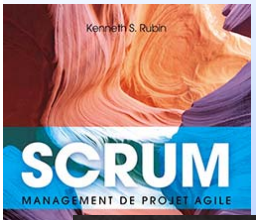
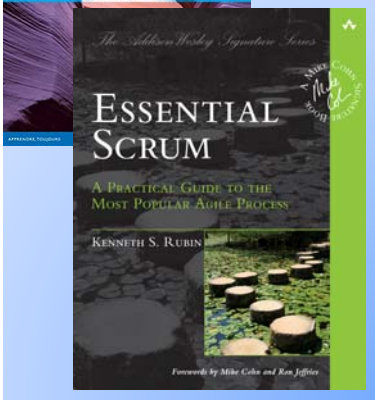
Agile IS Risk Management
April, 2014
by Ken Rubin

www.innolution.com

Background of Ken Rubin

Author





Trainer/Coach

Trained more than 20,000 people in Agile/Scrum, SW dev and PM

Provide Agile/Scrum coaching to developers and executives

CST ScrumAlliance CERTIFIED ScrumTrainer

Experience

Former Managing Director

ScrumAlliance transforming the world of work

My first Scrum project was in 2000 for bioinformatics

GENOMICA

IBM **Executive**

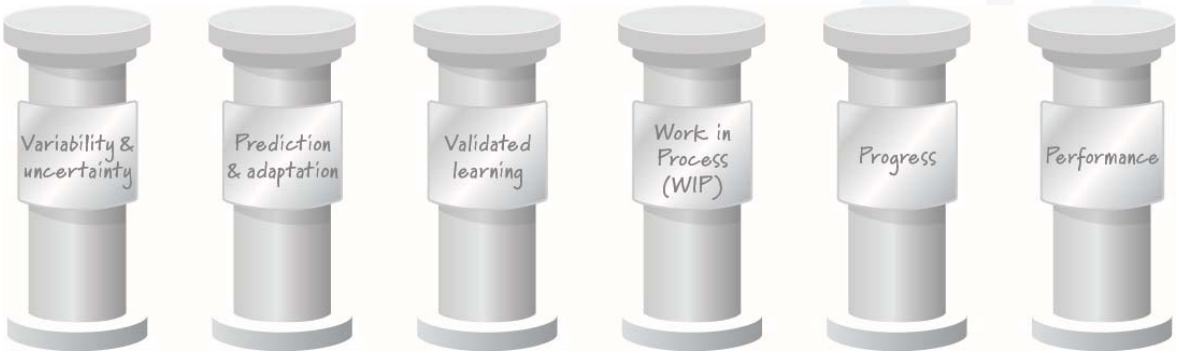
ParcPlace

Agenda

- Overview
- Uncertain Events
- Maximize Expected Monetary Value
- Risk Management Assumptions
- Traditional Risk Management
- Using Agile to Avoid Some Uncertain Situations
- Agile Principle-based Risk Management

Our goal today is to discuss…

How applying core agile principles make the development process robust and at times antifragile to the disorder of uncertain events, allowing us to avoid harm and reap the benefits of uncertainty, without the need for heavyweight risk management processes



What are we interested in?

When appropriate, applying simple traditional risk management techniques in a parsimonious (simplest possible) way

Applying agile principles to avoid the self-creation of inherently risky or uncertain situations

Applying agile principles to avoid the harm (be robust) and reap the benefits (be antifragile) from uncertainty in our environment

Many words for the same concept

- Risk
- Randomness
- Volatility
- Variability
- Uncertainty

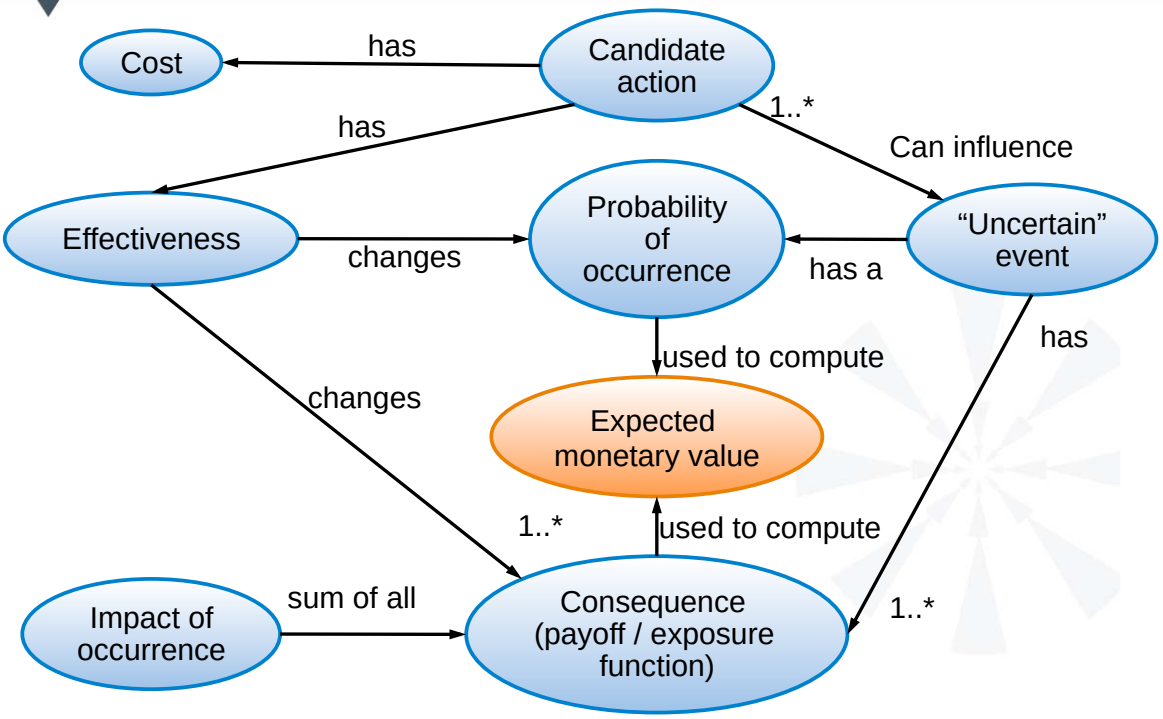
For our purposes we will treat them the same

Boils down to a lack of knowledge regarding uncertain events

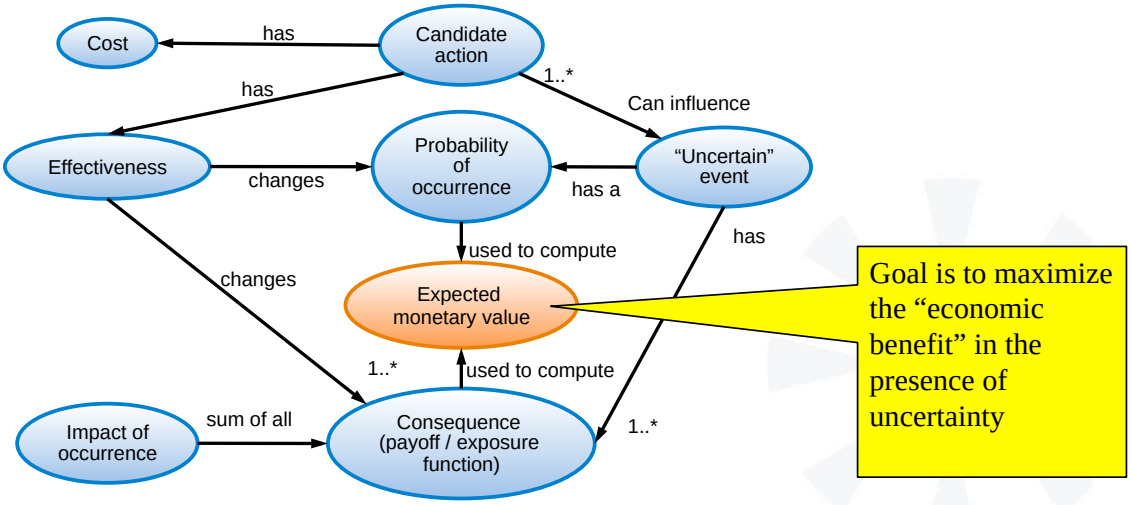
Example uncertain events

- Earthquake disables California data center housing the development servers
- Vendor fails to deliver a component when promised
- Application fails to scale to 10m current users
- Changing requirements
- Building the wrong product
- Knowledgeable people leaving company

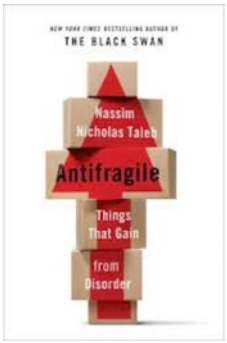
Typical mental model of uncertain events



