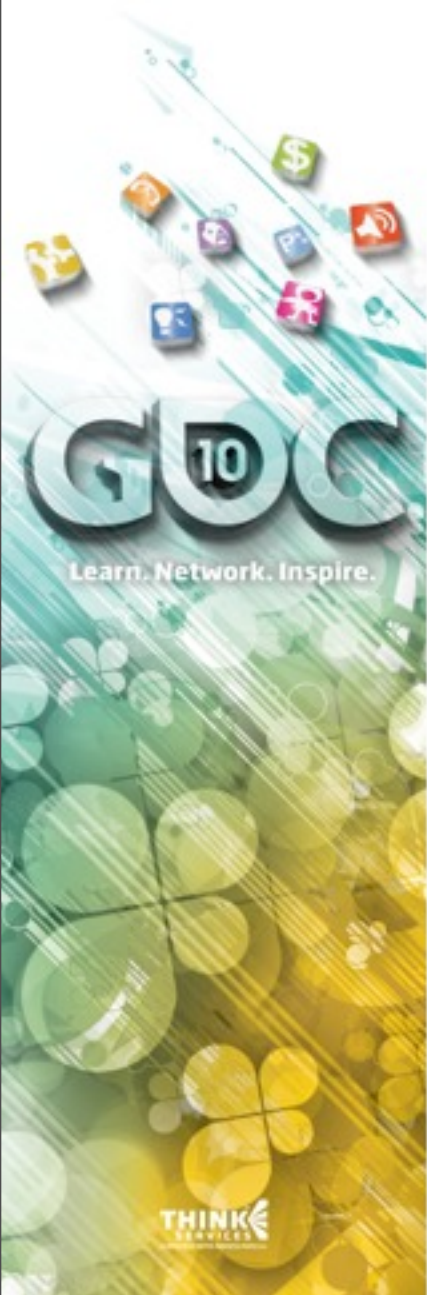


GD10C

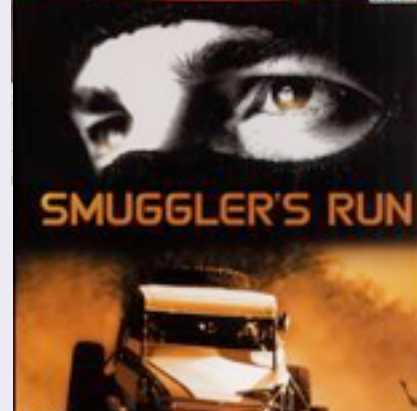
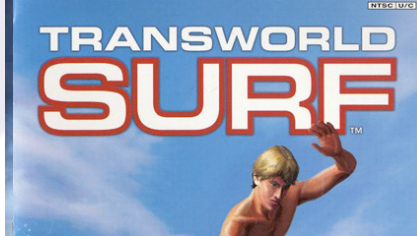
Learn. Network. Inspire.

www.GDConf.com



Beyond Scrum

Clinton Keith - Background



Agile coach and Scrum trainer

- 24 years of development experience
- 14 years of game development experience
- 7 years of agile development experience in the game industry
- Introduced agile to the industry in March 2005

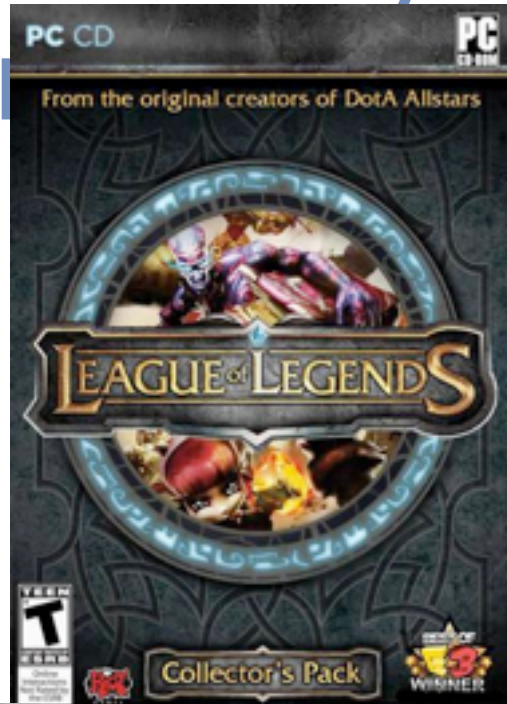
Beyond Scrum

- Filling out the framework
- Applying project management practices where needed
- Using common sense, not dogma



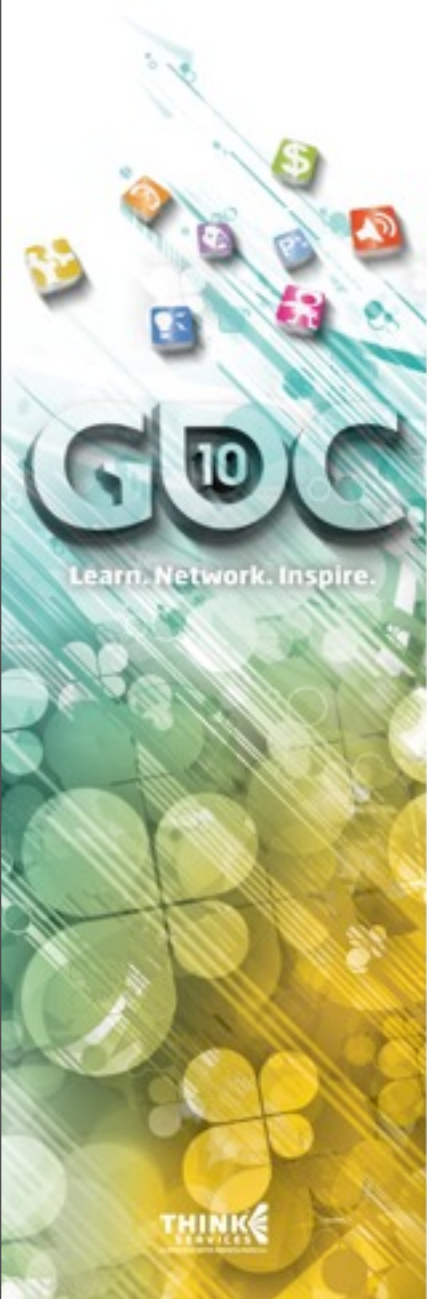
No rules, just shared practices

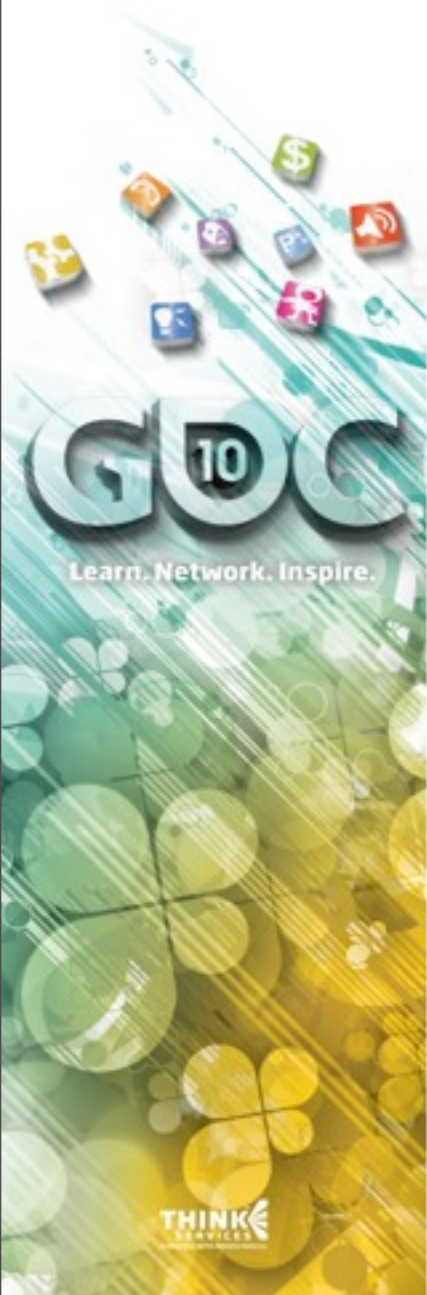
Games Recently Shipped That



Game Development Challenge

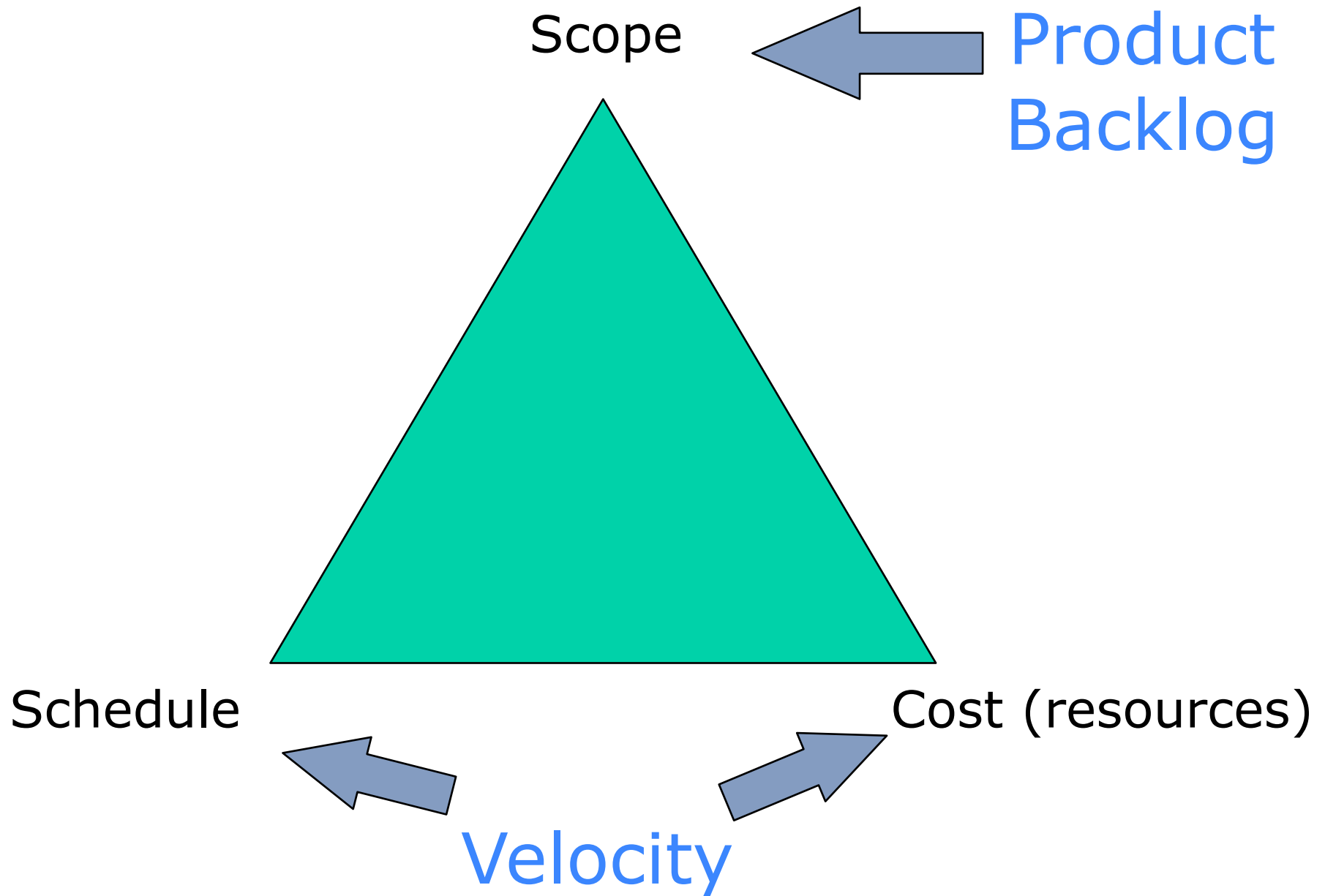
- Manage
 - Fixed ship dates
 - Minimum required feature sets
 - Long project cycles with production
- While avoiding crunch



