Implementing Risk Management as part of a Systems Engineering Process

Claude Y. Laporte
Yortar Technologies
481 Bissett
Saint-Jean-sur-Richelieu (Québec)
Canada
J3A 1W6
claporte@yortar.com

Guy Boucher
Oerlikon Aerospace Inc.
225, boul. du Séminaire Sud
Saint-Jean-sur-Richelieu (Québec)
Canada
J3B 8E9
gboucher@oerlikon.ca

ABSTRACT

During the last five years, Oerlikon Aerospace developed and implemented engineering processes. In this paper, we discuss the application of risk management to the re-engineering of operator console stations of a missile weapon system. We briefly describe the systems engineering process. Finally, twelve lessons learned are discussed.

PROCESS DEVELOPMENT BACKGROUND

Oerlikon Aerospace (OA) is the integrator of an air defense missile system. The system consists of a missile launcher mounted on a tracked vehicle or a fixed platform, together with radar and optical sensors, electronic control systems and communication equipment. Over 120 systems and software engineers are involved in the development and maintenance of the system.

The organization has been ISO 9001 certified since 1993. In 1997, the organization has also been assessed as CMM ({Paulk 93) level 2 by independent assessors certified by the Software Engineering Institute. In addition to satisfying level 2 goals, the organization also met 8 of the 17 level 3 goals.

In 1995, it was decided that a formal systems engineering process had to be developed and implemented in order to seamlessly integrate disciplines associated with systems engineering. The development effort was initiated by the performance of an internal assessment of our systems engineering practices. A decision was made to use, as frameworks, the Systems Engineering Capability Maturity Model (SE-CMM) and the Generic Systems Engineering Process (GSEP) developed by the Software Productivity Consortium (SPC 1995).

A systems engineering process (Laporte 1997) describes management and technical activities, roles and responsibilities, metrics and artifacts produced by each activity. The management activities of the Systems Engineering Process (SEP) major steps are summarized in table 1 while table 2 illustrates the technical activities (steps 210 through 270). The process had been applied to the re-engineering of two sub-systems: the launcher control electronics and the radar and electro-optical operator consoles (Laporte 1998). The launcher control subsystem is composed of a main data processor which coordinates the operation of the sensors and the launch and guidance of the missiles, a missile tracker processor, a target tracker processor, and a servo control processor. The operator consoles consist in a radar console to control the radar and communication subsystems, and an electro-optical console to control optical sensors and missile launcher.

THE RE-ENGINEERING OF OPERATOR CONSOLES

The reengineering of the consoles was divided into two increments: beginning in March 1997, a system definition increment of the subsystem in its new configuration; and beginning in January 1998, a detailed hardware/software development increment. The identification of each increment was based on the nature of the deliverable products at the end of the increment. In both cases, the first increment deliverable would be a system requirement specification, and the second increment deliverables would be a set of design and equipment specifications, plus a qualified working pre-production prototype.

The following paragraphs describe what was accomplished during increment one as well as what is being planned and performed for increment two. The emphasis of the paper will be put on the risk activities that have been performed so far.

Major Steps	Sub-Steps
110 Understand Context	111 Define Approach
	112 Estimate of Situation
	113 Review Context
120 Analyze Risk	121 Perform Risk Analysis
	122 Review Risk Analysis
	123 Plan Risk Aversion
	124 Commit to Strategy
130 Plan Increment Development	131 Execute Risk Aversion 132 Review Development Alternatives 133 Plan Increment Development 134 Commit to Plan
140 Track Increment Development	141 Monitor and Review Increment Development
	142 Update Increment Plan
	143 Review Technical Product
150 Perform Increment Closure	151 Baseline System Definition
	152 Assess Increment Closure
	153 Update External System Plan
	154 Commit to Proceed

Table 1. The Management Activities of the Systems Engineering Process

Major Steps	Sub-Steps
210 Analyze Needs	211 Determine Stakeholders
	212 Define Problem Domain
	213 Develop Informal Functionality
220 Define Requirements	221 Determine Behavioral Requirements
	222 Determine Performance Requirements
	223 Map Behavior to Performance
	224 Refine Requirements
230 Define Functional Architecture	231 Partition Requirements into Functions
	232 Define Lower Level Functions
	233 Define Functional Interfaces
240 Synthesize Allocated Architecture	241 Allocate Functions to Alternative Solutions
	242 Define Physical Parameters
	243 Define Physical Interfaces
	244 Integrate Design
	245 Refine Physical Architecture
250 Evaluate Alternatives	251 Assess System
	252 Perform Sensitivity Analysis
	253 Allocate Performance to Technical Parameters
	254 Assess Technical Risks and Problems
	255 Identify and Perform Trade-off
	256 Select best System Solution
260 Verify and Validate Work Products	261 Define V&V Procedures
	262 Verify System
	263 Validate System
270 Release System Definition	271 Control Technical Decision Data
	272 Control System Configuration

