DRAFT

full paper published in:

proc of. 7th European Conference on Software Quality

June 10-12, 2002 Helsinki, Finland

Paper presented at the Quality Forum poster session

Towards an integrated environment for risk management in distributed software projects

Janusz Górski[1], Jakub Miler[2]

[1] Technical University of Gdansk, Poland
jango@pg.gda.pl
[2] Technical University of Gdansk, Poland
jakubm@eti.pg.gda.pl

Abstract. The paper presents a concept of risk management in distributed software development projects. It recognizes effective, continuous and open communication as the prerequisite for successful risk management. Therefore, it concentrates on providing to the project stakeholders a broad and highly available communication channel through which they can communicate risk-related information. The channel has unlimited memory – it registers all incoming information much as the "black box" device memorizes all relevant data during an aircraft flight. The stored information can then be analyzed from different angles, e.g. to select and prioritize the most important risks or to analyze the project history in order to find out how risk perception developed during the project course. The description of a tool that embodies those concepts and preliminary reports from some validation experiments are also included.

1 Introduction

The importance of risk management has been well recognized by the project management community. In [8] risk management is listed among nine key knowledge areas related to project management. In relation to software project risks, much work has been done at Software Engineering Institute (SEI) [1, 2, 3, 9, 10, 11].

Software projects are exposed to various risks and risk management in such projects is still inadequate as is shown by the percentage of failed, delayed or too expensive projects [4]. The goal of a project is to deliver, in time and within the budget constraints, a product that meets stakeholders' needs and expectations. The essential factors of the project success are the quality, the time and the budget [10]. Present software projects are often facing expanding and changing client demands and are put under schedule pressure. The systems are growing in size and become increasingly complex. To shorten the development time, the systems are built out of reused (but often not reusable) components. The personnel turnover is high and the size and diversity of project groups is growing.

Risk management means that we change our attitude towards risks. A project without risk management faces serious problems only after the risks came to the surface as a material fact (the deadline is not met, the budget is overrun, the quality is poor). Then, the only thing to do is to strive to minimize the negative impacts of those facts on the project. The reaction is always expensive and time-consuming. A project with risk management aims at early identification and recognition of risks and then actively changes the course of actions to mitigate and reduce the risk. This requires open communication, forward-looking view and team involvement in the management and the knowledge base of typical problems. The lack of these exposes a project to a great risk of failure [2].

The objective of this paper is to present a concept of risk management in distributed software development projects. We have recognized effective, continuous and open communication as the prerequisite for successful risk management. Therefore, we concentrate on providing to the project stakeholders a broad and highly available communication channel through which they can communicate risk-related information. The channel has unlimited memory – it registers all incoming information much as the "black box" device memorizes all relevant data during an aircraft flight. The stored information can then be analyzed from different angles, e.g. to select and prioritize the most important risks or to analyze the project history in order to find out how risk perception developed during the project course.

In the subsequent sections, we introduce the concepts related to risk communication and point out to some techniques supporting risk identification. We also present some results of experiments performed to validate our approach and the plans for the future.

2 Risk communication and risk memory

Open and unrestricted communication facilitates the key activities related to risk management. It seems, however, that the "bandwidth" of the communication channel that is necessary to support effective realization of the basic risk management activities is not even, as is shown in Fig. 1.

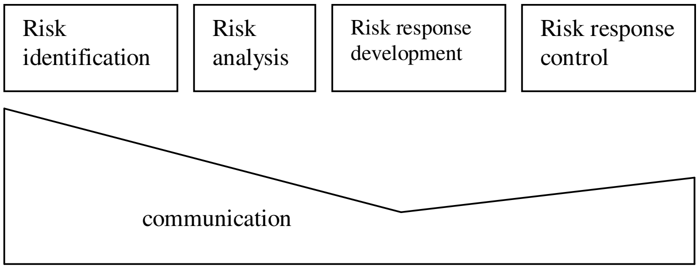


Fig. 1. Communication in risk management activities

The most broad communication channel is necessary for thorough risk identification. The channel should be open to any project stakeholder and should provide for communicating risk-related information from any relevant viewpoint. It should "absorb" information generated by using diverse identification techniques such as checklists, questionnaires, brainstorming sessions and individual observations. Moreover, it should be constantly open to protect against the risk-related information being lost (e.g. a risk has been recognized but there was no input to pass this information to, so nothing was done and the information "disappeared").