We strive to…



Fragile, Robust, Antifragile

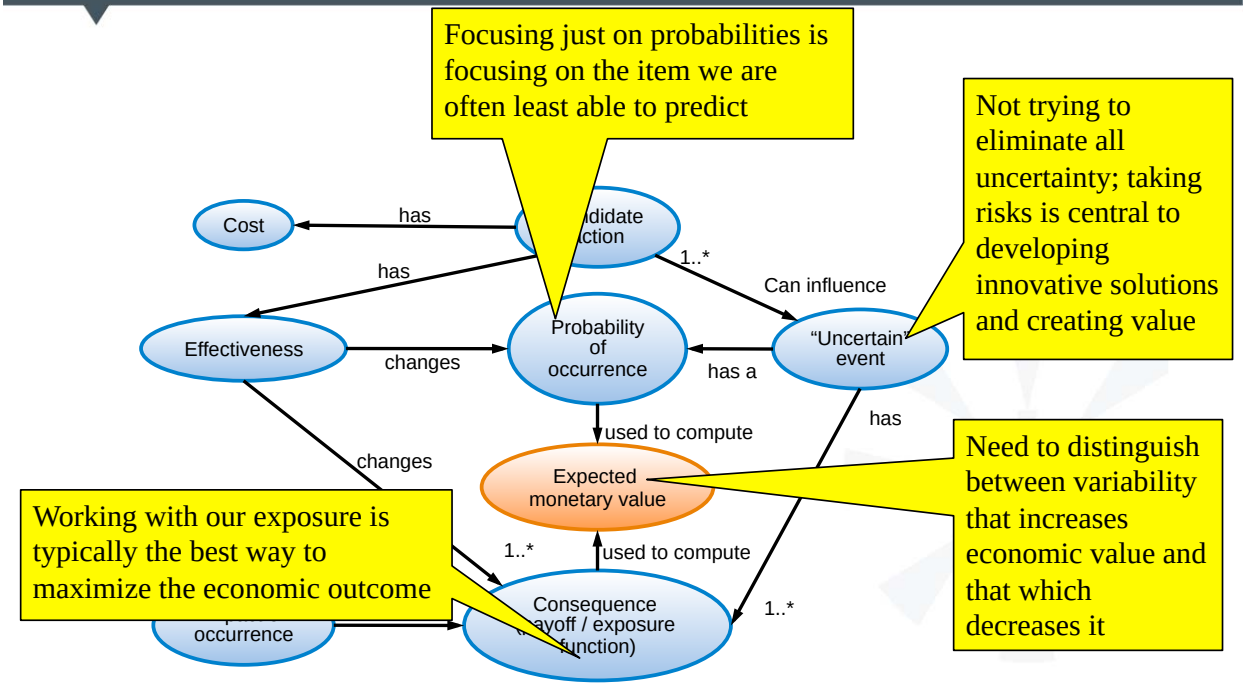


Waterfall

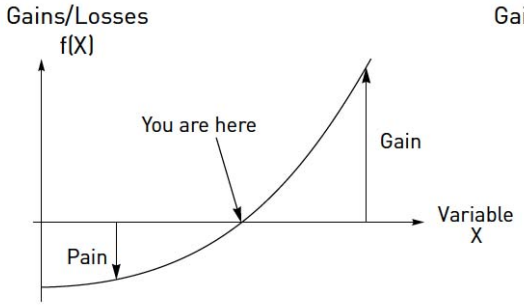
Agile

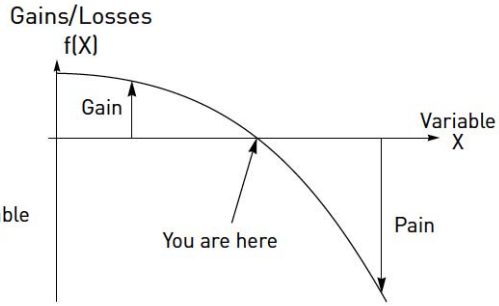
Fragile	Robust	Antifragile
Harmed by disorder	Resilient to disorder	Benefits from disorder

Goal is NOT to eliminate uncertainty, risk, or variability



Asymmetric payoffs create economic value or harm

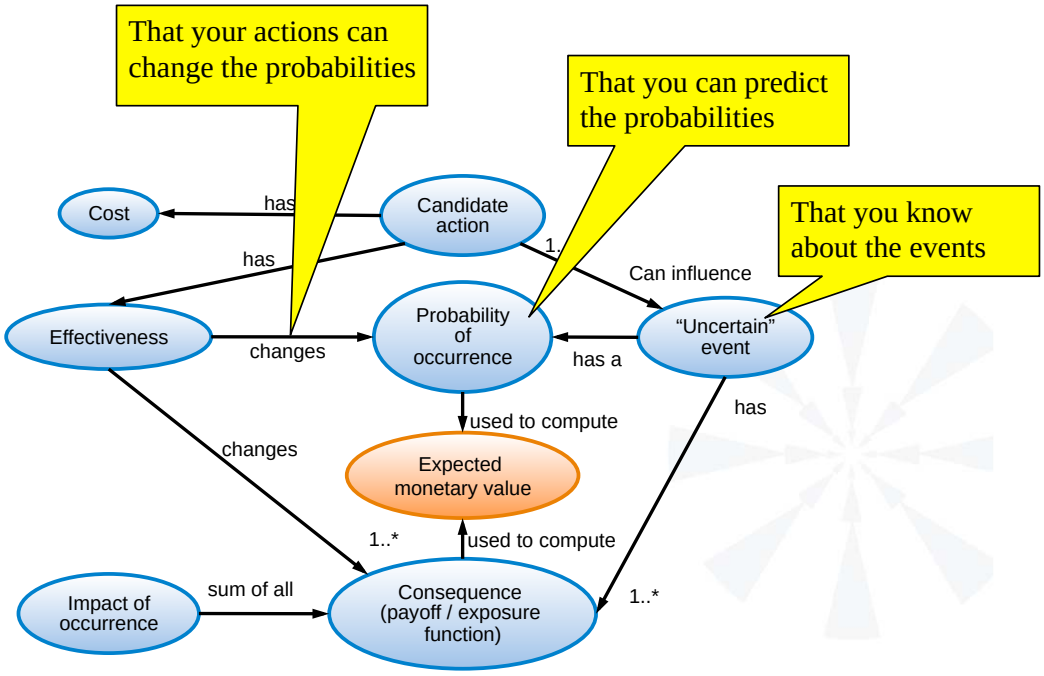




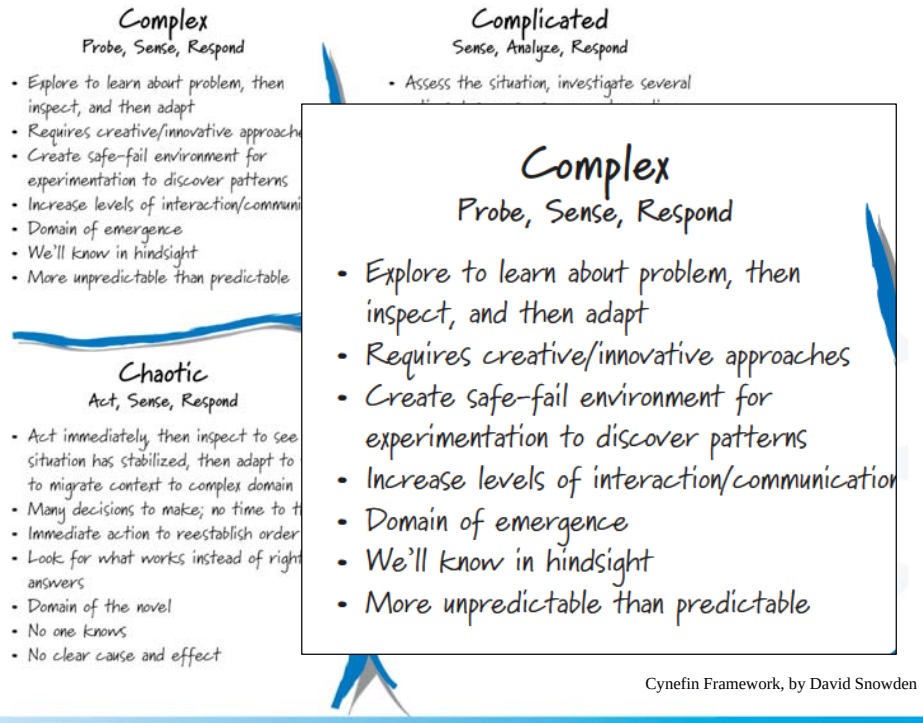
Positive asymmetric payoff (**antifragile**) anything that has more upside than downside from random events (variability)

Negative asymmetric payoff (**fragile**) anything that has more downside than upside from random events (variability)

Assumptions in traditional risk management



Operate in a complex domain



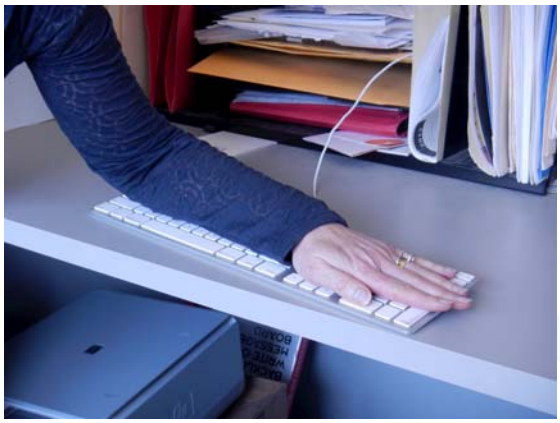
Cynefin Framework, by David Snowden

Unknown unknowns

Many uncertain events aren't predictable proactively

These are the unknown unknowns

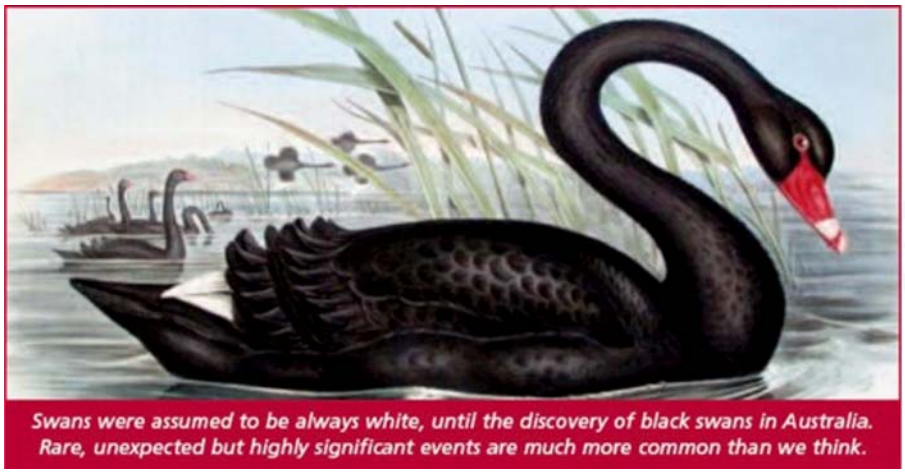
Some are inconsequential



Don't design a risk management system to deal with these, and even if we did these items would not be in our tables and charts

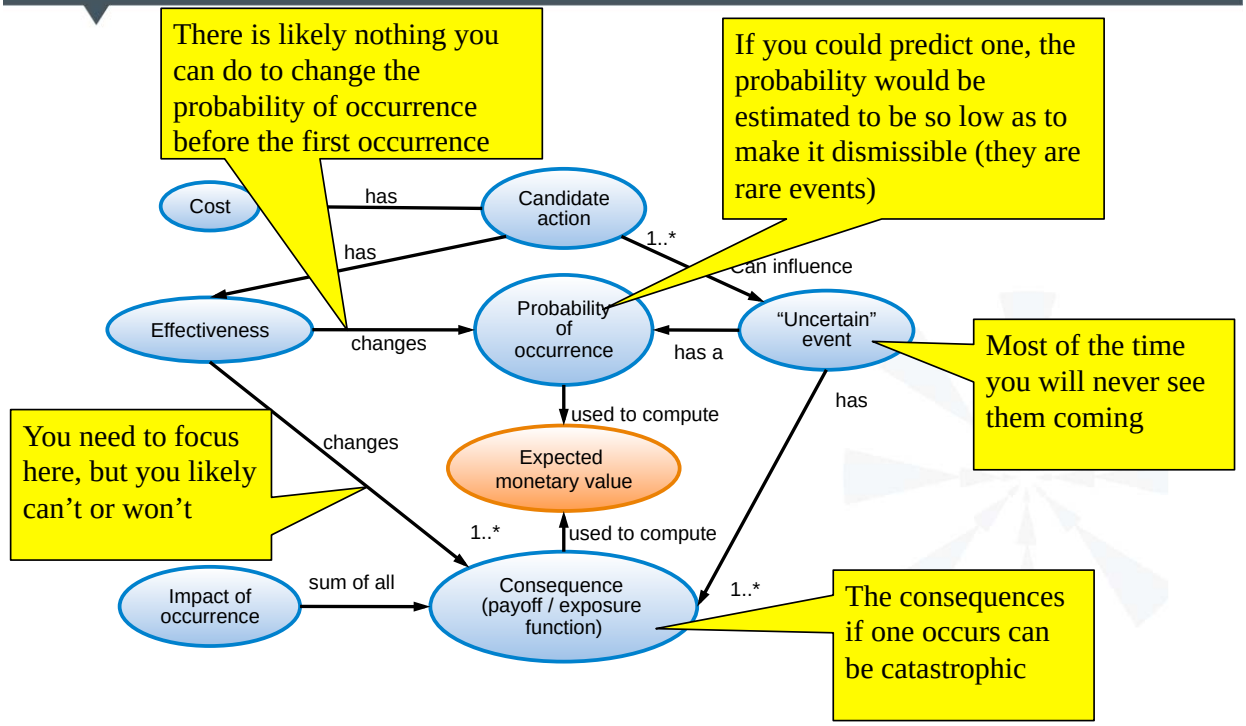
Others are Black Swans

Large-scale unpredictable
(or very hard to predict)
events of massive consequences



Swans were assumed to be always white, until the discovery of black swans in Australia. Rare, unexpected but highly significant events are much more common than we think.

