

Comprehensive Documentation Made Agile – Experiments with RaPiD7 in Philips

Ko Dooms¹,
Roope Kylmäkoski²

¹ Philips, Building SFJ-3,
Glaslaan 2
5616 LW Eindhoven, The Netherlands
ko.dooms@philips.com

² Nokia, Hatanpäänvaltie 30,
33101 Tampere, Finland
roope.kylmakoski@nokia.com

Abstract. This paper addresses the almost never-ending headache the role of documentation has given for software projects. *Working software* has been given recently a focus over *comprehensive documentation*, yet the required documents should be authored. This paper “revisits” the approach developed by Nokia improving the documentation work without scarifying the quantity or quality of documentation. The method is called RaPiD7. The cases presented are from Philips Digital Systems Laboratory. This paper elaborates the method by providing insights to applying RaPiD7 in practice, explains the encouraging results of the experiments and gives tips for practitioners of the method by explaining the lessons learned in Philips.

1 Introduction

In recent years agile methods (see for example [3]) have been the focus of discussion in the area of software engineering. Agile methods seem to be a response to the heavily prescriptive processes (see for example [5]) that dominated the field in the early 1990s. Agile methods focus more on the human aspects of software engineering than these so-called prescriptive processes, and place human interaction over tools and processes [1]. Agile methods also state that documentation is often a heavy and unneeded form of communication.

In [4] another method for documentation work is presented that combines to a certain extent the good sides of both directions, in other words, from prescriptive processes and from agile methodologies. The method addresses the challenge of creating understanding, sharing understanding and storing the created understanding in software projects. These challenges are addressed by a stronger focus on planned human interaction and early joint decision-making by the project development team. In practice, this method is about planning the needed human interaction and decision-making as part of project planning in the form of facilitated workshops, and then

2 Ko Dooms¹, Roope Kylmäkoski²

subsequently carrying out this plan. The method is called RaPiD7 (Rapid production of documentation – 7 steps), and it has been developed by Nokia during 2000-2001. Similar methods to RaPiD7 exist (see for example [2], [7], [6] and [8]), therefore the intention of this paper is not to provide a totally new or unique approach for software engineering. Consequently, this paper is not comparing the different methods, but is rather a lessons-learned paper providing insights on a particular case with a selected collaborative method for software related documentation and design. A similar method to RaPiD7 called JAD [6] has existed since the 1970s and recently agile methods have been addressing the same field especially by a method called Agile Modeling [2]. However, it does not appear that any of these methods, be it JAD, AM or RaPiD7, have gained the industry de-facto status they should have. We believe that methods like RaPiD7 should be an integral part of almost any software process. Thus we also see publishing the usage results as paramount. There have been too few concrete steps taken on ensuring effective human interaction in the field of software engineering. Teamwork has often been encouraged, but the real support has been missing.

Early results from the use of RaPiD7 have already been presented in [4]. Work is still continuing inside Nokia, with more results to be published. RaPiD7 use has now spread outside Nokia too. First trials of the method outside Nokia were carried out in Philips during April 2004. This paper explains this case in detail, giving concrete suggestions how the method can be applied (something that was not presented in [4]) with some encouraging results similar to what has been published by Nokia before. Furthermore, the analysis of the applicability of the method in another environment besides Nokia is briefly discussed, too.

This paper first briefly explains RaPiD7 and the baseline status in Philips. Then the actual cases from Philips are presented with details about the planning and implementation phase. The results from the cases are presented and analyzed. Finally conclusions are drawn from the cases and from the whole paper.

2 RaPiD7 in Philips – a Case Study

2.1 Brief Introduction to RaPiD7

In RaPiD7, creating understanding for the specifications is mostly done jointly, information sharing is done continuously early on and document writing is done as far as possible jointly, too. Furthermore, the quality assurance is built into the way of working. In practice, all this is carried out in the form of facilitated workshops. The approach is presented briefly in Figure 1.

