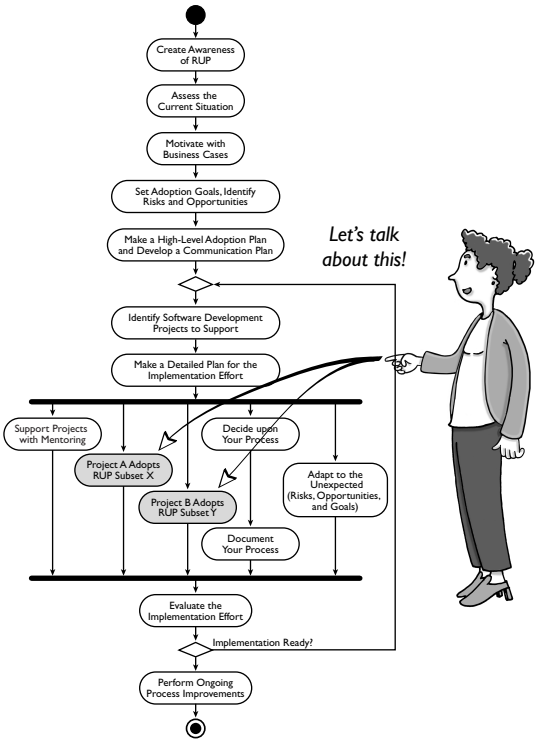
8

How to Adopt RUP in Your Project



Implementing RUP in a single software development project is *not* a minor task, even though it's less demanding than implementing RUP in an organization. Many factors impact the size of this task. One important factor is your current *process maturity*. Are the people who will staff your project used to working in a structured and predictable manner? Is it pretty clear what kind of documentation will be useful? How much of RUP do you plan to use? Actually, it is very important *not* to make the RUP adoption too big of a task! One of the secrets of a successful RUP implementation lies in the ability to decide upon a "just enough" amount (right-size) of RUP for your project to adopt without jeopardizing any of the objectives.

08Bergstrom.C08 Page 128 Thursday, December 4, 2003 12:06 PM

The main objective for every software development project—regardless of whether or not you're adopting RUP—should always be to successfully develop the intended software. Especially for that reason, it is important to *carefully plan* for your project's RUP adoption as well as *get support* for it so that it does well. Do not attempt too much, and make sure you benefit from what you do take on. Of course, your project's success is of interest to you and all the stakeholders, but it is also important for a potential rollout of RUP in your organization. (See the discussion of the sample goal "Ensure that every project that uses RUP succeeds" in the "Setting Adoption Goals" section in Chapter 6, "Planning the RUP Adoption.") To help you, find someone with experience, preferably a RUP mentor (also see Chapter 11, "A Guide to Successful Mentoring").

Perhaps your software development project is *part of an organizational initiative* of rolling out RUP across the organization. Perhaps your project is the pilot project. In any case, collaborating with the RUP implementation team is probably new to you. Just to set things straight here, your project manager is always in charge of your project's success in developing software. But because the real RUP adoption takes place on the project level, the success of the RUP implementation at the organization level therefore depends on your project. Ask for more help if you feel it's needed. Say no if you feel the implementation team pushes too many process demands on you. In this way you help your organization the most. And please be open minded and take some time to share your experiences of RUP, especially after the project has finished. The implementation team should help you with this (see Chapter 7, "Obtaining Support from the Organization").

So how do you adopt RUP within your project? In Figure 8.1 we present a simple flow of adoption activities that we recommend you perform. The subsequent sections in this chapter explore the details of the activities and guide you to further reading elsewhere in this book. The first thing your project team should do, preferably with some help, is to assess your project. Find out how you normally work and what your needs will be in this project. With that as input, you should select portions RUP to implement and plan how to achieve the required RUP knowledge. You do this in parallel with the ordinary project planning. After that, you run your project and get support on RUP. Finally, you share experiences with other people in your organization. This probably applies only if your project is part of an organizational RUP adoption.