Table 2. The Technical Activities of the Systems Engineering Process

Overview of Increment One

Management of Requirements

The system engineering CASE Tool CORE® has been used to develop the console requirements. The database included the following type of information:

- Originating requirements (behavioral and non-behavioral).
- Interface requirements.
- Verification requirements.
- Physical architectures.
- System diagrams.

The CORE database was baselined in December 1997 after the completion of increment one.

Development of an Engineering Model

An engineering model was developed during increment one. The model ran on standard PC and its purpose was to show the new concept of operation and the proposed Man-Machine Interface (MMI). The model was formally shown to stakeholders. Comments were collected and analyzed in order to modify and improve the system requirements in a second iteration.

Technology search

A series of technologies related to either hardware or software has been researched and trade-off analyses have subsequently been documented. Many potential suppliers were met and a few employees attended real-time embedded conferences as well as VME and high tech shows.

Training

Beside the training provided on the new systems engineering process, the only formal training provided in 1997 had been on tools: VAPS, the GUI (MMI) CASE tool, and CORE, the system definition CASE tool. Training was performed in early 1998 on VxWorks operating system, Rhapsody® software development CASE tool and UML software development methodology.

Estimate Reviews

Two formal recurring and non-recurring development effort estimates were performed during increment one, the first review was performed in August 1997 and the second review occurred in December 1997.

Overview of Increment Two

The plan for increment two consisted of proceeding with both the hardware and software detail design based on the interim system definition and the engineering model generated during increment one. The detailed development will include the construction of an engineering unit to support the hardware and software development and the construction of a pre-production unit that will support system integration and qualification activities. In addition simulators will be built in parallel to support development, integration and validation effort.

The plan for increment two also included other non-recurring activities such as the production jigs, tooling and logistic activities; LSA, technical publications and training.

THE APPLICATION OF RISK MANAGEMENT ACTIVITIES

Systems Engineering Process Step 120: Analyze Risk

In step 120 of the systems engineering process, risks were analyzed, risk mitigation strategies were developed, and stakeholders commitment were made on mitigation strategies (Sub-steps 121 and 122). The process describes what risk management activities should be performed, but it does not prescribe any particular method. The members of the project were aware of the method used by software engineers since a method was described in the project planning and tracking activities of the company software engineering process. After a brief discussion, the team decided to use the method proposed by USAF (USAF 1988). At the beginning of the project, it was felt that this step looked liked a paper exercise and was not very useful. It was, in fact, the first development project to proceed with a formal way to handle risks.

Since the project had been divided in two increments, it was decided to focus risk analysis to the issues most pertinent to increment one. Note that the risk management plans for both increments are similar in terms of how the risks are managed, but the risks themselves are very different in each increment. For example, there are no high risk issues regarding production during the concept definition phase (part of increment one).

A Risk Management Plan (RMP) was developed. The RMP had two main sections. The first section described the program overview and defined terms such as:

- Type of risk (cost, program, schedule, supportability, technical)
- Assessment of risk impact (catastrophic, critical, marginal, negligeable)
- Overall categorization of risk (high, moderate, low)

The RMP specified who was responsible for the risk management and how the risks were to be managed during the increment. This section was quite generic, it could be reused by other projects.

The second section of the RMP was really specific to the project. It was mainly composed of a single matrix that list all of the identified risks. The risk identification process was performed through brainstorming sessions with both the development team members and stakeholders. Along with the list of risks, in the same matrix, were the following elements of information:

- Type of risk (i.e. cost, schedule, program, technical)
- Probability of occurrence (i.e. very low, low, medium, high, very high)
- Impact (i.e. negligible, marginal, critical, catastrophic and cost)
- Overall risk (i.e. low, medium , high and cost)
- Identification of impact on other projects
- Brief resolution plan
- Drop-dead date
- Person(s) responsible (e.g. member of the project team, functional manager, project manager, director of engineering)
- Hours or resources required to perform the project
- Resolution Status (i.e. open, close)

The Implementation of the Risk Management Plan

The actions and status of the risks were then reviewed on a weekly basis during the project reviews. When a mitigation plan was required, e.g. special resources and a considerable amount of hours, then specific risk activities were directly integrated in the detailed Work Breakdown Structure (WBS) and scheduled like any other major development items.