Black Swan characteristics



Exercise – Identifying Black Swans

Identify Black Swans that you have seen affect software development efforts

Dealing with Black Swans

Leverage Agile principles to be robust or antifragile to things we can't predict or the very improbable

Focus our efforts on modifying our exposure the Black Swans instead of trying to quantify their statistical properties

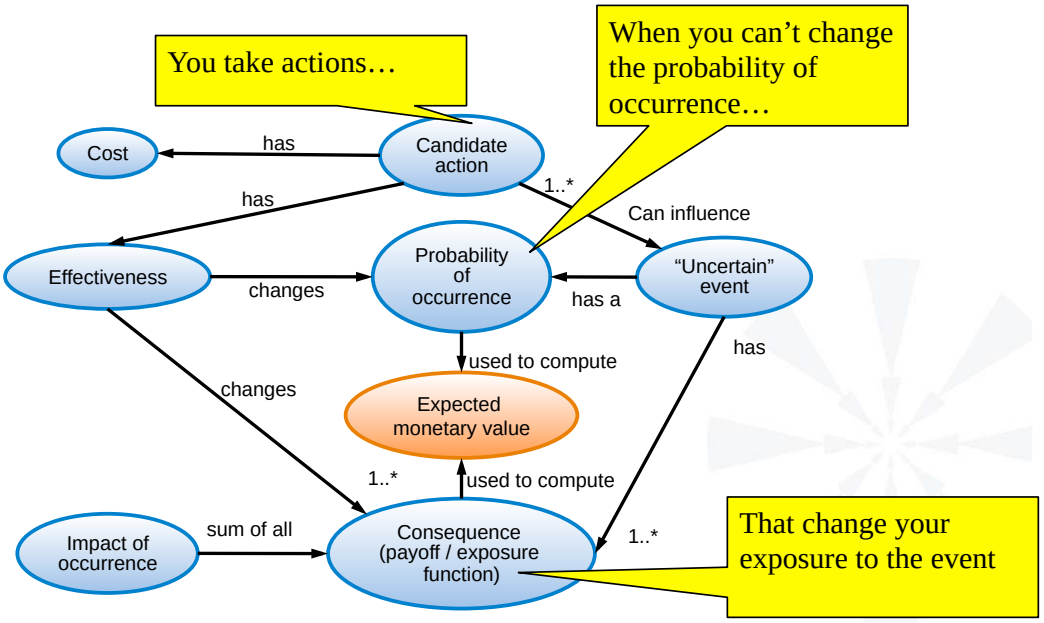
We know some events will happen, but we can't predict or change probabilities

We can predict earthquakes will happen in California

We can't predict the occurrence of a specific earthquake of a given magnitude, or change the probability of it happening

We can describe the consequences to our business via a disruption in our California-based data center if we are affected by an earthquake

Adjust your exposure



More sophisticated process does NOT solve this problem

Mistaken belief that we need better computation in order to more accurately predict the event and figure out the probabilities

More effective approach is to modify your exposure and learn to get out of trouble fast

Traditional risk management

Traditional risk management in Agile?

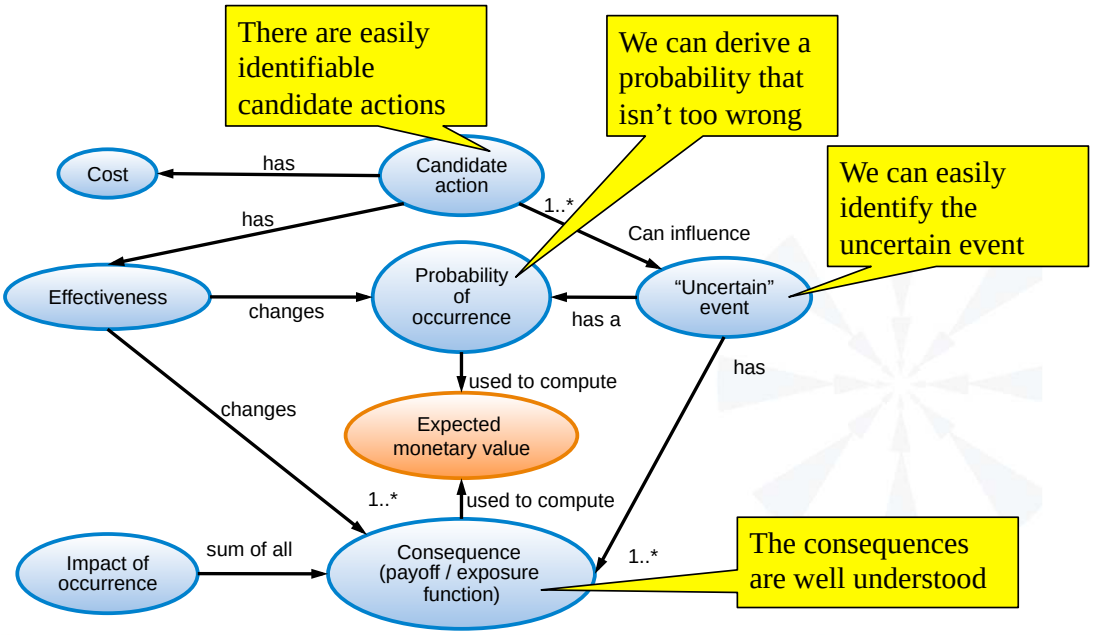
Yes, it's a matter of degree!

Domains where human lives are at risk will likely employ a more heavyweight risk management process

Most other domains can get along with very simple traditional risk management techniques and rely on effective application of agile principles as an a primary means of dealing with uncertainty

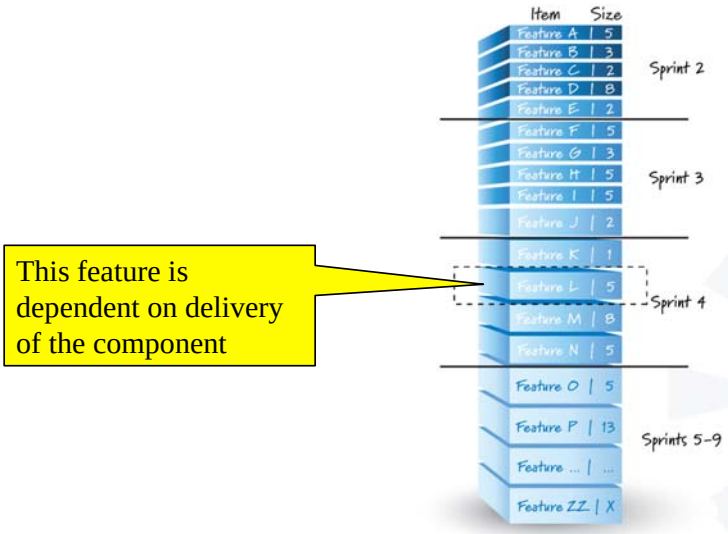
Example of simple risk

Vendor might fail to deliver a component on a promised date



Candidate Action 1 – Manage the dependency risk

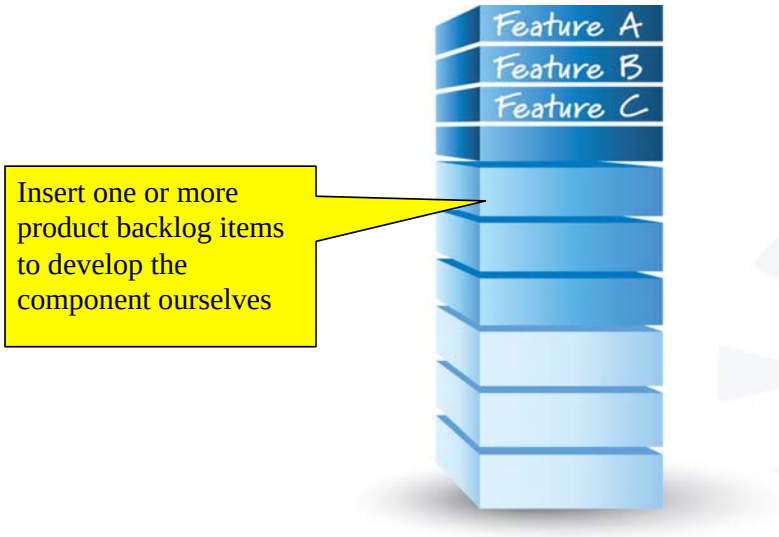
Affects prioritization of other items in the product backlog



Manage risk via product backlog grooming

Candidate Action 2 – Create new product backlog items

Employ the parallel hedge strategy



Manage risk via product backlog grooming

Candidate Action 3 – Not shown in our backlog

Send one or more of our employees to vendor to help expedite

Risk	Prob	Exposure	Mitigation
Vendor fails to deliver Component X	50%	$1m/month	Send Barbara to vendor to help expedite

Manage risk via lightweight traditional techniques

Candidate Action 4 – Also not shown in our backlog

Pay expedited charge to move to head of queue

Risk	Prob	Exposure	Mitigation
Vendor fails to deliver Component X	50%	$1m/month	Pay more money to get head of queue privileges

