MANAGEMENT SYSTEM TAILORED TO RESEARCH INSTITUTES

I. Verstovsek*, J. Kamenik, P. Kolaric, Cosylab, Ljubljana, Slovenia

Abstract

As all disciplines, project management has a set of rules that must be followed and set of recommendations that make work easier. But as in all engineering, there is no single magical formula or equation, no matter how much managers and physicists alike would love to have it. We present a working solution tailored to academic projects that requires only a minimum of effort and discipline and results in huge benefits, which will be presented in this article. Commercially available project management tools are not suited to manage the diversity of work in research institutes. We have therefore adopted a set of open source tools, implemented some custom additions and integrated the tools into a coherent product to suit our purpose. It enables developers to track their work and communicate effectively; project managers to monitor progress of individual projects; and management to supervise critical parameters of the company at any time. In the article, the experiences gained by using the system are presented. As it has turned out in practice, the product is ideal also for research institutes, as it is demonstrated by its use in control groups of DESY and ANKA.

INTRODUCTION

We can look at project management from two basic perspectives: *bottom-up* and *top-down*. In the bottom up perspective, the main emphasis is on organizing, tracking and classifying all activities within an organization.

In the top-down approach, we start from the highest possible level (this could be an accelerator control group or the whole institute) and try to define and organize the highest level activities in the organization. These activities are usually complex projects, themselves made of subprojects.

The challenge for all members in the organization: developers, project managers and organization leaders is to find the right balance between day to day tracking of activities and the requested level of oversight. Research institutes have their own specific requirements for project management: day to day tracking of activities is not as desirable as in private companies; a large proportion of work is dedicated to research activities, which do not have a well specifiable outcome at the beginning, etc.

Generic project management solutions are not tailored to the needs of research institutes and tend to be quite rigid in their design. If you add to this the cultural resistance to any "big brother" software; you soon have a large problem introducing off-the-shelf project management software into research organizations.

The solution presented in this article has evolved over 6 years within Cosylab, first a research group within Jozef Stefan Institute (Slovenian largest institute) and later a privately owned company, retaining the spirit of work in the research institutes.

*igor.verstovsek@cosylab.com

PROJECT MANAGEMENT CHALLENGES IN INSTITUTES

In a research environment the most crucial step of a project is not the actual implementation - scientist are very good a doing things: thinking, coding, debugging, etc. and usually they perceive this as fun. The critical parts of projects are the things around the fun stuff, in particular:

- **conception** (evaluating the idea before jumping into implementation)
- **planning** (how to get things done in terms of time, resources)
- **closing** (to finish all work on a project without a need for constant support)

Conception

The main pitfalls here are that:

- projects are started too easily (without basic thinking about purpose and scope),
- too many projects in parallel: focus of work is lost.

What one can do about it is:

- **a quick sanity check**: why start something? Think about the big picture as well. How will this help the work on other projects of the institute, etc.
- Prepare a **well rounded proposal**. By putting things in writing motivates you to think also about the not-so-fun stuff.
- Use **formal decision process** to start a project. This does not have to be a meeting with all the managers of an institute. Presenting the proposal in a well prepared presentation to your group colleagues can do the trick as well. During the discussion a decision whether the project is worth a shot or not will most probably spring up by itself.

Planning

The pitfalls of not doing planning are:

- people get over allocated and ineffective,
- there is no reference to track progress,
- there is no “satisfaction of a job well done”.

What to do about it:

- prepare **at least the initial project plan.** Even if the plan is never updated later, there is at least one reference point to relate to.
- Make a **risk plan**.
- **Communicate the plan** and get commitment. This very important and often neglected: the plan will hard to carry out if your team does not about or if the plan is not accepted.

Closing

This is the "sugar on top" that should come after the work on a project is coming to an end. Closing the project formally prevents the never-ending work and allows the whole team to learn from past experience on the project. You should at least take care of two things:

- **Acceptance process.** Acceptance should be made for internal projects as well - these projects are the most prone to an eternal life.
- Close project, **plan resources for support**. Important here is that the resources for support are well defined and are planned for a limited period of time.