08Bergstrom.C08 Page 129 Thursday, December 4, 2003 12:06 PM

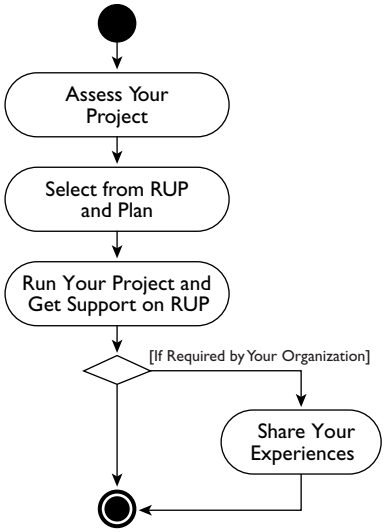


Figure 8.1 *A recommended flow of activities while adopting RUP in a single software development project*

8.1 Assessing Your Project

The project assessment forms the basis of your project's RUP adoption and enables you to "back the right horses" from the start. During the assessment, you look into the characteristics of your project and find out how the project members usually develop software. With that information at hand, the project manager can proceed with the next activity—planning the parts of RUP to introduce during the project.

If your project is part of an organizational RUP adoption, the implementation team should assist your project and provide assessors (RUP mentors) who make the project assessment for you.

8.1.1 Is RUP a Good Idea?

The first thing to figure out is whether RUP could apply at all to your project. (For simplicity, we'll talk about projects, but this information could apply to any software development assignment.) Answers to the fol-

08Bergstrom.C08 Page 130 Thursday, December 4, 2003 12:06 PM

lowing list of questions should provide you with the information you need to judge the applicability of RUP to a particular project.

1. Will the project develop software?
2. Will the project be staffed with people who have developed software before?
3. In general, are the project members interested in using RUP?
4. Does the project manager have a special interest in using RUP?

RUP includes guidance for software development. In order for your project to "match" and benefit from RUP, you should develop software of some kind.[1] That is not to say that the project has to involve completely new development starting from scratch! The project could just as well deal with *changing the configuration of a standard system* in order to match a certain need or problem. It could be evolution or *maintenance of existing software*. As long as the project has to do with finding a computerized solution to a set of functions defined as matching a certain need—for a business, for a market, or inside a technical system—RUP can be used.

Your project team should include people who have former software development experience. RUP won't teach you how to become a good programmer, for instance. RUP will guide you about *what* to do and even *how* to do many things, but basic software development knowledge is a prerequisite.

It is very hard to implement RUP within a team that is not at all interested in using RUP. Actually, we recommend that you do not use RUP if neither you nor the people who assist you can create any kind of interest.

It's very important to have a *project manager* who wants to use RUP. The project manager is in charge of all activities that take place in a project and makes them happen, including the RUP adoption. A RUP mentor only *supports* an adoption. The project manager has the most critical role during a RUP adoption. If the project manager doesn't want RUP, there will be no RUP.

---

1. Although RUP is intended for software development, many organizations get inspired by the concepts and ideas found in RUP and, for instance, use the terminology of artifacts, iterations, and the names of the phases for other areas as well. The use case modeling techniques described in RUP have been used for everything from laying out apple gardens to supervising a telecom network. Naturally, the amount of advice these applications get from RUP is very small. . . .

08Bergstrom.C08 Page 131 Thursday, December 4, 2003 12:06 PM

8.1.2 What Are the Characteristics of the Project?

The next thing to determine is what the project will do. Your project manager usually knows the answers to the questions listed below, but sometimes you'll need to consult the software architect.

1. What kind of system will you build (assemble, evolve, maintain)?
2. What will the technical platform be?
3. How big is the task? How difficult?
4. How long should it take?
5. How many people will be involved?

Read more about this in the "Types of Products and Projects" section in Chapter 4, "Assessing Your Organization."

8.1.3 How Do the Project Members Usually Develop Software?

The final thing to find out is how software is currently developed, including which aspects people think work well already and which don't, people's current knowledge, and their willingness to learn RUP. By asking the project members individually the following questions, you can survey the current process.

1. What are your roles? How do you normally work?
2. What documentation do you normally use? What documentation do you normally produce?
3. Which tools are used?
4. With whom do you normally communicate? Is it the same in this project?
5. How do you know what to do?
6. What is your view on how the overall development of software proceeds?
7. What do you know about RUP? What are your feelings about RUP in general?
8. What three things do you consider *good* about your work and this project? What three things do you consider *bad*?

