Using RUP/UP: 10 Easy Steps

A Practical Guide

Software Development Best Practices

A Call to Uniformity Not Conformity!

Industry Background: The Unified Process is a process architecture for business modeling and software development; created from the software engineering industry's best practices. The UP has become the standard process or the process of choice used by advanced software engineering professionals for implementing object oriented analysis, design, and programming concepts in an effort to construct complex software that will be realized as agile applications, systems, and subsystems.

Influencing Criteria: Unlike other engineering and architectural disciplines spanning industries, the software industry has not mandated, nor enforced, a standard of accountability for a unified approach to quality control. Thus the idiom of code first and model/document later has become pervasive within the software industry. This has led to a host of software quality and software development issues and a waste of resources. Specifically time and money.

Enter the Unified Process or UP. The UP is an iterative process initially developed by the IBM Rational Software Corporation (formerly Rational Software) as its Rational Unified Process or RUP. Created from the software industries time-proven best practices, it was adopted by the Object Management Group (OMG) and has since become the software industry's standard process for governing software engineering. The RUP/UP is the process not only for implementing an object oriented analysis, design, and programming methodology, but for modeling business processes as well. The RUP/UP is Use-Case driven and heavily reliant on the Unified Modeling Language or UML. UML is an industry agnostic notation for communicating visually.

Adoption Justification: As the complexity of business operations and associated software products increase, a process that helps guide the capture of business rules and efficiently relate them to software development is a must. Because the one constant is change, the process must be agile. Furthermore, this process must communicate in a manner understood the same way, all the time, by all involved parties; this, regardless of role or discipline. The RUP/UP uses UML to satisfy this need. UML is a universally understood notation for communicating visually through images with assigned meanings. Because the RUP/UP is an extension of the Meta Object Facility's Process Metamodel, aspects of it can therefore be applied across industry boundaries. The tools supporting UML are the most prevalent of any diagramming, modeling, and visual communication tool to date. Contrary to the belief of some, the UP/RUP is an Agile Process. Agile Process + Agile Language + Agile Software = Agile Business.

The Unified Process VS the Rational Unified Process: The UP had its beginnings as RUP. The UP is RUP without the copyrighted Rational Unified Process products; in essence…the process only without the tool mentors, artifact templates, guides, etc. For this step-by-step guide, the names of the artifacts produced bear the same names of those in RUP. However, apply any desired naming convention with the artifacts.

Copyright ©2003-2005 by X-tier SAE Inc., all rights reserved
Rev 9/12/2005-5

Implementation of RUP/UP in 10 Easy Steps

Briefly Understanding the Process; a Recap

An iteration is a completion of a sequence of disciplines also called a workflow. The resulting output for the completion of an iteration is an artifact or series of artifacts. An artifact can be source code, a compiled build, a model, log, or any other pertinent media**.** *Once the artifacts for a specific phase have been completed, transition to the next phase in the process occurs.*

In the RUP/UP there are 4 phases: Inception, Elaboration, Construction, & Transition. Each phase has its focus on reaching its Lifecycle Milestone. In Inception the focus is mainly on concept, vision, risks, budget, and requirements; the objective the software project must accomplish. In Elaboration it is on proving the software's architecture. In Construction it is building the software and Transition is concerned with releasing or roll-out of the final product. These milestones are reached when a predetermined number of ***n*** of artifacts for that phase reach acceptable levels of detail (completion).

Below is a very simplified overview of how this is organized by phase (***discipline*** denotes where the majority of effort is to be focused for the corresponding phase):

1. **Inception: Project Objectives Milestone (project viable or non-viable)**
    a. Create/Detail Development Case artifacts for this milestone
        i. Iterate through each discipline (workflow) to produce/refine artifacts for each discipline
            1. ***Business Modeling***
            2. ***Requirements***
            3. Analysis & Design
            4. Implementation
            5. Test
            6. Deploy
2. **Elaboration: Product Architectural Milestone (architecture is proven)**
    a. Create/Detail Development Case artifacts for this milestone
        i. Iterate through each discipline (workflow) to produce/refine artifacts for each discipline
            1. Business Modeling
            2. Requirements
            3. ***Analysis & Design***
            4. ***Implementation***
            5. Test
            6. Deploy
3. **Construction: Operational Capability Milestone (all functionality developed)**
    a. Create/Detail Development Case artifacts for this milestone
        i. Iterate through each discipline (workflow) to produce/refine artifacts for each discipline
            1. Business Model
            2. Requirements
            3. Analysis & Design
            4. ***Implementation***
            5. ***Test***
            6. Deploy