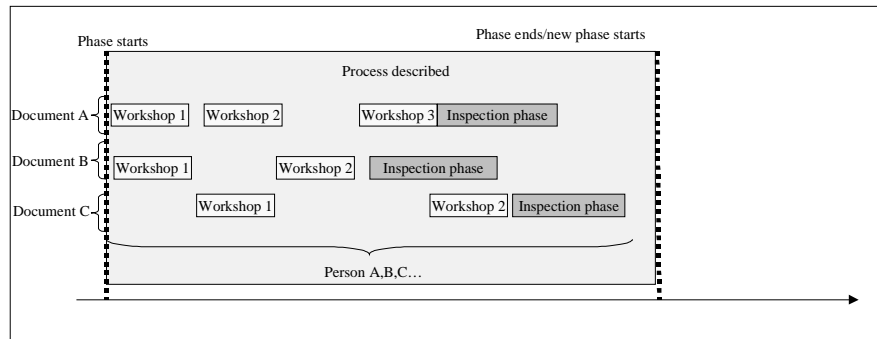


Figure 1, RaPiD7 approach in brief

Furthermore, RaPiD7 provides a three-layer structure. First the *project layer* describes how human interaction and joint decision-making is planned for software projects. In practice this means identifying the cases for applying RaPiD7. The *case layer* describes how the selected cases such as documents are to be created in consecutive workshops, and the *workshop layer* describes how the actual work is carried out in form of facilitated workshops.

RaPiD7 workshops comprise from seven steps. The steps aim at providing information on how to organize efficient workshops in software projects. The steps of RaPiD7 are shown in Figure 2.



Figure 2, Steps of RaPiD7 [4]

The workshops are planned in detail (1. *preparation*), then initiated properly (2. *kick-off*) and different ideas are gathered (3. *idea gathering*). Problem solving (4. *analysis*) and decision-making (6. *decisions making*) techniques are used in the workshops. The decisions are written down to a desired level (5. *detailed design*) and workshop results are verified and next steps are agreed on (7. *closing*). RaPiD7 workshops typically produce documents that can be finalized after the workshops in a few hours. Although

4 Ko Dooms¹,
Roope Kylmäkoski²

the workshops play the role of an inspection as well, a short inspection is typical for the documents produced with RaPiD7.

2.2 Background

We, at Philips Applied Technologies, translate innovative product ideas from research into real product implementations, such as flat TV's and DVD recorders. During the creation of these products we have to write an extensive amount of project documentation. This documentation is required to create a common understanding of what has to be developed, what the proper system architectures are, and provide details on the design for development. Furthermore when a product reaches the maintenance phase the maintainers of the product have to use the documentation to solve problems in the product or extend the products with new features. Philips Applied Technologies is a CMM level 3 organization, which means projects should be created in a repeatable and defined way. To achieve this, an efficient documentation process should be in place.

To achieve this we first looked into challenges and common mistakes in the writing of project documentation. These are listed in Table 1.

Table 1, Challenges and common mistakes in writing project documentation

Challenges	Common mistakes
Readability by the customer (for example product management and business owners)	Unnecessary documentation produced by inexperienced project people
Being able to cover everything that is required in the document e.g.: What is essential and what isn't Are the captured items actually describing what the stakeholders of the documentation need Where do we get the right input	Too little documentation written by experienced project people Experienced people can either be too busy with other tasks or see some aspects as too obvious to be written down
How to keep/make it consistent with other documents or project deliverables, for example code or test scripts	
Achieving common understanding of the decisions written down	

We, at Philips Applied Technologies, have been looking for ways to improve our way of working and to address the mentioned challenges. When we first heard of RaPiD7, the method developed by Nokia, we were attracted by its structured workshop approach used in the creation of project documentation. Including workshops as part of the way of working was not new to us, but using a structured approach where the document at hand is really written inside the workshop was. We decided to get more out of the workshops by integrating them better into our daily way of working and add more structure to the workshops. In addition, we were intrigued by the possibility

of finalizing the required documentation almost completely in the workshops. Furthermore, we also see inside Philips Applied Technologies an increasing number of requests from our customers to “do more with less”. There are high pressures especially in:

- Shortening the time to market
- Reacting faster to changes of the product specification during development
- Spend less Non Recurring Engineering money

Due to these requirements, the introduction of a more agile approach for our software process is needed and we see that RaPiD7 can play a role in this. At the same time we need to maintain our CMM level. However, this should not be an issue with RaPiD7, as Nokia is also using RaPiD7 in parts of their organization that are concerned with CMM level too. No negative consequences on the achieved CMM level have been noticed; on the contrary an improved effect on the KPA review process was the case, due the intensive collaboration and reviews in the RaPiD7 workshops.