During risk analysis there is still much communication necessary as this activity is much like a consensus building process during which the parties involved communicate their views of the identified risks in order to agree on risk evaluation, priorities and possible remedying actions.

Risk response development is more related to building plans, securing resources and assigning responsibilities for handling the high priority risks. It is more on the managers' side and therefore the need for open communication is lower.

Risk response control is again the activity that heavily relays on communication, as it needs to trace the evolution of the identified risks and eventually trigger actions that are included in the risk response plans. Nevertheless, the "bandwidth" of the communication channel is not as wide as during risk identification, as here the attention is mainly focused on the already recognized risks.

From the above model, we can observe that providing a broad and highly available communication channel as early as possible is a necessary condition to successful risk identification. It strongly influences the success of all other risk management related activities. This channel should allow project stakeholders to apply diverse risk identification techniques. It should in particular allow to pass information that reflects a given person's intuitions and concerns even if this information were not obtained with the help of any defined risk identification techniques.

The conditions of successful risk identification can be summarized as follows:

- providing a constantly open communication channel,
- involvement of all relevant viewpoints,
- application of diverse identification techniques,
- effective control of the scope,
- learning from the past ("memorizing" risk related information).

The idea of having a constantly open and highly available channel for communicating and memorizing risk-related information is shown in Fig.2. As the project advances, risks can be identified either during scheduled project activities or informally, e.g. when people talk to each other at lunchtime, travel or during their leisure time. The idea of *risk black box* comes from the fact that memorizing this risk-related information should be effective and as complete as possible (much like it is done during the aircraft flight). The difference to the aircraft black box is that we want to use this information with the proactive attitude, although we do not exclude its use for retrospection (e.g. to analyze the risk history after the project success/failure).

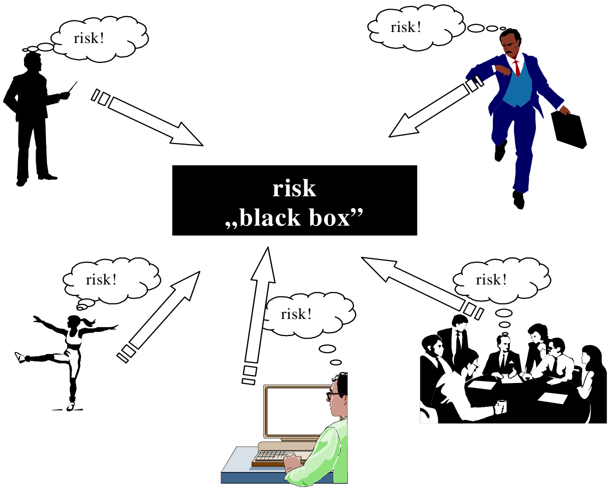


Fig. 2. Constantly open risk memorizing channel

3 Risk assessment

Our risk assessment is based on three concepts: *reviews*, *snapshots* and *reports* that underpin the three layers of processing the risk-related information: identification, analysis and reporting. Reviews establish the framework for risk identification. Snapshots pass the identified risks for further analysis. Reports communicate the results of risk assessment. The three layers are presented in Fig. 3.

