

Agile at ON.Lab

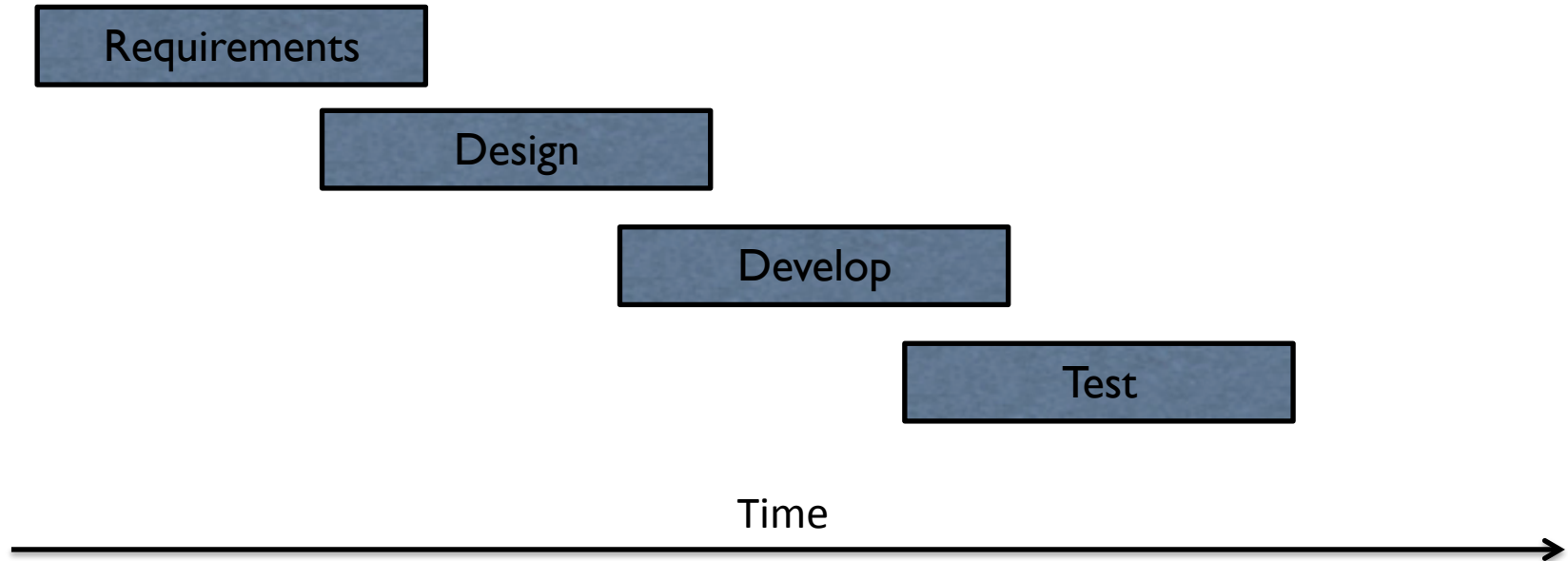
Bill Snow
VP of Engineering



What is waterfall?