Some of the risks identified were: project risks such as budget overrun, schedule delays mostly due to lack of dedicated resources, and technical risks such as the lack of experienced personnel in using a new process and a new CASE tool (CORE®). Also, since this project was performed concurrently with another project, it was necessary to closely monitor integration, validation and verification activities, and interfaces definition with the rest of the missile system. Finally specific risks like availability of COTS hardware, mastering of new technologies such as VME, development of new custom circuit card assembly (CCA) and development of new communication bus were also identified.

Risk impacts were represented by a weighted probability of occurrence and consequence index. This risk matrix was stored in a database and was continuously updated during the two increments.

In some cases, the same mitigation strategy addressed several risks. Mitigation strategies included activities such as pilot projects, engineering models and mock-ups, additional analyses and subsystem modeling. Specific participant training was also planned in some areas. Finally, a formal review with stakeholders helped to identify other risks, gather mitigation suggestions, and obtain final commitment (Sub-step 124).

LESSONS LEARNED

Quantification of Risks Issues

During increment one, risk only had a "qualitative" score i.e.: high, medium or low. We found that this had two major drawbacks compared to quantitative evaluations:

- It did not have the same weight or necessary attention from management
- No money/resources were set aside should the risk issue occurred. This could lead to budget overruns.

Evaluation of Risks in a Systemic Perspective

For increment two, we quantified and costed all risks, even the ones that the team had not the control over such as hiring or allocating budgets and expenses. The fact and the matter is that the development team would be ultimately impacted should the risk occur. As a result, the company decided to put money aside for risks in the budget for increment two.

Risk Management was not Free but was a Wise Investment

We quickly found out that some risks required relatively a lot of effort to mitigate. One example was the activities related to the engineering model in increment one above. It was decided to proceed with an engineering model to mitigate a risk previously identified that related to the fact that we had no customer requirement. The fear was then that we would proceed with a design that would not meet any potential customer wishes.

Approximately 800 hours were spent to model a new concept of operation and Man Machine Interface. This included activities such as design of model and, even more important, validation of the concepts with selected group of operators from inside and outside the company.

This model allowed us to develop and refine the system requirements as well as define software use cases with a very high confidence level that they would remain stable throughout the entire design chain. Although it was difficult to precisely assess the amount of time/money that has been and will be ultimately saved, one can imagine what would be the cost of delivering a product that would not meet customer expectations.

Another example is a pilot project performed in increment two. This pilot project came as a result of a risk identified that expressed the concerns that we would enter the software design phase with a new methodology, new CASE tools (design and GUI) and new development environment. All ingredients that can lead to failure. About a thousand hours were spent on a mini-project that had the main objectives to verify the capabilities of the tools, to verify the integration of the tools, and to propose a design method. The answer to all these questions was crucial to a generation of a proper Software Development Plan (SDP) that needs to clearly show, organize and plan the work of a group of more than 20 persons for a 24-month timeframe. The pilot project represented about 1.5% of the total software design effort, but it was sure worth since it ensured that the remaining 98.5% of the project would be done properly and correctly.

Pilot Projects as a Risk Mitigation Strategy

It was very important to carefully select pilot projects and their participants since these projects would foster adoption of new practices throughout the organization. Also, first time users of a new process would make mistakes; it was therefore mandatory to properly coach the participants. If participants sensed that mistakes would be used to learn and make improvements to the process instead of "pointing fingers", the level of anxiety was reduced. This also led the individuals to bring forward suggestions instead of "hiding" mistakes. Most of the participants for both projects were therefore selected within the working group who developed the SEP. Other participants were given a two-day training session on the SEP.

Response of Management to Risks

Dealing with formal risk management represented a mentality change not only for the project team but also for the entire organization. However, when risk management activities were done properly by the development team, management was more prone to agree and support the risk activities that resulted from the analysis of the risks.

Risk Mitigation lead to design decisions and development strategies

The results of the risk mitigation activities related to technical risks will necessarily lead, or as a minimum be an input, to design decisions and will provide direction for follow on activities. In fact, whether a mitigation plan was the generation of an analysis, conduction of a test or construction of a physical or behavioral model, the result will be the confirmation of a hypothesis or the identification of the best design alternative. This ultimately lead to design decisions and subsequent development strategies.