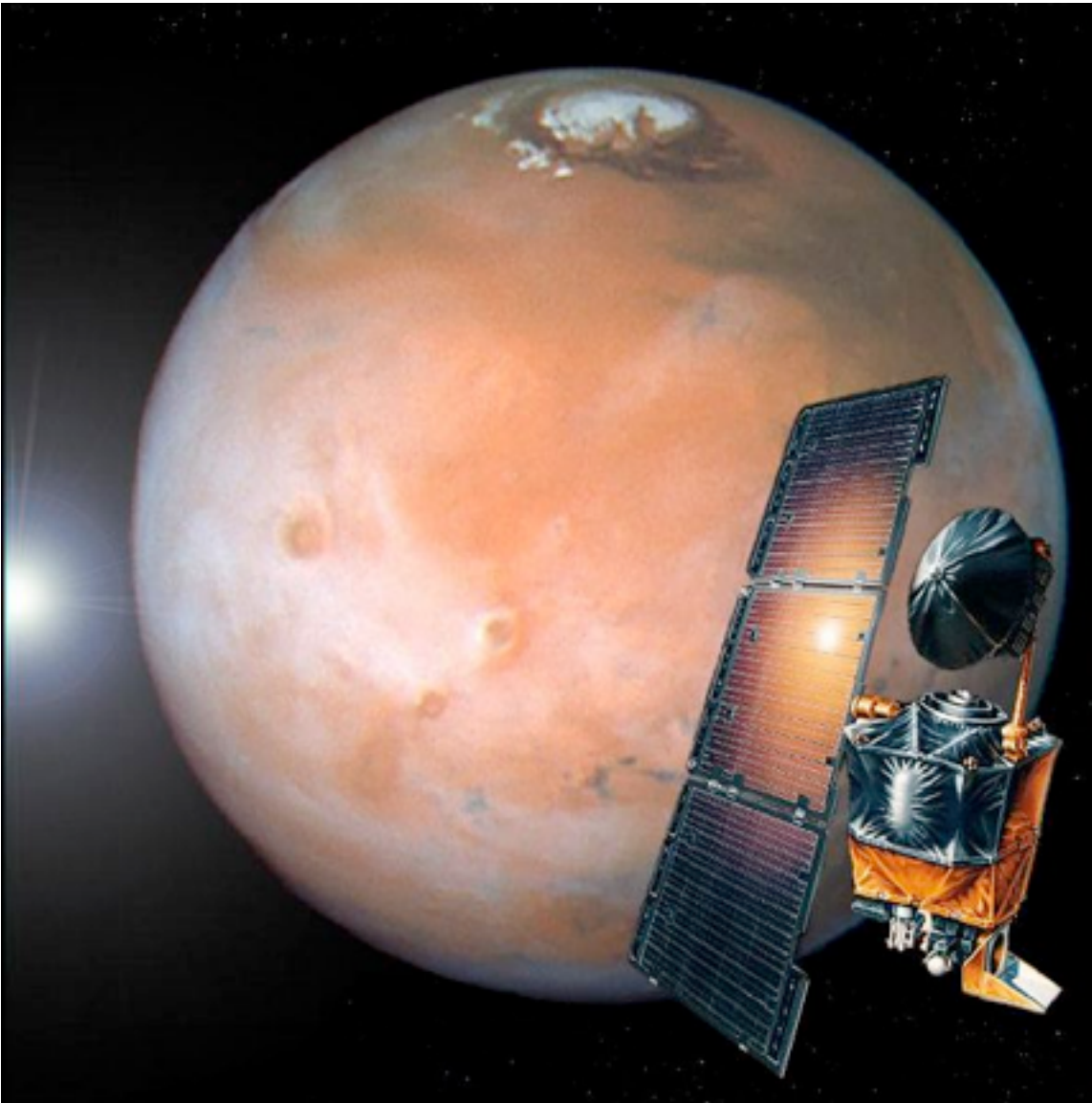


Fixed Ship Dates

The Project Management Triangle

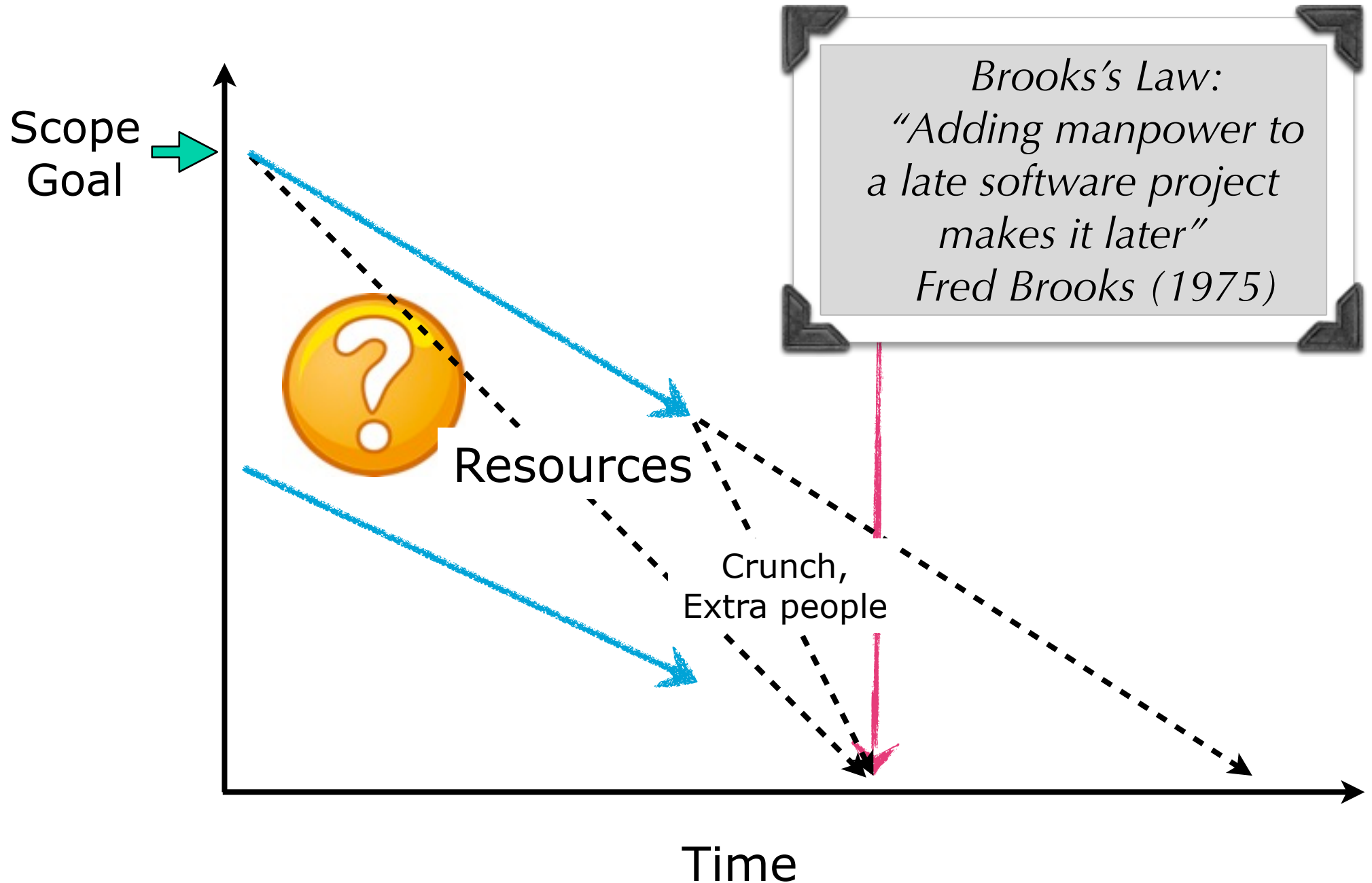


Better & Cheaper & Faster



© 2010 Clinton Keith Consulting

Velocity & Fixed Schedules



There are often scope limits

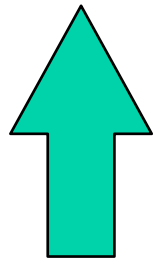
- Minimum marketable features
 - Cutting scope by 50% to meet a date may produce a product that won't sell
 - Example: A Word Processor without a printing feature

