The Sheffield University Maxi Project
The Industrial Project Manager's Perspective

Stan Price, BSc[Eng.]; C.Eng.; M.R.Ae.S; M.B.C.S.
Price Project Services Ltd, 23 New Mount St., Manchester M4 4DE

Abstract

The Sheffield University Computer Science Department Maxi Project for MSc Students is managed by an experienced industrial IT project manager. This paper describes how he does this and the thinking behind it and the way it has evolved.

1 Introduction

Sheffield University, Department of Computer Science conceived the idea for the Maxi Project for it's MSc students in 1988. The concept involved the use of an industrially experienced Information Technology [IT] project manager [1] to manage teams of students as if they were his or her staff working in an industrial environment. I have performed the role since it's conception on my own, apart from two years when student numbers were such that a second industrial project manager was required.

I have therefore played a significant role in deciding how the Maxi Project operates. This modus operandi has been continuously developed since it's conception. New ideas have been introduced, evaluated and if found not to be sufficiently beneficial they have been dropped. This paper does not therefore just describe how the Maxi Project is currently operated, but also describes it's development including features that have been tried and rejected and the reasons why.

The Maxi is a mandatory module for most of the Department of Computer Science MScs. Over the years these MScs have addressed topics including software engineering, advanced software engineering, software systems technology, artificial intelligence and expert systems and, in recent years, telematics.

2 The Author

The Author has been involved in IT for thirty-two years. After an initial three years as first a programmer and then a systems analyst in an engineering environment I started to manage small to medium sized projects. After a period of some six years I became one of the first in the United Kingdom [UK] to have the job title Software Engineer. Despite this technical title, I was the project officer instrumental in procuring the UK's en-route air traffic control system software from the United States [US] Federal Aviation Agency [FAA]. This software represented some 3,000 man-years of development activity. My ongoing relevant work included a similar role on the Metropolitan Police Command and Control System Project.

After a period involved in the sales and marketing of a variety of bespoke IT systems I became an independent consultant. In this capacity I have been involved in IT projects in a number of ways viz:-

-controlling a number of IT projects,
-monitoring a number of IT research projects,
-acting as an expert witness in litigation subsequent to the failure of IT projects.

I have also conceived and taught IT project management courses for a number of organisations in both higher education and commercial training. Some of my previous ideas are contained in a book [2]. However since it was written in 1984 my ideas have developed and been refined.

3 Education Value

Despite the above training experience I would not describe myself as a professional educator. The educational value of the Maxi Project is in the student's learning by doing. However, a project manager has to, at times, act as a coach and in this capacity I illustrate concepts by anecdotes from my experience. I have the conceit to believe that these may have some educational value.

Apart from this, it is for others, especially my employer, Sheffield University Department of Computer Science, and my students, to comment on the educational value of what I do.

4 Philosophy

The way I manage the Maxi Project is partly dictated by what the University requires and the constraints specific to the academic environment as well as the normal project constraints. However subject to that, it embodies a number of my beliefs about IT project management. Incidentally, most of these are applicable to project management generally. The principal relevant beliefs are:-

- Project Management is not a mechanistic task which consists simply of a number of administrative activities. It involves activities such as creating a culture, team building and staff motivation [principally by carrots but occasionally and if necessary by sticks].

- An IT project involves a number of creative activities which cannot always be scheduled and put into a strict chronological order. Therefore, although a framework for any particular project is required this must allow a high degree of flexibility so that creativity can flourish. It is interesting that the World's largest software producer, Microsoft, attempts to achieve the same balance [3].

- Any project is a social activity and it is important to take measures against anything that would impact the social cohesion of the project team and so threaten the attainment of the project's objectives. Note however, a little creative tension is not necessarily a bad thing.

- It is impossible to foresee everything that must be done at the start of the project. Even if one did not forget anything unforeseeable threats and opportunities will arise during the course of a project. It is therefore important that the project manager aided by his or her team has formal and informal processes for the timely detection of events that require actions and then their timely enactment.

- The informing of the project team of the actions that they will have to carry out and the subsequent requests to do so should be done in such a way that the project team is challenged and not threatened to a consistent level throughout the project.

- Progress should be measured against both time and resource expenditure. Also work should be constantly checked to see that either or both are not being achieved at the expense of quality.

5 Planning the Maxi

The Maxi for the following academic year is planned in August/September. Every year the Maxi has run from the start of the first term or semester to the end of the second term or semester.

