Software Development Methodologies

Lecturer: Raman Ramsin

Lecture 6

Integrated Methodologies: USDP
Unified Software Development Process (USDP)

- Also known as Unified Process (UP)
- First introduced in 1999
- A refined, simplified, and non-proprietary version of the Rational Unified Process (RUP)
- UML-Based
- Use-Case-Driven
- Architecture-centric
- Iterative and Incremental
Unified Software Development Process

- Software lifecycle is decomposed over time in four sequential phases
  - Inception (Vision Milestone)
    - Define the vision of the product, scope of the project and the business case
  - Elaboration (Architecture Milestone)
    - Refine the definition of the product
    - Define and baseline an architecture
    - Develop a more precise plan for its development and deployment
  - Construction (Initial Operational Capability Milestone)
    - Build the product to the point where it can be delivered to its end-users for the first time
  - Transition (Product Release Milestone)
    - Transition the product to the user community; this includes manufacturing, delivering, training, and planning for supporting and maintaining the product.
Iterations and Workflows

Phases

Inception | Elaboration | Construction | Transition

Iterations

Preliminary Iteration(s) | iter. #1 | iter. #2 | iter. #n | iter. #n+1 | iter. #n+2 | iter. #m | iter. #m+1

An iteration in the elaboration phase

Workflows

Requirements
Analysis
Design
Implementation
Test

Department of Computer Engineering
Inception Phase

The purpose of Inception is to "get the project off the ground":

- establishing feasibility - this may involve some technical prototyping to validate technology decisions or proof of concept prototyping to validate business requirements;
- creating a business case to demonstrate that the project will deliver quantifiable business benefit;
- capturing essential requirements to help scope the system;
- identifying critical risks.
Inception – Concerns

- The inception phase is a preparatory stage that attempts to answer the following questions:
  - What is the purpose and objectives of the project? Is it worth the effort?
  - Is the project feasible (e.g. technologically, financially, with current personnel)?
  - Should we buy the system, or build it?
  - Will it be developed now, or built from an existing system?
  - What are the estimated costs and risks?
  - Should we proceed with the project?

- This phase also deals with project planning and project management
  - This includes Gantt charts and plans, budgets, etc.
## Inception – Postconditions and Deliverables

<table>
<thead>
<tr>
<th>Conditions of satisfaction</th>
<th>Deliverable</th>
</tr>
</thead>
<tbody>
<tr>
<td>The stakeholders have agreed on the project objectives</td>
<td>A vision document that states the project’s main requirements, features, and constraints</td>
</tr>
<tr>
<td>System scope has been defined and agreed on with the stakeholders</td>
<td>An initial use case model (only about 10% to 20% complete)</td>
</tr>
<tr>
<td>Key requirements have been captured and agreed on with the stakeholders</td>
<td>A project glossary</td>
</tr>
<tr>
<td>Cost and schedule estimates have been agreed on with the stakeholders</td>
<td>An initial project plan</td>
</tr>
<tr>
<td>A business case has been raised by the project manager</td>
<td>A business case</td>
</tr>
<tr>
<td>The project manager has performed a risk assessment</td>
<td>A risk assessment document or database</td>
</tr>
<tr>
<td>Feasibility has been confirmed through technical studies and/or prototyping</td>
<td>One or more throwaway prototypes</td>
</tr>
<tr>
<td>An architecture has been outlined</td>
<td>An initial architecture document</td>
</tr>
</tbody>
</table>
Inception Timeline

- An important idea with Inception is that we do not yet know if a project will take place!
  - Often 1 or 2 iterations are required for Inception

- Therefore, since a project may be rejected, it makes sense that the Inception phase should be very short
  - Therefore, if the project gets scrapped, little time (and money) would have been wasted
  - It is not uncommon for Inception to last a few days to a few weeks, maximum
Elaboration Phase

- The purpose of Inception is to understand the problem, whereas Elaboration explores the solution:
  
  - create an executable architectural baseline;
  - refine the risk assessment;
  - define quality attributes (defect discovery rates, acceptable defect densities, and so on);
  - capture use cases to 80% of the functional requirements;
  - create a detailed plan for the construction phase;
  - formulate a bid that includes resources, time, equipment, staff, and cost.
Elaboration and the Workflows

In the Elaboration phase, the focus in each of the core workflows is as follows:

- requirements - refine system scope and requirements;
- analysis - establish what to build;
- design - create a stable architecture;
- implementation - build the architectural baseline;
- test - test the architectural baseline.
Elaboration - Concerns