Active Projects (39) (2296 man-days)

Queue	Name	Reports	Master Ticket	Project Manager
BNL-ID_Control_Study	BNL: IDControlStudy for NSLS II	Brief Long	35968	Mha Pelko
CERN-MiddlewareAnal	CERN: First contract	Brief Long	36159	Matej Sekoranja
DESY-Applications2	DESY-Applications2	Brief Long	33444	Igor Kriznar
DESY-CSE	Control System Engine w/ CAS transport implementation	Brief Long	36140	Matej Sekoranja
DESY-Matlab-Study	DESY: Matlab Study	Brief Long	36619	Igor Kriznar
DESY-SmallProjects-2007	Small projects with DESY in 2007	Brief Long	32295	Klemen Zagar
DESY-VDCT-CSS	VDCT-CSS integration design	Brief Long	27367	Matej Sekoranja
DLS-Support-Q12007	Additional CAJ Support for DLS in Q1 2007	Brief Long	30118	Matej Sekoranja
EMBL-TineTango-Gateway	EMBL: Completition of TINE2TANGO Gateway	Brief Long	36496	Rok Stefanic
ESO-ACS_7	ESO: ACS 7.0	Brief Long	36175	Matej Sekoranja
ESO-ALMA-Jul-Oct-06	ACS July-October 2006	Brief Long	25660	Klemen Zagar
FNAL-DAL	FNAL: DAL testing consultancy	Brief Long	36385	Jaka Bobnar
FZK-ANKA-2007	FZK-ANKA Support 2007	Brief Long	32274	Igor Kriznar
FZK-ANKA-BPM-BLM-upgrade	ANKA: BPM and BLM upgrade	Brief Long	34346	Primoz Kolanic

Figure 1: High level overview of projects, serves as an entry point for all project reports.

COSY PROJECT MANAGER

Tickets as Basic Units of Work

Any project is composed of a number of more-or-less well defined units of work, assigned to developers, usually by the project managers. There is a multitude of tools that manage "work units", many of them freely downloadable from the Internet. We chose the Request Tracker (RT) [1] because of one simple reason – this tool can manage e-mails really well – not only sending but also receiving. RT was picked as our main tool and we adapted all the other ones to it. Every now and then, when experiencing problems with RT's code written in Perl, a question appears why we did not rather write such tool by ourselves, but the final statement remains that RT is really well structured and extremely useful.

RT is (as described in its manual) an "enterprise-grade ticketing system which enables a group of people to intelligently and efficiently manage tasks, issues, and requests submitted by a community of users".

Its main unit is a ticket, which represents a specific task to be done. It has several fields:

- status (new, open, resolved, stalled, dead),
- estimated time,
- time spent (increases whenever a user reports a transaction),
- dates: start date, due date,
- priority,
- keywords (severity of a bug, type of ticket: QA ticket, Master ticket for the project, etc.), and
- links - how does this ticket relate to other tickets (parent - child, "depends on" and "refers to" relationships are supported).

Each ticket can have one or more parents, children or brothers. Setting also dependencies, a clear structure can be made. RT can warn us by email of a creation or modification of some ticket. The best feature of RT is the possibility of managing tickets via e-mail – every project has its e-mail address to which we can send a request for creation, correspondence or comment and set just about every field of the ticket.

RT has easy-to-use search functionality, which allows one to quickly find a ticket. With all its features RT can be used for handling support requests, ordinary tasks and bug tracking. We create about 15.000 tickets per year in Cosylab.

1.4. Contract tasks

Ticket	Task name	Spent / total estimated / contract task time (days)	weeks	spent / contract	total estimated / contract	spent / total estimated	Responsible	Status	Activity
13688	Quality Assurance Ticket	0.00 / 0.00 / 1.10	0.00 / 0.00 / 0.22	0%	0%	0%	Jansa Gasper	new	View
14438	Customization work for EPICS	1.61 / 3.12 / 5.00	0.32 / 0.62 / 1.00	32%	62%	51%	Jansa Gasper	new	View
13689	EPICS MPS device driver modifications	1.39 / 2.75 / 2.00	0.28 / 0.55 / 0.40	69%	137%	50%	Jansa Gasper	new	View
13722	microIOC development software	4.46 / 7.09 / 0.00	0.89 / 1.42 / 0.00	Inf%	Inf%	62%	Jansa Gasper	new	View
13687	Project Management	1.76 / 3.17 / 1.56	0.35 / 0.63 / 0.31	112%	202%	55%	Jansa Gasper	new	View

Completed 0 out of 5 items.