The most important expected benefits from working in a more agile way and from RaPiD7 in specific are:

- More interaction expected between the project stakeholders: Philips Applied Technologies project members, customer representatives and third parties if involved
- Focus on writing only the essential documentation
- Create a common understanding, share knowledge better
- Create more focus, which should result in quicker results
- Respond better to changes

As these were exactly the benefits we were looking for we started to prepare ourselves for the trials with RaPiD7.

2.3 Preparing for the Case

In preparation for the introduction of RaPiD7 in Philips Applied Technologies, we (two Philips Applied Technologies employees) first followed a two-day in-house training course at Nokia. During the training we used this method in the creation of several documents such as:

- Requirements for a mobile application
- Most important “use cases” for this application
- High level architecture
- Interface description of the building blocks
- Estimates for the development of the application

The eight trainees all played a specific role such as project manager, product manager, architect and so on. In addition, we were confronted with practicing the facilitator role, which was seen as very important for the success of the workshop. We were surprised that in only a two-day workshop (where most people were not familiar with each other and were still learning the method) we got both usable output and quality of the deliverables as mentioned above.

After the training our thoughts about the usability of the method in Philips Applied Technologies were positive. However, to avoid initial high costs we experimented in

6 **Ko Dooms1,**
Roope Kylmäkoski2

one of the software groups within Philips Applied Technologies. The following describes the first trial case in more detail.

2.4 The case

2.4.1 The Baseline Status

Most of the employees in Philips Applied Technologies have a personal objective each year to write one or more so called “white cards”. White cards are the starting point for intellectual property patents. Before RaPiD7 was used, a year ago, a group of 10 people had the idea to brainstorm together to generate new ideas for white cards. The group used one afternoon and first visited *the room of the future* in our Philips research laboratory to stimulate the flow of ideas for brainstorming. After this visit they did a brainstorm session and discussed the generated ideas. The end result of the afternoon was an Excel-sheet with around 20 ideas written down. A single line described the idea. Furthermore each member of the group was assigned one idea to expand it into a white card after further investigation about the usefulness of the idea. After one year a few (3-4) ideas were converted into white cards and the Excel-file was passed as “a hot potato” between the 10 people until the process came to completion.

The promise of RaPiD7 to really finish some documentation work inside the workshop by collective decision and communication was the reason we wanted to try out a similar session with the help of this agile method.

2.4.2 The planning Phase

As a first step we planned a one-hour **preparation** meeting (step 1) together with the project leader and the team leader of the project team (in total 11 people) selected to generate white cards on the specific technology they were working on. During this preparation we went through the whole process (steps 2 to 7) and estimated time for the steps and assigned roles. Furthermore we decided upon techniques to be used and the materials we needed.

The most important outputs of the preparation were:

- The goal set by the manager of the group was: “Generation of a number of white cards related to the technology that the group was working on”.
- The target set was: “3 to 4 detailed and submitted white cards, plus several ideas ready for next workshop”.
- The project leader should present a short presentation to set the scene: known technology and trends of the subject.
- The team leader should present one patent (granted) in this field as an example (5 minutes needed).

- A detailed agenda (describing 4 hours of work) with all the steps and methods to be used was made.
- The list of needed material was as follows:
 - Enough laptops connected to the network
 - Post-it notes
 - A room
 - Invitations for the workshop and copies of the agenda

We found that the preparation of the above-mentioned subjects and the following of the process resulted in high and promising expectations for the workshop itself.

2.4.3 The Workshop

The workshop attendees were the manager of the department (who played the facilitator role), the project leader (who played the secretary role), the team leader and the team itself containing an architect and 7 software developers.