Minimum Marketable Features
+ Fixed Schedule = RISK

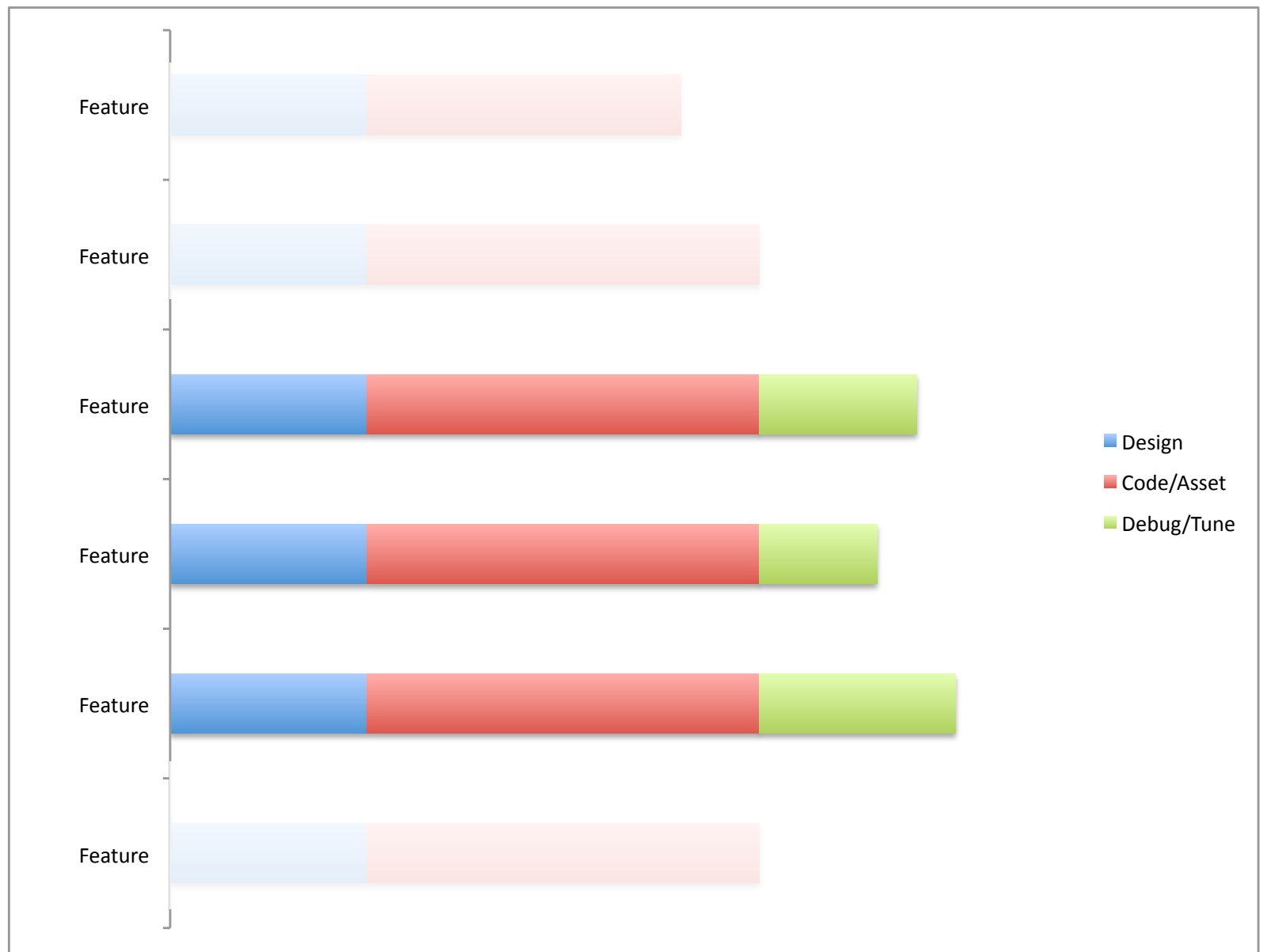
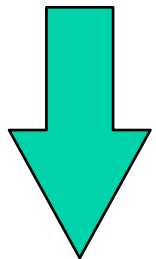
Applying PM, Attacking Risk

- Identify minimum marketable feature set
 - MuSCoW analysis
- Identify critical paths
 - Example: Game running on the PS3
- Identify path risks
 - Example: Graphics throughput on the RSX
- Prioritize work based on risk
 - E.g. Spike to demonstrate demo level on PS3 with maximum desired detail
- Find and address bad news early
 - Example: Need to create lower resolution levels

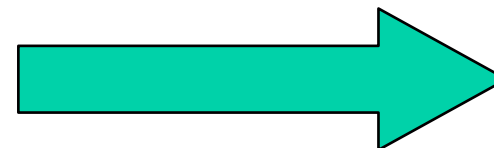
High
Value



Low
Value



Progress



MuSCoW Analysis

Must Have

Should Have

Could Have

Won't Have



Minimum Required Feature Set Must Haves

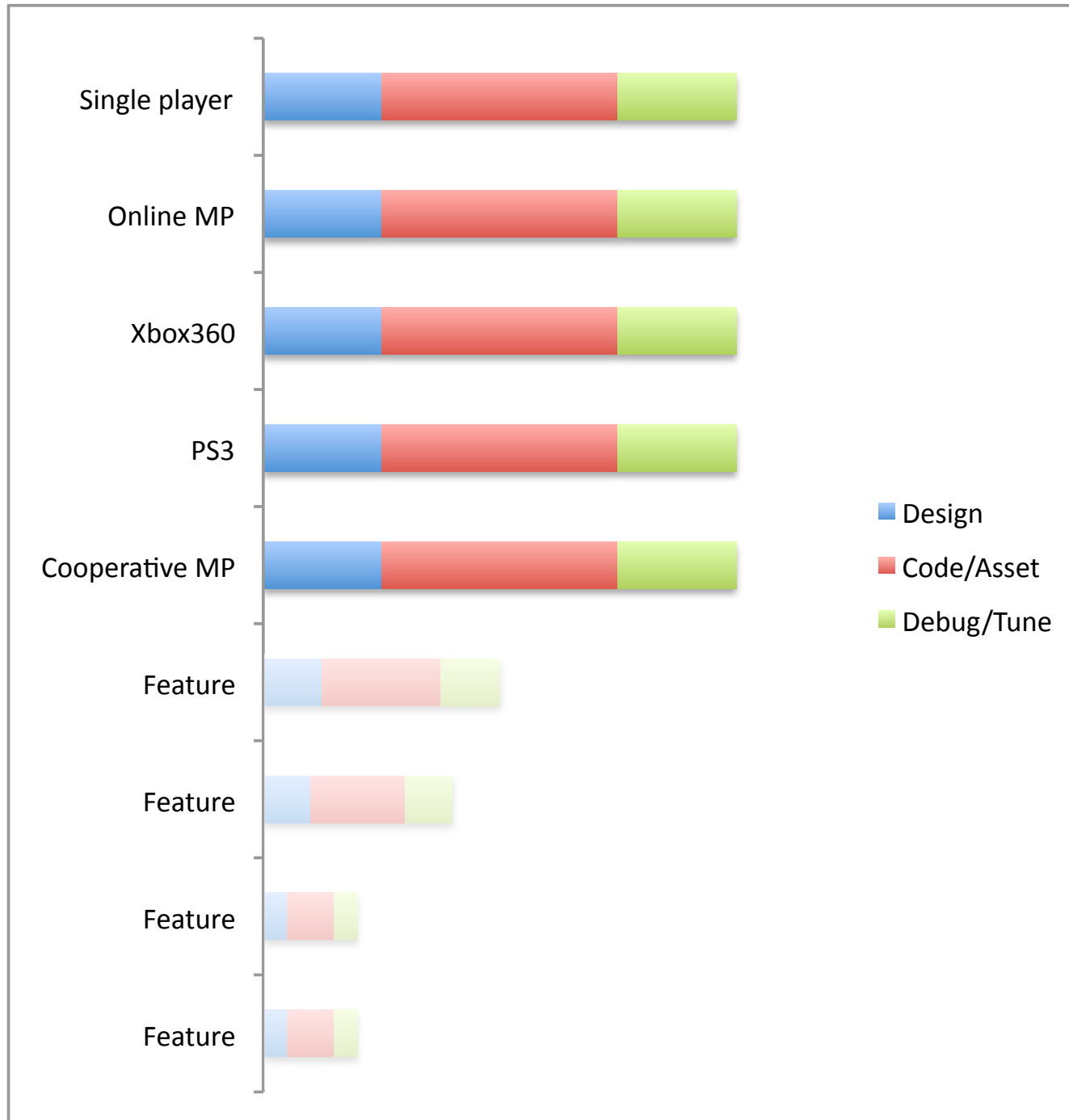
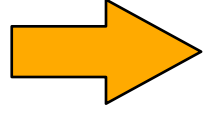
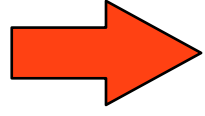
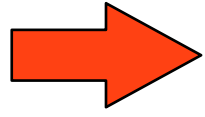
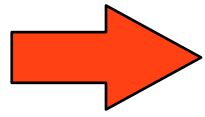
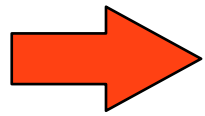
- Example: Console FPS
 - 8 + hours single-player campaign
 - Online multiplayer
 - XBox 360
 - PS3