Manage risk via lightweight traditional techniques

Traditional risk management summary

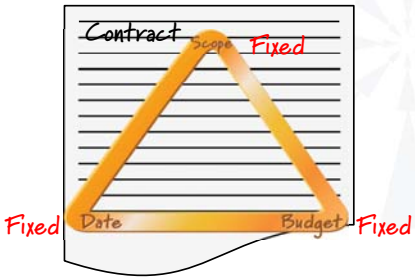
Like anything else in Agile, we would embrace the minimum (barely sufficient) amount of process that would be sufficient for dealing with the risks in our particular environment

Some uncertain events can be avoided altogether

Avoid the self-creation of inherently risky or uncertain situations

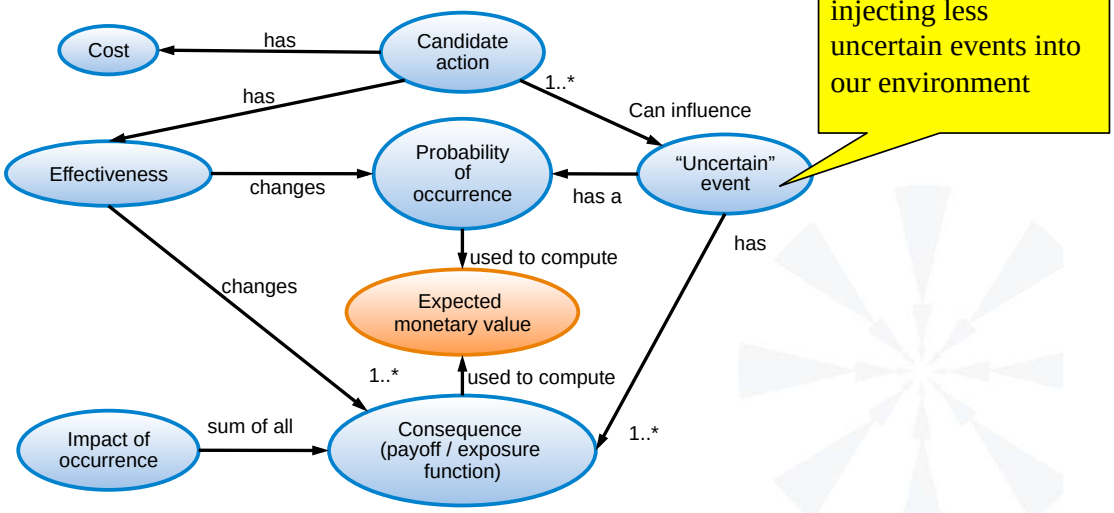
- If you don't go into space, you don't have to worry about the risk that your spaceship could run out of fuel

- If you don't write fixed price subcontracts, you can avoid the risks of fixed price contracts!



Effort saved not having to "manage" uncertain events

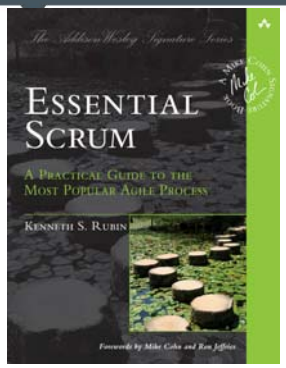
Think of the effort saved if we end up injecting less uncertain events into our environment



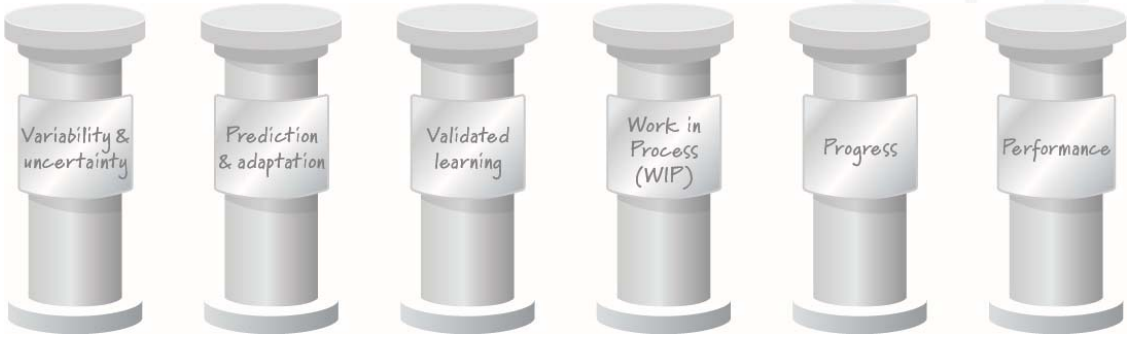
Exercise – Uncertain events avoidable by Agile

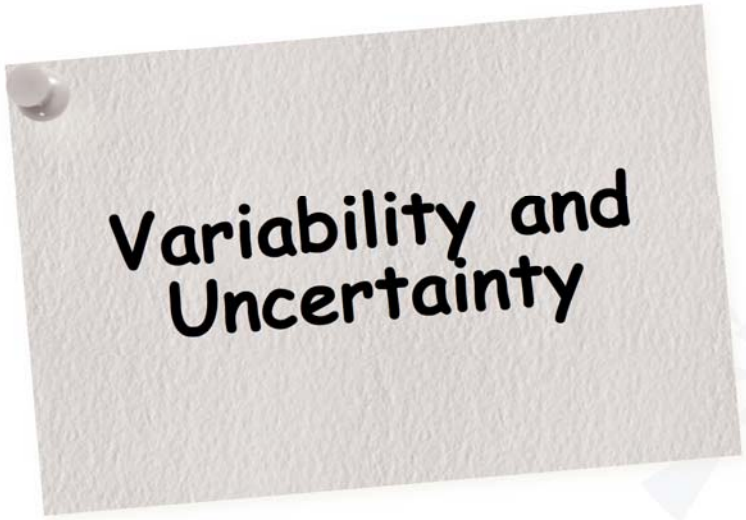
What are some known risks or uncertainties that we can avoid just by applying Agile development?

Agile principles

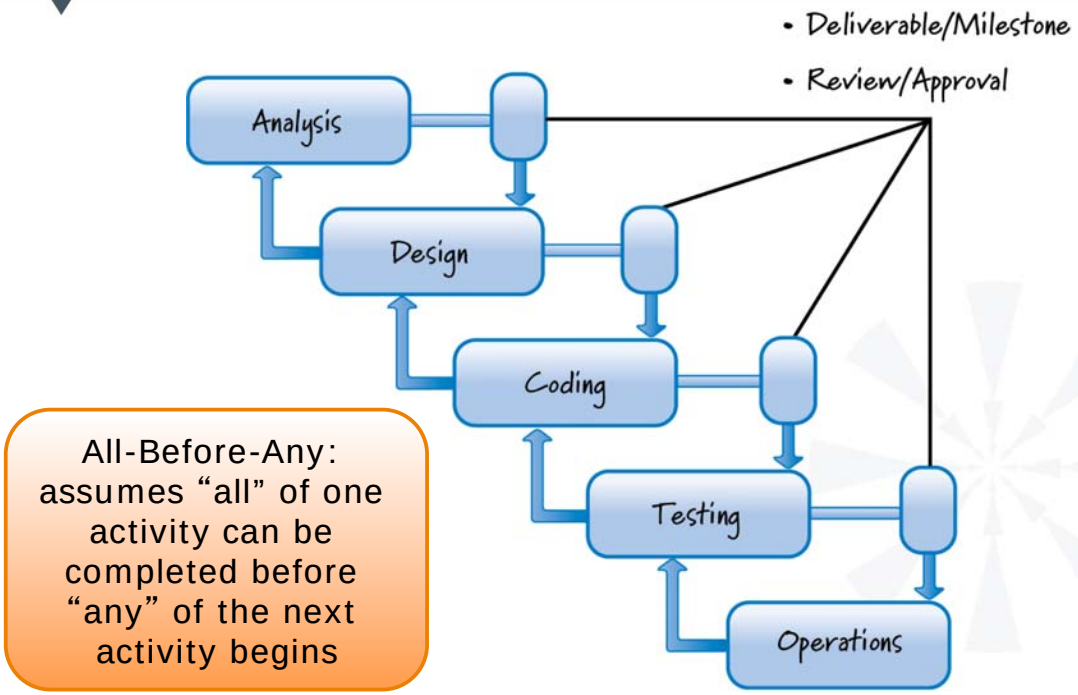


Described in detail in Chapter 3 of my book



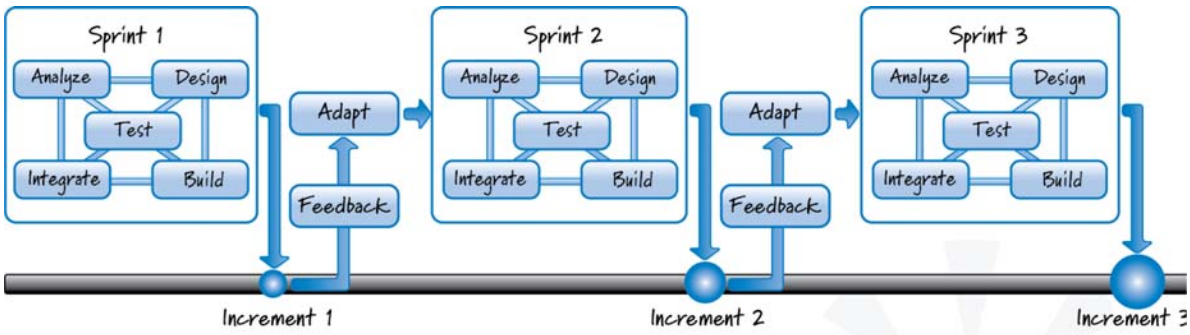


All before any is “risk generating”



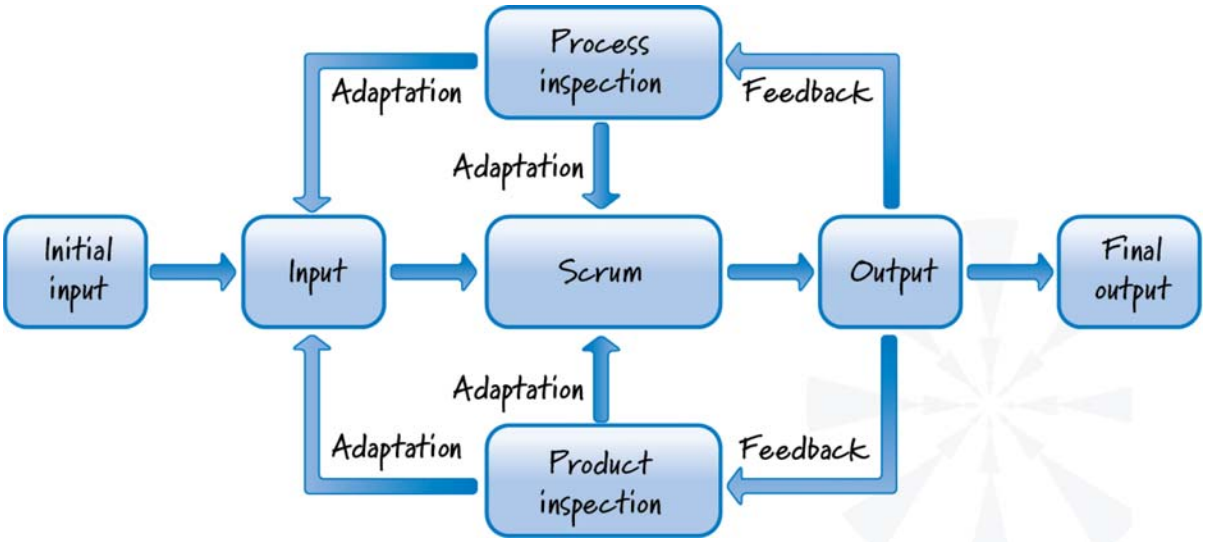
All-Before-Any: assumes “all” of one activity can be completed before “any” of the next activity begins