Copyright ©2003-2005 by X-tier SAE Inc., all rights reserved
Rev 9/12/2005-5

4. **Transition:** Product Release Milestone **(product released into production)**
    a. Create/Detail Development Case artifacts for this milestone
        i. Iterate through each discipline (workflow) to produce/refine artifacts for each discipline
            1. Business Modeling
            2. Requirements
            3. Analysis & Design
            4. Implementation
            5. Test
            6. ***Deploy***

Each Phase has the same defined workflow (iteration) which is the same set of sequential disciplines that are followed every time for every iteration, always. The time and effort expended within each discipline for an iteration depends on the Phase. Again, the completion of a workflow (steps 1 through 6) is an iteration. The number of iterations one goes through in a Phase theoretically results in the outcome of a higher quality product, but also increases development time. So there is a direct trade-off between quality and time.

Using the UP/RUP Process

The main confusion with RUP/UP is in artifact production, iterations, and phase transitions. Therefore think of each phase having 15 steps to an iteration. Multiple iterations per phase simply means repeating the steps for an iteration ***n*** times per phase until you produce the artifacts for that phase (at an acceptable percentage of completion) to transition to the next phase. Only those artifacts identified in the Development Case are to be produced.

> **Note:** A Target Organizational Assessment should be accomplished in the Inception Phase to aid in the base-lining of the organization and defining the Development Case before starting the process.

For each iteration there should be a tangible outcome; i.e. something should be produced. Some artifacts are independent of other artifacts, and some are interdependent on other artifacts that may only be produced is later phases. Because of this a determination should be made that defines the artifact's completion threshold for a phase. That is to say that when most of the artifacts for a given phase have reached, lets say, a 90% completion level, with the remaining 10% to be satisfied with the input of artifacts from another phase, the transition from the current phase to the next can and should occur as long as the milestone objective for that phase has been satisfied. These objects are simply the completion of the artifacts for that UP/RUP Phase as defined in the projects Development Case.

Table A below holds the workflow as a series of steps mapped to specific workflow disciplines. Again, completion of a UP/RUP Phase means the milestone for that phase has been reached. Since the RUP/UP is configurable this means that all the agreed upon project artifacts for a phase have been completed or are so near completion that Phase transition can occur. *One cannot stress enough that the artifacts for a development project by phase are defined in the Development Case for the project before the development begins!*

> **Note:** It is a common thought that all RUP/UP artifacts must be produced. *This is not true and grave misconception that can ultimately lead to process overload and subsequent process failure.* Yet it is the most common factor with improper RUP/UP implementations

Copyright ©2003-2005 by X-tier SAE Inc., all rights reserved
Rev 9/12/2005-5

While Project, Configuration, Change, and Environment Management gain focus at the beginning of every phase of the RUP/UP and are prevalent across every phase, they are mostly periphery efforts of the software engineering process. Because of this they are not part of the 10 Simplified Steps as addressed within this document. Still, they are identified within the workflow of an iteration as indicated within **Table A** below. This does not mean they are less important as software engineering cannot progress or will be extremely limited without them.

