From Inception to Elaboration

Chapter 8
Applying UML and Patterns
Craig Larman
Objectives

- Elaboration is the initial series of iterations during which the team does the following
  - Serious investigation
  - Discover & stabilize major requirements
  - Mitigate/retire risks (business value)
  - Build core architecture elements
  - Estimate overall schedule and resources
  - Establish a supporting environment
Inception Checkpoint

- Brief & incomplete artifacts
- Quick phase
- Shallow investigation
- Determine basic feasibility, risk & scope
- Decide if project is worthwhile
Inception - Artifacts and Activities

- Requirements workshop
- Name actors, goals, use cases
- Keep use cases brief
- Identify most risky & influential quality requirements
- First version of Supplementary Specification and vision
Inception - Artifacts and Activities (2)

- Risk list
- Technical feasibility
- UI oriented prototypes
- Buy/build/reuse components
- High-level candidate architecture
- Plan first iteration
- Candidate tools list
Elaboration - Key Ideas

- Not a waterfall model!
- Two to six weeks for each iteration
- Timeboxed iterations
- Each iteration ends in a stable and tested release
Architecture
Prototype/Baseline

- Not a partial system
- Evolutionary prototype
- Don’t create throw-away prototypes
- Production subset of final system
- Also called Executable Architecture
Best Practices

- Start programming early
- Adapt based on feedback
- Design, implement and test adaptively
- Test early and realistically
- Requirements and use case details through series of workshops
Architecturally Significant Features

- Wide and shallow design
- Refine inter-module interfaces
- Integrate existing components
- Simple scenarios
Essential Activities

- Define, validate and baseline Architecture
- Refine Vision
- Create and baseline detailed iteration plans for Construction Phase
- Refine development case and proper development environment
- Refine Architecture and select components
Rank Criteria

- Risk
- Coverage
- Criticality
Ranking

- Rank work across iterations
- High ranking scenarios in early ranking
- Rank adaptively
UP Artifacts

- Iteration Plan
- Change Request
- Software Development Plan
Iteration 1 Requirements

- Implement basic key scenario
- Start-Up use case
- KISS
- No external collaboration
Incremental Development

- Handle requirements across iterations
- Varying features over iterations
- Complete short, simple use cases in single iteration
Artifacts starting in Elaboration

- Domain Model
- Design Model
- Software Architecture Document
- Data Model
- Test Model
- Implementation Model
- Use-Case Storyboards and UI Prototypes
Inception and Elaboration

- Main output is a stable software architecture, that enables quality planning of Construction and Deployment
- 15 to 25 percent of total project cost
Lifecycle Architecture

Milestone

- Stable product vision and requirements
- Stable architecture
- Proven approaches for test and evaluation
- Major risks addressed
- Sufficient detail & feasibility for iteration plans of construction phase
- Stakeholders agree to current vision
- Actual Vs planned expenditure acceptable
You didn’t Understand Elaboration When ...

- No Timeboxed schedule
- Single Iteration
- Most requirements already defined
- No Risk mitigation/resolution
- No Executable Architecture
- Requirements Phase
- Attempt full and careful design
You didn’t Understand Elaboration When (2)...

- Minimal feedback and adaptation
- No early and realistic testing
- Frozen Architecture
- No Proof-of-concept programming
- No multiple requirements workshops
References

- Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process - Craig Larman
- http://www.arcorn.com/approach/approach_2_01.html
References (continued...)

- http://www.therationaledge.com/content/dec_01/f_projectConsole_pw.html