



# RUP@EC

## Overview

---







## Table of Contents

1	Objective and Target Audience of the Document .....	2
2	What is RUP@EC .....	3
3	Positioning of RUP@EC.....	4
4	Specific Practices for RUP@EC.....	6
4.1	Service Modeling.....	6
4.2	Requirements Management .....	6
4.3	Use Case Driven Development .....	6
4.4	Risk Management .....	6
4.5	Project Process Tailoring .....	6
4.6	BPM@EC.....	6
4.7	Security@EC .....	7
4.8	Use Case Business Driven Modeling.....	7
5	Project Organization .....	8
6	About RUP .....	9
7	Benefits of RUP@EC.....	10

## **1 Objective and Target Audience of the Document**

The goal of this document is to provide a quick overview on the RUP@EC methodology.

RUP@EC is a very well structured and complete methodology and it's fully documented as a library. Due to copyright reasons, it can only be accessed inside the Commission network in the context of project implementation.

People with no knowledge on RUP@EC (or other EC methodologies) should be able to understand this document. Although no specific knowledge is required with RUP@EC, a solid background with Unified Process is recommended.

## 2 What is RUP@EC

The RUP@EC is a tailored publication of the RUP Practices Library to be used with any software development project initiative for the Commission.

As described in the next section, RUP@EC is now moving towards the integration of practices from OpenUp and other publicly available practices, including **DAD**. This means that several references to **OpenUp** may be found.

As an Agile methodology by nature, RUP@EC relies on a set of Agile principles and values that are described and explained in the **Agile@EC** guide and its tools and techniques.

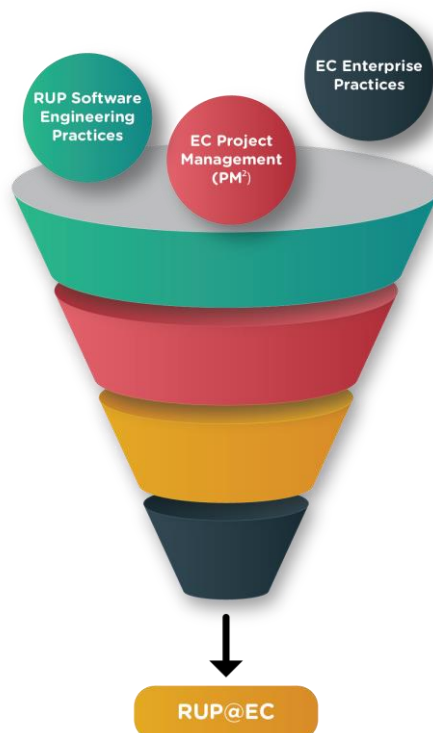
The RUP@EC Library is well described and provides all the necessary information to help those who need to use it. This doesn't mean that it should be seen as a fully rigid methodology that has to be adopted as it is. With RUP@EC, tailoring should be made in order to fit the specifics of a project taking into account its dimension, complexity, etc.

### 3 Positioning of RUP@EC

The RUP@EC is not a brand new methodology or approach to software development. RUP@EC tailors RUP to the EC ecosystem by:

- Adopting a relevant portion of the **standard practices**;
- Adopting and **customising** some of the standard practices;
- Incorporates several **specific** practices in use in the EC, like **BPM@EC**, Security@EC, etc;
- Incorporates specific Project Management principles from the **PM<sup>2</sup>** methodology.

As part of the strategy defined by the Commission, the RUP@EC will keep evolving towards a completely open source Library, adjusting the core principles and practices. This means that a transition will be made from an extensively RUP based library to a more **OpenUP** based tool, while continuing to take into account the EC specifics.