Training as a Risk Management Issue

One important aspect of risk management was training. Previously, most plans showed a nice flow down of activities with associated efforts as it should be. However, these plan also reflected the fact that they were all conducted by high skilled personnel that knew exactly what to do at all times. This obviously did not represent reality. Therefore, appropriate training became mandatory to manage the risks and training activities were built in the project plan.

Dividing a Project in Increments as a Risk Management Strategy

The project increments must be carefully defined so that they remain of manageable size. Their associated activities not too long to be properly tracked, on the other end not too small, so their activities require micro-management. Project manager experience was found a critical asset for project and increment definition. A manageable increment size was also critical for the proper performance of design reviews, in that participants to the review kept focus on the increment scope.

Risks about the Implementation of a New Process

It was found that for some areas of the systems engineering process, specific deliverables were difficult to determine precisely. This situation happened because the end-products (i.e. project documents) grew iteratively as process steps were performed. It was therefore difficult to closely measure the progress of the activities and report progress to management.

Risk about the People Issues

Managing the human dimension of the project was found to be an element which not only fostered the adoption of the new process but also created an environment where changes were introduced at an increasingly greater rate. Members of the engineering organization realized that managing the "soft stuff" was as important as managing the "hard stuff".

Risk Management Activities were Planned and Included in the Project Plan

Since a substantial amount of energy was expanded in risk management activities, those activities were identified, estimated and incorporated in the project plan.

Appointment of a Risk Officer in a Project

When a project is composed of many projects similar to the one described in this paper, all risk activities may represent a substantial effort. Also, the risks have to be analyzed at the project level since risks in one sub-project may create risks at the project level. Risks from different sub-projects

may be analyzed and mitigated at the project level instead of being mitigated individually. It was found that all project risk activities were better managed by one individual. A project role called risk officer had been established. The risk officer hat was allocated, as a secondary duty, to a member of the team who was interested by this role and had a lower load in the project.

CONCLUSION

A new systems engineering process involving the management of risks had been deployed and used in the redesign of a missile system operator console. The risk management activities were found very useful to plan activities and collect technical and managerial information more formally in the course of the projects. It did also help to manage and improve the dynamic human dimension of the development project.

REFERENCES

Forsberg, K., Mooz, H.., "Application of the 'Vee' to Incremental and Evolutionary Development", Proceedings of the Symposium of the International Council on Systems Engineering, St.Louis, MO, July 1995.

Laporte, C.Y., Guay, A., Tousignant, J., "The Application of a Systems Engineering Process to the Re-Engineering of an Air Defense System", Proceedings of the Eight Annual International Symposium of the INCOSE, July 26-30, 1998, Vancouver, British Columbia, Canada.

Laporte, C.Y., Papiccio, N.R., " Development and Integration of Engineering Processes at Oerlikon Aerospace", Proceedings of the Seventh International Symposium of the INCOSE, August 3-7, 1997, Los Angeles, California.

Paulk, M. et al, 1993. "Capability Maturity Model for Software", Software Engineering Institute, SEI/CMU-93-TR-24.

SPC, "A Tailorable Process for Systems Engineering", Software Productivity Consortium, SPC-94095-CMC, January 1995.

USAF, "USAF's Software Risk Abatement Handbook", AFSC/AFLC Pamphlet 800-45, September 30 1988.

BIOGRAPHY

Claude Y. Laporte obtained a Bachelor in Science from le Collège Militaire Royal de Saint-Jean in 1973. In 1980, he obtained an MS in physics at Université de Montréal, and in 1986, an MS in Applied Sciences from the Department of Electrical and Computer Engineering at École Polytechnique de Montréal. He was an officer within the Canadian Armed Forces during 25 years and a professor for over 10 years. He left the Canadian Forces in 1992 at the rank of major. He joined Oerlikon Aerospace where he coordinated the development and implementation of processes, methods and tools. He left Oerlikon Aerospace in 1999 to launch consultation services as a Partner of Yortar Technologies.

Guy Boucher obtained a Bachelor in Electrical Engineering from the Royal Military College of Canada in 1982. He then served the Canadian Forces for a period of five years as a radar engineer on a long range radar station and a Canadian representative on a joint development program of the US Over-the-Horizon-Backscatter Radar in Bangor, Maine. He left the Forces in 1987 at the rank of Captain. Since then, he has joined Oerlikon Aerospace as first Radar Principal Engineer and subsequently Project Engineer.