Read more about this in the "What to Assess" section in Chapter 4, "Assessing Your Organization."

8.1.4 Documenting the Project Assessment

You should document your assessment findings in a bulleted list of the 15–20 topmost facts that summarize your project's situation (see Figure 8.2)

08Bergstrom.C08 Page 132 Thursday, December 4, 2003 12:06 PM

- *The team is new and on its way to establishing common routines.*
- *There is no collected set of requirements. Nonfunctional requirements are not documented.*
- *The test team asks for better specifications.*
- *A consultant who is good at RUP's Test discipline has been hired for the project.*
- *There is no change management of documents. Code is stored in tool X, documents on a common server and people's hard drives.*
- *The team will soon have a situation of more customers and several parallel installations.*
- *The project manager is responsible for requirements, project management, and delivery.*
- *The project develops Web applications with a lot of user interface requirements.*
- *There is a complex deployment environment whose setup is dependent on one individual.*

Figure 8.2 *Sample of project assessment findings*

and that might affect the RUP adoption in some way. Make sure everyone in the project agrees on that list.

If assessors from an organizational implementation team assist you, they should document their findings in a project assessment report. It's important that your project team agrees on that document and the summarized list of findings; ask for changes if you don't feel comfortable with these.

8.2 Selecting from RUP and Planning the Implementation

When your project's situation is known, you should choose a relevant subset from RUP for your project to adopt. Focus on areas where the need for a new process is substantial. You also need to plan for the process support required in order for the project members to learn and start working according to the selected parts of RUP. That support must be coordinated with other project activities; sometimes the scope of the RUP subset must

08Bergstrom.C08 Page 133 Thursday, December 4, 2003 12:06 PM

be cut down because it becomes obvious that the project team hasn't got enough time to adopt it all. Pay attention to the tool support required by the chosen RUP subset. You probably need to plan for installation, configuration, and setup of a number of tools as well.

If your project is part of an organizational RUP adoption, the implementation team will provide RUP mentors to help you make a good selection from RUP as well as a list of recommended process support activities for you to put in your project plan.

8.2.1 Deciding upon and Documenting Your Process, Part 1

Depending upon what areas you consider to be weak, you choose applicable parts from RUP. Your project-specific process will be a mix of RUP and your old way of working. You should not take on too much RUP at once—you need to master the situation and not let the development be shaken to its foundations. At this stage (part 1) you make a rough selection most likely based on the disciplines[2] in RUP. You probably also cut out parts of each discipline you select to follow. Later (part 2) you will learn more details of your process selection (see "Running Your Project and Getting Support on RUP" later in this chapter). But disciplines are not at all the only way to express your process; read more in Chapter 9, "Deciding upon Your Process."

Many projects have big needs, and although the *main* recommendation is to adopt the selected parts of RUP rather deeply to become "a process in the heart of the employees," there are *some* cases where a wide and shallow strategy might be necessary; see Chapter 6, "Planning the RUP Adoption."

Your project-specific process, or at least the parts you select from RUP, should be documented in a *development case*. There are different ways to present the development case, including through the use of a normal document. Read more in Chapter 10, "Documenting Your Process." The development case not only documents your process but also should tell the readers which *tools* support which parts of your process.

2. The disciplines in RUP are Business Modeling, Requirements, Analysis & Design, Implementation, Test, Deployment, Configuration & Change Management, Project Management, and Environment.

08Bergstrom.C08 Page 134 Thursday, December 4, 2003 12:06 PM

8.2.2 Planning the Process Support

It is very hard to change a way of working if you don't get some *extra time* and some support in terms of training as well as practical assistance from someone with experience, preferably a RUP mentor. You ought to plan for process support on the parts from RUP selected for your project-specific process. Which RUP subsets require the most support, and how is that support best given? What training and mentoring is needed and when?

Training will most likely consist of a course in RUP basics for everyone as well as one or more in-depth courses for some project members. Mentoring can be provided individually to project members acting in certain RUP roles and sometimes to a group of people in a workshop (e.g., on how to perform use case modeling). RUP mentors can help people to get started with RUP and also to conduct reviews of documents, models, and project plans, ensuring that the guidance of RUP has been followed in a beneficial way in order for the project to reach its software development objectives.