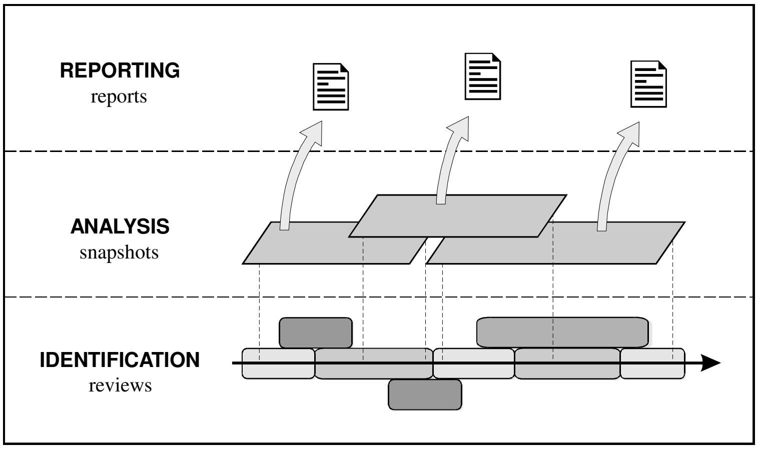


Fig. 3. Three layers of risk assessment

The risk identification layer uses *reviews* to gather risk-related information from a project. Reviews differ in terms of their scope, duration, participants and identification techniques. It is possible that two reviews overlap in time, however differing in their scope and/or participants. Risk-related information collected during a review is represented as *risk indication* and identifies a particular risk, the involved project stakeholder, timestamp, the identification technique and possible comments. The idea of risk indications is presented in Fig. 4.

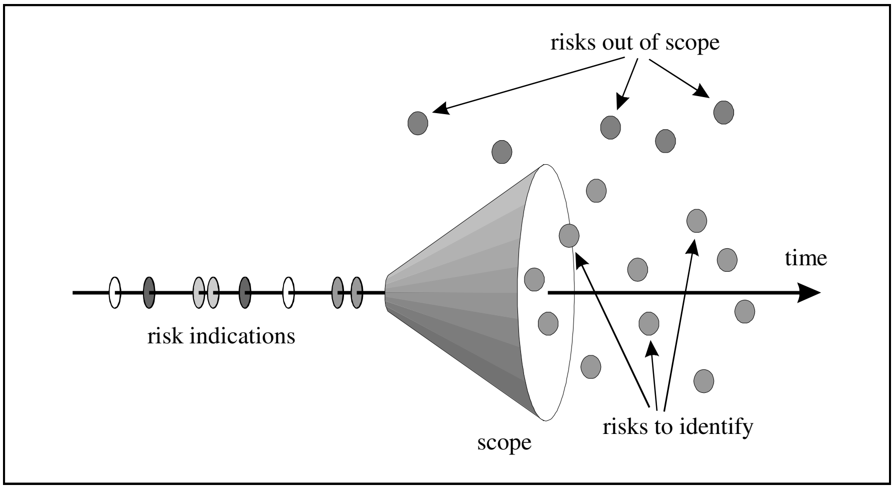


Fig. 4. The idea of risk indications

For any defined period we define *risk snapshot* as a report showing, in a predefined form, all the risks identified during this period. Thus, a snapshot is a sort of the “map of identified risks” with removed redundancies and timing information. A snapshot can be constantly “open” presenting to the risk manager how the situation (in terms of risk indications) changes during an active review. This provides a deeper insight into the risk-related information collected during the review and may be used to decide on closing the review and passing to the assessment phase. The idea of risk snapshot is illustrated in Fig.5.

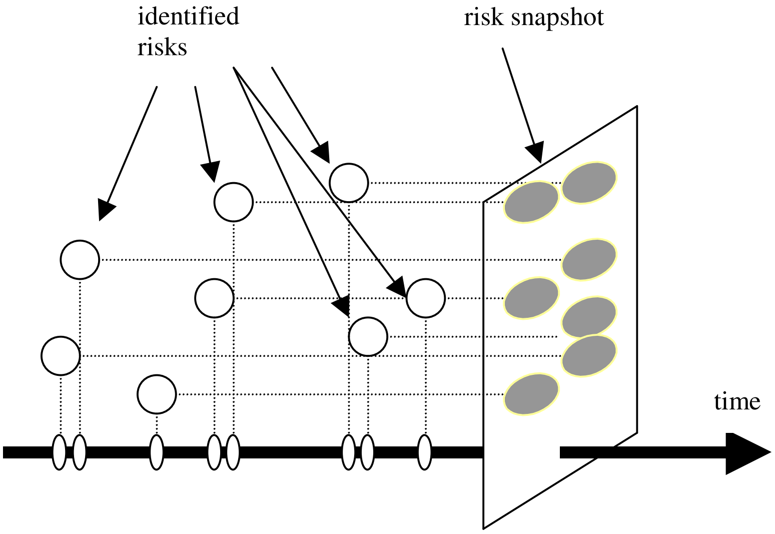


Fig. 5. The idea of risk snapshot

Risk snapshots form the input to the risk analysis. After the analysis, the risk assessment report is generated. The report is a sort of “risk summary” of the present view at risks. It can then be used as input for risk mitigation related activities. It may also be taken as an input to the next risk review action. The cycle of risk identification and analysis is shown in Fig. 6.