During the **kick-off** (step 2) the manager explained the goals and target of the workshop. A one-slide introduction was then given about RaPiD7 (explaining the steps). This was followed by two other short presentations about stimulating creativity (as already described above in the planning phase).

We then continued with **gathering ideas** (step 3). During this step, each person received a pen, an empty A3 sheet of paper and some post-it notes. The task was to think quietly, write a new idea on the note and then place the sticker on the A3 sheet. The sheet would then be passed to your neighbor and the process repeated. The idea was to stimulate further ideas on the same theme. This process continued for 25 minutes and after that each of the 76 stickers was placed on the white board. We then continued by clustering the ideas on the white board. After a further 10 minutes the facilitator took the lead and asked the contributors to clarify their ideas if necessary. We finished with 7 clusters of related ideas. The group was divided into three sub teams and each team picked a cluster that they liked to work on further. The remaining four clusters were saved by the scribe for later investigation (possible input for next workshop). After these two steps the team took a well-deserved break.

Each of the three sub-teams began with **analysing ideas** (step 4) of the cluster they picked. They asked for clarification when necessary and removed duplicates; they also wrote a one-line description of the ideas that were left. The final result was that two teams ended up with two ideas and one team with nine. Although the teams had different numbers of ideas to work on, the work seemed equally divided because the nine ideas were more easily to refine than the others.

Next step was **detailing** (step 5) the one-liners into a full description (text plus drawings). In this step, laptops were used so that the text was immediately written down in the right format. During this writing process an automatic review process took place. The fact that a single PC was used meant that review comments could immediately be taken into account. The team finished with a **decision** (step 6) on each produced white card and the result was submitted online. Each team member's personal opinion was taken into account. This way the work was completed in 4 hours, and the workshop finished on time. During the **closing** session (step 7) the

8 **Ko Dooms¹,**
Roope Kylmäkoski²

team with nine ideas made an appointment (next day) to finish and submit the last white cards.

Looking back on the workshop the most important team remarks were:

- Surprised at the number of generated white cards. The target was to have 3-4 white cards, but the workshop finished with 13.
- The focus of the team and energy used was high during the whole workshop
- Good help with the structure (agenda) of the meeting
- Clear what was expected
- Very satisfied feeling at the end result

This was our first exposure to using RaPiD7 in the authoring of documentation within Philips Applied Technologies. We saw that the workshop provided a productive and very interactive way of working.

2.4.4 The Results

Although in the above case we did not produce any project documentation such as an architectural document or a design document, we could still draw important conclusions from this experiment. We could assume that if the method was successful in the creation of white cards it might also be successful in the creation of more technical project documentation later. This could be proven in another case.

Conclusions of this first RaPiD7 trial within Philips Applied Technologies:

- Compared to the white card generation workshop of one year ago, we accomplished much more output using the RaPiD7 method. In the initial trial only one-liners were described and no white cards were submitted. In the RaPiD7 case most of the work was done inside the workshop
- Almost no “homework” was required after the workshop, which gave people a satisfied feeling
- Proper preparation is essential to success
- RaPiD7 seems usable for generating documents needed in Philips Applied Technologies
- The decision was made to deploy the method further

Successes often flourish, and the same happened in this case. The team members talked to other software people in the department that liked to copy the workshop approach for white cards. The manager networked with a manager from another Philips division and explained to him the results and method used. So far we have heard of 4 successful copies of the white card creation workshop with similar results.

2.4.5 A Step Further – Another Case

The next step was to invite the project team to develop a part of their technical documentation using the RaPiD7 method. A training/workshop session of two days was organized.

The workshop was held at Philips Applied Technologies premises with support from Nokia. The goal was to write an architectural document (describing the functionality of the system to be build, the major subsystems and their interfaces) and two detailed design documents of important components of the system. A template for both types of documentation was available. Normally this kind of documentation takes 40-100 hours to finish (write, review and rework). During this workshop we managed to come up with the most important parts of the documents, but not the full document. Finalizing the documents can be done in follow-up workshops or by the author himself. The most important conclusions of this workshop are listed in Table 2.

