The Learning House
427 S. 4th Street #300
Louisville KY 40202

SOFTWARE GUILD

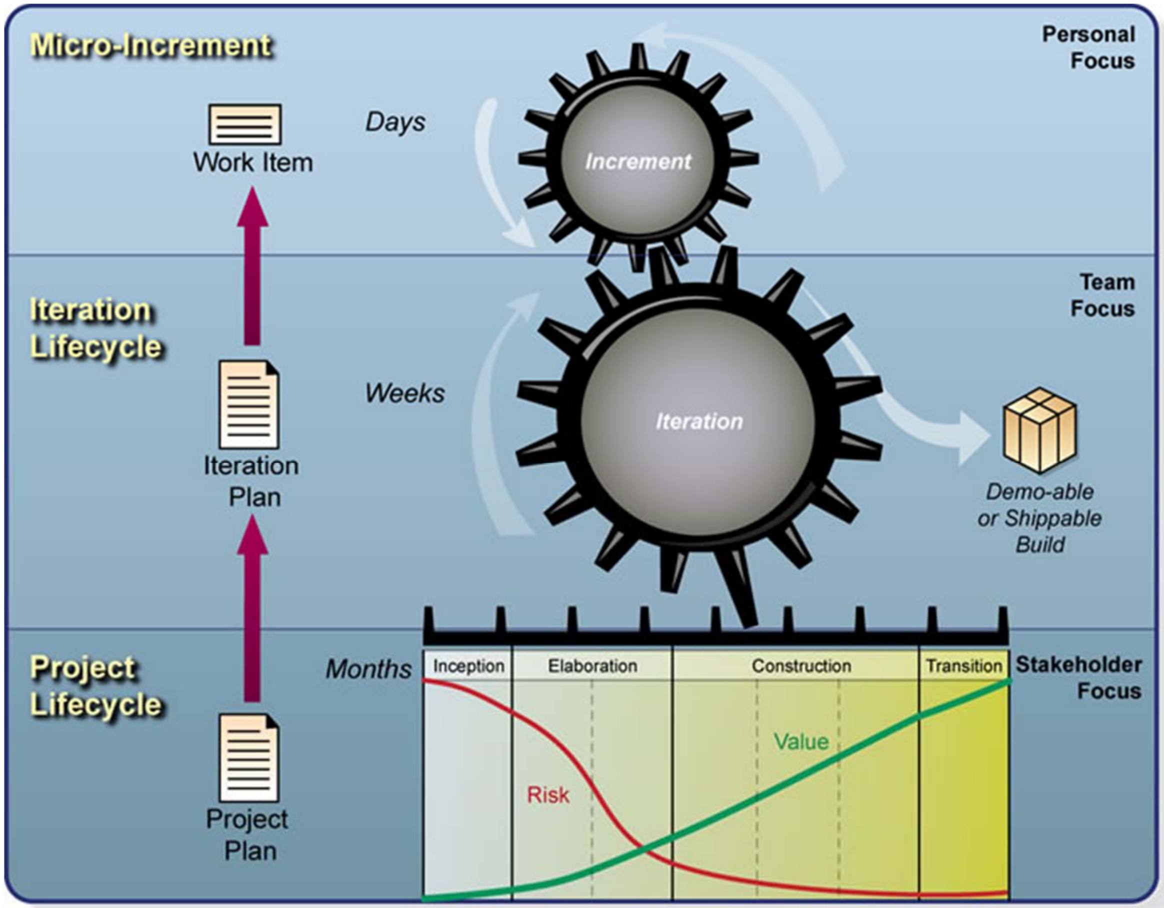
Software Development Lifecycle

Lesson 2 - Agile Life Cycle

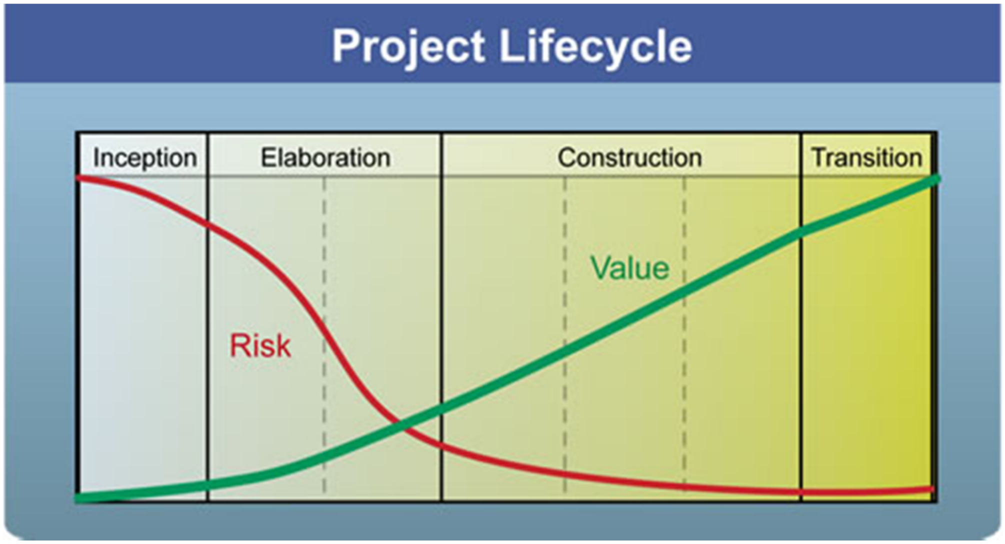
Agile Life Cycle

- We’ll look at some concepts for OpenUp and Extreme Programming
- Each methodology has its own terms for these concepts - methodologists are very zealous…
- We’ll try to show concepts in a method-agnostic manner

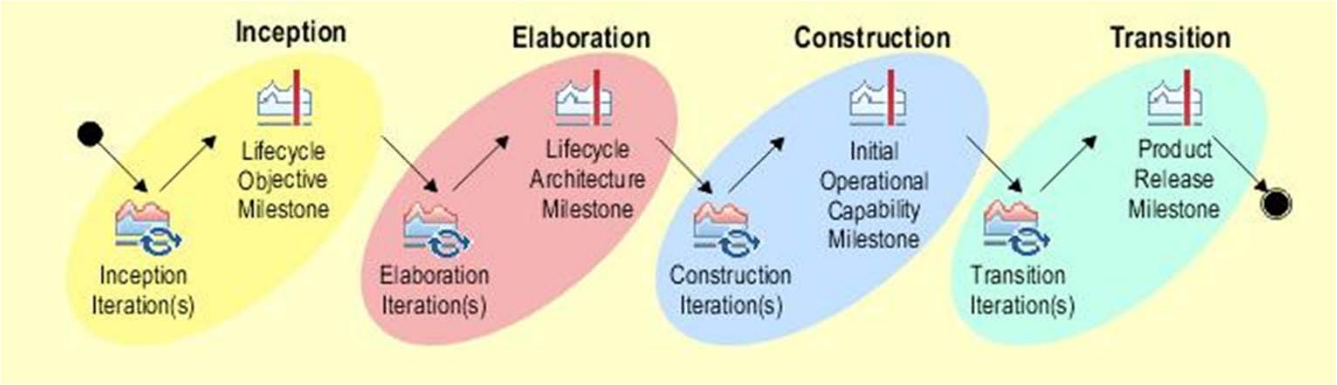
Life Cycle Overview



Risk/Value Curves



Phases


Inception
Elaboration
Construction
Transition
Lifecycle Objective Milestone
Inception Iteration(s)
Lifecycle Architecture Milestone
Elaboration Iteration(s)
Initial Operational Capability Milestone
Construction Iteration(s)
Product Release Milestone
Transition Iteration(s)

Inception

- Do we agree on scope and objectives?
- Should we proceed?
- We must:
  - Understand what to build
  - Identify key system functionality
  - Determine at least one possible solution
  - Understand the high-level estimate for cost, the schedule and the risks for the project

Elaboration

- Do we agree on the executable architecture to be used for developing the project?
- Do we find the value delivered so far and the remaining risk is acceptable?
- We must:
  - Get a more detailed understanding of requirements
  - Design, implement, validate, and establish a baseline for the architecture
  - Mitigate essential risks and have a good idea of costs and schedule