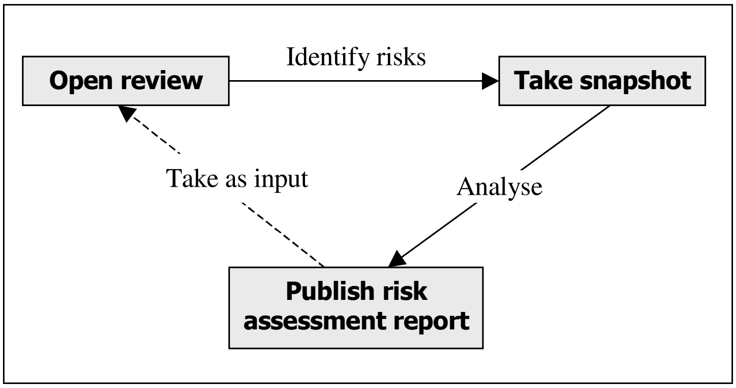


Fig. 6. The risk identification and analysis cycle

4 The process

Opening a channel to communicate and memorize risk related information is not enough, as it does not guarantee that anything is actually communicated and memorized. Causing that the information is actually generated is the manager's task. We assume that there is a risk identification and analysis process *performed* by the project stakeholders and *controlled* by the risk manager (the role usually played by the project manager except large projects where it could be assigned separately). The process is structured as a sequence of *reviews* as is shown in Fig.7.

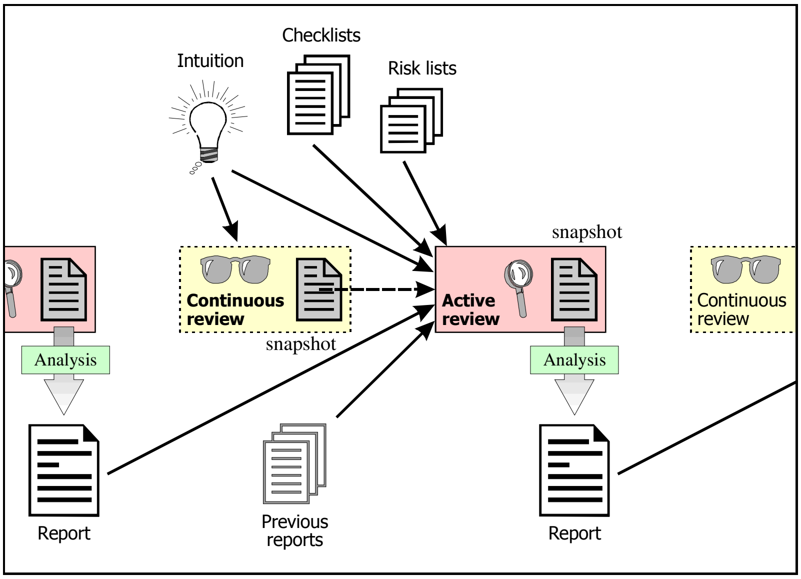


Fig. 7. Review-based risk identification and analysis process

It is assumed that at any time some review is *open*. The review remains open over its *time window*. Time windows of subsequent reviews are adjacent. We distinguish between two types of reviews:

- *active review* – its starting and ending times are set by the risk manager as well as its scope and participants (the stakeholders involved in the review). The review has a defined set of inputs (reports, checklists, questionnaires, etc.) and associated risk identification techniques. As a rule, the *snapshot* from the last *continuous* review, is included as an input of the active review. The active review ends with the risk analysis session that aims at assessing and prioritizing the identified risks and produces a relevant report.
- *continuous review* – it starts with the end of the previous review and ends with the start of the next review (being it active or continuous). It just keeps the communication channel open enabling the communicated risk information being memorized.

The set of its input documents is not controlled by the risk manager. Any project stakeholder can pass risk-related information disregarding the way of its generation.