Agile is iterative & incremental

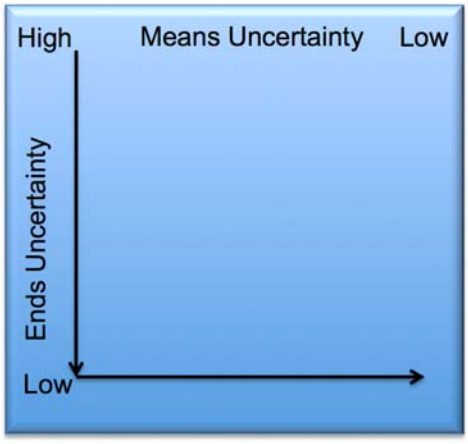


- Get things wrong before we get them right
- Build some of it before you build all of it
- Reduces forecasting errors
- Offers opportunity for continuous deployment

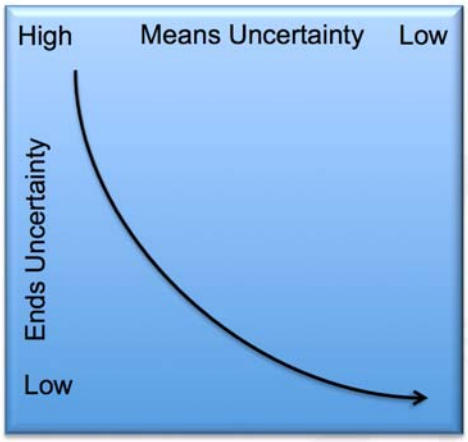
Agile is an empirical process model



Address all forms of uncertainty simultaneously

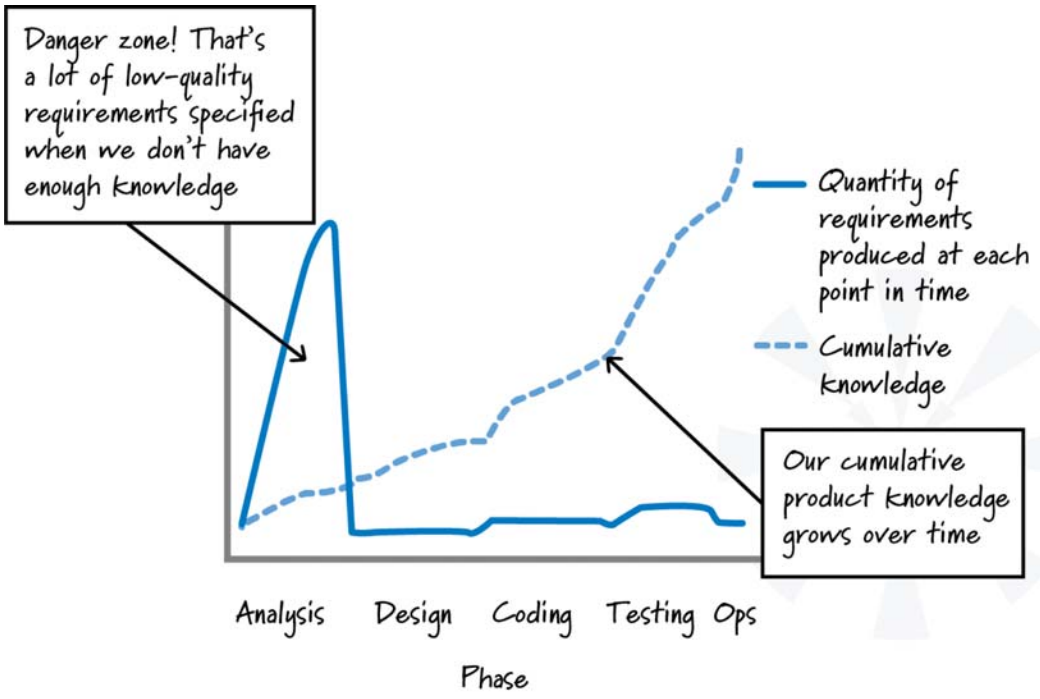


Defined

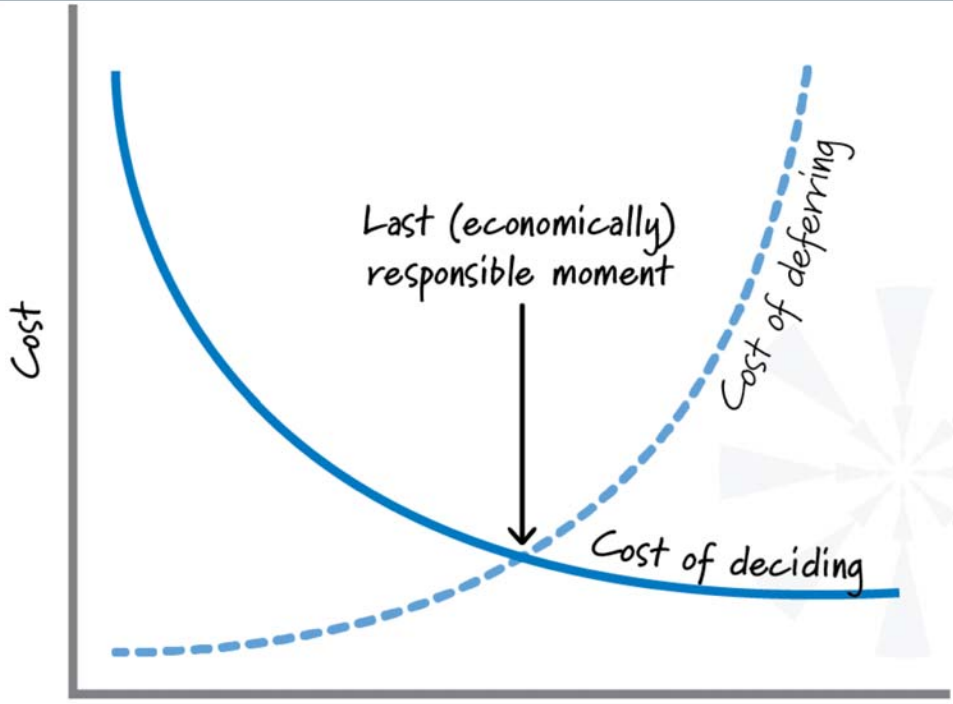


Empirical

Risk of trying to get it right upfront



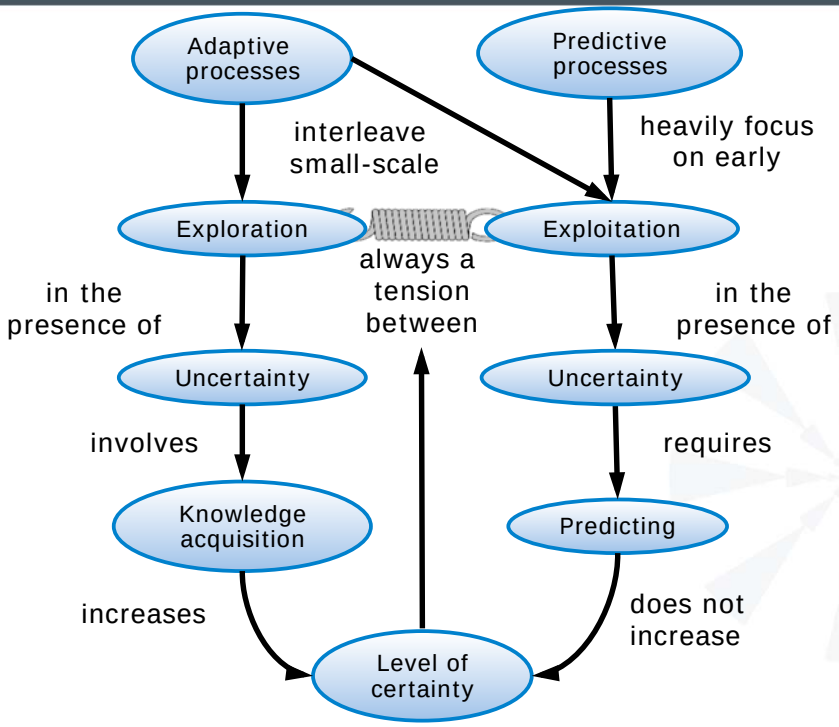
Keep options open (last responsible moment)



Exercise – Architecture A vs. B

First day of a new product development effort. There are two architectural choices: A or B. Each appears to have viable characteristics. Which one should we select?