Dropping any of these means failure

Minimum Required Feature Set Should Haves

- Example: Console FPS
 - Cooperative Online Multiplayer

**These add great value, but aren't
absolutely necessary**



Backlog Prioritization



Value



Cost

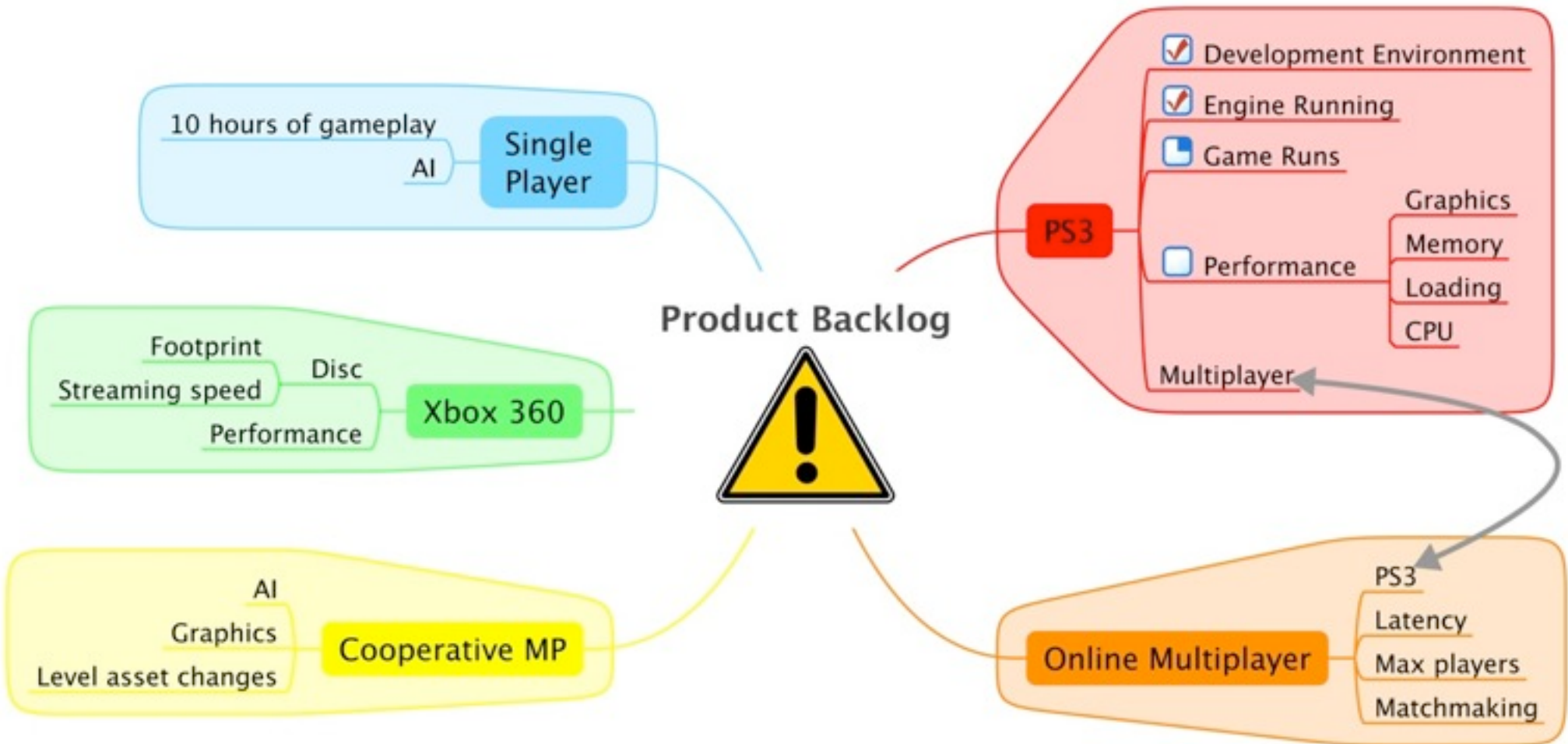


Knowledge



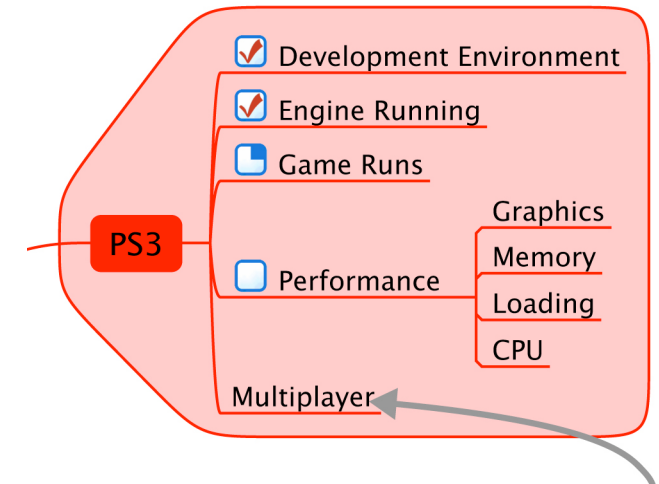
Risk

Risk Prioritization

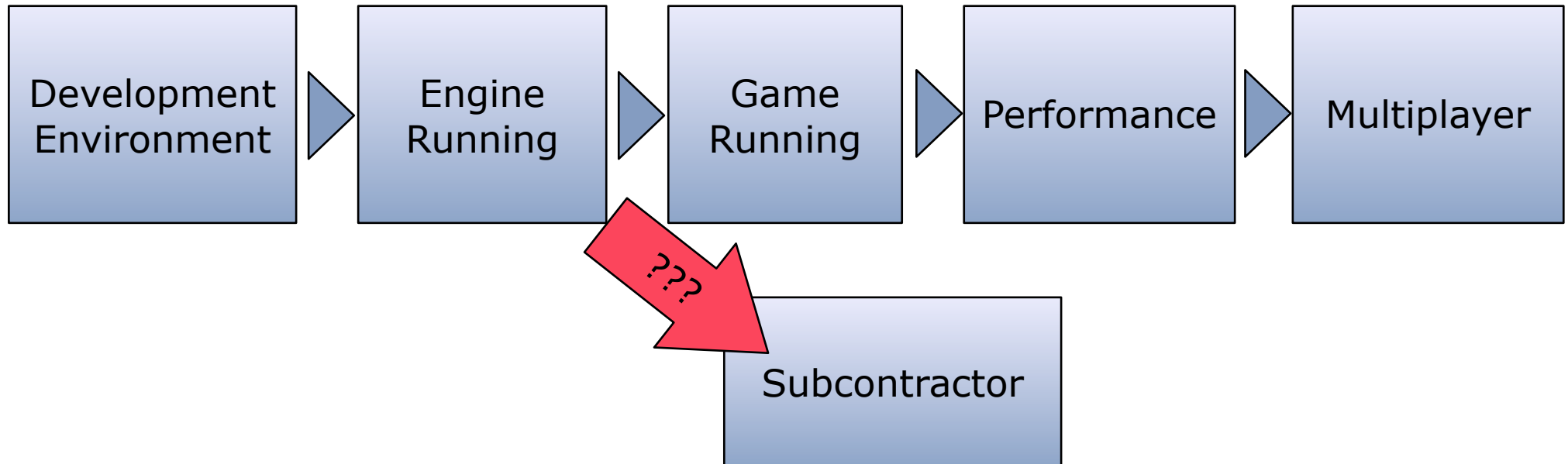


Critical Path

- The things that need to happen, in sequence.
 - Usually dependent on one another



PS3 Critical Path



Critical Chain Staffing

PS3
(6 people x 10
months)

Single Player
(20 people x 12 months)

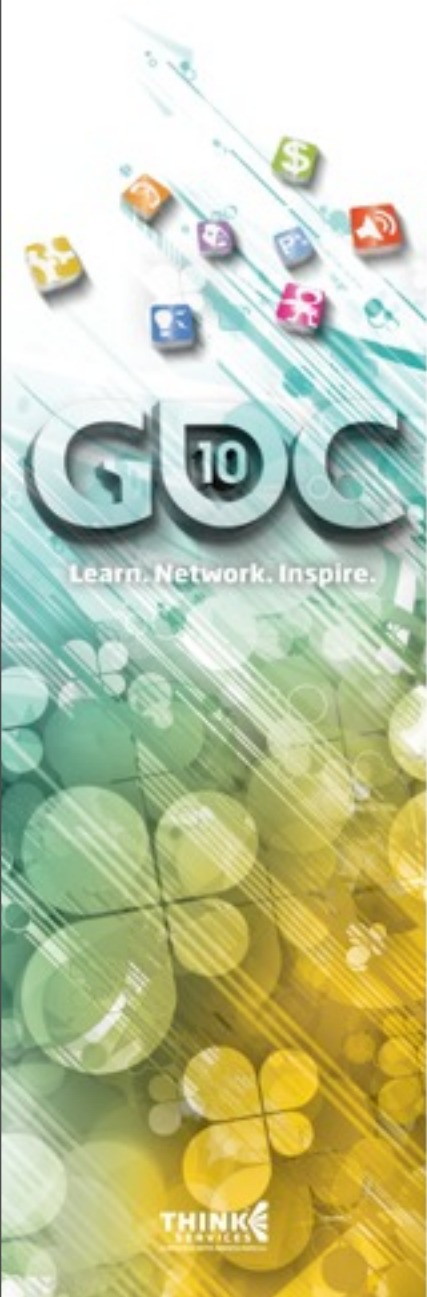
Xbox 360
(4 people x 16
months)

Multiplayer
(12 people x 8
months)

Cooperative MP
(4 people x 6
months)

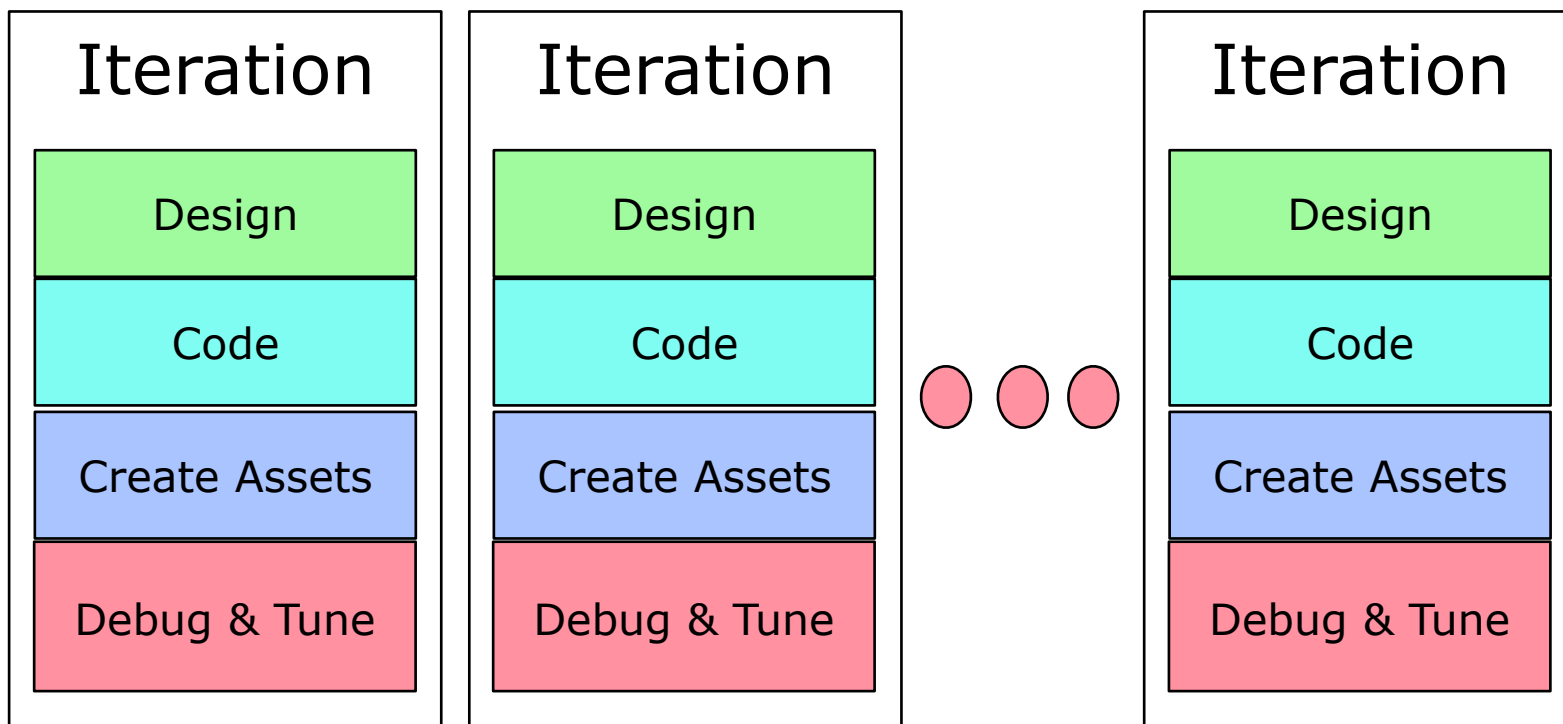
Putting it all together

- Have dedicated people/teams/time for each major feature
- Have them work off their own backlog
 - Watch out for
 - Inconsistent progress
 - Great Multiplayer, bad single player
 - “Feels like” different experiences
 - Not enough sharing
 - Example: Multiplayer on the PS3