Typically, a snapshot is taken at the end of the continuous review to provide an input to the subsequent active review. A snapshot is also taken at the end of an active review to summarize the effects of risk identification activities (as shown in Fig.7.). The risk assessment report is generated at the end of an active review.

We assume that the process has the active and continuous reviews interleaved, their extent (in time) and scope (in terms of inputs and participants) being controlled by the risk manager. This way we achieve the following benefits:

- the communication channel is constantly open,
- the identification actions are being planned (active and continuous reviews),
- all communicated risk-related information is being memorized,
- the identified risks are periodically reviewed and assessed and the frequency and scope of those assessments is under control of the risk manager,
- the results of the analyses are kept in the form of reports and are available downstream of the process (can support further identification and analysis)

5 Representation of risks

The risk management process must be supported by adequately defined data structures maintaining the risk-related information generated and used in this process. The proposed representation is presented in Fig. 8 in a form of class diagram using UML notation.

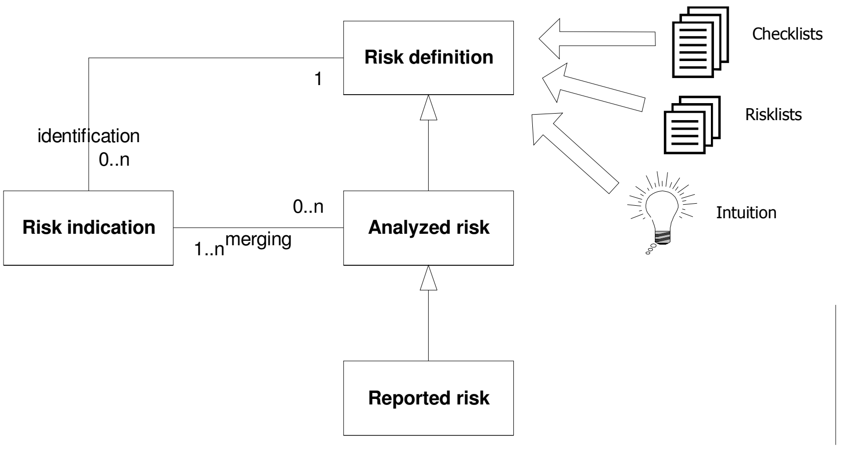


Fig. 8. Representation of risks

First, the risk itself must be defined giving the description of the undesired event that may possibly occur. The *risk definition* can be a simple statement expressed in a natural language, a formal expression in a certain notation or a scenario showing how we can get to the undesired state/event. In our present process, risk definitions primarily come from checklists and lists of risks already built and published, but we also recognize the necessity and value of defining new situation-specific risks that are recognized by project stakeholders using intuition and engineering judgement.

Risk indications point to a risk definition and declare that this particular risk is being present in the project. Many stakeholders may indicate the same risk as well as it may be reported many times by the same stakeholder (e.g. in different reviews).

To process the risk-related information we distinguish two types of risks, namely *analyzed risk* and *reported risk* that inherit a risk definition to point to the risk being analyzed or reported. The indicated risks are mapped to the analyzed risks using the concept of the snapshot as presented earlier in the paper. An analyzed risk comprises additional information like: priority, evaluation of its likelihood, certainty etc. It is important to provide for backward and forward traceability between the various risk representations to provide for visibility of the analyses and retrospection. The most important analyzed risks are selected for publication in the risk assessment report. A reported risk is an analyzed risk extended with a risk response related information, e.g. contingency plans.

6 Tool support: the Risk Guide system

To experiment with the concepts proposed above, we have elaborated and implemented a risk management tool named □Risk Guide. It is an Internet application and can be accessed simply by a web browser [12]. It makes it applicable in distributed software projects. It supports risk reviews, indications, snapshots and reports to publish assessment results. Multiple project members can post risk indications simultaneously and those are then automatically lined up in the risk repository. The tool can support multiple projects at a time with independent risk identification and assessment processes.

