How to introduce industrial/ client-led projects into the computing curriculum

1 Identifying your project objectives

The first issue to be addressed is the nature of the objectives that client-based projects involve.

A key aspect will be the introduction of new skills – interpersonal, organizational, managerial; that can sometimes be achieved through the medium of game type activities but the fact that the context is n real seems to reduce their impact amongst many students.

Further more, these projects will provide practice and help to extend recently acquired skills, in realistic context.

Don't forget that the teachers will also benefit by also developing many new skills and insights software engineering, project management and related areas.

In the final analysis the objectives must sit comfortably with the department's own, those departmer that desire engineering accreditation should note that many of the practical skills required can be found such projects.

2 Gaining the support of colleagues

2.1 Find a champion

Any new initiative is more easily established in a university department, when it is championed by one or more members of staff who understand clearly the benefits to the department and its students of the initiative, in this case, the advantages of building working relationships with industrial collaborators.

In many cases of which we are aware, an industrial project initiative has been introduced by staff senior level, by an enlightened Professor or Head of Department. Alternatively, a member of staff v specific responsibility for teaching and teaching quality may have the degree of influence and res necessary to drive the initiative successfully. If the department already employs a co-ordinator to organize student placements, this person is likely to have valuable personal contacts and experience of negotiati with industrial partners that will increase the chances of project success in its introductory stages.

2.2 An open minded culture

Whoever the champion of industrial projects, they will be more easily embedded into the curricul where the departmental culture prizes the quality of teaching and the authenticity of students' le experiences highly. As we remark throughout this handbook, industrial project organization and outcomes are less easily controlled than activities in the lecture theatre. Defining and assessing success for th students can be less straightforward than it is when marking traditional assignments.

We sense that a number of university departments have resisted the introduction of industrial partners in their teaching for fear of an adverse effect upon their teaching quality assessment. Conversely, we feel that with appropriate contingency planning for projects that might otherwise fail and with appropria student and staff review mechanisms in place, this fear is groundless.

2.3 Some persuasive arguments

Where resistance to the idea of industrial projects is anticipated or met, it is worth bearing in mind that a number of external demands on university teaching can be satisfied by the introduction of industri project work. Both the Dearing review and the Committee of Vice Chancellors and Principals envisaged a substantial increase in work experience and industrial placements for students, that can partly be met by our model of "client-led" projects run at the university site.

As mentioned earlier, our model of industrial projects finds favour with accreditors at the BCS and IEE.

There are a number of initiatives that seek to introduce entrepreneurship into the curriculum. One way in which to do this is to introduce a course similar to our fourth year "Set up and run your own sc company" course.

There are also a number of initiatives that seek to increase universities' role in the local and re economy. We feel we have been very successful in raising our department's profile in the local SME sector, as a result of advertising for companies to become clients for our students' software developmen and consultancy projects. The end result is that the University benefits from excellent public relations. The local business community sees that the "ivory tower" can provide a practical and effective sou expertise for many local organizations, both large and small.

2.4 Side effects for other courses

If you are about to embark on an industrial project course then your lecturing colleagues should warned of some anticipated side-effects, not all of which are desirable!

When we surveyed our second year computing and software engineering undergraduates abo motivations to study, the approval of a real industrial client was judged to be the most important motivating factor of six alternatives. We have to face it: students prefer to impress company representatives than us, their lecturers. As a result, they will devote hours of their energy to completi project for an industrial collaborator, often exceeding our recommendations for man hours and sometimes to the detriment of effort expended on other courses.

In fact, it is common for students to ask for extensions to deadlines for other course work because they have failed to contain their dedication and enthusiasm for pleasing their industrial project clients. colleagues who are committed to students learning by doing a job in a realistic setting are by and la tolerant of these requests for course extensions. Other colleagues are not at all tolerant. After all, they argue, the ability to plan ones time to accommodate the demands of all courses, is itself a skill that students must learn. What's more, they are right. Some students have learned this lesson the hard succeeding in the industrial project and failing another course.

Another side-effect is the tendency of students to seek out lecturers who are not involved with th project course, but whose subject expertise is related to the project's demands, for help and advice. Ag some colleagues may have the inclination and interest to help, but this tendency clearly needs to controlled. We would advise careful prescription of project rules at the start of the project course, rul which clearly identify permissible sources of help and advice.

3 Identifying suitable student cohorts

Throughout this handbook, we are keen to acknowledge that every university computing departmen and curriculum is different. Each department will tailor its teaching content to the strengths of its staff. We know that what works for us is likely to need some adaptation to fit with the curriculum and resources available in other institutions. That said, there are some pre-requisites for running successful industrial projects that have general application.

Industrial projects usually provide a very challenging experience to students, but it is our duty educators to ensure that students are armed with some, if not all, of the technical skills required for projects to succeed. So, first year undergraduate cohorts are not suitable for industrial projects that require an understanding of the software lifecycle and a rudimentary ability to analyze, design, code and tes working software system.

We do, however, subject our conversion Masters students to an industrial project, virtually from we one. This is partly a pragmatic decision, determined by the fact that they spend only a total of eight months on lecture courses. We justify throwing these students in at the deep end, on the grounds that, whilst their technical skills are weak or non-existent, their life and work experience will already ha exposed them to some of the communications and managerial skills that industrial projects de Additionally, our Masters projects extend over the better part of two semesters, with very rigo management by an experienced commercial project manager.