If an organizational RUP implementation team assists your project, you should ask for *proactive support to be planned in advance* (to some extent). The initiatives for assistance should not come only from your project team calling in RUP mentors when you do not know what to do. Read more about this in Chapter 7, "Obtaining Support from the Organization." Figure 8.3 shows the RUP and tools adoption support given to a sample project adopting parts of RUP's Project Management, Requirements, and Test disciplines and a tool for Configuration Management. As you can see, the support is rather extensive in the beginning during the project's Inception and Elaboration phases.

8.2.3 Documenting the Process Support

Your project's process will be documented in a development case. But where should you document the activities for the process and tool support? Usually you document the process and tool support activities in the project's *normal project planning* only. If you use the Project Management discipline from RUP, the planning documents will consist of the software development plan (SDP) and a number of iteration plans. Among the other "normal" project activities, you will find RUP-related

08Bergstrom.C08 Page 135 Thursday, December 4, 2003 12:06 PM

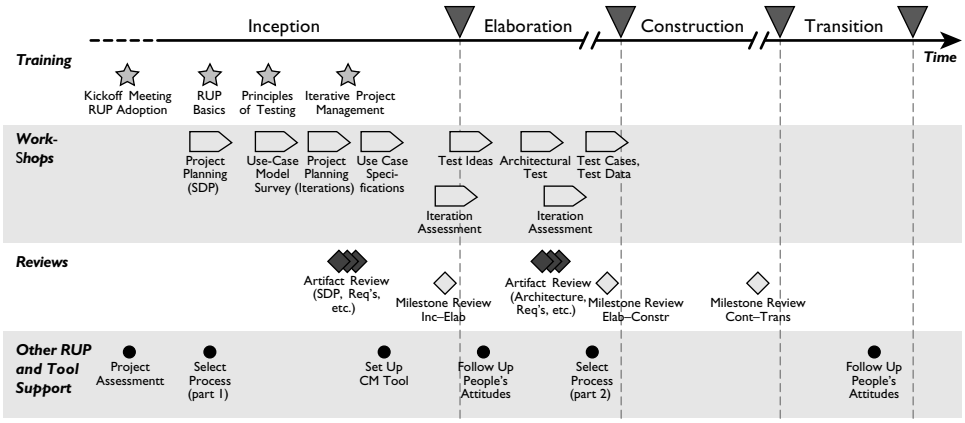


Figure 8.3 *Examples of the RUP adoption support given to a software development project*

activities—RUP training, RUP workshops, RUP reviews, and the RUP artifacts that are expected as deliverables. The SDP refers to the development case mentioned above.

If an organizational process team supports your project, the team may choose to document the process and tools support in a separate project implementation plan; see Chapter 7, "Obtaining Support from the Organization." No matter how the overview planning is documented, the activity list will look the same because all activities in a project should be included in an iteration plan.

8.3 Running Your Project and Getting Support on RUP

Your project is running and you start working according to the subset of RUP you have selected to be part of your project-specific process. A RUP mentor provides support on all RUP-related activities along the way. This support should be both proactive and reactive. *Proactive support* is the kind of support you can plan for in advance (see Figure 8.3 above); it consists of training, workshops, and reviews.

08Bergstrom.C08 Page 136 Thursday, December 4, 2003 12:06 PM

But because it is impossible to foresee the project team's needs completely, the RUP mentor also needs to give *reactive support* either individually to project members acting in certain RUP roles or sometimes to a group of people by conducting an extra workshop. Make sure you get enough support! A RUP mentor should be able to support the project manager by defining all the RUP-related detailed activities in the project's iteration plans (see below).

Another thing that is impossible to foresee is the final process your project will use. You have made a first selection from RUP, but you need to find out how that selection will look when applied to your case when the project is running. Probably several situations will appear and raise questions: "How shall we do this?" "What would this be for us?" A RUP mentor should help you fine-tune your process and really listen to you and your project members to make the process fit your needs.

