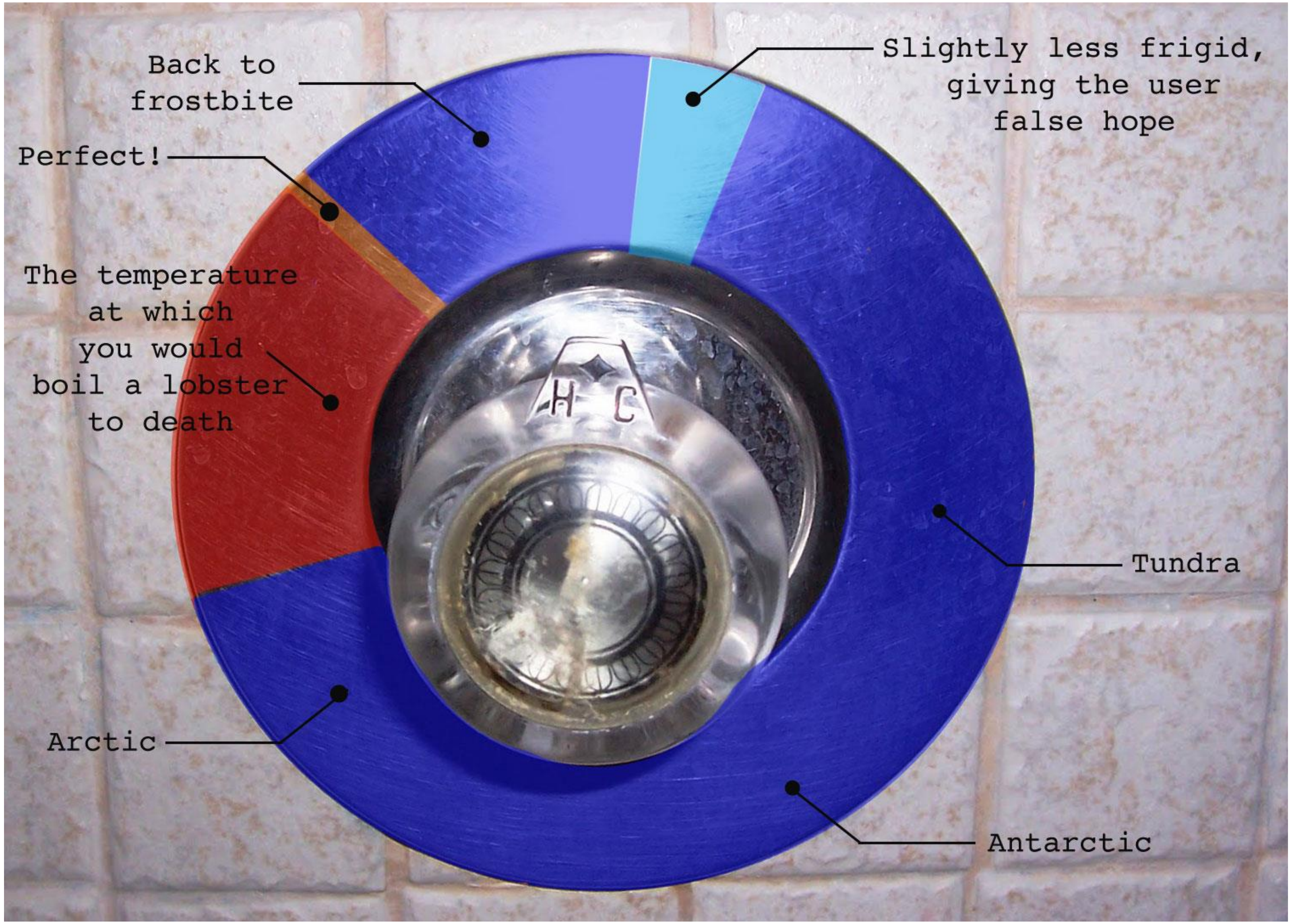
Original source: http://www.versionone.com/
Agile101/Agile-Software-Development-Benefits/

Why is This Important?