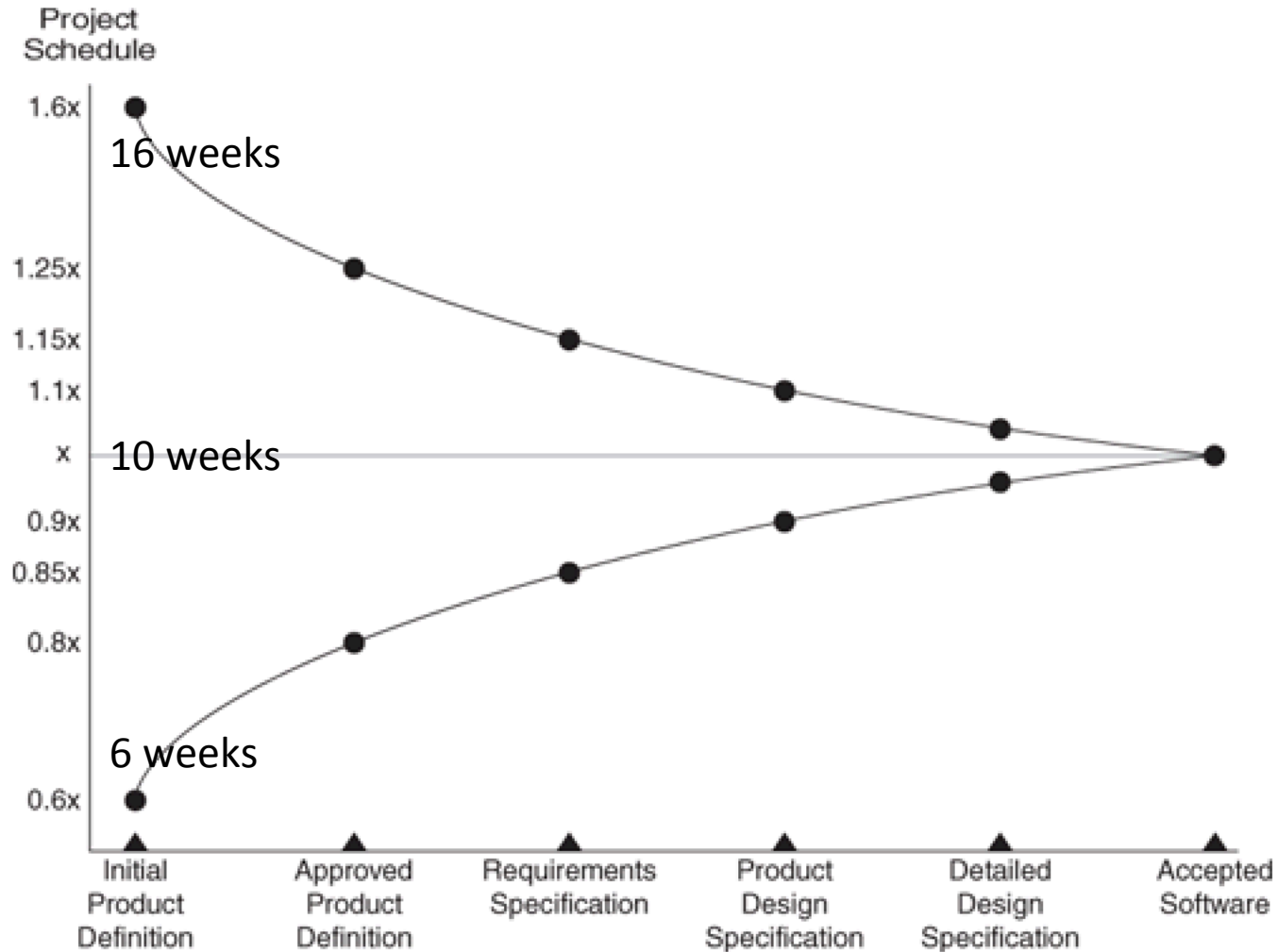
Or



Assertion

- We cannot precisely anticipate
 - Our users' requirements
 - How long it takes to develop a feature or system
 - Which design will be best
 - What tasks are needed to develop a feature

The cone of uncertainty



From the book *Agile Estimating and Planning* by Mike Cohn

Principles behind Agile

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

Business people and developers must work together daily throughout the project.

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

The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Principles behind Agile

Working software is the primary measure of progress.

**Agile processes promote sustainable development.
The sponsors, developers, and users should be able
to maintain a constant pace indefinitely.**

**Continuous attention to technical excellence
and good design enhances agility.**

**Simplicity--the art of maximizing the amount
of work not done--is essential.**

**The best architectures, requirements, and designs
emerge from self-organizing teams.**

**At regular intervals, the team reflects on how
to become more effective, then tunes and adjusts
its behavior accordingly.**