Table 2, Pros and challenges of RaPiD7 as seen by the workshop attendees

Pros of the method	Challenges of the method
Resulting documents are small in size, only the essential documentation is recorded and therefore easier to maintain	Trained and capable facilitators are needed
Results are written down in reviewed documentation	Risks if planning of workshop is not done properly are: Decisions are made too quickly Conclusions are drawn too quickly resulting in sloppy documents
Lots of collaboration and interaction took place, which stimulates discussions, common understanding and knowledge sharing of the problem at hand	Problem how to handle different knowledge levels of participants
Detecting of important issues (any kind) early in the process	Mismatch with quality system templates, these are not workshop friendly
Fast results when writing documents	Workshop teams should be formed carefully, only real stakeholders should be selected

If we study the pros in the table above we can conclude that using the RaPiD7 method we see some pretty results on the documentation process. Namely: Documents will be small, consist of the essential information, are reviewed and delivered quickly.

2.4.6 Suggestions, Conclusions and the Future

After the above-described workshops were completed we at Philips Applied Technologies drew up some tips for the further introduction and deployment in our lab. These are listed in Table 3.

Table 3, General workshop and deployment tips for RaPiD7

General tips from the team	Deployment Tips from the team
Clearly define the goals of the workshop	1 hour RaPiD7 training for every employee should be enough, then practice in projects
Consider to have longer (30 minutes) breaks for: Looking up things Thinking silently Check things with others	Select cases carefully To maximize success of deployment choose a case which is not too difficult
Use the seven steps liberal (combine were needed)	Setup/organize training/coaching for facilitators at department level
The use of templates is helpful	Consider how to create a balanced workshop team?
Try to use (parts of) the method in all types of meeting	Consider how to arrange facilitator assignment

In conclusion, Philips Applied Technologies in conjunction with the SPI steering group decided to proceed with the introduction of the RaPiD7 method. A further deployment plan will be made. We expect to introduce it using a bottom-up approach by selecting newly created project-teams and then deploying the method within these teams. As with the white card examples success should spread itself. At the same time we will support from top-down, an action to make the design templates more workshop-friendly.

3 Conclusions

The results found were only from a few cases of using RaPiD7 in Philips. Nevertheless, the presented workshops have been mostly successful and detailed descriptions of the way the workshops have been organized are presented. Furthermore, the results are a step forward from the baseline situation in Philips for the mentioned cases. We have created more interaction between the project stakeholders within the few cases and we have been able to focus on writing only the essential documentation. Common understanding has improved and reaching the results has been faster. However, there are definitely more steps we can take within Philips with RaPiD7. This approach needs to be integrated into the general way of working rather than just having ad-hoc workshops occasionally. The work towards this approach is in progress. This is the way to systematically reach the expected benefits we set for the improved way of working.

On the other hand, when the results are compared to the results Nokia has published, we can present similar results from our individual cases already now. In addition, this paper provides a concrete and pragmatic view on organizing RaPiD7 workshops and thus provides part of the missing guidance on using RaPiD7. The results give us the confident feeling that, in fact, RaPiD7 can be applied in other environments not

typical to Nokia. Naturally, the results cannot be generalized either by providing a single example outside Nokia. Nevertheless, the results emphasize the need for methods like RaPiD7 in overall.

References

- 1, Agile manifesto in web, <http://agilemanifesto.org/>, last visited in December 2004
- 2, Ambler Scott, Agile Modeling, John Wiley & Sons, Inc., 2002
- 3, Cockburn Alistair, Agile Software Development, Addison-Wesley, 2002
- 4, Kylmäkoski Roope, Efficient Authoring of Software Documentation Using RaPiD7, 25th International Conference on Software Engineering - Proceedings, IEEE, 2003
- 5, Sommerville Ian, Software Engineering, Addison-Wesley, 1996
- 6, Wood Jane, Silver Denise, Joint Application Development, John Wiley & Sons Inc., 1999
- 7, Coughlan Jane, Macredie Robert D., Effective Communication in Requirements Elicitation: A comparison of Methodologies, Springer-Verlag London Limited, 2002
- 8, Gottesdiener Ellen, Requirements by Collaboration: Workshops for Defining Needs, Addison-Wesley, 2002