Project Support and IPSEs

Paul Rook, GEC Software Ltd

System Development projects are rapidly increasing in size and complexity, presenting a growing problem to Business Management. In order to remain competitive it is necessary to meet the customer's expectations by providing sophisticated facilities with state-of-the-art implementation. Thus the development of a large system is usually technically complex and may involve hundreds or even thousands of man-years of skilled effort with correspondingly large budgets. In addition, the management of such a project is itself a formidable and intrinsically complex task, making it even more difficult to bring the project to a successful completion.

Nearly every large development project is faced with numerous difficulties. When a project is successful it is not because there were no problems but because the problems were overcome. Many of the problems are technical but often the critical ones are managerial.

Project management cannot be simply separated from the technical work that is being managed - despite the complexity and depth of the technical skills involved it is necessary for the project manager to recognise the interactions inherent in the technical work. Simply making management decisions based on reported progress and naïve plans without adequate realisation of the full implications of the technical issues has the consequence that technical problems are turned into project crises.

The successful completion of a project can be briefly defined as the development of a product on time, within budget and which meets the customer's requirements for functionality, reliability and quality. With this as our objective, let us look at the means available to overcome the problems presented by the Herculean task of managing the development of a large and complex system. These are threefold:

* define the project process for all the technical engineering disciplines involved also including project management, quality assurance and configuration management.

* establish the organisation to match the process and support it, making the most effective use of real feedback, enabled by the explicit definition of the process, to control the project.

* use tools to do the work within the structured process more efficiently by automating the tasks and supporting the process.

Definition of the Process

It is management's job in a project to steer the creative individuals and teams in a common direction so that their products interface with each other, are finished within the project cost and schedule constraints, and together accomplish the project goals. Achievement of the objective of a successful project depends on effective control of the project throughout its duration. Project control is based on the project manager setting a structure for the interactions within the project which includes responsibilities, recognition of risks and needs for decisions, commitments, achievements, change control and the acquisition of information to enable the project to be managed.

A project can only operate effectively when each member knows the answers to basic questions regarding the job such as:

- who do I work for?
- what is expected of me?
- why is it expected of me?
- what tools and facilities are available to me?
- how do I do what is expected?
- what must I produce?
- when must it be produced?
- who do I give it to?
- how will my product be evaluated?

Establishment of the relationship between the functions of project management and the technical activities depend on the engineering professionalism of the technical staff. Codes of practice, accepted standards, training in these and professional integrity in working to industry procedures are the mark of a mature engineering discipline. Where these exist then it is unambiguous as to what is meant when a task is assigned and unarguable as to whether the product of the task has been completed to the standard required.

Thus the codes of practice of mature engineering disciplines are part of answering the questions (listed above) posed by individuals in the project. A large system development will include work from a number of different technologies and where engineering practices are mature they will all be able to support explicitly defined processes within the overall process of the project.

Increasingly, large systems depend on software to achieve the functionality required. Contributing to the difficulties of control of the development project is the much publicised view of the programmer as the unbridled genius, whose creative process will be stifled by any of the recommended project management controls, standards and engineering disciplines. The worst fears are confirmed by the reputation of the software parts of projects for being the most difficult to estimate and control to planned dates for integration with the hardware and the difficulty of meeting the goal of reliable operation in a controlled timescale. In such cases, it has to be questioned whether the software development process used corresponded to a mature engineering discipline.

The fact that software will continue to be an increasingly important part of system development cannot be ignored, nor can the problems be wished away. The business importance is indicated by the fact that software is a significant component of large projects costing billions of pounds annually.

There are genuine differences between software and hardware, as follows:

- software has no physical appearance
- few software quality metrics exist
- software has much higher complexity than hardware
- it is deceptively easy to introduce changes into software
- effects of software change propagate explosively
- software includes data as well as logic
- software development has traditionally made very little use of pre-existing components

However, in many important ways software development is like hardware development and ought to be managed and controlled using very similar techniques to those used in any other engineering discipline. The genuine dissimilarities listed above are the very factors which make an engineering approach much more critical for successful software development. The principles of an engineering approach to software development are now well defined and provide a basis for the explicit determination of a development process backed by a code of practice with standards and procedures [1].

Since software is intangible, it is necessary to obtain precise definitions of products of tasks in the software development process, the achievement of which then provide unambiguous milestones for measurement of progress. Baselining the parts of the software definition, design, code and documentation so that they can only be subsequently changed under control of the project management provides the final link for a foundation of the technical software development within the overall project control. This can be established and carried out by professional software development staff as reliably as any other engineering discipline.