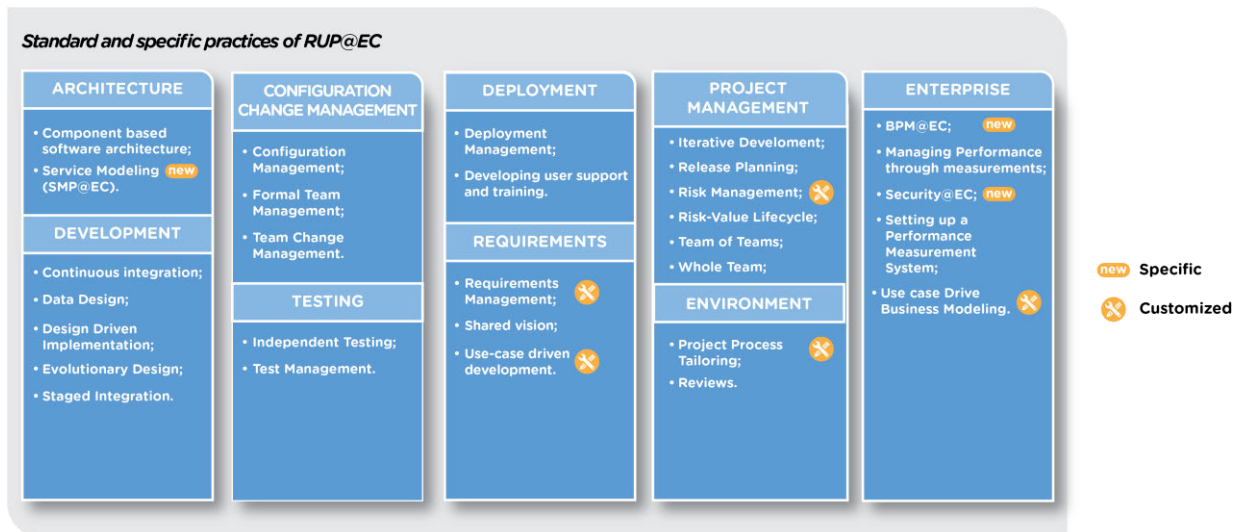


**Fig. 3.1:** RUP@EC is a set of well-defined practices and principles.

With the RUP@EC, a very important focus is put on the **Practices**. These “chunks” of processes are the foundation of RUP@EC and each one is described as a standalone capability adopted by the EC. As an example is the adoption of several practices in order to comply with the specific security standards of the EC or the integration of Business Process Management (BPM) standards as complementary techniques to describe business processes.

In a nutshell, the RUP@EC practices, all together, support the proper implementation of the methodology. Because RUP@EC is foreseen to become an open source methodology, several of the practices adopted are also common with the **OpenUP** implementation.





**Fig. 3.2:** Standard, customized and specific practices of RUP@EC

## 4 Specific Practices for RUP@EC

### 4.1 Service Modeling new

Service Modelling Practice (SMP@EC) can be used by IT units that wish to apply the Service Oriented Architecture (SOA) style to the development of IT Systems. SMP@EC provides guidelines for IT units that want to create a layer of abstraction including software services into their IT models. Services will bring more agility and flexibility to the IT landscape and the way to manage its evolution. Services will help to achieve the strategic objective of IT Rationalization, endorsed by the IT Governance put in place at the European Commission.

### 4.2 Requirements Management

Requirements management is a systematic approach to finding, documenting, organizing and tracking the changing requirements of a system. This practice provides a good starting point for requirements management. It recommends basic information to track about your requirements, including a standard set of traceability information to help you assess change impact.

### 4.3 Use Case Driven Development

Although this practice also exists in the OpenUP publication, in the EC context, this practice describes essential use cases and use case granularity, which encourages a team to create the right amount of detail for their use cases. This reduces development time by detailing only those sections of use cases that need to have significant thought and collaboration applied to them.

This practice relies on three important concepts:

- **Functional Decomposition:** an anti-pattern;
- **Use Case Granularity:** Concept that refers to the way in which information is "chunked" or organized within use case specifications, and to some extent, the level of detail at which they are written;
- **Storyboards:** technique that can be used to capture the operations that a system can perform for a particular purpose.

### 4.4 Risk Management

Every project contains some measure of uncertainty. The Risk Management practice deals with this uncertainty, trying to understand its potential influence on the project, such as the [risk consequence cost](#). Throughout the project, the project manager, team, and stakeholders are involved in identifying and prioritizing risks, proposing strategies to respond to them, and monitoring the events that might trigger the risks to happen. The goal in performing this practice is to increase the probability and impact of positive events and decrease the probability and impact of events adverse to the project.

### 4.5 Project Process Tailoring

The purpose of this practice is to guide the project team on tailoring and deploying the process to be followed. The team also makes decisions such as how to use work products, what are the project-specific guidelines, and so on. The tools to support the project are then installed, configured, and used by the team. It's also very important to mention that although this practice has its major effort in the beginning of the project, process tailoring should be an ongoing activity based on the constant learning and reflection that the team does after each iteration.

### 4.6 BPM@EC new

BPM@EC is an EC methodology for Business Process Management (BPM). This practice aims to support current users of BPM@EC methodology run their IT projects following RUP@EC. It can also be used by traditional RUP@EC users interested in alternative ways of describing business processes and other elements of business architecture. The objectives may vary from the need to reuse artefacts to expressing specific workflow patterns. They may be related to requirements management, testing or user documentation.

#### 4.7 Security@EC new

Security should be considered before and during the system development process to ensure sufficient security is designed and implemented in the system, to protect the information managed through it. The security considerations defined as integral part of the RUP@EC will help the system owner to consider corporate requirements and good practices and incorporate measures deemed necessary during the development process.