Requirements

Design

Code

Test

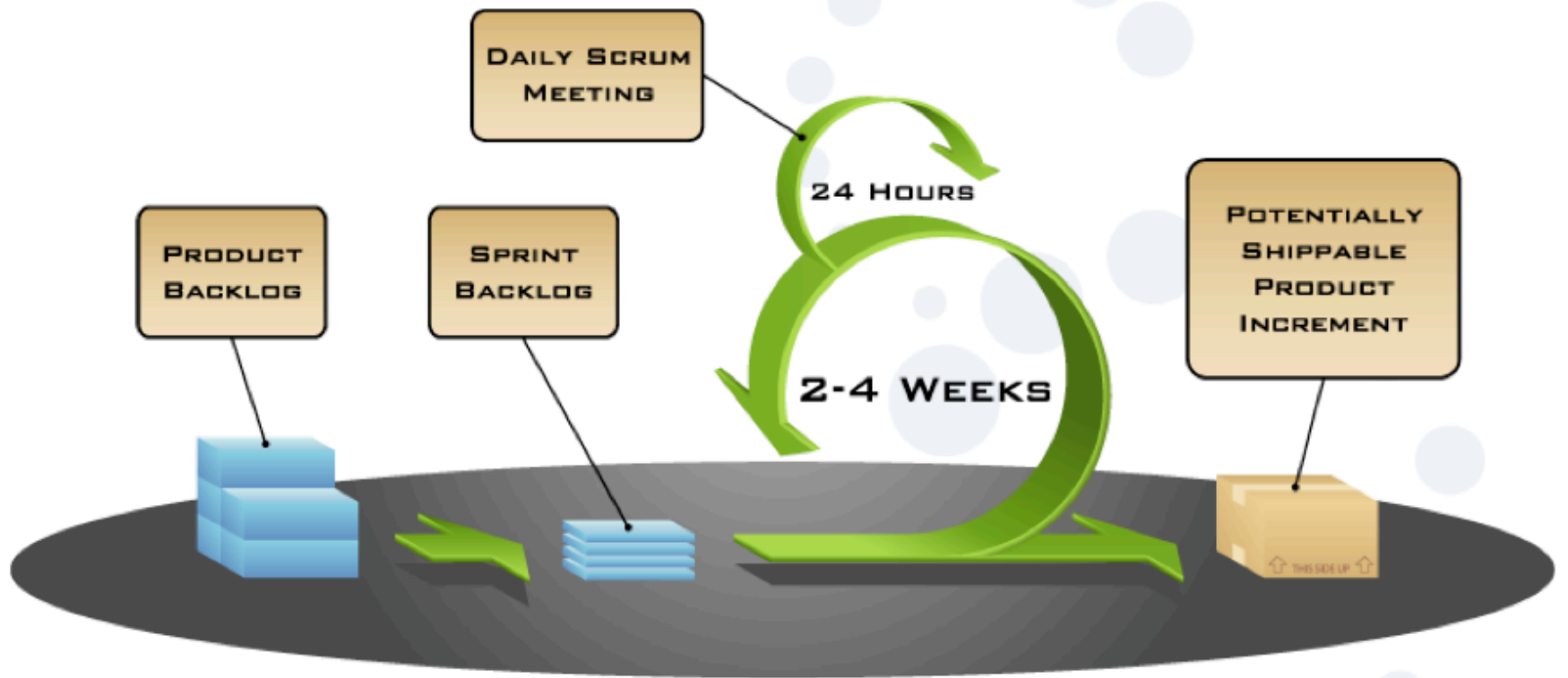
Rather than doing all of
one thing at a time...

...Scrum teams do a little
of everything all the time



Source: "The New New Product Development Game" by Takeuchi
and Nonaka. *Harvard Business Review*, January 1986.
Mountain Goat Software, LLC