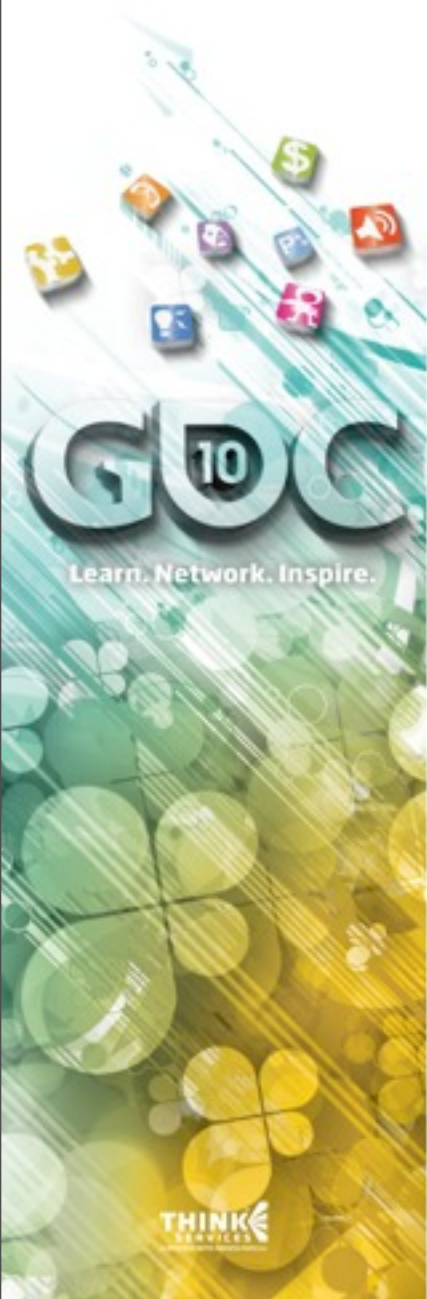


Production Risk

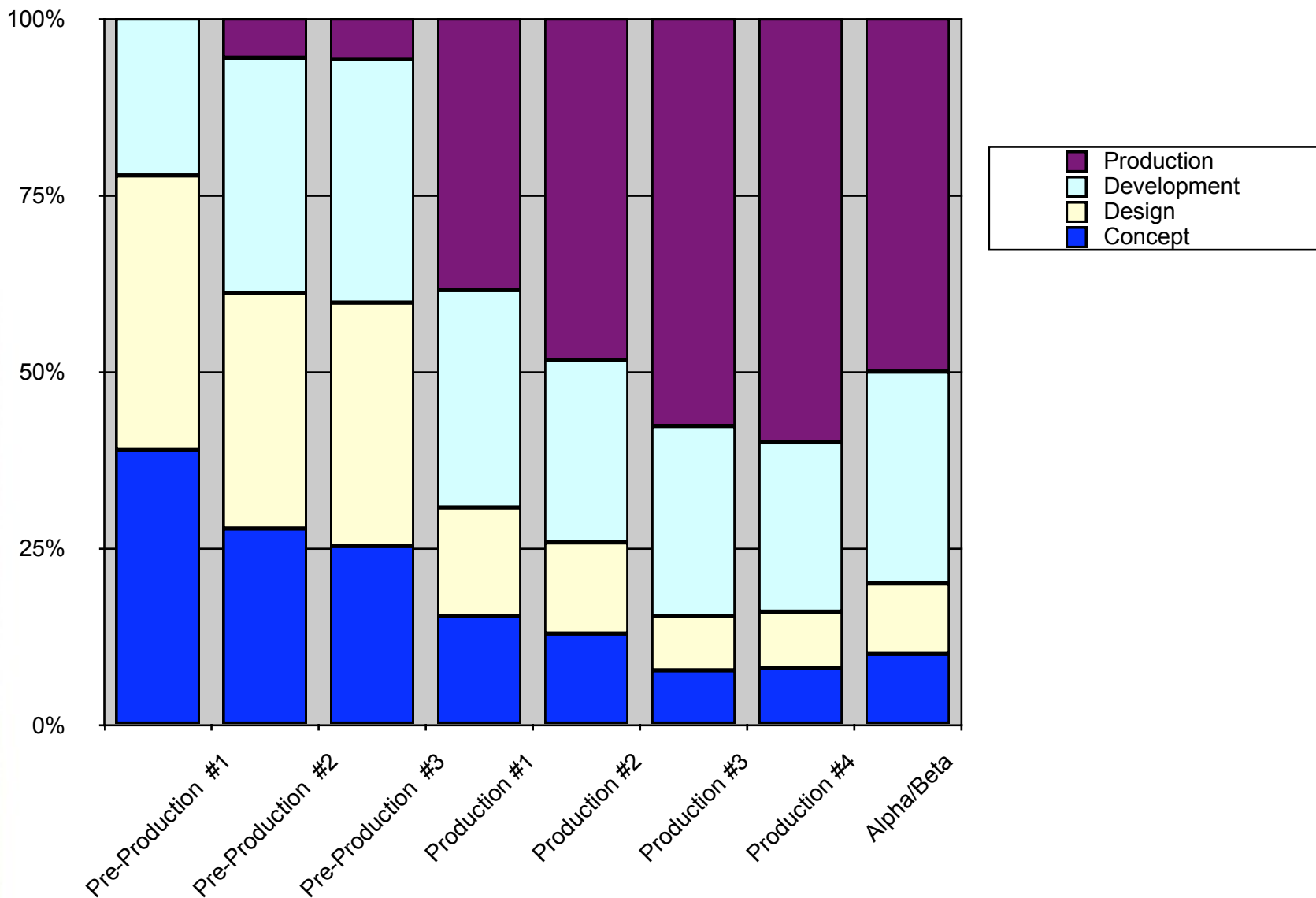
Agile is phase-less



...is game development?