The activities that require face-to-face contact between myself and the students are scheduled for one afternoon per week throughout this period. Not all of these afternoons are in fact allocated. There is a gap between semesters and some weeks it is more sensible to let the students get on with their work. This must however be balanced against the importance of regular meetings to ensure a satisfactory rate of progress is achieved.
The first afternoon is primarily occupied by my delivery, for some 2 to 3 hours, of background lectures plus the project's overall briefing which is also given by me. All other afternoons are divided between half hour lectures, termed briefings, and tutorials with each student team. The briefings are intended for and attended by all the students i.e. all the teams and are the forum for my dissemination of information which is common to all students and all student teams. These include briefings on what is required in each stage of the Maxi, arrangements for the semester break and the final afternoon of the Maxi etc.

The tutorials last 15 to 20 minutes and are held with each team. Given that normally 12 teams have to be seen within a single afternoon I ensure that I am completely prepared and the students are "encouraged" to be the same. The tutorials as well as including normal discussions of their progress and identifying problems, also incorporate such activities as Fagin's Inspections, End of Stage Reviews. Towards the end of the project, when there is actually work to be seen, these tutorials are held in the computer science laboratories.

Apart from my face-to-face work with students significant time has to be spent in my own Manchester office planning, administrating and marking. These are taken account of but not included in the Maxi planning.

Despite my reservations about how realistic it is to a real project, the Maxi is based on a waterfall model embodying five stages. These stages are typically Feasibility, Requirements Capture, Design and Test Preparation, Coding and Implementation. The break between semesters presents a real problem. Ideally the coding stage should start before the break. However, the difficulties of a stop-go stage are felt to be too severe and so the start of the Coding stage is deferred to the second semester. This tends to mean more is required of the

students during the second semester. It also allows a less arduous synchronisation of the Maxi requirements and the timing of when students are taught relevant material within the academic modules of their MSc.

The planning activity results in a two page briefing document with diary and plan plus an example stage plan which are distributed in document form to the students on the first afternoon of the Maxi. A User Briefing is also produced and distributed to the appropriate users.

Fig. 1 shows a typical MAXI project diary.

SHEFFIELD UNIVERSITY - DEPARTMENT OF COMPUTER SCIENCE
MASTER OF SCIENCE COURSES

MAXI PROJECT 97 DIARY [REVISED 05/12/97]

Week	Date Wed	Project Stage	Lecture/Briefing	Tutorials	Deliverables
1	1/10/97	1	Project & Stage 1 Plan	No	
2	8/10/97	1	No	Yes	Project Budget, Stage 1
3	15/10/97	1	Stage 2	Yes	
4	22/10/97	2	No	Yes	Stage 1, Stage 2 Plan
5	29/10/97	2	No	Yes	
6	5/11/97	2	No	No	
7	11/11/97*	2	-	-	Draft Stage 2 (DS2)
	12/11/97		Stage 3 Inspection	DS2	
8	19/11/97	3	No	Yes	Stage 2, Stage 3 Plan
9	26/11/97	3	No	Stages 1/2 Review	Stages 1/2 Stats
10	3/12/97	3	No	No	
11	9/12/97*	3	-	-	Draft Stage 3(DS3)

	10/12/97			Stage 4	DS3	
				Inspection		
12	17/12/97		4	Yes	Yes	Stage 3, Stage 4 Plan
20	11/2/98		4	Stage 5	Stage 3	Stage 3 Stats
				Review		
21	18/2/98		4	No	Yes	Stage 4A, Stage 5 Plan
22	25/2/98		4/5	No	No	
23	4/3//98	4/5	No		Yes	Stage 4B
24	11/3/98		4/5	No	No	
25	18/3/98		4/5	Yes	Yes	Stage 4
26	25/3/98		5	No	Product	Stage 5 (everything)
				Marking		

Apart from 1/10/97 all briefings will be at 1.30pm. The timings of individual team tutorials and the lecture rooms for briefings and the seminar rooms for tutorials (1997 only) will be given on the appropriate notice boards.

* Tuesdays

PLEASE ENTER THESE DATES IN YOUR DIARY

Fig. 1

For the first few years of the Maxi student teams were encouraged to produce their own overall schedule but this made the overall management of the Maxi within the budget impossible. Also, many students found it too difficult.

6 Choice of Application and User

An application project and associated user is selected each year for each type of MSc. Typically there are three applications being addressed in one year with several student teams allocated to each one.

Getting real industrial applications is extremely difficult. Although the applications owner may be said to be getting something for nothing in fact they have to devote significant time,

particularly for their requirements to be elicited. This is more than the normal industrial situation where they would have to do this only once, whereas with the Maxi they have to do it for each student team. This presents a significant obstacle to industrial co-operation.

My own involvement with the users is limited. It is the job of the student teams to handle the face-to-face contact, as would be the case in the industrial situation assuming I was the project controller i.e. the project manager's superview. What involvement I do have is generally limited to problems with the user which the student teams cannot resolve e.g. the user's typical tardiness in providing test data.