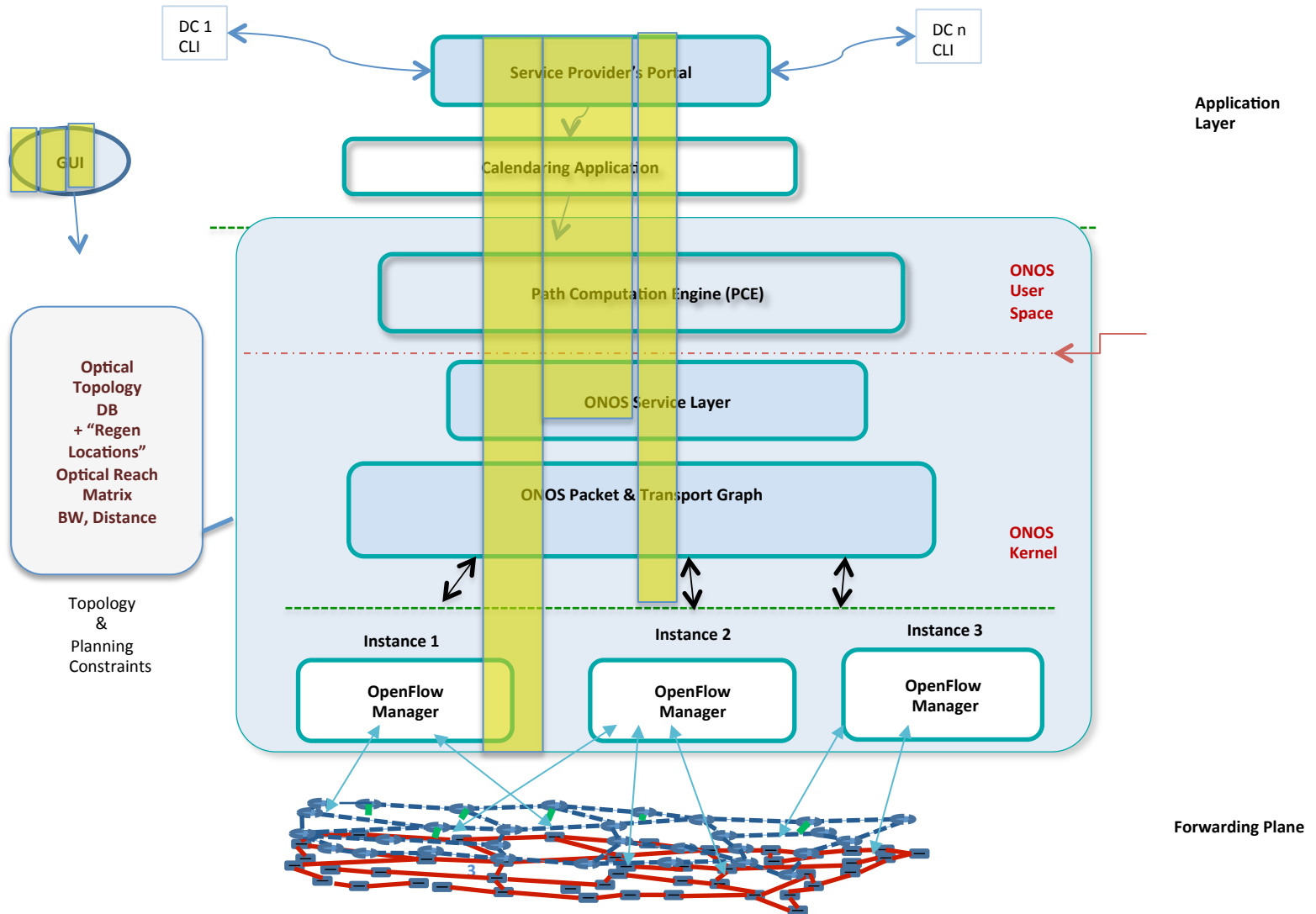
COPYRIGHT © 2005, MOUNTAIN GOAT SOFTWARE

Image available at
www.mountaingoatsoftware.com/scrum

Differences

- Team is cross functional – artifacts are not handed off between functional groups
- Team works in short iterations of 2-4 weeks – not in long periods of months or years.
- Working code is delivered every iteration. Not just at the end.
- Planning is done every iteration, not just at the beginning.
- Requirements change over time and are not set in stone at the beginning.
- Customers are continually involved in the cycle, not just in beta and the end.
- Less documentation
- More “shoot first, aim later”
- Less pretense of long term foresight