The Process in 10 Steps:

1. Define a Development Case for the project. That is "the artifacts to be produced for the software development lifecycle." Make sure Use-Cases are one of the artifacts as RUP/UP is Use-Case driven.
2. Identify most if not all of the Use-Cases for the project.
3. Classify the Use-Cases into levels of Risk from highest to lowest.
4. Classify Artifacts by Disciplines.
5. *Iterate through the RUP/UP disciplines (see table below) to CREATE artifacts that satisfy each Use-Case, beginning with the high risk Use-Cases first.*
6. *Iterate through the RUP/UP disciplines (see table below) to DETAIL (refine - add info) artifacts to their completion thresholds (levels) and/or the maximum number of iterations needed to satisfy each Use-Case as defined by the Development Case.*
7. Inception Phase Milestone Objective complete, transition to the Elaboration and repeat steps 5 & 6.
8. Elaboration Phase Milestone Objective complete, transition to the Construction and repeat steps 5 & 6.
9. Construction Phase Milestone Objective complete, transition to the Transition Phase and repeat steps 5 & 6.
10. Transition Phase Milestone Objective complete, Change Manage and maintain the product.

Below is the workflow for each iteration with the resulting artifacts classified by workflow-discipline. Remember, the workflow is always the same for each iteration within each phase. Again the amount of time and effort expended for each workflow-discipline is determinant on the Phase in which the iteration is being executed; for instance, a focus on Use-Cases is negligible in the Transition Phase but predominant in the Requirements discipline of the Inception Phase.

TABLE A: WORKFLOW STEPS FOR AN ITERATION

Process Disciplines	Steps	Human Actions	Artifacts Produced*
Business Modeling (Business Understanding)	1. 2.	For initial iteration, **ELICIT Business Rules, Business Needs, Business Understanding** ; for all subsequent *x* iterations **DETAIL Business Rules, Needs, Understanding** For initial iteration, **IDENTIFY** all significant **Business Use-Cases, Specifications, Models, Rules, Vision, and Architecture**; for all subsequent *x* iterations **DETAIL Business Use-Cases, Specifications, Models, Rules, Vision, Architecture**	Target Organizational Assessment Document, Business Glossary Document, Business Rules Document, Business Use-Case Model, Business Vision, Object Model, Business Architecture Document, Supplementary Business Specification, Business Glossary
Requirements (User/System Requirements Gathering)	3. 4.	For initial iteration, **ELICIT Requirements (Requests), & Rules**; for all subsequent *x* iterations **DETAIL Requirements (Requests), & Rules**. For initial iteration, **IDENTIFY** all significant **Use-Cases and classify by risk**; for all subsequent *x*	Stakeholder Requests Requirements Management Plan, Supplementary Specification, Use-Case Specification, Use-Case Model, Glossary, Software Requirements Specification, Storyboard, Use-Case Package Diagrams, User