Problem	**Solution**
Communication overload	Small teams
Cost of poor communication	Maximize signal-to-noise ratio
Lack of transparency	Increase visibility

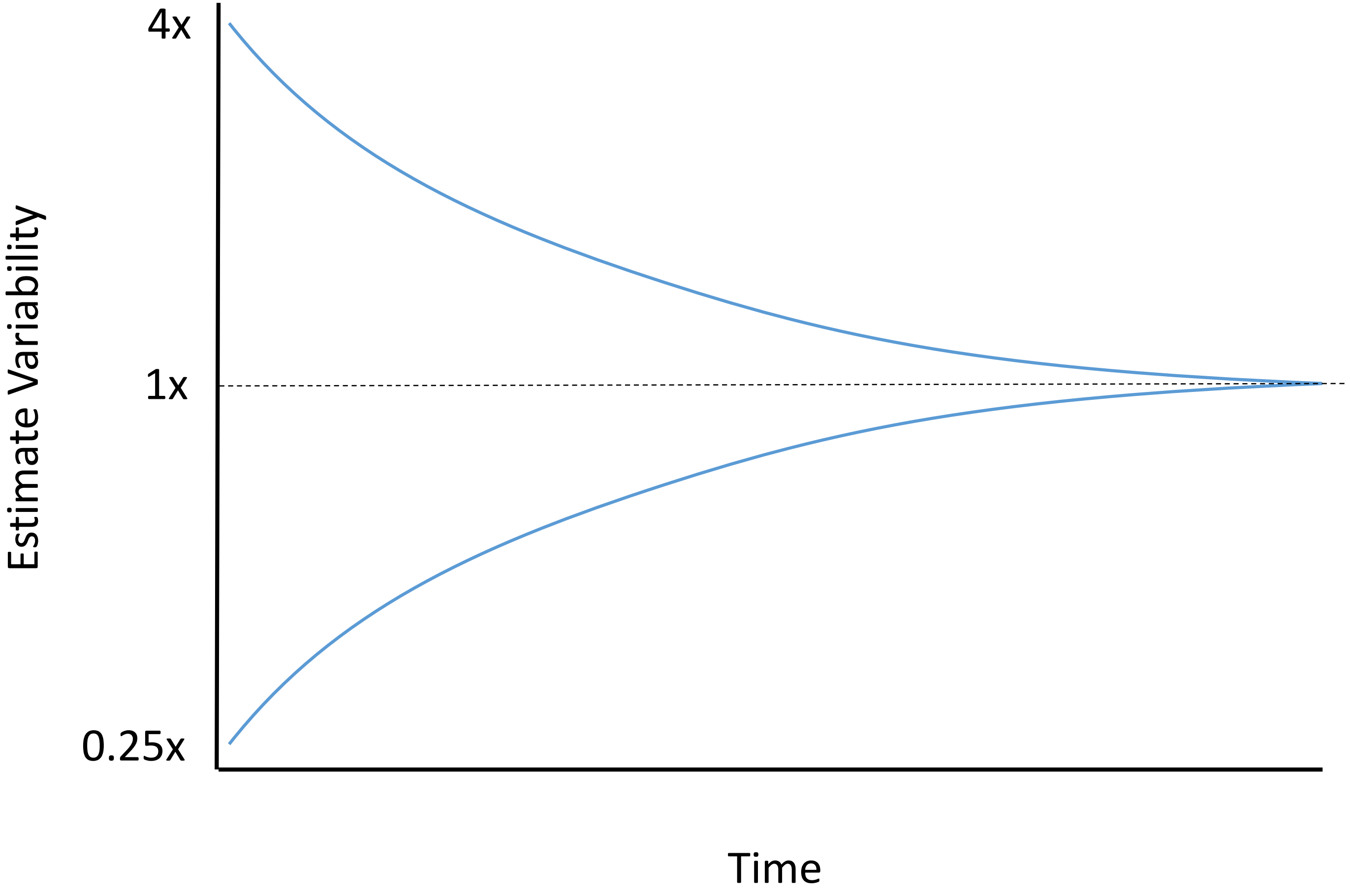
Agile teams communicate more effectively

7. Feedback


Back to frostbite
Slightly less frigid, giving the user false hope
Perfect!
The temperature at which you would boil a lobster to death
H C
Tundra
Arctic
Antarctic

Source: http://www.letterstobuffoons.com/wp-content/uploads/2012/09/ShowerHandle.jpg