If an organizational implementation team supports your project's RUP adoption, this team will provide you with RUP mentors. Read more about the support to expect in Chapter 7, "Obtaining Support from the Organization." In Chapter 11, "A Guide to Successful Mentoring," you can learn why RUP mentors are important and how they should work and behave.

8.3.1 Getting Support Down to the Activity Lists in the Project's Iteration Plans

It is important to receive *concrete* support from your RUP mentor. Figure 8.4 shows an iteration plan for the only iteration in the Inception phase for the sample project mentioned earlier (a project that adopts parts of RUP's Requirements, Project Management, and Test disciplines and a tool for Configuration Management).

8.3.2 Deciding upon and Documenting Your Process, Part 2

You will work out the final project-specific process while running the project. Your project's way of using RUP will be decided upon and documented in one way or another. As mentioned earlier, at the start of the project you will have a rough selection (ideas or drafts) of what your process will be (part 1 of process tailoring). When exposed to the specific problems of your project, you will refine and detail that first selection to

08Bergstrom.C08 Page 137 Thursday, December 4, 2003 12:06 PM

1. Introduction

1.1 Purpose

To describe what will happen short-term in the project. There will be one iteration plan for each planned iteration (see Ref 1, SDP). The project manager uses the iteration plan to plan all detailed activities and follow them up. The project members use the iteration plan to see what they are expected to do in the project and which other activities they might depend on.

1.2 Scope

The first iteration in Inception (hopefully the only one!).

1.3 References

Ref 1 SDP (Software Development Plan)

Ref 2 Risk List

Ref 3 Use Case Model Survey

1.4 Overview

At first the objectives of the iteration are described as a list of artifacts and to what extent these should be ready (expressed in %). The risks addressed and use cases realized during the iteration are listed. The resources working are listed by name, role, and level of involvement (expressed in %). At the end you will find a complete activity list, a "to do" list including activity, who the activity is assigned to, when it is expected to be ready, and a reference to the corresponding RUP activity (if any).

2. Iteration Objectives (Evaluation Criteria)

To roughly define the requirements and to suggest ideas for a solution, a "paper-only" architecture. To conclude Inception. The iteration will end June 13, 2003.

The following artifacts will be produced/worked on during the iteration:

– Use Case Model

– Use Case Model Survey (~80% ready)

– Use Case Specifications for prioritized use cases (~20% ready in total)

– Supplementary Specification (~80% ready)

– Glossary (~90% ready)

– Paper-only architecture (~100% ready)

– SDP (~80% ready)

– Iteration Plan for e1 (first iteration in Elaboration) (~80% ready)

– CM Plan (~40% ready)

– Master Test Plan (~10% ready)

3. Risks

#5, Not enough resources (see Ref 2).

4. Use Cases

Not applicable in Inception.

5. Resources

Name	Role	Working x-%
Peter	Project manager	90%
Sandra	Software architect	60%
Sid	System analyst, requirements speicifer	90%
Rebecca	Requirments specifier	90%
Timothy	Test manager	10%
Lotta	RUP mentor	20%

(Continues)

Figure 8.4 *Extracts of an iteration plan during the Inception phase*

08Bergstrom.C08 Page 138 Thursday, December 4, 2003 12:06 PM

6. Activity List

#	Activity	Resp.	Start	End	Comment (▷ = RUP activity)
1	Workshop, Use Case Model Survey	Lotta	3 Jan.	3 Jan.	Participants: Everyone ▷ *Find Actors and Use Cases* ▷ *Structure the Use Case Model*
2	Write Use Case Model Survey	Sid	4 Jan.	11 Jan.	▷ *Find Actors and Use Cases*
3	Write Supplementary	Rebecca	6 Jan.	15 Jan.	▷ *Detail the Software Requirement*
4	Write Glossary	Sid	4 Jan.		▷ *Capture a Common Vocabulary*
5	Write Use Case Specifications for "Register new member"	Sid	7 Jan.		▷ *Detail the Use Case*
6	Workshop Project Planning SDP, 1st	Lotta, Peter	10 Jan.		▷ *Plan Phases and Iterations*
7	Workshops Project Planning SDP, 2nd	Lotta, Peter	11 Jan.		▷ *Plan Phases and Iterations* ▷ *Identify and Assess Risk* ▷ *Prioritize Use Cases* ▷ *Develop Iteration Plan*

Figure 8.4 *Extracts of an iteration plan during the Inception phase (continued)*

make up your final process (part 2 of process tailoring). Any need to customize the process (extend it or change it slightly) should be taken care of as soon as possible in order for you to create an effective process. A RUP mentor should be able to help you. But remember that the main purpose of the project is to deliver software; the delivery must never be postponed by any long-lasting process documentation activities. The decisions about how to do things should be documented "quick and dirty" (but still documented!). Later this quick-and-dirty documentation may evolve to proper guidelines, checklists, templates, and so on.