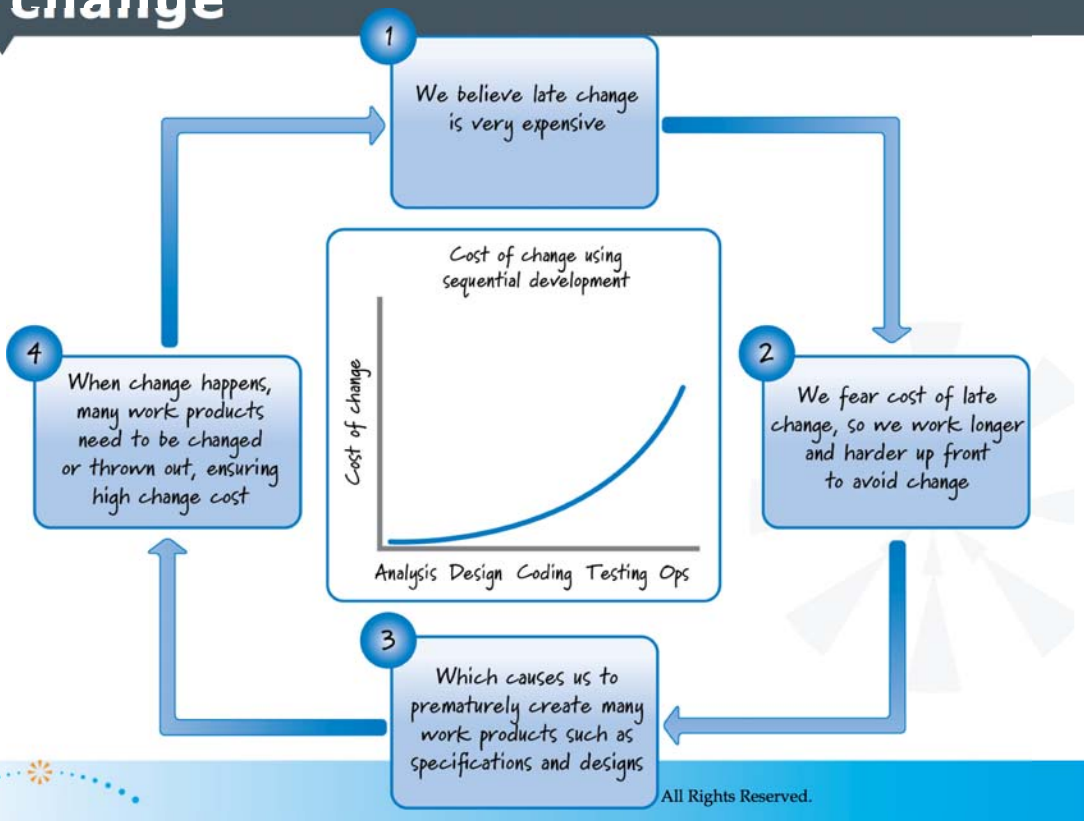
Rapidly intermingle exploration and exploitation



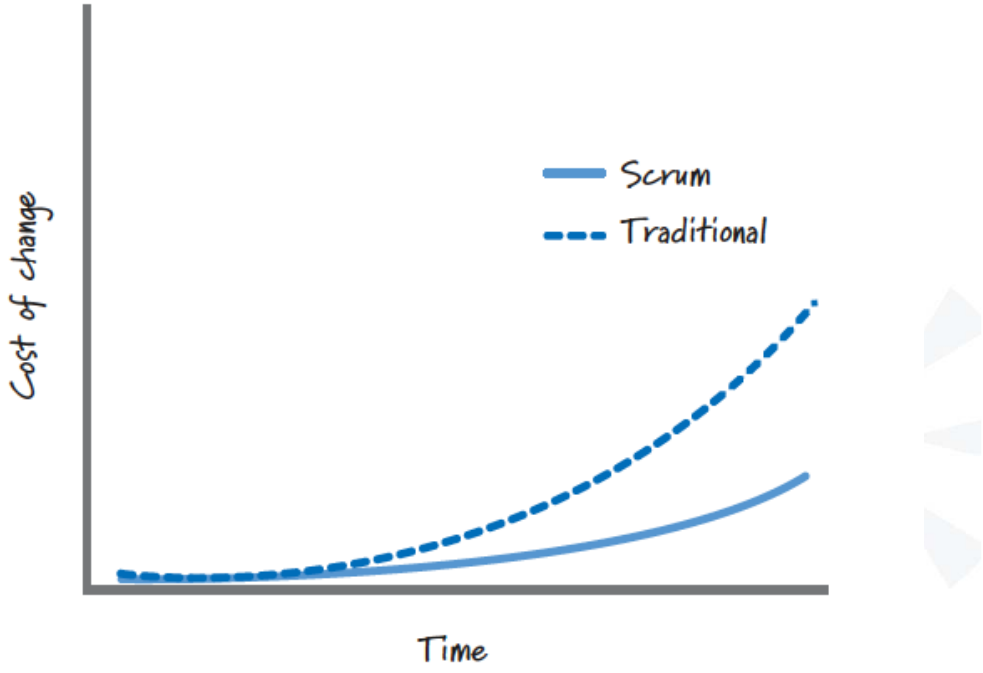
Real options

- The right but not the obligation to do something
- Options have value
- Options expire
- Never commit early to an option unless you know why

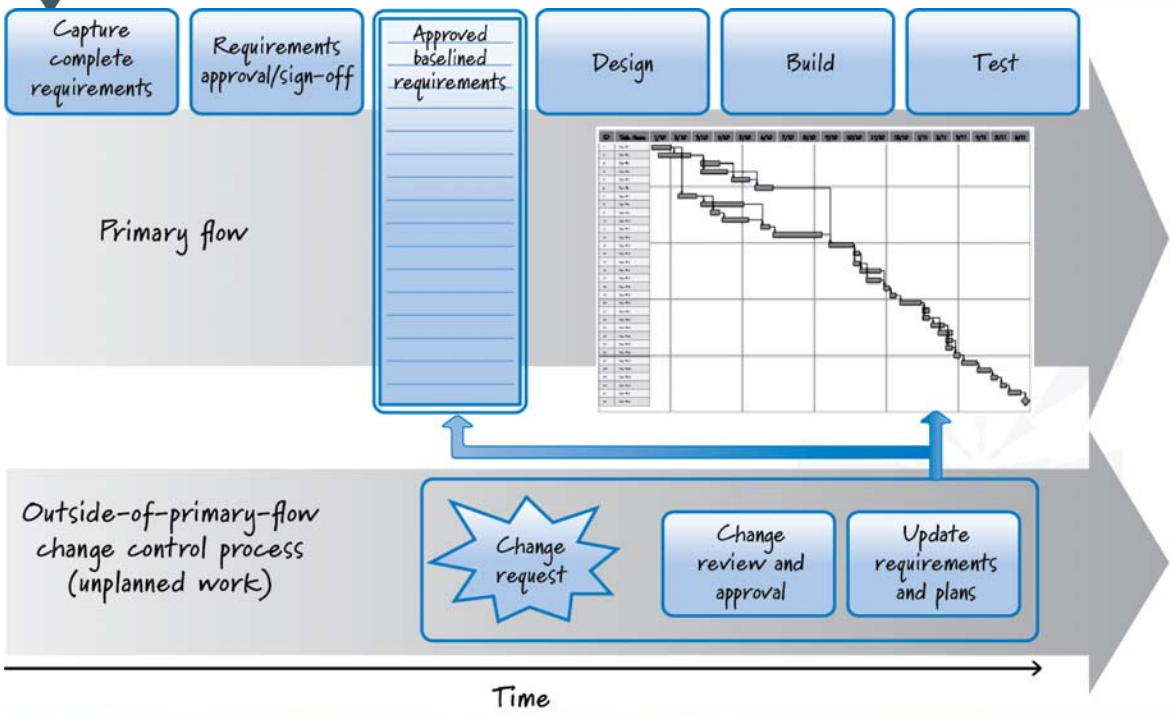
The risk of trying to eliminate change



Reduce risk by flattening the cost of change curve



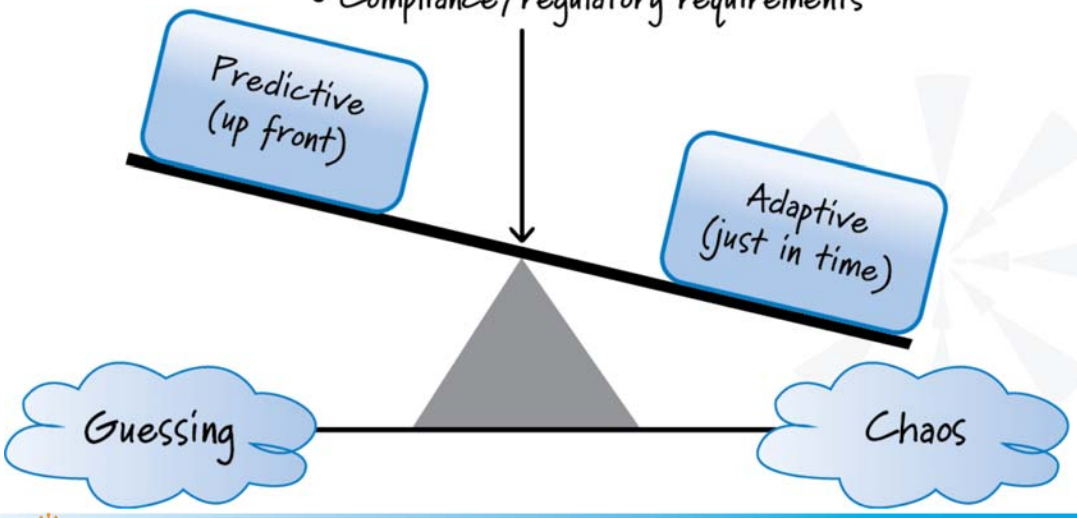
Managing change risk during a traditional development project

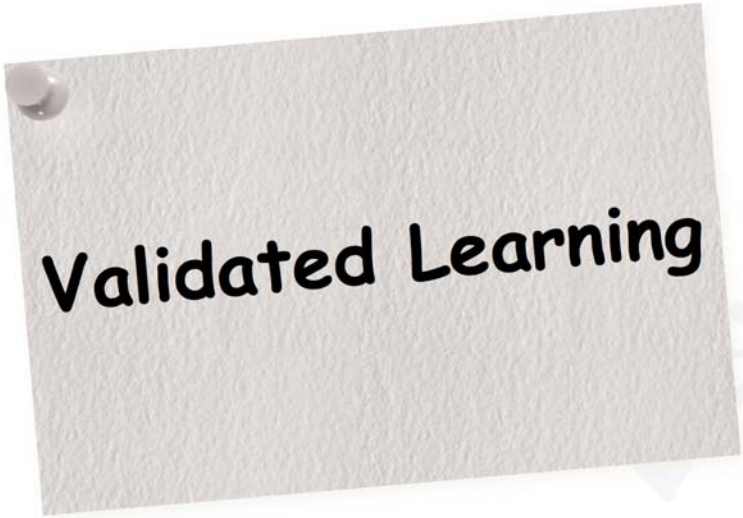


Managing change risk using Scrum

Upfront work should be helpful without being excessive

- Type of product
- Degree of end uncertainty
- Degree of means uncertainty
- Constraints on development
- Compliance/regulatory requirements