Cone of Uncertainty



Original Source: Barry Boehm, Software Engineering Economics (1981)

Feedback and Learning

Learning reduces uncertainty

Feedback is necessary

Continuous and rapid feedback

Source: http://www.icanhascheezburger.com

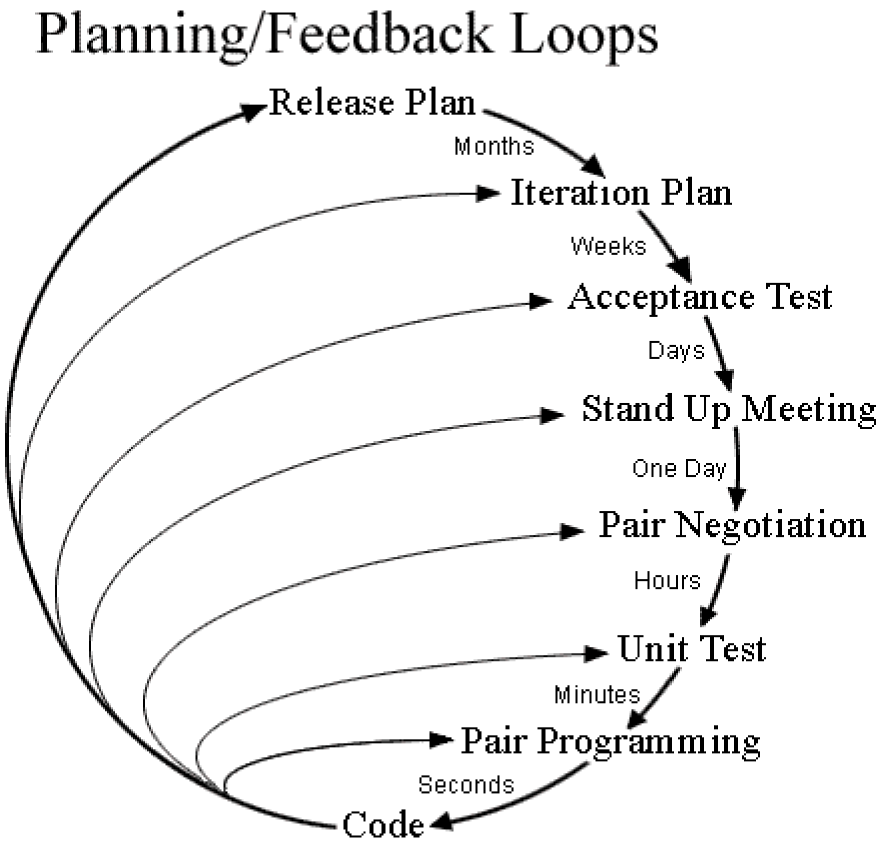
Agile Feedback

Continuous and rapid feedback

Multiple timescales

Powerful for:

- Learning
- Reducing risk
- Eliminating Uncertainty



Source: http://www.agile-process.org/communicate.html

Smart Failure

Short and frequent experiments

Low cost and high value

Old world vs. new world

Requires mindset change

Source: http://craftfail.com/2011/08/cookie-monster-cupcake-fail/

It’s Not OK to Fail BIG!

