

# Unified Process

The **Unified Software Development Process** or **Unified Process** is an iterative and incremental software development process framework. The best-known and extensively documented refinement of the Unified Process is the Rational Unified Process (RUP). Other examples are OpenUP and Agile Unified Process.

## Contents

Overview

Unified Process characteristics

Iterative and incremental

Architecture-centric

Risk-focused

Project lifecycle (Phases of Unified Process)

Inception phase

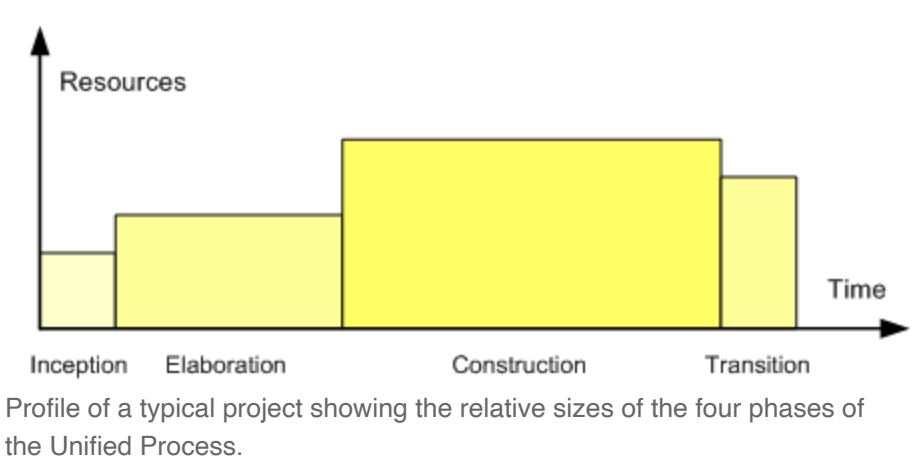
Elaboration phase

Construction phase

Transition phase

Refinements and variations

References



## Overview

The Unified Process is not simply a process, but rather an extensible framework which should be customized for specific organizations or projects. The *Rational Unified Process* is, similarly, a customizable framework. As a result, it is often impossible to say whether a refinement of the process was derived from UP or from RUP, and so the names tend to be used interchangeably.

The name *Unified Process* as opposed to *Rational Unified Process* is generally used to describe the generic process, including those elements which are common to most refinements. The *Unified Process* name is also used to avoid potential issues of trademark infringement since *Rational Unified Process* and *RUP* are trademarks of IBM. The first book to describe the process was titled *The Unified Software Development Process* (ISBN 0-201-57169-2) and published in 1999 by Ivar Jacobson, Grady Booch and James Rumbaugh. Since then various authors unaffiliated with Rational Software have published books and articles using the name *Unified Process*, whereas authors affiliated with Rational Software have favored the name *Rational Unified Process*.

In 2012 the Disciplined Agile Delivery framework was released, a hybrid framework that adopts and extends strategies from Unified Process, Scrum, XP, and other methods.

## Unified Process characteristics

### Iterative and incremental

The Unified Process is an iterative and incremental development process. The Elaboration, Construction and Transition phases are divided into a series of timeboxed iterations. (The Inception phase may also be divided into iterations for a large project.) Each iteration results in an *increment*, which is a release of the system that contains added or improved functionality compared with the previous release.

Although most iterations will include work in most of the process disciplines (*e.g.* Requirements, Design, Implementation, Testing) the relative effort and emphasis will change over the course of the project.

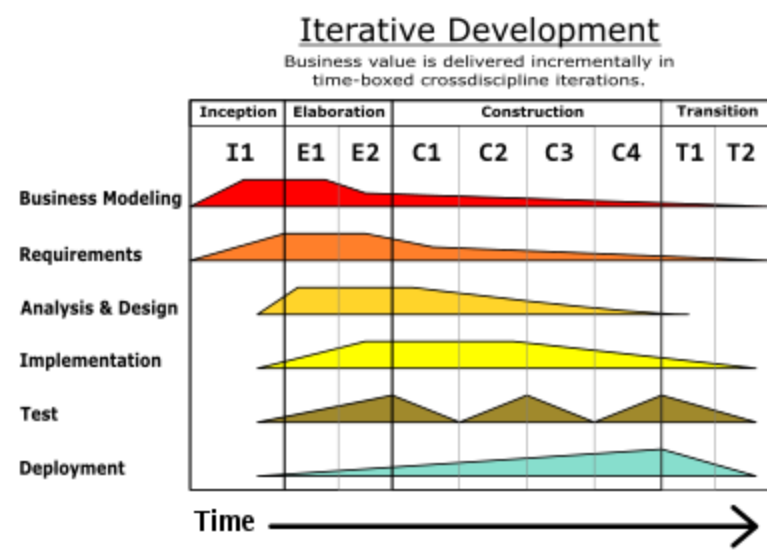


Diagram illustrating how the relative emphasis of different disciplines changes over the course of the project

**Architecture-centric**

The Unified Process insists that architecture sits at the heart of the project team's efforts to shape the system. Since no single model is sufficient to cover all aspects of a system, the Unified Process supports multiple architectural models and views.

One of the most important deliverables of the process is the executable architecture baseline which is created during the Elaboration phase. This partial implementation of the system serves to validate the architecture and act as a foundation for remaining development.

**Risk-focused**

The Unified Process requires the project team to focus on addressing the most critical risks early in the project life cycle. The deliverables of each iteration, especially in the Elaboration phase, must be selected in order to ensure that the greatest risks are addressed first.