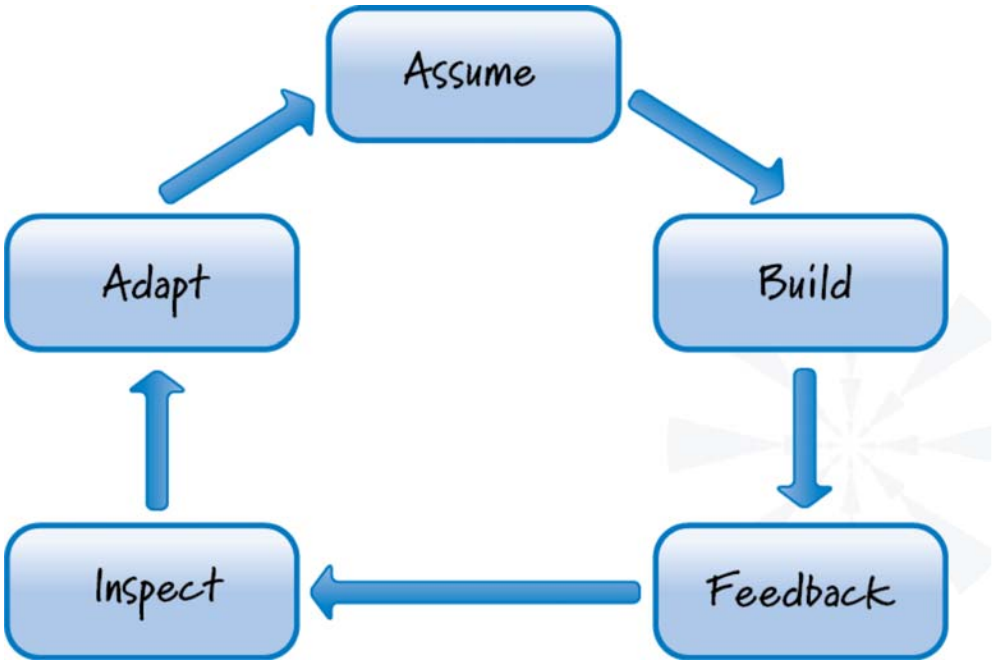
Assumptions = accrued risk

Assumption is a guess or belief that is assumed true, real, or certain

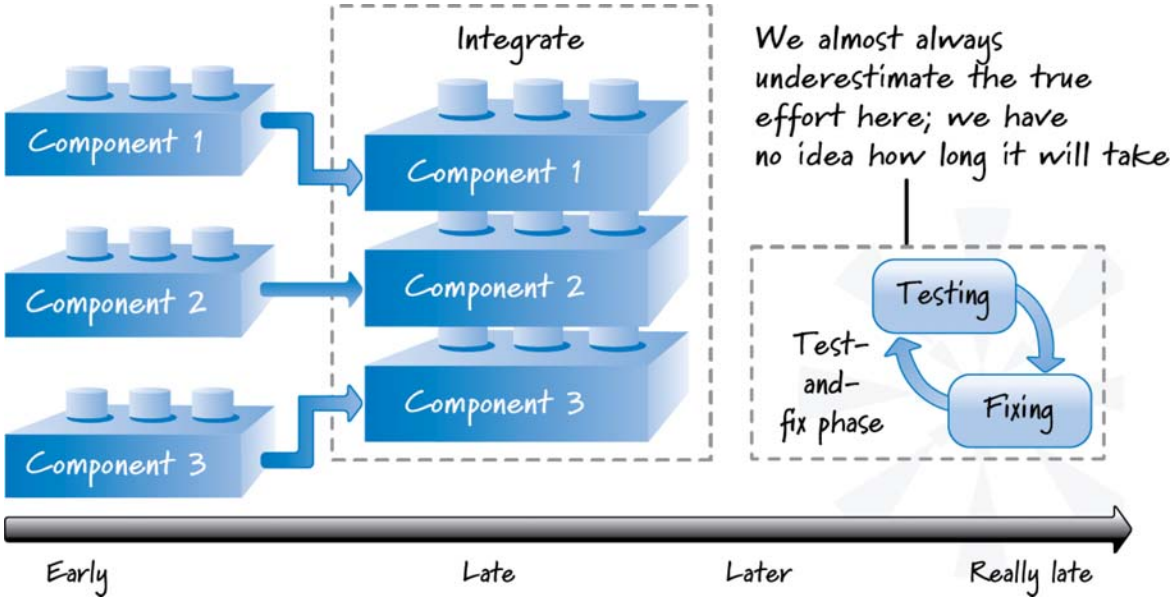
Not-yet-validated assumptions represent significant accrued risk during development

Don't let important assumptions live long without validation

Reduce risk by going fast through the loop



Organize flow of work for fast feedback



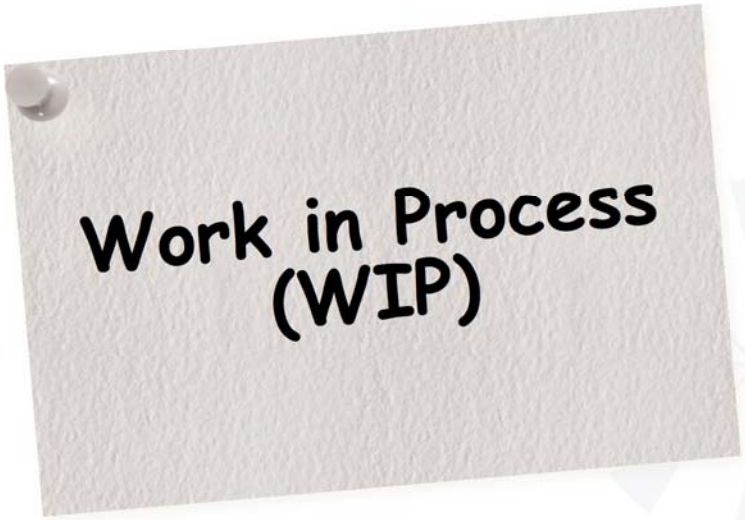
Fast feedback is antifragile

Agile benefits from the uncertainty (unpredictable things we learn) in fast, frequent feedback

Learn fast you are going down the wrong path you can truncate the path

Exploit newly acquired knowledge to realize an emergent opportunity

Asymmetric payoff by limiting downside harm and providing much greater upside potential



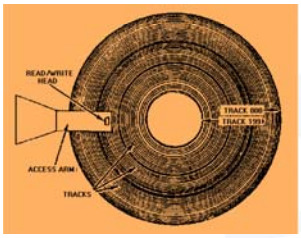
Use economically sensible batch sizes

- Reduced cycle time
- Reduced flow variability
- Accelerated feedback
- Lower risk of failure
- Reduced overhead
- Increased motivation & urgency
- Reduced cost and schedule growth

Inventory (WIP) represents a flow risk

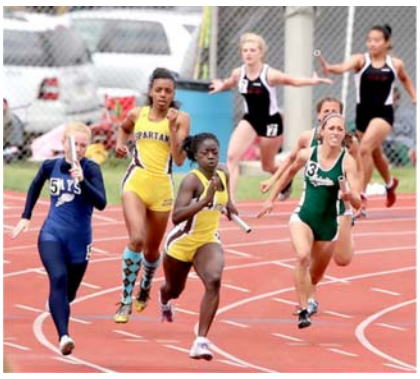
Manufacturing inventory is both physically and financially visible

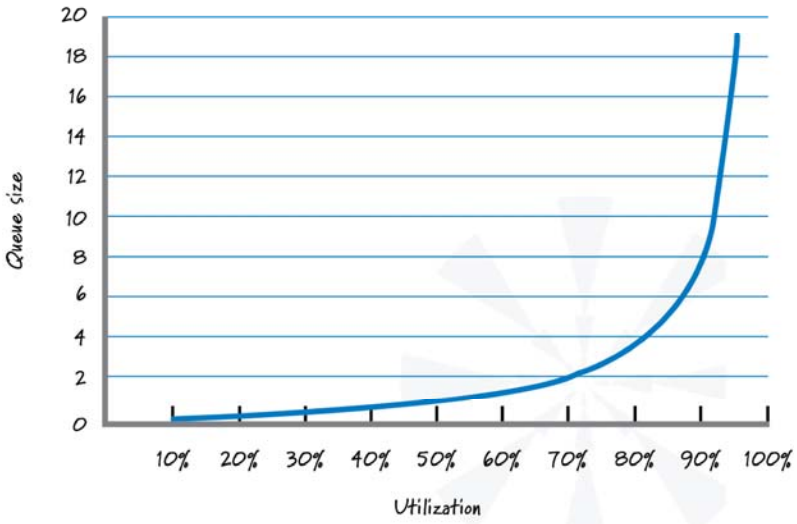
Product-development inventory are knowledge assets that aren't visible in the same way as physical parts