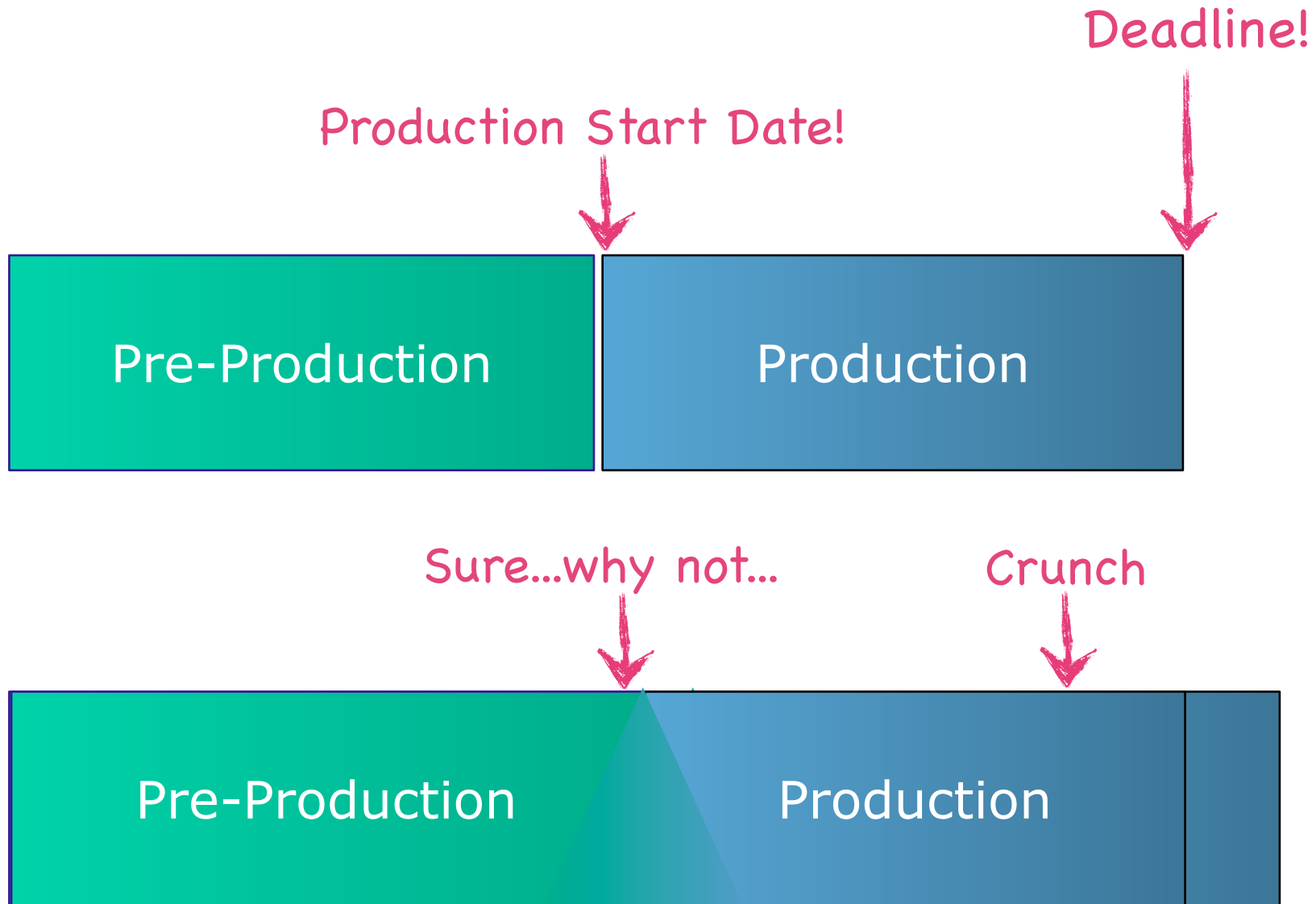


Not Quite

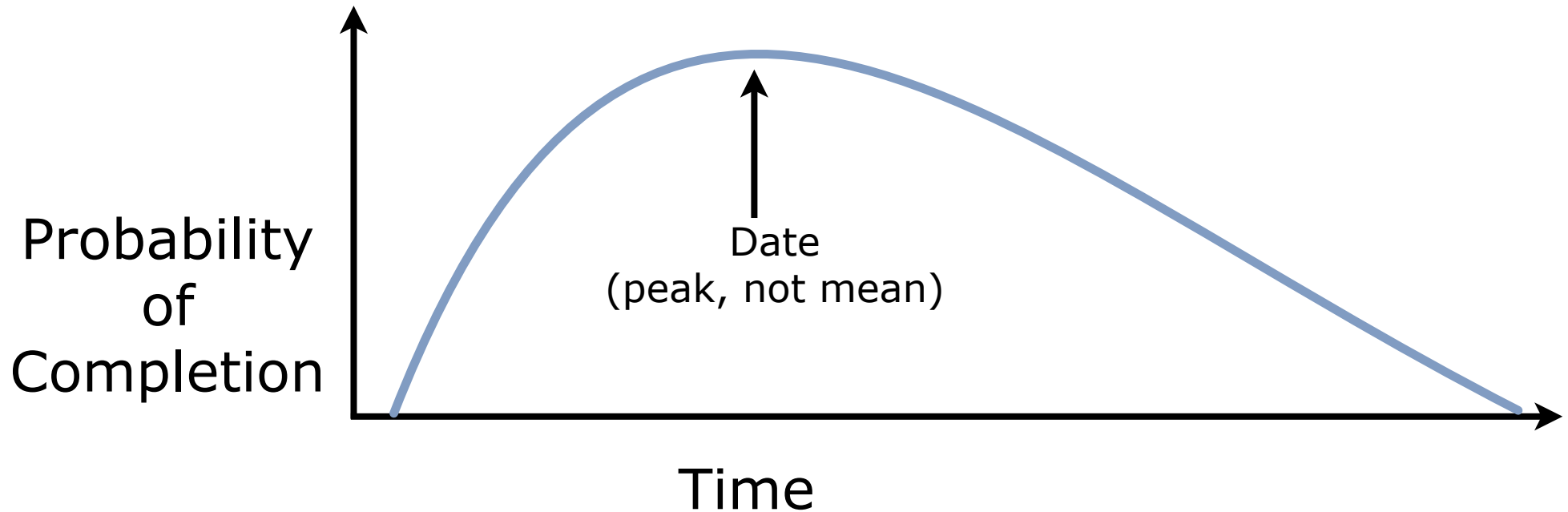
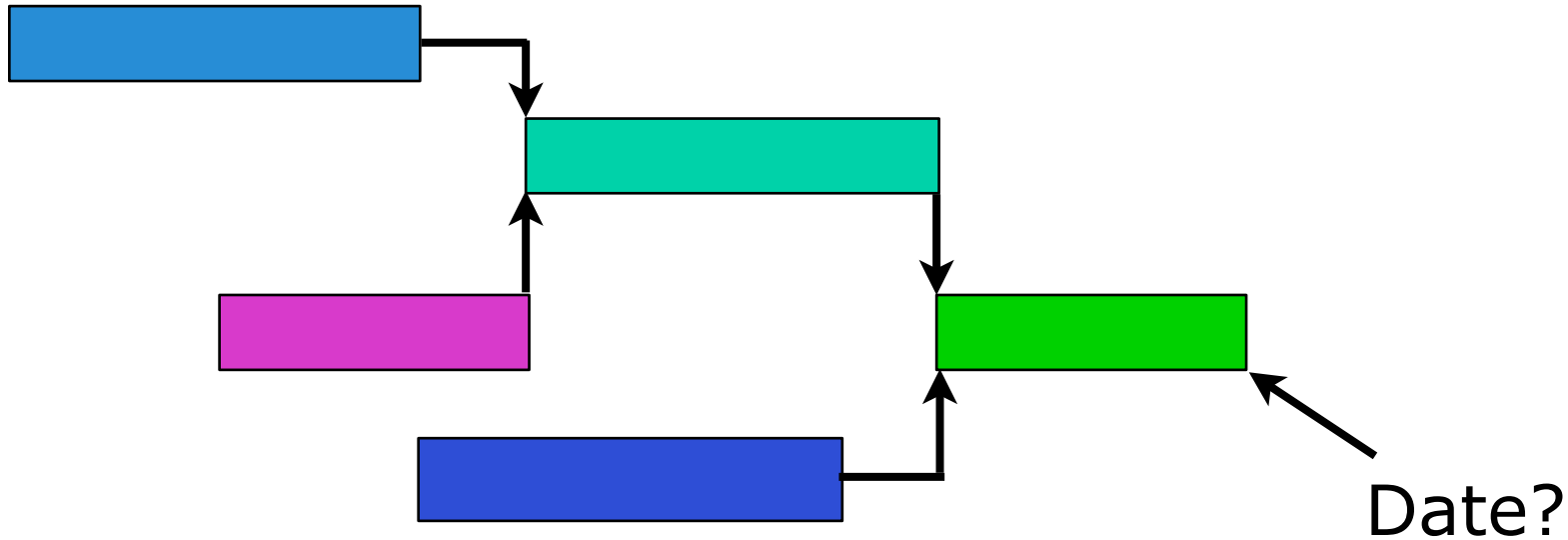