**Project lifecycle (Phases of Unified Process)**

The Unified Process divides the project into four phases:

- Inception
- Elaboration (milestone)
- Construction (release)
- Transition (final production release)

Each phase will generally contain multiple iterations (named I1, E1, E2, C1, etc. in the UP phase illustration). The exact number of iterations in each phase depends on the scale and nature of the project. Note that the UP phase illustration here contains exactly 1, 2, 4 and 2 iterations in the four phases, but this is merely an example of how a specific project could look.

**Inception phase**

Inception is the smallest phase in the project, and ideally, it should be quite short. If the Inception Phase is long then it may be an indication of excessive up-front specification, which is contrary to the spirit of the Unified Process.

Develop an approximate vision of the system, make the business case, define the scope, and produce a rough cost estimate and project schedule.

## Elaboration phase

During the Elaboration phase, the project team is expected to capture a healthy majority of the system requirements. However, the primary goals of Elaboration are to address known risk factors and to establish and validate the system architecture. Common processes undertaken in this phase include the creation of use case diagrams, conceptual diagrams (class diagrams with only basic notation) and package diagrams (architectural diagrams).

The architecture is validated primarily through the implementation of an Executable Architecture Baseline. This is a partial implementation of the system which includes the core most architecturally significant components. It is built in a series of small time-boxed iterations. By the end of the Elaboration phase, the system architecture must have stabilized and the executable architecture baseline must demonstrate that the architecture will support the key system functionality and exhibit the right behavior in terms of performance, scalability, and cost.

The final Elaboration phase deliverable is a plan (including cost and schedule estimates) for the Construction phase. At this point the plan should be accurate and credible since it should be based on the Elaboration phase experience and since significant risk factors should have been addressed during the Elaboration phase.

## Construction phase

Construction is the largest phase of the project. In this phase, the remainder of the system is built on the foundation laid in Elaboration. System features are implemented in a series of short, time-boxed iterations. Each iteration results in an executable release of the software. It is customary to write full-text use cases during the construction phase and each one becomes the start of a new iteration. Common Unified Modeling Language (UML) diagrams used during this phase include activity diagrams, sequence diagrams, collaboration diagrams, State Transition diagrams and interaction overview diagrams. Iterative implementation for the lower risks and easier elements are done. The final Construction phase deliverable is software ready to be deployed in the Transition phase.

## Transition phase

The final project phase is Transition. In this phase the system is deployed to the target users. Feedback received from an initial release (or initial releases) may result in further refinements to be incorporated over the course of several Transition phase iterations. The Transition phase also includes system conversions and user training.

## Refinements and variations

---

Refinements of the Unified Process vary from each other in how they categorize the project *disciplines* or *workflows*. The Rational Unified Process defines nine disciplines: Business Modeling, Requirements, Analysis and Design, Implementation, Test, Deployment, Configuration and Change Management, Project Management, and Environment. The Enterprise Unified Process extends RUP through the addition of eight "enterprise" disciplines. Agile refinements of UP such as OpenUP/Basic and the Agile Unified Process simplify RUP by reducing the number of disciplines.

Refinements also vary in the emphasis placed on different project artifacts. Agile refinements streamline RUP by simplifying workflows and reducing the number of expected artifacts.

Refinements also vary in their specification of what happens after the Transition phase. In the Rational Unified Process the Transition phase is typically followed by a new Inception phase. In the Enterprise Unified Process the Transition phase is followed by a Production phase.

The number of Unified Process refinements and variations are countless. Organizations utilizing the Unified Process invariably incorporate their own modifications and extensions. The following is a list of some of the better known refinements and variations.

- Agile Unified Process (AUP), a lightweight variation developed by Scott W. Ambler
- Basic Unified Process (BUP), a lightweight variation developed by IBM and a precursor to OpenUP
- Enterprise Unified Process (EUP), an extension of the Rational Unified Process
- Essential Unified Process (EssUP), a lightweight variation developed by Ivar Jacobson
- Open Unified Process (OpenUP), the Eclipse Process Framework software development process
- Rational Unified Process (RUP), the IBM / Rational Software development process
- Oracle Unified Method (OUM), the Oracle development and implementation process
- Rational Unified Process-System Engineering (RUP-SE), a version of RUP tailored by Rational Software for System Engineering

## References

---

- Kroll, Per; Kruchten, Philippe (2003). *The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP*. ISBN 0-321-16609-4.
  - Kruchten, Philippe (2004). *The Rational Unified Process: An Introduction* (3rd Ed.). ISBN 0-321-19770-4.
  - Ambler, Scott (2002). *Agile Modeling: Effective Practices for EXtreme Programming and the Unified Process* (<http://www.ambysoft.com/books/agileModeling.html>). J. Wiley. ISBN 0-471-20282-7.
  - Scott, Kendall (2002). *The Unified Process Explained*. ISBN 0-201-74204-7.
  - Bergstrom, Stefan; Raberg, Lotta (2004). *Adopting the Rational Unified Process: Success with the RUP*. ISBN 0-321-20294-5.
  - Ambler, Scott; Constantine, Larry (2002). *The Unified Process Transition and Production Phases* (<http://www.ambysoft.com/books/transitionProductionPhase.html>). CMP Books. ISBN 1-57820-092-X.
  - Larman, Craig (2004). *Agile and Iterative Development: A Manager's Guide*. ISBN 0-13-111155-8.
- 

Retrieved from "[https://en.wikipedia.org/w/index.php?title=Unified\\_Process&oldid=1089527307](https://en.wikipedia.org/w/index.php?title=Unified_Process&oldid=1089527307)"

---

**This page was last edited on 24 May 2022, at 09:20 (UTC).**

Text is available under the Creative Commons Attribution-ShareAlike License 3.0; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.