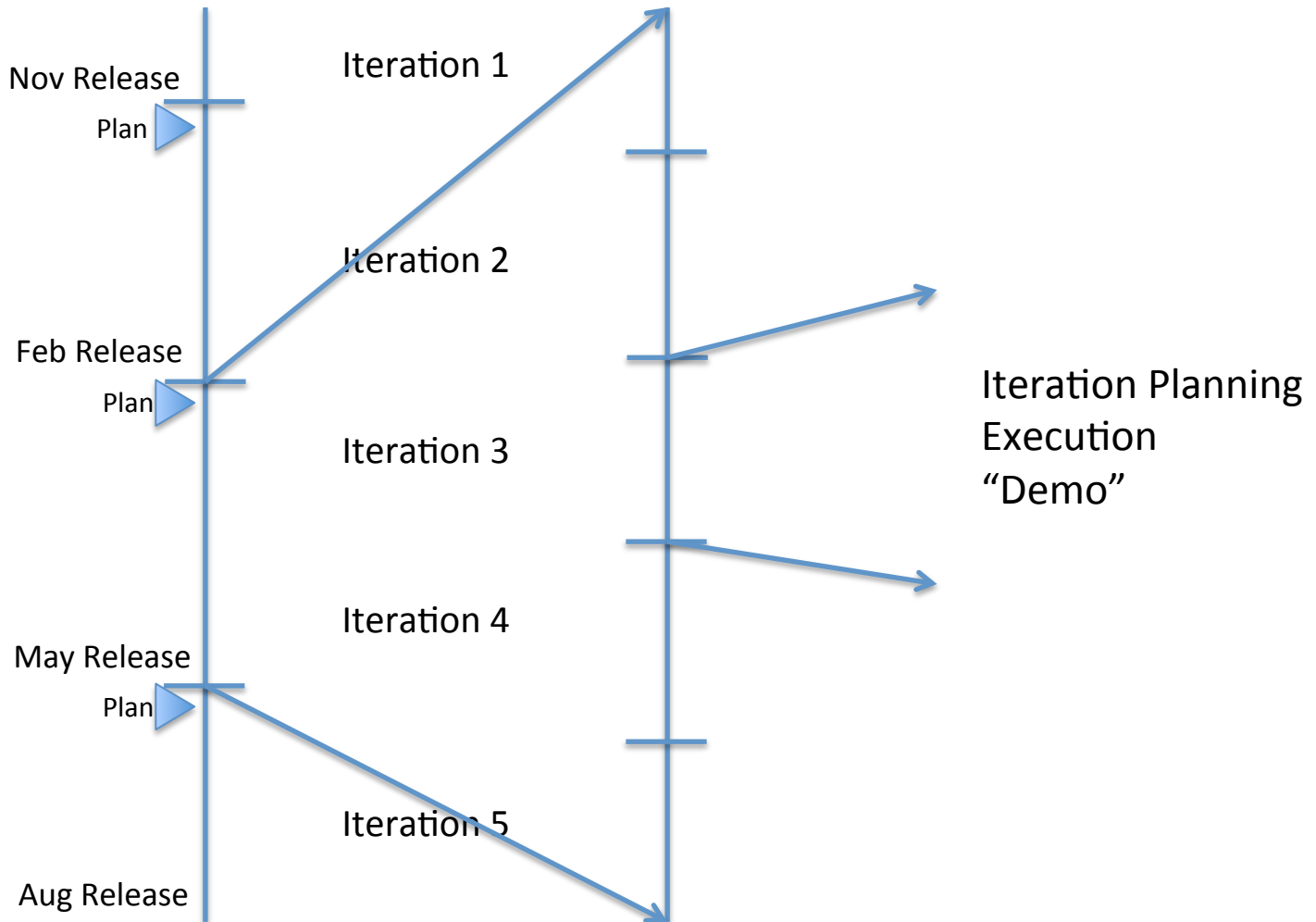
Real World Example



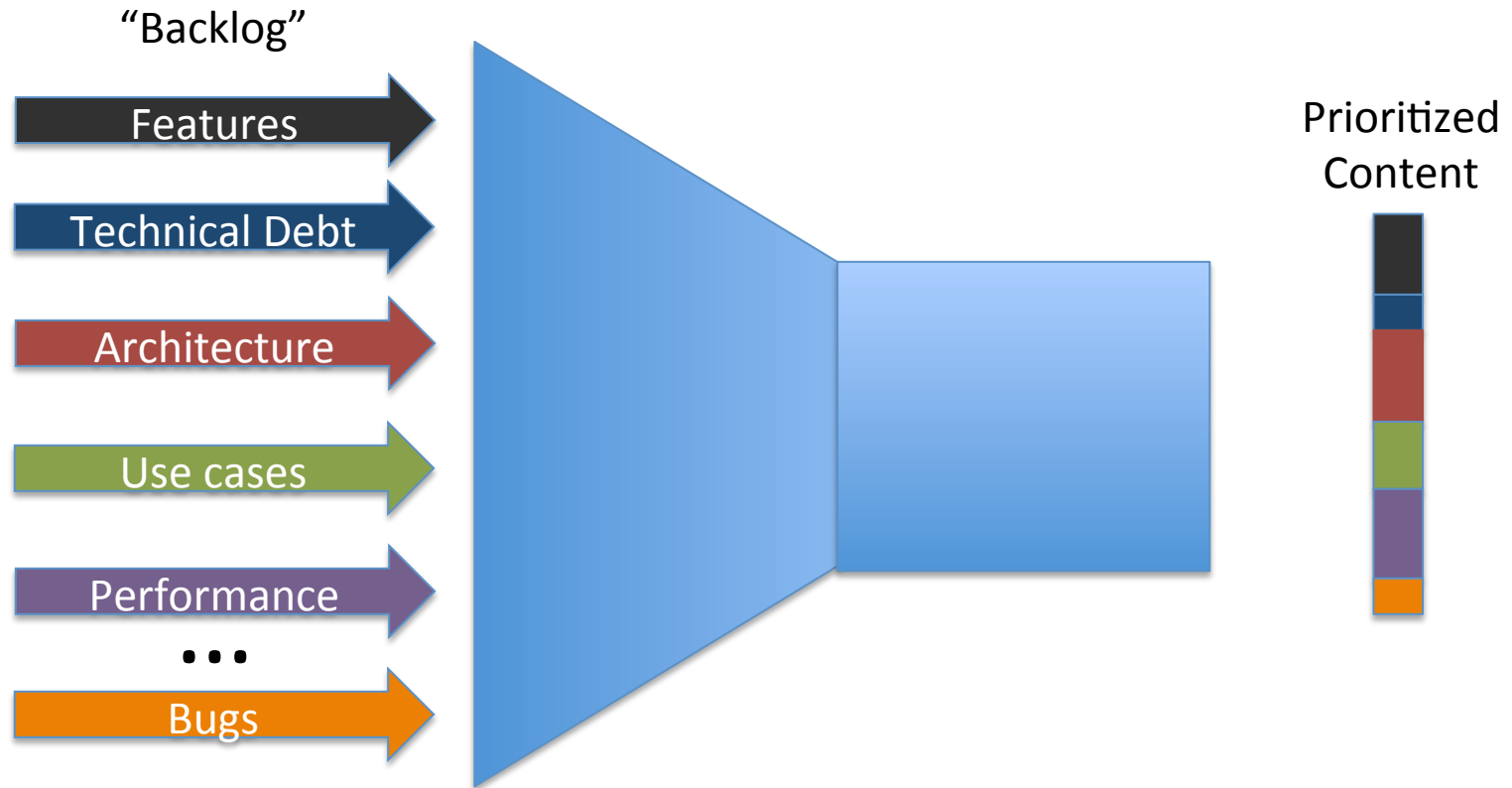
Release Model

Quarterly
“train” Releases

2-3 Week
Iterations



Release and Iteration Planning



Jira

Database and Workflow

Tracking Release Content

- Jira – issues tracking, workflow
- Hierarchy of Jira issues
 - Epic – work happening throughout a release
 - Stories – work that fits in a sprint
 - Tasks – work that is done in 1-2 days
- See wiki for more details

Summary

- ON.Lab uses a version of Agile Scrum
 - 3 month releases, 3 week iterations
- People unfamiliar with agile, will find it different (and initially uncomfortable)
- We all work together to learn and improve
- Watch this 10 minute video
 - <https://www.youtube.com/user/axosoft?v=XU0IIRltyFM>