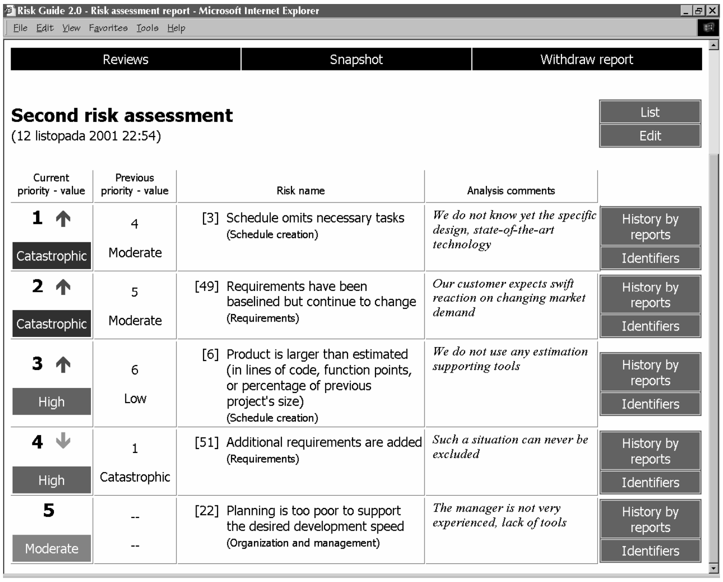
The system offers a knowledge base of checklists and the lists of common risks e.g. Taxonomy-Based Questionnaire [9] or Complete List of Schedule Risks [5, 6]. Three techniques to identify risks are presently supported:

- automatic generation of risk indications based on the answers to a questionnaire,
- explicit selection of a risk from a list of risks,
- supplying a new definition of a specific risk identified by intuition and/or engineering judgement.

The tool supports management of checklists and lists of risks as well as provides for evolution of risk definitions (by means of versioning).

Risk manager can take a snapshot of identified risks at anytime. As soon as the snapshot is taken, all open reviews are closed and further identification requires opening a new review. The analysis of risks is carried out in two phases: evaluation of each risk and assignment of priorities to risks. The risks can be evaluated in three dimensions: possibility, severity and timeframe. For each dimension □Risk Guide offers a qualitative evaluation scale. An overall risk evaluation is calculated from individual assessments using the risk evaluation matrix. In addition, a comment can be added to the evaluation to justify it or to express its certainty. The priorities can be assigned automatically according to the evaluation or based on manager's decision. The list of risks analyzed in a snapshot is ordered, so the most important risks are available on top of the list.

Once the analysis is completed, the resulting list of the most important risks can be published in a risk assessment report. An example of a risk assessment report is presented in Fig. 9.


Risk Guide 2.0 - Risk assessment report - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Reviews
Snapshot
Withdraw report
List
Edit

Second risk assessment
(12 listopada 2001 22:54)

Current priority - value	Previous priority - value	Risk name	Analysis comments	
1 ↑ Catastrophic	4 Moderate	[3] Schedule omits necessary tasks (Schedule creation)	*We do not know yet the specific design, state-of-the-art technology*	History by reports Identifiers
2 ↑ Catastrophic	5 Moderate	[49] Requirements have been baselined but continue to change (Requirements)	*Our customer expects swift reaction on changing market demand*	History by reports Identifiers
3 ↑ High	6 Low	[6] Product is larger than estimated (in lines of code, function points, or percentage of previous project's size) (Schedule creation)	*We do not use any estimation supporting tools*	History by reports Identifiers
4 ↓ High	1 Catastrophic	[51] Additional requirements are added (Requirements)	*Such a situation can never be excluded*	History by reports Identifiers
5 Moderate	-- --	[22] Planning is too poor to support the desired development speed (Organization and management)	*The manager is not very experienced, lack of tools*	History by reports Identifiers

Fig. 9. Example of a risk assessment report

In addition the tool offers various options supporting risk tracking and risk history analysis and includes adequate access control mechanisms.

7 The experiments

On our way towards an integrated environment for risk management we plan and carry out experiments to test its concepts. As for November 2001, one experiment is already finished, another was just started and another is being planned.