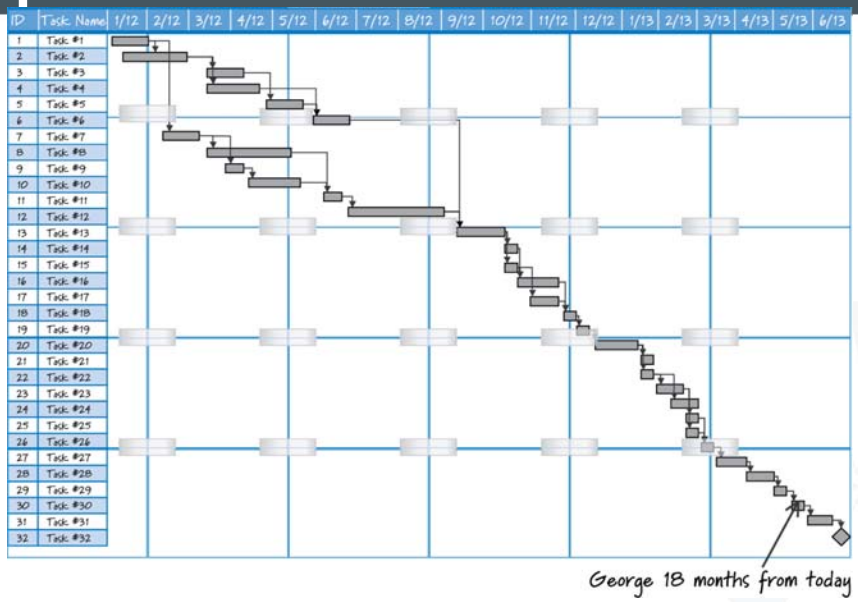
Risk is idle work, not idle workers

Watch the Baton Not the Runners



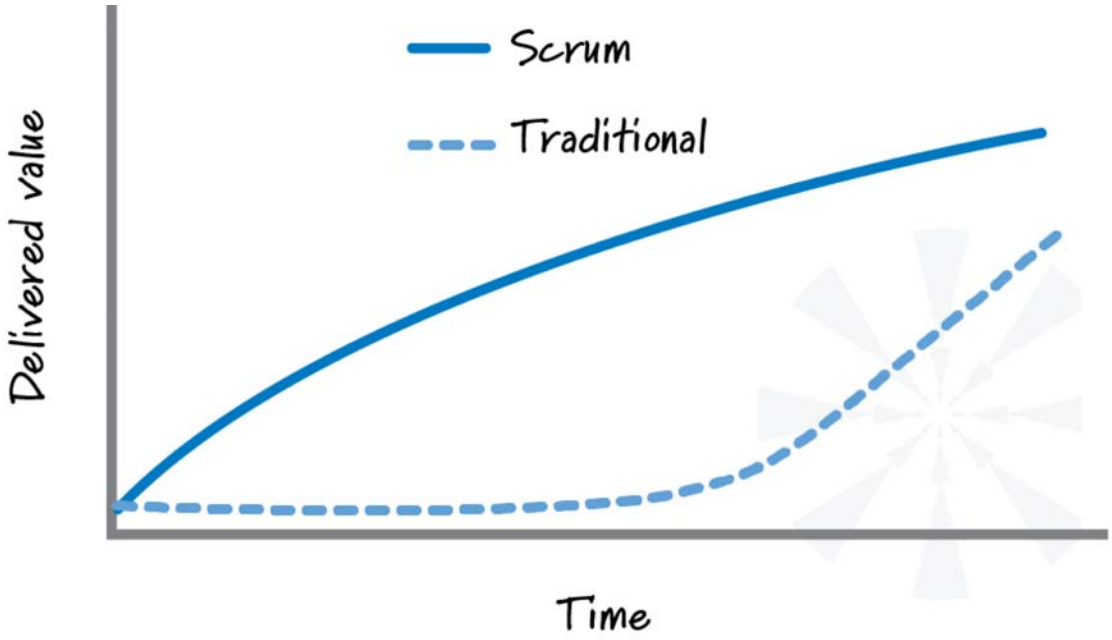


Adapt to real-time information and replan

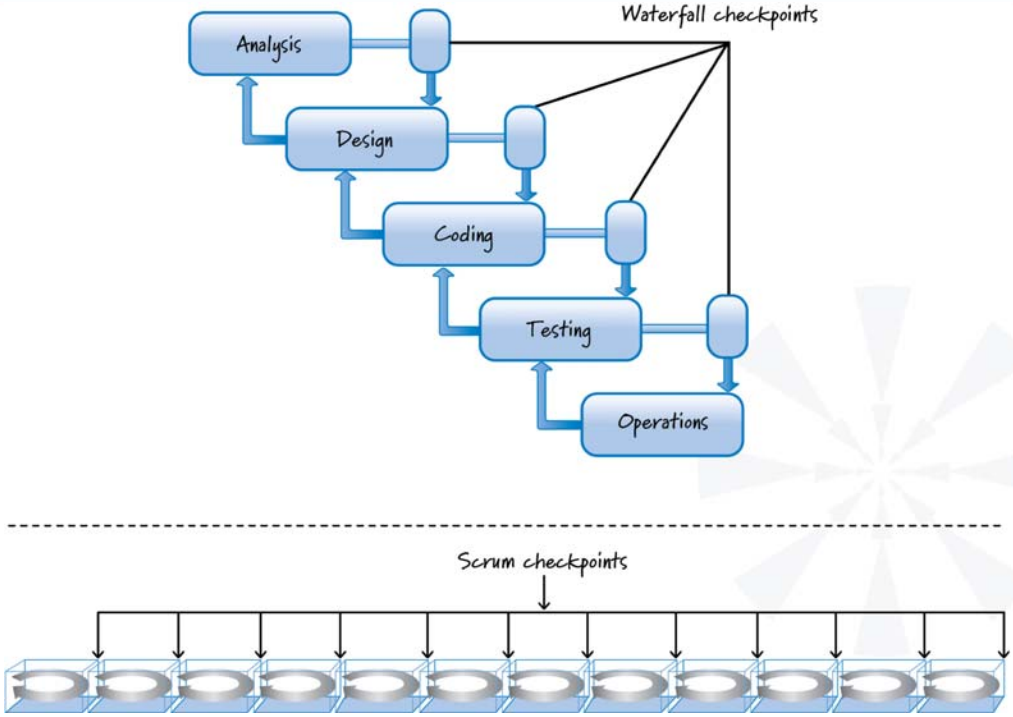


Using Agile, progress is measured by validating working assets (e.g., features)

Reduce risk by focusing on value-centric delivery



More frequent checkpoints reduce risk



Belief that loading planning on the front-end reduces risk

- Better the planning the better the understanding and therefore the better the execution
- Give appearance of orderly, accountable, and predictable approach
- Developing a product rarely goes as planned
- Beliefs don't match uncertainty in product development

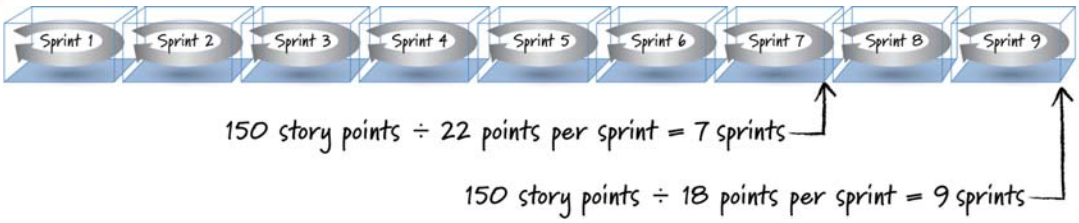
Scope is the risk-reducing degree of freedom

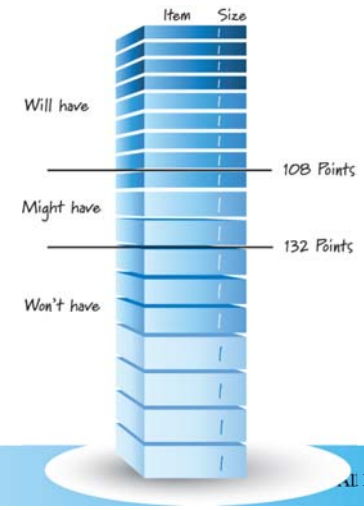
Scope can be binary

Scope can be shades of grey

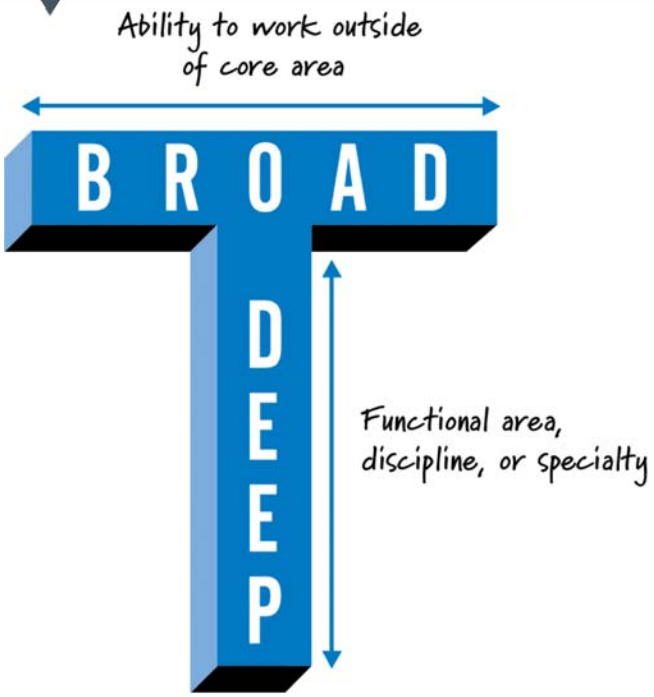
Allows us to bound the downside on the asymmetric payoff function

Communicate uncertainty with range answers to questions





Teams with T-Shaped skills



Can offset random increases in demand by quickly compensating with changes in capacity

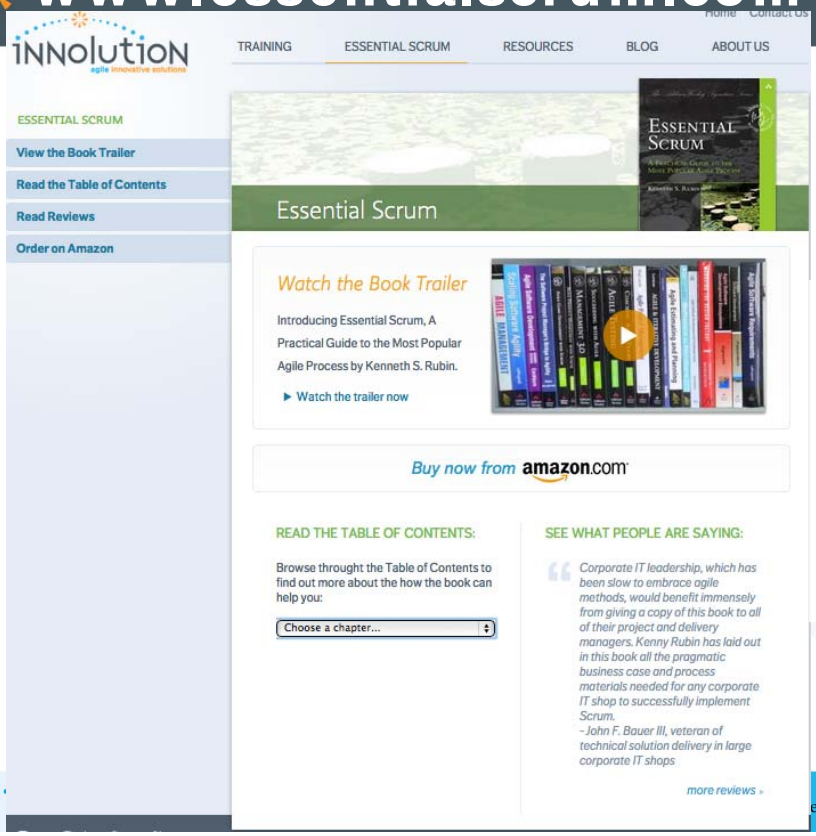
Example of negative covariance or counter balancing (one random variable counterbalances the other)

Visual AGILExicon®

- Slides in this presentation contain items from the Visual AGILExicon®, which is a trademark of Innolution, LLC and Kenneth S. Rubin.
- The Visual AGILExicon is used and described in the book: ***Essential Scrum: A Practical Guide to the Most Popular Agile Process***.
- You can learn more about the Visual AGILExicon and permitted uses at: http://innolution.com/resources/val-home-page

Visual AGILExicon®

www.essentialscrum.com



Contact Info for Ken Rubin

iNNOlution
agile innovative solutions

Email:	krubin@innolution.com
Website:	www.innolution.com
Phone:	(303) 827-3333
LinkedIn:	www.linkedin.com/in/kennethrubin
Twitter:	www.twitter.com/krubinagile
Facebook:	www.facebook.com/InnolutionLLC
Google+	plus.google.com/+KennyRubin1/
Essential Scrum: A Practical Guide to the Most Popular Agile Process	www.essentialscrum.com
Comparative Agility Website	www.comparativeagility.com