We have stages

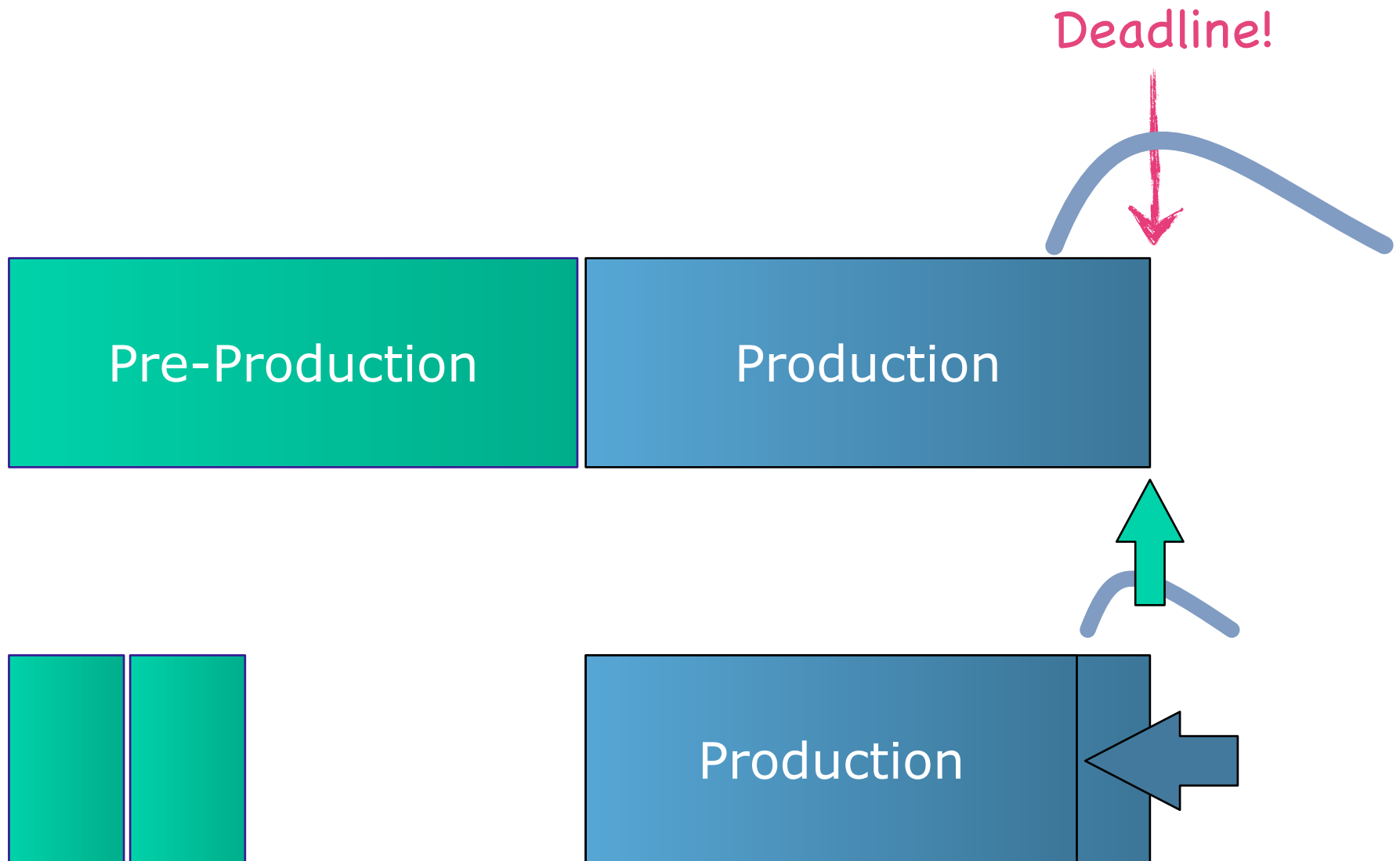
Production is Debt

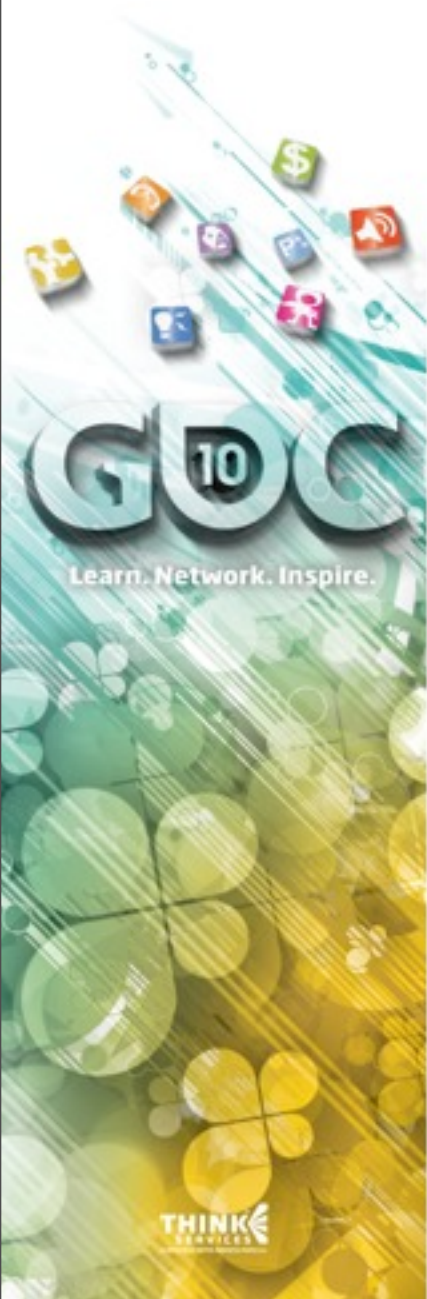


Schedule Probabilities



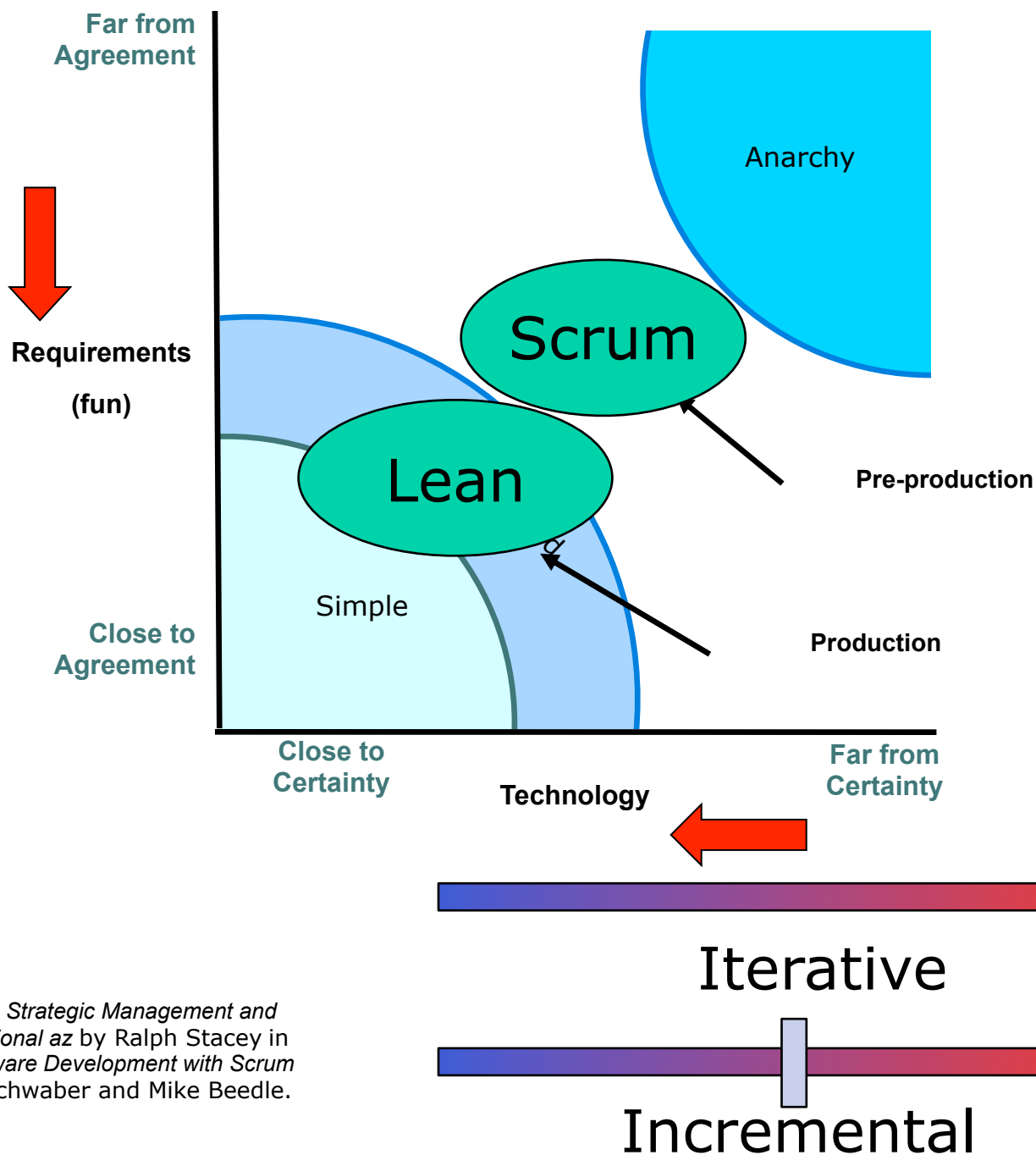
Refining Production Dates





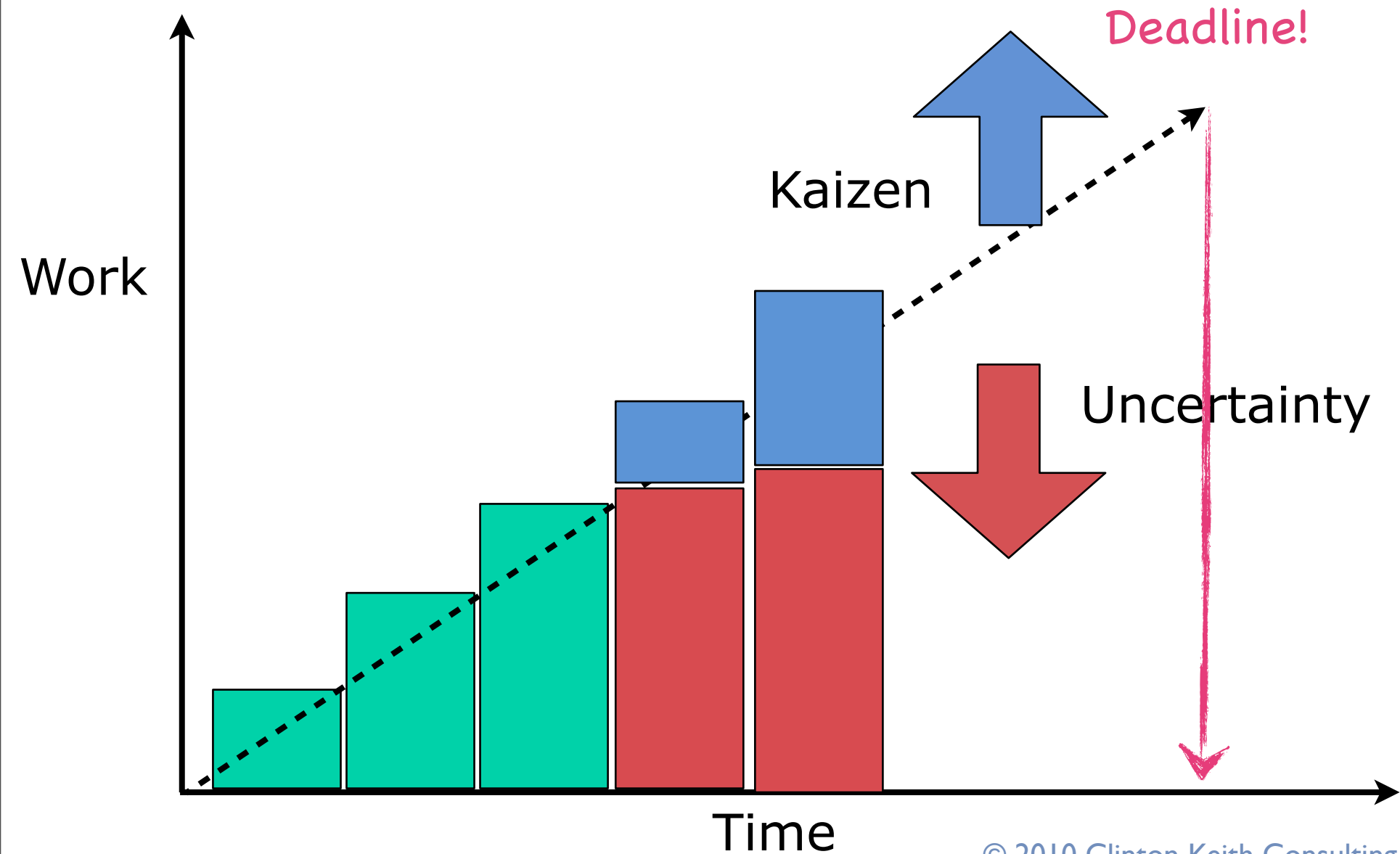
Selecting the right tools for pre-production and production

Process tools driven by certainty



Source: *Strategic Management and Organizational az* by Ralph Stacey in *Agile Software Development with Scrum* by Ken Schwaber and Mike Beedle.

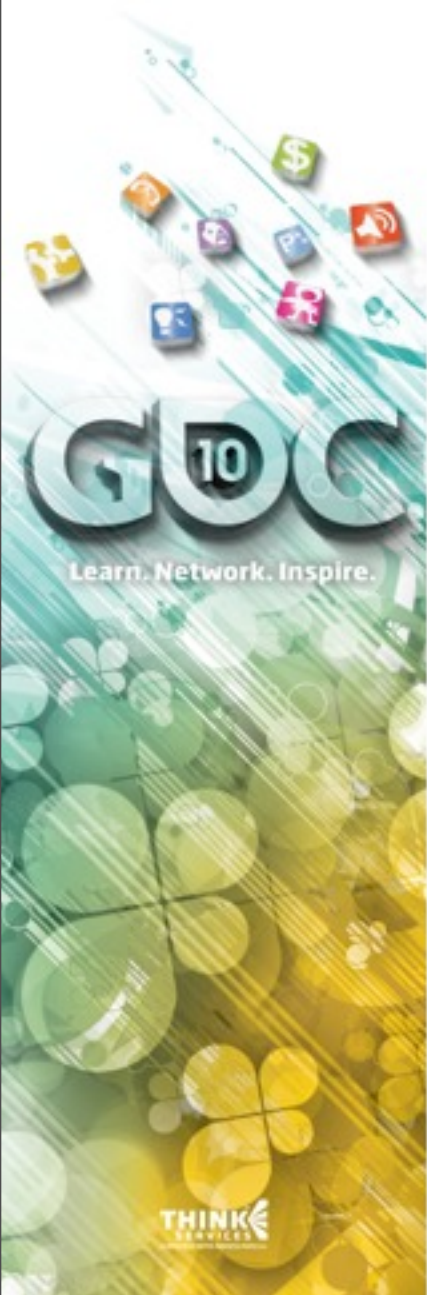
Lean/Kanban



More on Kanban...