Apart from test data the only items generally required from the user are a "wish list" at the start of the Maxi and a mark for each team at the end. The former is given to the students simultaneously with my briefing documents.

7 Formation of Student Teams

A team is generally composed of individuals on the same MSc course. The ideal number for a student team is five. This is based on general industrial experience where the communication overhead starts to become excessive within larger teams. It also fits in with the five Maxi stages in that it allows each student, within a team, to act as a stage manager. Teams of a different number are formed where the numbers on a particular MSc course make it inevitable. Six is the largest number allowed in a team. If this is the case, the third stage of the Maxi is divided into two viz Design and Test Preparation with a student managing each so that each student in the team still has an opportunity of stage management. Teams of three and four are common. Here some students have the misfortune to have to manage two stages, however they are compensated by having the higher of their two stage manager marks used in the overall marking. As an inducement they are told of this in advance.

A team of two has proved to be too risky apart from anything else it only requires one to drop out and the team is unviable. In these circumstances, and in others e.g. a breakdown of relations in a team, team transfers are contemplated although the later these occur in the life of the Maxi the greater the difficulty.
Apart from seeing that teams are largely composed of students on the same course and that teams are equally endowed with English speaking competence no other criteria is used in deciding the constituents of teams. Early in the Maxi a much more complex team selection

process was attempted. This categorised students as leaders, doers, communicators etc. but it was abandoned as it did not offer any apparent benefits.

8 Stage Managers

Subject to issues related to team size, as described above, and the requirement that each student will be stage manager, once the students agree amongst themselves which stages they will manage each stage manager has to produce a stage plan by the date shown in the overall plan produced by me. This will be based on the stage briefing which I will have normally given them a week before. At the end of the stage the stage manager has to produce statistics showing how predicted effort in the stage plan compared with actual effort. Each student in the team is required to sign this to indicate they agree on the hours worked. Based on a notional hourly charge rate the cost of the stage is thus determined.

9 Fagin's Inspections and End-of-Stage Reviews

The critical requirements and design and test documents are subject to a simulation of a Fagin's Inspection which I conduct in the relevant tutorials. In order for me to achieve this the overall plan requires drafts of these documents to be submitted a week in advance with sufficient lead time for me to examine them before the Inspection [see fig. 1]. The students have the opportunity to improve the documents before their final submission a week later. In accordance with the spirit of Fagin's Inspections they are not allowed to influence the Maxi Marking. The students, in addition to these inspections, are encouraged to perform their own within the teams on other documents and code.

Typically two end of stage reviews are held. A combined Feasibility and Requirements i.e. Stage 1 & 2 Review and a Design and Test Document i.e. Stage 3 Review. The Agenda followed is a cut down and modified version of that advocated by the PRINCE project management methodology. It is:-

- Actual against Planned Progress in terms of time and resources [money]
- Quality
- Outstanding concerns
- Updated Risk Analysis
- Go/No Go Decision
- Future Plans & Concerns.

10 Financial Control

As part of the Feasibility Stage the student teams are encouraged to produce a guestimate of the cost of their Maxi Project. The Key element is of course the number of man-hours required. This is converted to money by an arbitrary fee rate per hour which I give the students. The cost of the project is broken down into stage costs and it is these that are compared with actual costs, both stage and accumulative, at the end-of-stage reviews. Whilst the dubious nature of the figures is acknowledged to the students, the importance of financial control in real life industrial projects is constantly emphasised.

11 The Project Stages

Brief details follow of what is required in each stage given the Feasibility, Requirements, Design & Test Preparation, Coding and Implementation life cycle alluded to earlier.

11.1 Feasibility

A Feasibility Report has to be produced by each team approximately three weeks into the project. This should address the feasibility of and the justification of and for the project and include an overall estimate of the cost of the project and a simplified risk analysis.
Apart from my briefing the teams are ill-prepared for this. The advantages of it being a normal industrial requirement and acting as an immediate spur to get the teams working together are felt to outweigh any disadvantages stemming from this ill-preparation.

11.2 Requirements

A Report agreed and signed by the user is produced approximately seven weeks into the project. This should address both the functional and non-functional requirements and also any constraints on the design of how these requirements are to be met. The students are expected to negotiate with the user what these requirements will consist of. Unless there are significant problems I do not get involved. I do however advise the student teams how to go about the negotiations.

11.3 Design and Test Preparation

A Report showing how the system will be designed to meet the requirements allowing for the constraints is produced approximately eleven weeks into the project. Amongst the issues it will address will be:-

-Choice of Hardware.
-Programming Language Chosen.
-Design of User Manual.
-Choice of System Operatives and their training requirements.