- After Elaboration, project risks are essentially eliminated
  - The Architecture and UI have been approved by customers and managers
  - Technically difficult software components have been implemented, or proof-of-concept code has been created to prove it was possible
  - Cost estimates are finalized, so budgets can be approved
  - Preliminary user manuals have been created and analyzed

- Analysis, architecture and design well underway after Elaboration
### Conditions of satisfaction

<table>
<thead>
<tr>
<th>Conditions of satisfaction</th>
<th>Deliverable</th>
</tr>
</thead>
<tbody>
<tr>
<td>A resilient, robust executable architectural baseline has been created</td>
<td>The executable architectural baseline</td>
</tr>
<tr>
<td>The executable architectural baseline demonstrates that important risks have been identified and resolved</td>
<td>UML static model, UML dynamic model, UML use case model</td>
</tr>
<tr>
<td>The vision of the product has stabilized</td>
<td>Vision document</td>
</tr>
<tr>
<td>The risk assessment has been revised</td>
<td>Updated risk assessment</td>
</tr>
<tr>
<td>The business case has been revised and agreed with the stakeholders</td>
<td>Updated business case</td>
</tr>
<tr>
<td>A project plan has been created in sufficient detail to enable a realistic bid to be formulated for time, money, and resources in the next phases</td>
<td>Updated project plan</td>
</tr>
<tr>
<td>The stakeholders agree to the project plan</td>
<td>Business case</td>
</tr>
<tr>
<td>The business case has been verified against the project plan</td>
<td></td>
</tr>
<tr>
<td>Agreement is reached with the stakeholders to continue the project</td>
<td>Sign-off document</td>
</tr>
</tbody>
</table>

Department of Computer Engineering
Construction Phase

- The purpose of Construction is to iteratively enhance and evolve the previously created artefacts into the target system:
  - complete all requirements, analysis, and design
  - evolve the architectural baseline generated in Elaboration into the final system.
Construction and the Workflows

- We can summarize the kind of work undertaken in each workflow during Construction as follows:
  - requirements - uncover any requirements that had been missed;
  - analysis - finish the analysis model;
  - design - finish the design model;
  - implementation - build the Initial Operational Capability;
  - test - test the Initial Operational Capability.
Construction – Postconditions and Deliverables

<table>
<thead>
<tr>
<th>Conditions of satisfaction</th>
<th>Deliverable</th>
</tr>
</thead>
<tbody>
<tr>
<td>The software product is sufficiently stable and of sufficient quality to be deployed in the user community</td>
<td>The software product</td>
</tr>
<tr>
<td></td>
<td>The UML model</td>
</tr>
<tr>
<td></td>
<td>Test suite</td>
</tr>
<tr>
<td>The stakeholders have agreed and are ready for the transition of the software to their environment</td>
<td>User manuals</td>
</tr>
<tr>
<td></td>
<td>Description of this release</td>
</tr>
<tr>
<td>The actual expenditures vs. the planned expenditures are acceptable</td>
<td>Project plan</td>
</tr>
</tbody>
</table>
Transition Phase

The purpose of Transition is the ultimate deployment of the software produced at the end of Construction:

- conduct beta test and acceptance test, and correct defects;
- prepare the user sites for the new software;
- tailor the software to operate at the user sites;
- modify the software if unforeseen deployment problems arise;
- create user manuals and other documentation;
- provide user consultancy;
- conduct a post-project review.
Transition and the Workflows

We can summarize the kind of work undertaken in each workflow during Transition as follows:

- Requirements - not applicable.
- Analysis – update the analysis model, if required.
- Design - modify the design if problems emerge in testing.
- Implementation - tailor the software for the user site and correct problems uncovered in testing.
- Test - beta testing and acceptance testing at the user site.
## Transition – Postconditions and Deliverables

<table>
<thead>
<tr>
<th>Conditions of satisfaction</th>
<th>Deliverable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beta testing is completed, necessary changes have been made, and the users agree that the system has been successfully deployed</td>
<td>The software product</td>
</tr>
<tr>
<td>The user community is actively using the product</td>
<td></td>
</tr>
<tr>
<td>Product support strategies have been agreed on with the users and implemented</td>
<td>User support plan</td>
</tr>
<tr>
<td></td>
<td>Updated user manuals</td>
</tr>
</tbody>
</table>
USDP: Strengths and Weaknesses

**Strengths**
- Same benefits as RUP
- Complexity has been significantly reduced.
- Unlike RUP, Analysis and Design are separate workflows, each with its own specific set of work-units and products.

**Weaknesses**
- Same weaknesses as RUP, albeit to a far lesser degree.
- Business modeling, deployment, and management activities have lost their pivotal roles (as disciplines) in the process.
References