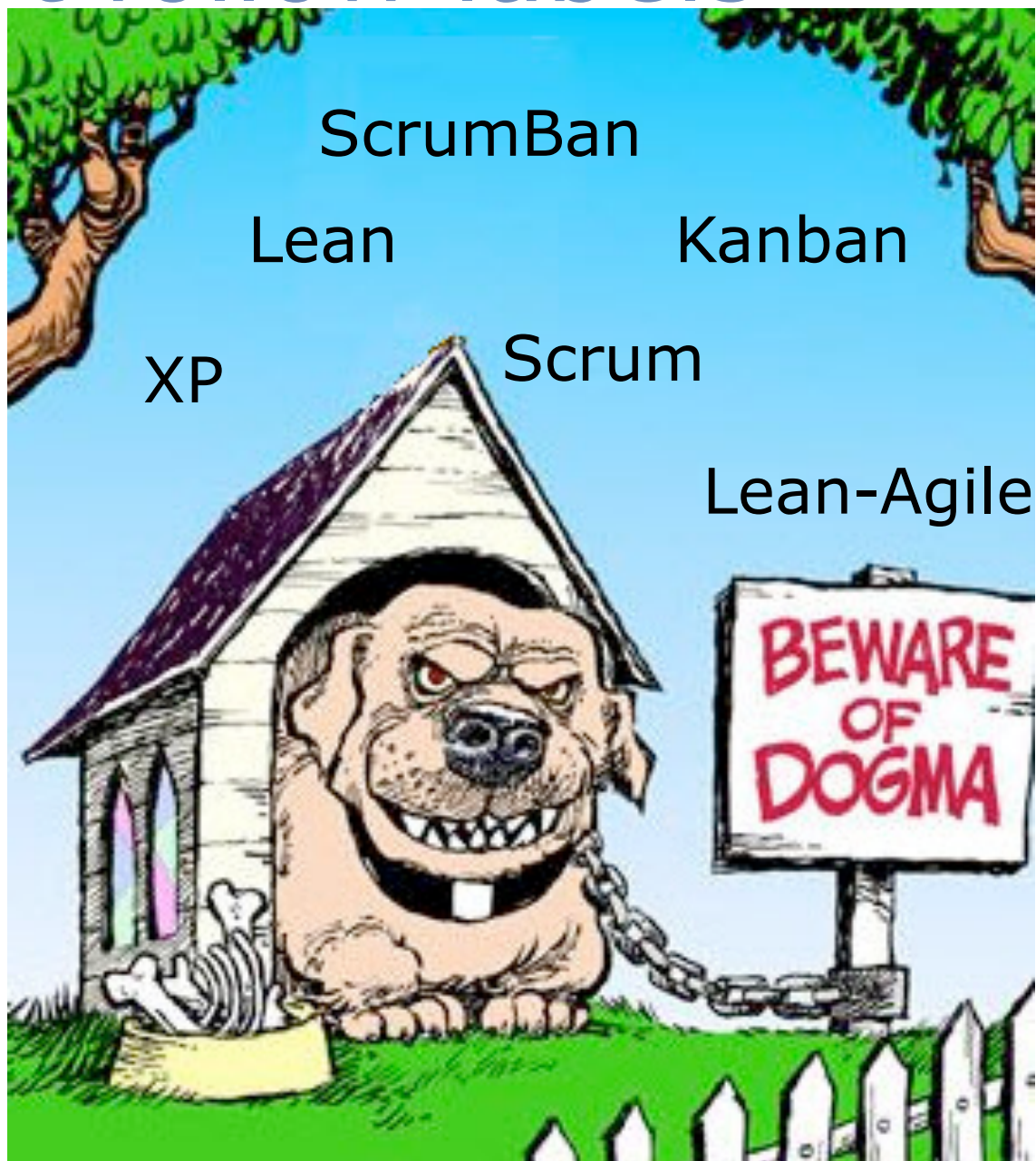
November 2008 feature article in Gamasutra

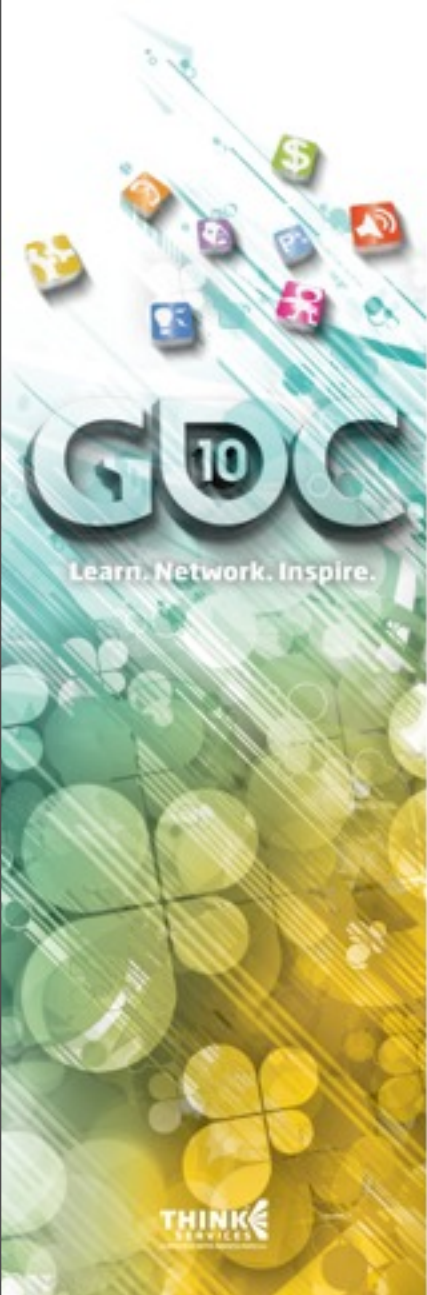
Chapter in upcoming book



Common sense

Don't follow labels

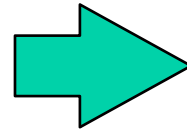




- Scrum isn't the goal
 - Following the rules no matter what
- Focus on the principles
 - Empiricism - Science vs. alchemy
 - Emergence
 - Time-Boxing
 - Prioritization
 - Self-Organization
- Iterate on the process
- Provide leadership
- Plan, plan, plan

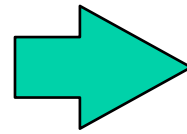
Summary. Changes.

Devs given suspect
dates & crunch



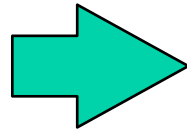
Iterate and we'll
measure velocity
towards the goal

Devs solely responsible
for achieving the goal



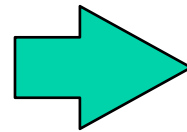
Biz is responsible for
achieving the goal

Completion of activities

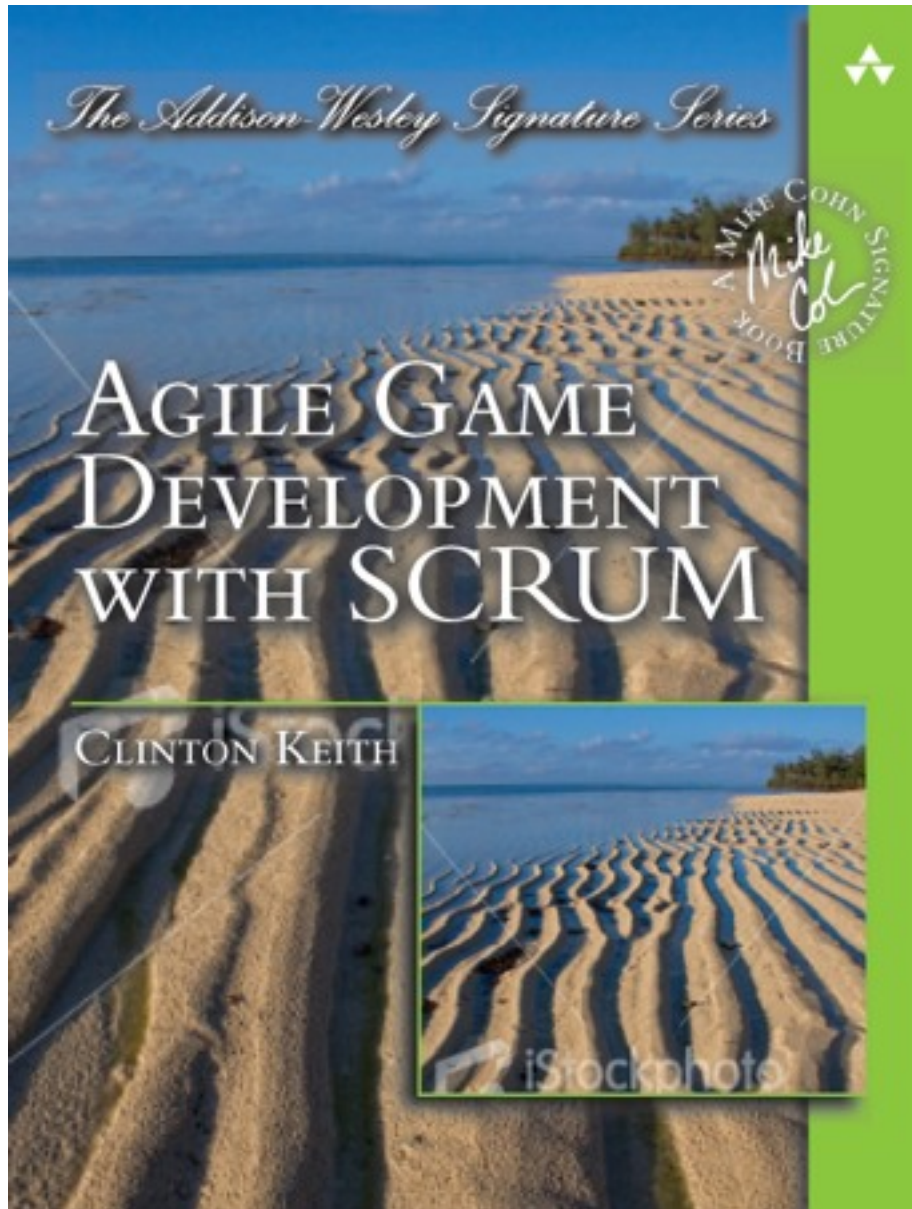


Delivery of
features

Devs make biz
decisions, biz makes
dev decisions



Devs make dev
decisions, biz makes
biz decisions



Questions?