Create a new task.

Figure 2: Basic project data, time progress and status of individual tasks are shown. Using this report, one is able to easily determine the status the project and the progress of each individual contract task.

Project is More Than a Collection of Tickets

RT and its web interface, however extremely useful, do not have everything one would wish for when managing and monitoring a project in progress, especially if this project is larger than a few man-weeks of work. The central entity of RT's web interface is a ticket, and just by browsing through different tickets, one can easily get lost by looking at the trees, and not seeing the entire forest.

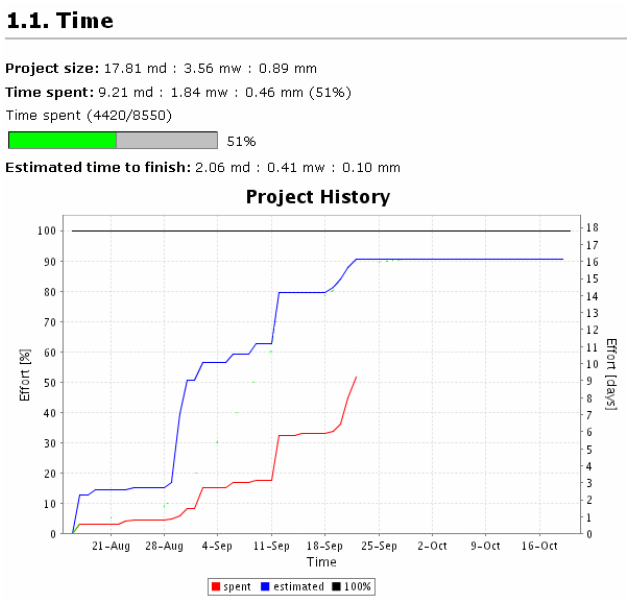

Figure 3: Real time tracking of project progress. Black line denotes project budget, blue estimated utilization of resources and red actual time spent on the project.

In addition, a project is much more than a collection of tickets, and the ability to easily manage projects that are run on the basis of a ticketing system is where we have not found an easy open source solution and decided to implement one ourselves.

In order to introduce the smallest possible overhead and use all the data already available in the RT database, we have added project management functionality to RT database itself by introducing a small number of special tickets (defined via existing RT keywords) and by adding additional fields to RT tickets. This has provided the basis for transformation of a ticketing system into a powerful project management tool. The additional information that defines a project is:

- project group (developers and the project managers),
- budget (in terms of work hours or / and money),
- deadlines and milestones,
- communication (team meetings, meetings with the customer or The Big Boss™),
- deliverables (SW and HW releases, etc.),
- quality assurance, and
- project management.

Project report tool is written in Java and generated entirely from the data in the RT database. Project report is a HTML generated from CosyDoc XML [2] that we use for writing documentation. The XML is transformed by means of a XSLT transformation and ANT scripts into HTML and PDF format. Figures 2 and 3 show some sections of this report.

SUMMARY

Cosy Project Manager is orientated more into the direction of flexibility, but it still gives enough order that it is possible to spot possible problems early, which is one of the goals of project management.

Each organisation must choose its place on this balance, based the nature of its work. Requirements of Cosylab and research institutes are very similar - we all like to do research, but we must still meet deadlines. Therefore we believe that our solution is ideal also for academic projects.

It requires only a minimum of effort and discipline and results in huge benefits, presented in this article.

REFERENCES

[1] Request Tracker, http://www.bestpractical.com/rt/
[2] I. Verstovšek *et al.*, Cosylab Management System, PCaPAC 2005, Hayama, Japan.
[3] G. Milcinski *et al.*, e-Management and Quality Assurance, ICALEPCS 2003, Gyeongju, Korea.
[4] G. Milcinski *et al.*, Developing a Control System from a Divan Bed, PCaPAC 2002, Frascati, Italy.
[5] Gantt Project, http://ganttproject.sourceforge.net/
[6] Successful Project Management, Trevor L. Young, Konan Page Limited, 2000.