Copyright ©2003-2005 by X-tier SAE Inc., all rights reserved

	5.	iterations **DETAIL Use-Cases (high risk Use-cases first), Specifications, Models, Realizations** to match all lower-level **Analysis Classes and Analysis Diagrams** and higher-level **Business Rules, & Requests**. **PRIORITIZE** or **REPRIORITIZE USE-CASES** by **RISK.**	Interface Prototype
Analysis & Design (Behavioral & Structural Modeling)	6. 7. 8. 9.	For initial iteration, **CREATE Collaboration Diagrams, Analysis Classes, Analysis Packages, Charts, Realizations, Definitions, & Analysis Models**; for all subsequent *x* iterations **DETAIL Collaboration Diagrams, Analysis Classes, Analysis Packages, Charts, Realizations, Definitions, & Analysis Models** to match all lower-level **Design Class Diagrams** and higher-level **Use-Case** Models. For initial iteration, **CREATE Sequence Diagrams, Analysis Classes, Analysis Packages, Charts, Realizations, Definitions, & Analysis Models**; for all subsequent *x* iterations **DETAIL Sequence Diagrams, Classes, Packages, Charts, Realizations, Definitions, & Models** to match all lower-level **Design Class Diagrams** and higher-level **Use-Case** Models. For initial iteration, **CREATE Design Classes & Class Diagrams**; for all subsequent *x* iterations ns **DETAIL Design Classes & Class Diagrams** to match all higher-level **Analysis Classes, Diagrams, & Models**. For initial iteration, **CREATE Data Models**; for all subsequent *x* iterations **DETAIL Data Models**.	Communication Diagrams, Object Diagrams, Sequence Diagrams, State Charts, Activity Diagrams, Package Diagrams, Class Diagrams, Software Architecture Document, Deployment Model, Analysis Model, Design Model, Proof-of-Concept Prototype, Use-Case Realizations, Design Packages, Subsystem Design Packages, Design Classes, Unit Test Classes, Analysis Classes, Data Models, Data Definitions
Implementation (Process Modeling)	10.	For initial iteration, **CREATE Component Diagrams & Models**; for all subsequent *x* iterations **DETAIL Component Diagrams & Models** to reflect any changes to **Design Classes, Diagrams, & Models**.	Implementation Model, Component Diagrams
Test (Quality Assurance)	11.	For initial iteration, **CREATE Class Diagrams, Logs, Lists, Components, Classes & Architecture**; for all subsequent *x* iterations **DETAIL Class Diagrams, Logs, Lists, Components, Classes & Architecture**.	Test Cases, Test Classes, Test Plan, Test Evaluation Summary, Test Scripts, Test Ideas List, Workload Analysis Model, Test Data, Test Results, Test Log, Test Guidelines, Test Classes, Test Components, Test Interface Specification, Test Automation Architecture, Test Environment Configuration
Deployment (Environmental Modeling)	12. 13.	For initial iteration, **CREATE Deployment Diagrams, Builds, Instructions, Plans, Notes, Releases**; for all subsequent *x* iterations **DETAIL Deployment Diagrams, Builds, Instructions, Plans, Notes, Releases**. For initial iteration, **CREATE Component Diagrams, Builds, Instructions, Plans, Notes,**	Deployment Diagrams, Alpha Software Build Releases, Beta Software Build Releases, Versioned Software Build Releases, Release Notes, Deployment Plan, Bill of Materials, Installation Instructions, End-User Support

Copyright ©2003-2005 by X-tier SAE Inc., all rights reserved

		Releases; for all subsequent *x* iterations **DETAIL Component Diagrams, Builds, Instructions, Plans, Notes, Releases**.	Material, Training Materials, Artwork
Change Management (Risk & Capacity Planning)	**14.**	For initial iteration, **CREATE Change Management Plan, Requests, Findings**; for all subsequent *x* iterations **DETAIL Change Management Plan, Requests, Findings**.	Change Management Plan, Change Request, Configuration Audit Findings
Project Management (Resource & Time Management)	**15.**	For initial iteration, **CREATE Project Management & Iteration Plans, Lists, Records, Cases, Orders, Assessments**; for all subsequent *x* iterations **DETAIL Project Management & Iteration Plans, Lists, Records, Cases, Orders, Assessments**.	Project Plan, Iteration Plan, Business Case, Software Development Plan, Iteration Assessment, Status Assessment, Problem Resolution Plan, Risk Management Plan, Risk List, Work Orders, Product Acceptance Plan, Measurement Plan, Quality Assurance Plan, Issues List, Project Measurements, Review Records

Resources/References

Software Process Engineering Metamodel (SPEM)

- Object Management Group: www.omg.org

Unified Modeling Language

- Object Management Group: www.omg.org

Rational Unified Process

- Rational Software Corporation: www.rational.com

About The Author

Rushton Prince is the Chief Architect and one of the founders of X-tier SAE Inc, a Kansas City based Software Engineering Firm. He began programming in 1980 on a beloved demonstrator TRS-80 at an Upland California Radio Shack; then, in his words, “upgraded to a HeathKit H-89 and Apple IIC.”

His experiences have been as a Principal Manager, Instructor, Mentor, Systems/Applications Architect, and Programmer. Since 1997 he has, as a consultant, helped Fortune 100 companies with the establishment of security centric engineering and operational environments for intelligent agents and distributed (Web Portal) object oriented systems and applications. Side-by-side with Rational Software’s experts, he has performed large and small scale RUP implementations.

Rushton is a published writer, is FCC licensed for the operation, maintenance, and repair of commercial radio based electronics (wireless), holds an A.A.S. in Avionics, a B.S. in Information Technology, and is currently pursuing his M.B.A.

Copyright ©2003-2005 by X-tier SAE Inc., all rights reserved
Rev 9/12/2005-5