The first experiment aimed at evaluation of the ease of use of □Risk Guide and assessing the effectiveness of its support in a single risk assessment cycle (according to Fig. 6). It was carried out in the academic year 2000/2001 during the Software Engineering Project Management course at the Technical University of Gdansk. □Risk Guide was used as a supporting tool at some stage in the student project associated with the course. Students, working in teams of 3-4 members, practiced selected project management activities such as effort estimation and project planning with respect to, in most cases imaginary, projects (6 of those projects were real and related to other courses). The size of the projects taken under consideration ranged from 3 man-months to 40 man-years. The risks were considered while building the detailed project plan and developing risk mitigation plans. In total, 38 groups took part in the experiment.

The experiment was then evaluated using a questionnaire that was distributed among the students to gather their opinions. According to the answers the experiment resulted in (more details can be found in [7]):

- better understanding of vulnerable project areas,
- bringing to the surface many potential problems that otherwise would have been omitted,
- increasing students' interest in risk management (increased awareness),
- elaboration of more realistic detailed plans,
- effective collaboration and parallel working,
- "common memory" of potential problems that can affect the future of the project.

The second experiment launched in October 2001 aims at the evaluation of the concepts of snapshots and reports as well as assessing the effectiveness of support in the entire project course. Full recurring assessment process (as presented on Fig. 7) is being followed in the experiment. This experiment is carried out within the Post-graduated Course on Software Engineering organized at the Technical University of Gdansk. The course includes Team Project as a substantial practical component. The participants of the project are professionals coming from the local software industry. The project covers all development phases from early conceptualization to implementation. Presently, 10 students divided into 2 groups each of 5 members attend the course and take part in the Team Project. Both groups use □Risk Guide to share opinions on risk perception and to assess risks in their projects. To evaluate the experiment, we plan to gather students' opinions with questionnaires at the end of the course. The course will finish in late January 2002 and by this time, the results of the experiment are expected.

The third experiment is just being planned and involves using □Risk Guide to support risk management in a real project(s). Several software companies expressed their interest in incorporating the tool into their projects. As this is a very "last minute" arrangement it is still not certain if the results will be available at the time of the conference (June 2002).

8 Conclusions

In the paper, we have pointed out that risk management can prevent schedule and budget overruns and low-quality products of software projects. We emphasized the essential role of communication in the risk management process and proposed a concept of a risk “black box” memorizing all the risk-related information arising in the project. We distinguished three hierarchical layers of risk assessment and explained how they interact. Finally, we presented a process of continuous risk assessment taking benefit from all the above ideas.

Risk management can benefit from tools that support communication and collaboration. We described an Internet tool that can be offered to a software development team disregarding the actual geographic dislocation of team members. We also described experiments that are performed or are being planned to check the validity of our approach in the real projects.

References

1. Galagher B. P., Software Acquisition Risk Management Key Process Area (KPA) – A Guidebook Version 1.02, SEI report CMU/SEI-99-HB-001, Carnegie Mellon University, Pittsburgh PA, October 1999.
2. Higuera R. P., Gluch D. P., Dorofee A. J., Murphy R. L., Walker J. A., Williams R. C., An Introduction to Team Risk Management, SEI report CMU/SEI--94-SR-01, Carnegie Mellon University, Pittsburgh PA, May 1994.
3. Higuera R. P., Haimes Y. Y., Software Risk Management, SEI report CMU/SEI--96-TR-012, Carnegie Mellon University, Pittsburgh PA, June 1996.
4. Jones C., Assessment and Control of Software Risks, Prentice Hall, 1994.
5. McConnell S., Code Complete, Microsoft Press, 1993.
6. McConnell S., Rapid Development, Microsoft Press, 1996.
7. Miler J., Górski J., Implementing risk management in software projects, Proc. of 3rd National Software Engineering Conference, Poland, 2001.
8. PMBOK Guide, 2000 Edition, Project Management Institute, 2000
9. Sisti F. J., Joseph S., Software Risk Evaluation Method, SEI report CMU/SEI-94-TR-19, Carnegie Mellon University, Pittsburgh PA, December 1994.
10. Van Scoy R. L., Software Development Risk: Opportunity, Not Problem, SEI report CMU/SEI-92-TR-30, Carnegie Mellon University, Pittsburgh PA, September 1992.
11. http://www.sei.cmu.edu/
12. http://mkzlway.eti.pg.gda.pl/riskguide