Source: http://t4toby.files.wordpress.com/2008/07/epicfail1.jpg/

Know When to Pivot

Pivot = change direction

Assumptions incorrect => pivot

Pivot early, not late

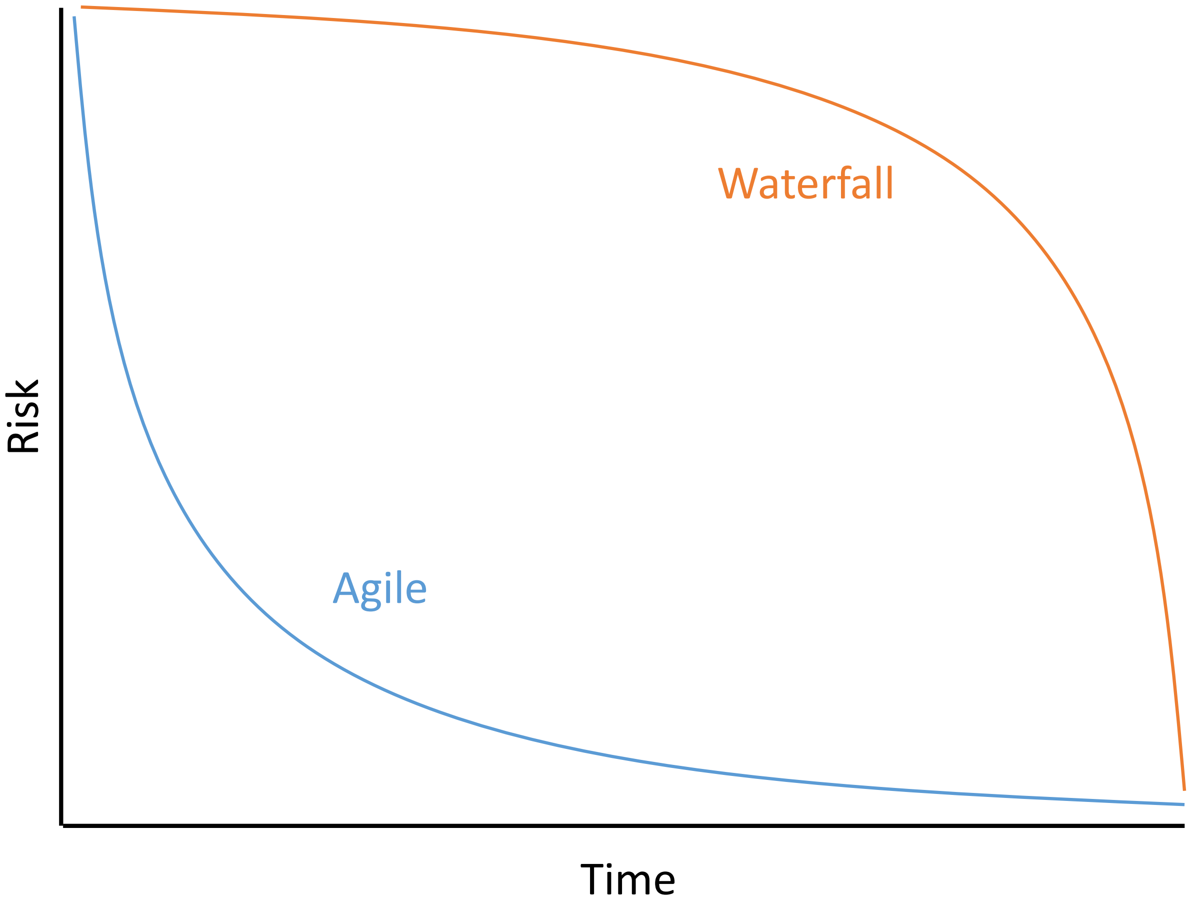
Minimize cost to pivot

Source: http://thesalespivot.com/wp-content/
uploads/2011/07/left-turn-sign.jpg

Why is This Important?