Simultaneously a short document describing how acceptance testing is to be carried out will be produced. Attached to it will be a number of Acceptance Test Specifications and a blank proforma to be used for the reporting on the outcome of these tests each time they are attempted.

11.4 Coding

The document deliverables from this stage are the program listings [possibly on a floppy] and a road map overview of them for software maintenance purposes. The stage should proceed with the evolutionary development of the software. Normally I inspect this at three discrete times [shown in the overall plan] and the User should do likewise. The student teams are encouraged to perform modular testing but this is not examined.

11.5 Implementation

This involves the carrying out, including completing test report proformas, of the Acceptance Tests specified earlier until they are successful. It also involves the production of a User Manual. It culminates in the delivery of the finished system and all it's attendant documents etc. on the final afternoon of the Maxi. The total list is:-

Feasibility Report
Requirements Document
Design Document
Test Document.
User Manual
Software Listings
Software Documentation [Roadmap]
Test Reports

Change Notes
Acceptance form
Outstanding Statistics
Demonstration

The first 4 will be the documents I have already marked and returned to the user.

The stage plans and stage 1, 2, 3 statistics I will have retained.

12 Marking

Each team is given an overall mark for the Maxi comprising individual marks for most of the main items referred to above, apart from Stage Plans and Statistics. The individual items are given weightings dependant on their importance to the whole project [e.g. Requirements carries a high weighting] and each item's mark is multiplied by it's weighting before it is accumulated into the overall team score.

The final product, as demonstrated to me and the user, is marked by myself and the user. These marks, which comprise two separate items, carry a high weighting such that:-

a team may score highly on documents but poorly on the product in which case it will tend to fail, or score poorly on documents but highly on the product in which case it will tend to pass.

An important point is that the product is marked in relation to it's requirements including any amendments documented in change notes. Thus when comparing product marks between teams, without reference to the requirements, it may seem that a "better" product has a lower mark to an "inferior" product, to some students chagrin.

The teams are marked out of 80%. The remaining 20% is for individual students. The team mark plus the individual student mark out of the remaining 20% represents the total individual student's mark for the Maxi Module.

Half [i.e. 10%] of this 20% is award on the basis of the student's performance as stage manager. If he or she has managed two or more stages the highest mark is used. This is derived from the quality of the stage plans and the presentation of statistics.

The other half [i.e. 10%] is awarded on the basis of the individual student's contribution to the team. I determine this from the individual's contribution:-

-as shown by the stage statistics,
-by the number of test documents, lines of codes [authorship is requested in the form of comment lines], and
-their performance in tutorials.

The imperfections and hence dangers of this marking schema are recognised. All I can say is that I believe I have never failed a student who deserved to pass but students have passed who may have deserved to fail.

13 Problems

Some of the problems in trying to simulate an industrial software product in an academic teaching environment have been alluded to previously. There are others. Overall the significant ones are:-

- The impact on the user and the teams of several teams addressing the same requirement simultaneously.
- The constraints imposed by the equipment/software choices available in the University.
- The normal employer's power of termination of employment or the threat of it is absent.
- The stop-go nature of the project as imposed by it being spread over two terms and the introduction of the semester system which has exacerbated this.
- Budgetary limitations which preclude the full gambit of stage reviews, inspections etc.
- The need to schedule the project mindful of an academic teaching schedule.

However, any industrial environment would impose similar problems and restrictions and therefore I believe the Maxi is a relevant exercise for the students in learning to produce something whilst attempting to overcome obstacles.

14 General Observations

When I first managed the Maxi I was surprised to find the students were not as self-motivated and confident as I would expect from those attempting to gain a Master degree. This was one of the factors why the overall plan became prescriptive in terms of deadlines.

The second major observation is the difficulty many students have in seeing what the purpose of an IT project is. Many see the introduction of any form of IT as an object in itself. Maybe this is just symptomatic of an underlying problem within the IT sector generally.

However, despite the difficulties and problems most student teams produce a credible working product at the end of the Maxi. Only in a few cases does this reach a marketable standard but this in itself is, I believe, a valuable lesson for the students

Acknowledgements

My thanks to my colleagues and friends and indeed students in the Department of Computer Science at Sheffield University for giving me the opportunity to pass on some of my experience.

References

1. Mike Holcombe and Hugh Lafferty. Using Computer Professionals for Managing Student Software Projects. Conference Proceedings Developments in Teaching computer Science. University of Kent Canterbury 1992.

2. Stan Price. Managing Computer Projects. John Wiley and Sons

3. Michael Cusumana and Richard Selby. How Microsoft Builds Software. Communications of the ACM vol. 40 no. 6 June 1997.