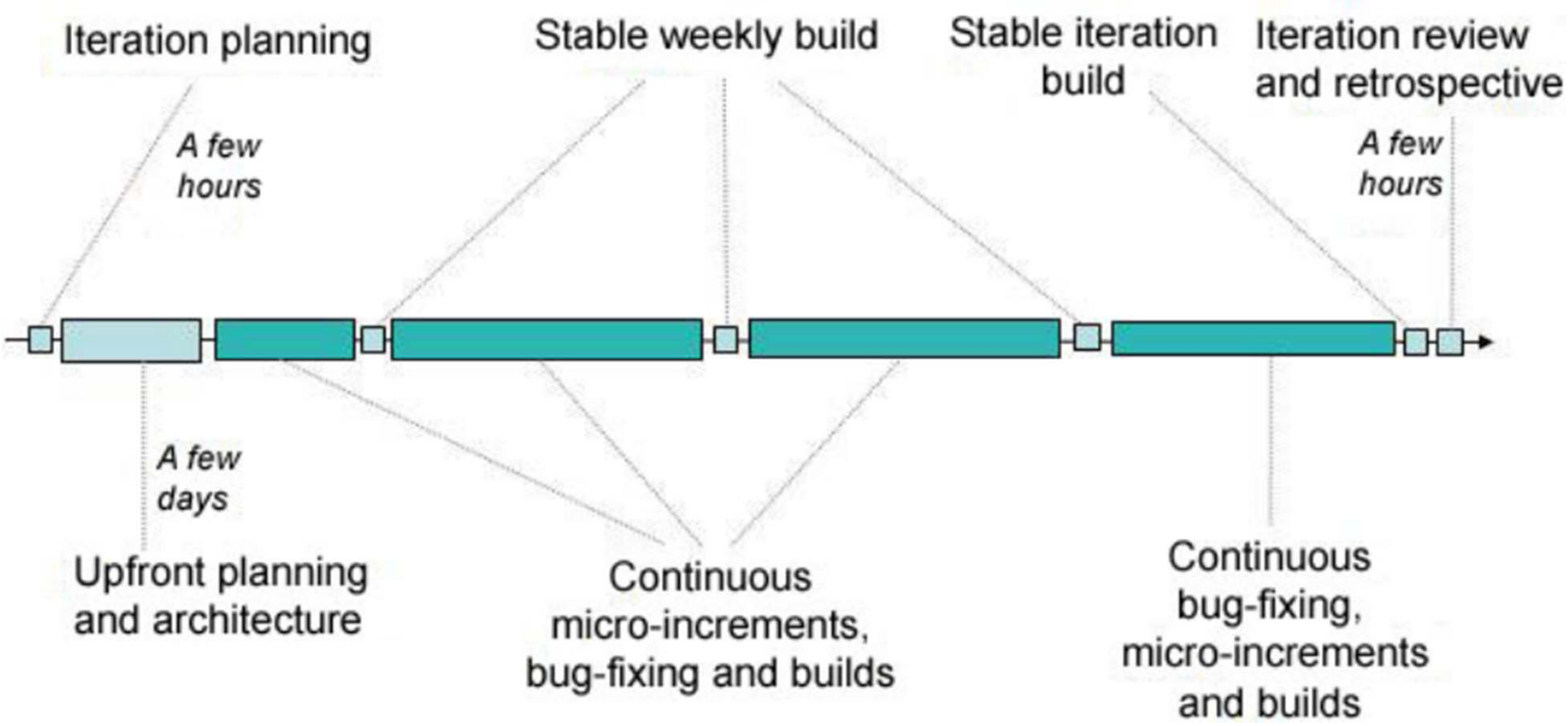
Construction

- Are we close to release?
- Should we switch focus to tuning, polishing, and deployment?
- We must:
  - Complete development of the system based on the baseline architecture
  - Deliver in an iterative manner - short iterations, frequent deliverables
  - Test as we go

Transition

- Are we ready for release?
- We must:
  - ensure software is ready for delivery
  - Beta test to ensure user expectations are met
  - Achieve stakeholder concurrence that deployment is complete
  - Improve future project performance through lessons learned

Iteration Life Cycle


Iteration planning
Stable weekly build
Stable iteration build
Iteration review and retrospective
A few hours
A few hours
A few days
Upfront planning and architecture
Continuous micro-increments, bug-fixing and builds
Continuous bug-fixing, micro-increments and builds