If your project is part of an organizational RUP adoption, you know your project's application of RUP will be of much interest to other people within your organization. If you have RUP mentors from an implementation team supporting you, they will help you with the application of RUP and your needs for tailoring/customization (if any). The RUP mentors should write down what have proven to be good ways to work in your project and then find ways to merge and insert these techniques into existing RUP activities. In this way the RUP mentors prepare for the harvesting and collection of working processes from the projects (see the next section).

08Bergstrom.C08 Page 139 Thursday, December 4, 2003 12:06 PM

Read more about tailoring/customization of RUP in Chapter 9, "Deciding upon Your Process." An easy way to document a project's application of RUP is to write guideline documents, but there are other ways too; read more in Chapter 10, "Documenting Your Process."

8.4 Sharing Your Experiences

Your experiences of RUP during your project can be helpful and interesting to others within your organization, especially if you are part of an organizational RUP adoption. Perhaps your project marks the first time RUP has been applied within your organization. You should spend some time sharing your experiences. You could do this in various ways: *talk informally* to other people about what you have done, how you liked it, and so on; *write something down,* hopefully in the form of a success story; or run a meeting with the goal of telling your story.

If your project is part of an organizational RUP adoption, the implementation team will *arrange opportunities* for you to share your thoughts. You could share your experiences in person formally by talking at RUP experience meetings or seminars or just informally by talking to another person who will take on the same RUP role in another project. You could share your experiences in written format via your organization's RUP Web site; see Chapter 10, "Documenting Your Process." If it hasn't already done so, the implementation team will *harvest* your project to find valuable material in the form of guidelines, checklists, templates, details, and customizations of various RUP activities; examples of RUP artifacts; and so on. These will be *merged* with your organization's process on the RUP Web site and shared with all people in your organization. Upcoming project teams will benefit a lot from examining what other project teams have done and reusing successful parts of the process, which will also speed up their usage of RUP.

Your project has applied and tried out the process. Delivering the project on time is often a good-enough result. However, if the RUP implementation team has chosen a more formal way to follow up the project's performance, this is the time to collect measurements and try to discover whether the results of the project are aligned with the organizational RUP adoption goals. Read more in Chapter 5, "Motivating the RUP Adoption."

08Bergstrom.C08 Page 140 Thursday, December 4, 2003 12:06 PM

This is also a good time to follow up people's attitudes regarding the changes in ways of working.

8.5 What Will Happen after the Project?

The project is over, and hopefully some parts of RUP were successfully adopted. You may wonder whether there will be any further benefits from RUP. In fact, we have some very good news. *You will benefit a lot from the RUP adoption during the maintenance of your software!*

How you will benefit depends of what parts of RUP you adopted. If you adopted use case modeling for the requirements, it will be much easier for new people to maintain the developed software and understand it. If you adopted complete model-driven development (use case model, design model, code generation), you can quickly and safely make functional changes to the software. If you adopted the Test discipline, you have documented test cases that will help you make sure the software still works after a change. Not to mention what a good architecture can do for a software system that will last and evolve for years!

8.6 Conclusion

It all happens in the projects! A project's adoption of RUP will differ depending on whether it is supported by the organization or is doing the adoption by itself. However, the same types of activities apply, namely, assessing the project, selecting a subset from RUP, and planning activities regarding the adoption together with the normal development activities. Hopefully the project team can get support from a mentor while the project is running. After the project ends, experiences regarding the process should be shared. This is especially important if the project is part of an organizational adoption of RUP. Knowledge about RUP and software development should be transferred from project to project.

In the next chapter, we will guide you in how to decide on elements of your process.