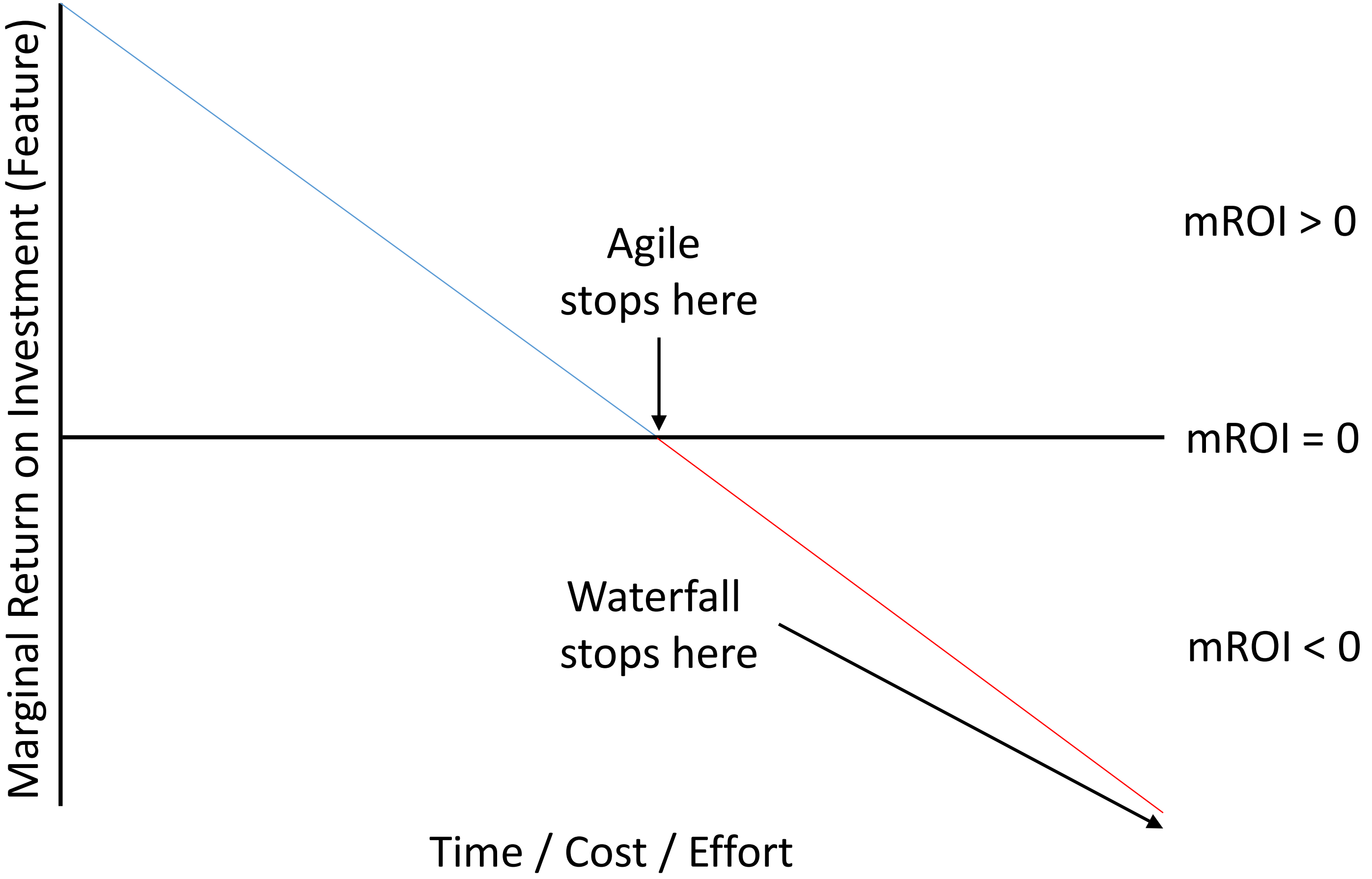
Problem	**Solution**
Cone of uncertainty	Feedback
Avoid epic failure	Embrace smart failure
Difficulty changing course late	Minimize cost to learn

Agile Teams Use Feedback to Reduce Risk



Source: http://www.versionone.com/Agile101/Agile-Software-Development-Benefits/

Know When to Stop



Original Source: http://www.versionone.com/Agile101/Agile-Software-Development-Benefits/

Know When to Stop

- Everything else:
  - The Cost of Complexity
  - Eliminating Waste
  - Inventory Hides Problems
  - Metrics Have Consequences
  - Embracing Human Factors
  - Information Gain / Entropy
  - Embedded Documentation
  - Kanban and Queuing Theory
  - TDD, Dopamine, and Crack
  - Sustainable Development
  - Agile is an Emergent Property
  - and much more…

Source: http://www.rounds.com/blog/wp-content/uploads/2010/11/stop-hammertime.png

Conclusion

Why is Agile so Successful?

1. It is well adapted to the world after midnight.
2. It inverts its constraints to be more flexible.
3. It maximizes ROI by prioritizing features by value.
4. It is more adaptable by embracing change
5. It utilizes the efficiencies of self-organization.
6. It produces more effective communication.
7. It reduces risk by continuous and rapid feedback.