In common with a number of other institutions, we let our second yearstudents loose on industrial group projects, having prepared them with a dummy, internal group project at the end of their first year. By the second semester of the second year we feel they have sufficient appreciation of lifecycle competence in at least one programming language and the experience of courses in human-comp interaction and professional issues, to develop these skills further through practical application to a problem. At one institution we know of, the industrial project is arranged for the summer vacation at the end of the students' second year.

In courses with a predominantly business and management orientation it is still possible to carry realistic projects. Even if in those student groups, where programming skills may be very weak it is possible for the groups to carry out business process analysis and perhaps develop simple web pag databases for small organisations.

In the States and Australia, there are a number of examples of industrial group project courses "capstone" courses in students' final year, courses that are the crowning glory of a student's training software engineer. In the UK, we know of one institution that runs a third year undergraduate group project course focusing on software maintenance. We, however, have not run group projects in the thi year, for two reasons. Our industrial projects tend to have specific and practical requirements: they do pose open-ended nor theoretical topics for investigation This undoubtedly distinguishes computer science education in the UK from software engineering training in the US.

Secondly, there remains the thorny issue of fair assessment of individual students on group projects. In the second year, when project work comprises roughly 5% of the marks that contribute to a stud eventual degree classification, some margin of error in the assessment of an individual's contribution t group effort is tolerable. In the third year, when project work comprises roughly 15% of a student's degree mark, it is less defensible to set group work and to involve an external industrial organization. Should group fail or the industrial contact prove unreliable, there is too much at stake.

Increasingly, universities are seeking to offer four year degree courses leading to the award of a masters
degree. This offers a further opportunity for students to do novel project work to industrially specifie briefs. Indeed, when we have interviewed our fourth year M.Eng and MSc students about their motivation for staying on another year, when quite lucrative employment prospects beckon, they cite the opportunity to involve themselves in our "Run your own software company course" as the greatest determining factor.

4 Assessing and obtaining resources

In seeking to introduce external industrial and commercial clients into software engineering proje work, the need for additional resources must be recognized and met. Project courses have a reputation being expensive in terms of staff time. When we surveyed other institutions already running projects wit an industrial flavour, a number of respondents asked "Is there any way of doing what we do more cheaply?"

4.1 Estimating staffing requirements

Recruitment, selection and support of clients; negotiation of potential project outcomes with assessment of large volumes of software documentation are all time-consuming activities that tend to outside the traditional expectations of lecturing staff. So, how much staff time is required, for each student group, for each project client? Who is best suited to these roles? Do they have to be full-time mem academic staff and can teaching assistants or post-graduate students be involved?

It is not always possible for university departments to find staff suited to running such projects, nor for them to allocate the extra time required for project administration. Staff at both Durham and Salford, for example, have echoed these sentiments quite forcibly.

At Sheffield, we have operated a ratio of one project supervisor to approximately twenty five student on our second year projects, where students work in teams of five. We find it is not too onerous f supervisor to monitor five development teams, all competing to build a system for the same client, with total of three hours' student contact time per week, (30-36 hours' contact time per supervisor per pr course). A similar time is required for assessment, marking and smoothing potential project probl supervisor per project course.

For the past 12 years the Department has employed a very experienced software consultant to manag the MSc Maxi project. This has been a very successful exercise.

These students are on an accelerated conversion degree and it is important that they get used formalities of software project management as quickly as possible. The manager establishes a routine them and has the credibility to do this since he is a professional manager and not an academic. If funds were available it would be nice to do this in the other areas as well but, to be realistic, it is unlikely to be possible. However, academics can play the role with a little preparation and thought.

On the more intensive fourth year course, "Run your own software company", it becomes difficult for a project supervisor to monitor more than about a dozen students. Typically, a dozen students will forn three teams and will work for between one and three clients during the course of an academic year. Thi may mean that a team works for all three clients simultaneously at some points in the year.

The ratios we suggest for project staff to students are quite demanding. They can be fulfilled when there are under 100 students on a second year undergraduate course and when there are only 25 or so fourth year M.Eng students. Project organizers in institutions where undergraduate class sizes are in the range of 150-2 students will probably have to delegate project supervision to teaching assistants or post-graduate students.
We have not been keen to do this but we recognize it is a pragmatic solution to increasing student numbers.

4.2 What space is required to run projects?

Need a space in which student groups can meet together and with their client to plan and to revie project progress. In effect, you need access to many small rooms which are conducive to small gr activities.

4.3 What are the likely hardware and software demands?

This question is, of course, highly specific to an individual project and a particular student col However, we would advise against taking on a project that utilizes software or hardware that the department does not already possess. Even if the industrial collaborator has offered to sponsor nev hardware or software to support a project, lack of familiarity with an environment may prove a headache for technicians and may prove an insurmountable obstacle to effective management of project progress, as we have found to our cost.

A recent fourth year project that we undertook with some hesitation, was required to use complex transaction and database management software with which we as staff were unfamiliar. The project began to founder and time-scales began to slip unacceptably when tested in a network environment. As managers of the project, we were unable to identify the root of the problem, which was eventually trac with the aid of a friendly commercial IT consultant, to an inefficient and inappropriate strategy fc managing transactions.

4.4 Are external resources available?

Software development for external clients may require “state of the art” technical resources that ar(short supply in university departments. This is particularly the case in the current climate where ind clients’ foremost demand is for internet based applications.