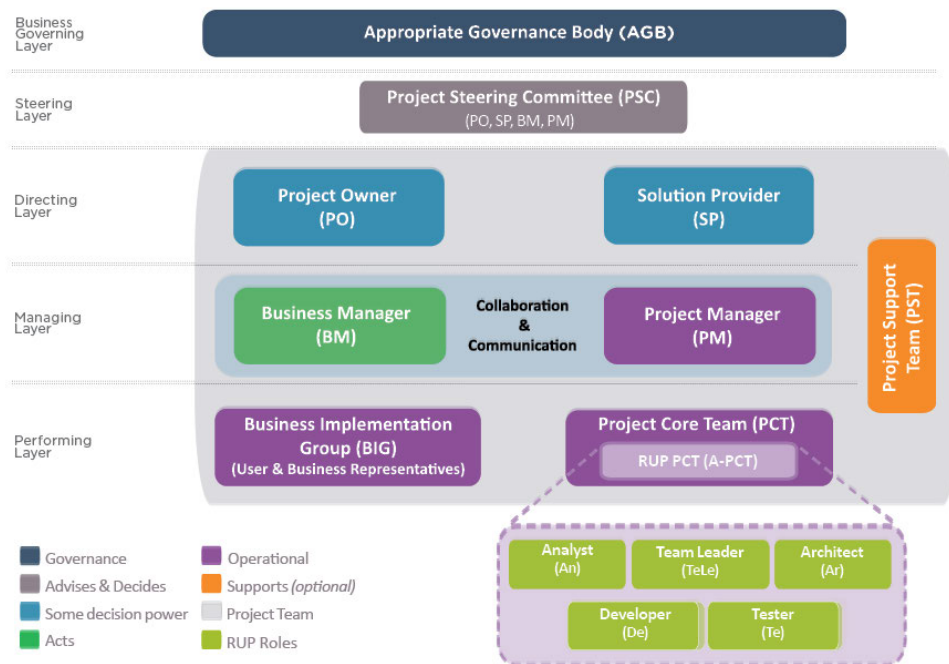
#### 4.8 Use Case Business Driven Modeling

Use Case-Driven Business Modeling (UCDBM) establishes a description of an organization for the purposes of business transformation, business process improvement and Service-Oriented Architecture (SOA) adoption.

The Use Case Driven Business Modeling practice improves consistency and predictability by clearly documenting as-is and to-be business processes with easy-to-follow use case and UML techniques. It improves oversight by assuring the value the business provides is realized by elements within the business. And it improves audit results by providing traceability from the business goals (use cases) to the description of how the business operates (realizations).

## 5 Project Organization

Another very important aspect in RUP@EC is the adoption of the governance model defined in the PM<sup>2</sup> methodology, having its principles reflected in the Project management discipline. As an example, a RUP team is seamlessly integrated in the overall project organization described by the PM<sup>2</sup> methodology.



**Fig.5.1:** How a RUP team fits inside the global PM<sup>2</sup> governance model

The Project Manager has a more active role in the operational perspective of the team, being more involved with the business solution of the system.

## 6 About RUP

RUP is a prescriptive, well-defined system development process, often used to develop systems based on object and/or component-based technologies. It is based on solid and proven software engineering principles such as taking an iterative, risk-value, scenario driven, and architecture-centric approach to software development.

The RUP has two main axes (or dimensions): One is the time, which is represented by means of Iterations, grouped into phases. These are ordered according to the key milestones being chased. The second one is the Content dimension. This one associates and puts together the activities into disciplines.

Progress is made by mitigating the most relevant risks and delivering value, through a product increment, after each iteration. After all the iterations completed, the product should be ready and delivered to be used by its users.

The outcome of each iteration is the result of the work synchronized between the several disciplines. In other words, all disciplines provide inputs towards the completion of the iteration's goal...

### Key Characteristics of RUP

- Short-term iterations with well-defined goals for continuous feedback collection;
- Decision points at the end of the various phases, to provide management visibility into the development process;
- Iterations are planned according to risk and value. The beginning of the project is more focused on mitigating higher severity risks as they may undermine the project success, while from Elaboration onwards, value is the iteration planning driver;
- Continuous delivery of value to the stakeholders in order to reduce risk related with uncertainty.

Because several generic practices of RUP are common to **OpenUP**, you can collect more information on those by visiting the site [here](#).

## 7 Benefits of RUP@EC

The benefits of RUP@EC come from the two main streams that support this methodology:

- Having the RUP as a foundation;
- Include key specific practices from the EC.

These two elements allow all the teams involved in IT projects to benefit, as a whole from:

- Constant feedback from the stakeholders;
- Enhanced ability to adapt to changing requirements and to incorporate change;
- Increased focus on REAL business value;
- Increased stakeholder satisfaction;
- Early, frequent and predictable delivery;
- Reducing delivery risk;
- Reducing architectural risk;
- Aligned with corporate management methodologies;
- Integrates EC project management philosophy and approach.