My Website

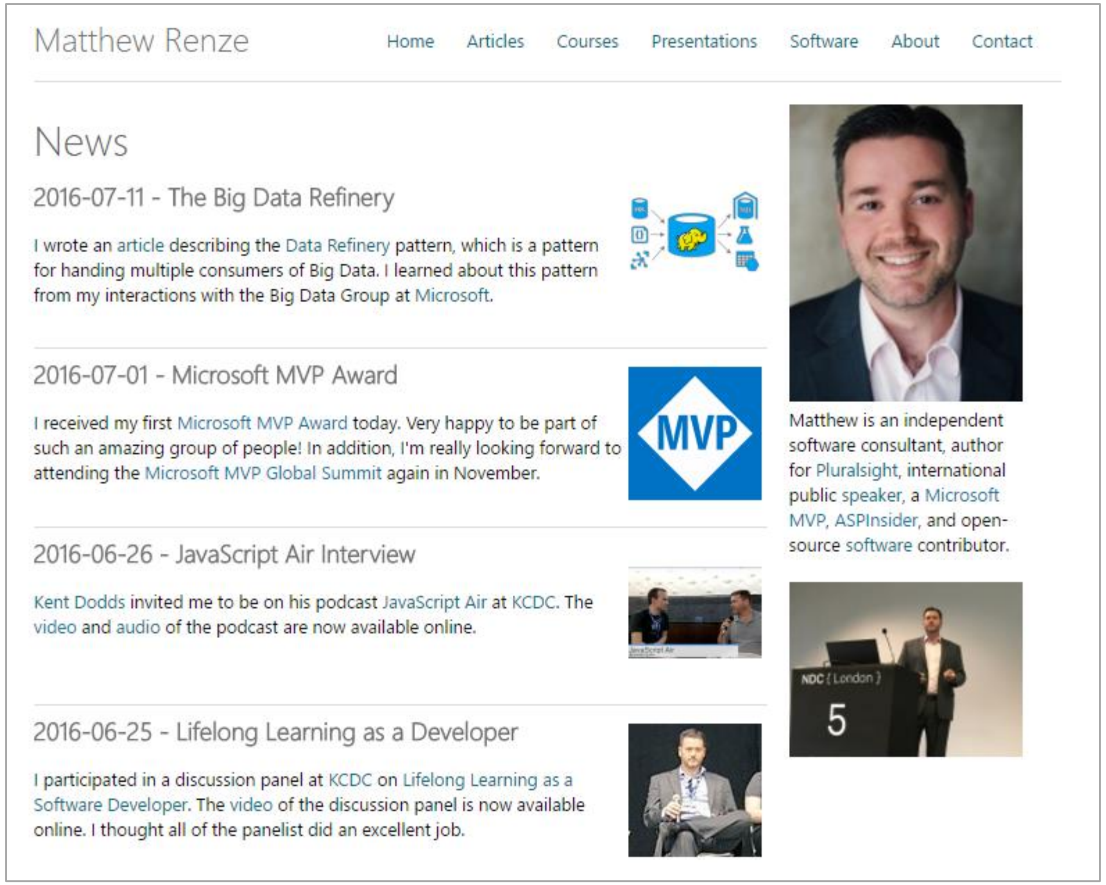
Articles

Courses

Presentations

Source Code

Videos



Matthew Renze

Home Articles Courses Presentations Software About Contact

News

2016-07-11 - The Big Data Refinery

I wrote an article describing the Data Refinery pattern, which is a pattern for handing multiple consumers of Big Data. I learned about this pattern from my interactions with the Big Data Group at Microsoft.

2016-07-01 - Microsoft MVP Award

I received my first Microsoft MVP Award today. Very happy to be part of such an amazing group of people! In addition, I'm really looking forward to attending the Microsoft MVP Global Summit again in November.

2016-06-26 - JavaScript Air Interview

Kent Dodds invited me to be on his podcast JavaScript Air at KCDC. The video and audio of the podcast are now available online.

2016-06-25 - Lifelong Learning as a Developer

I participated in a discussion panel at KCDC on Lifelong Learning as a Software Developer. The video of the discussion panel is now available online. I thought all of the panelist did an excellent job.

Matthew is an independent software consultant, author for Pluralsight, international public speaker, a Microsoft MVP, ASPInsider, and open-source software contributor.

www.matthewrenze.com

Feedback

Feedback is very important to me!

One thing you liked?

One thing I could improve?

Contact Info

Matthew Renze

Data Science Consultant

Renze Consulting

Twitter: @matthewrenze

Email: info@matthewrenze.com

Website: www.matthewrenze.com

Thank You! :)