On the basis that the work for all engineering disciplines within the project development process, including project management, quality assurance and configuration management, is defined by standards and procedures and carried out by staff working professionally to industry codes of practice, then we have the basis for fully defining the process to be adopted for the project.

Establishment of the Process

The establishment of a project environment with the process of working to cope with the risks, changes and surprises of a development project and the corresponding organisation with clearly defined responsibilities and disciplines is the responsibility of the project manager. He or she has to define the process and ensure that it is understood, put in place at the beginning of the project and fully supported throughout the duration of the project.

Setting up the organisation to correspond to the process is a growing trend. Since it has been observed that products tend to display only too well the structure of the organisation that produced them, it is now perceived as wise to set up the organisation to correspond to the product that it is wished to produce.

Everything that is done right in project management is done early - there is very little opportunity for catching up when things are discovered to be going wrong later in the development. If the necessary foundation is not adequately put in place at the beginning, then, on a large project working to meet a deadline, there is never sufficient time subsequently to back-track and set up better means of project control.

Just as full insight into the appropriate engineering technologies is a crucial part of defining the processes used in the project, so also it is necessary for the production of estimates on which meaningful plans can be based. Reliable estimates are best obtained (and for important projects second best should be unacceptable) from a combination of the methods of expert judgement, summation of effort for individual tasks and metrics using data from previous projects. All three methods depend on an appreciation of the project process as well as on the definition of the work being tackled.

A process which is fully integrated with the technologies involved together with plans based on estimates which take into account the required product, the technologies and the process, provide the project manager with the opportunity for real project control based on real feedback.

The technical codes of practice (which include standards and procedures) provide the proper basis for assignment of meaningful tasks according to the authorities and responsibilities of the project organisation. The quality system provides measurement of conformance to the procedures defined in the process and correctness of the products to the standards. Together they can be used to ensure that declared achievement of a milestone can be relied on. The configuration management system ensures reliable possession of baselined products and, allied with a change control board, ensures control of project and customer requests for changes. The review and testing system gives early warning of product deficiencies - they are much more expensive to correct if not discovered until later.

If the project manager ensures that the project process includes setting up efficient feedback paths based on the above and they are carefully nurtured and attended to throughout the project, then the project manager can obtain effective and early feedback on planned milestones achieved, slippage, changes and the necessary information for re-estimating, replanning, reprediction and decision making - all in the confidence that it corresponds to reality (past, present and future). Feedback provides knowledge; knowledge is power - or at least the basis for making rational and effective decisions.

Imposing a structure onto the process of a large development project reduces the complexity of the whole task into a number of tasks of reduced complexity. If this is carefully and properly done, following the principles discussed, then there need be no loss of the essential nature of task and the ability of the project manager to control the project to a successful completion. This must be emphasised; on a large project, simplistically separating project management into out-of-touch planning unaware that the various teams are digging themselves into private technical holes is unlikely to be anything other than a disaster.

For a well structured process, the parts are chosen so that the communication threads that must be attended to between the parts are minimised and therefore amenable to the organisational authorities and responsibilities. This is quite feasible at the higher levels of the structure but becomes increasingly difficult as the structure is taken to lower and lower levels. However well thought-out the structure and the organisation there are levels below which the communication required starts making the work more complex.

The difference between a good structure and a poor structure for a given project is that the good structure will be able to work effectively at lower levels of delegation without everybody having to climb into communication meetings to review each other's work in order to make valid progress. An attempt to run a badly structured project process at too low a level results in inefficienct progress because of trying to do all the necessary communication (meetings and paperwork), or overstaffing (which makes the problem worse) or pathological connections between groups (depending on the network between responsible technical staff to protect the project from technical incoherence - not exactly project control and usually doomed).

Thus, despite the advantages of carefully and explicity structuring the process and establishing a corresponding organisation, there is a limit to how far the overall complexity can be reduced. There will still be tasks which are large and complex within the structure. If an attempt is made to ignore this and reduce all the work to straight forward small tasks then the technical staff will feel that the business is trying to reduce them to mere automata without the opportunity to creatively apply their skills to the undoubted technical problems of design and implementation. If this is done on any significant system development not only is the project likely to fail, but also the staff will be very disgruntled, with the best ones probably leaving.

Tool Support for the Process

To deal with the remaining complexity of the significant tasks in the structured process, we turn to two further techniques: use as few as possible really good staff (to reduce the communication overhead on a complex task) and equip them with the best tools so that they are efficient enough to complete the task in the required timescale.

The tools not only increase the rate of work (incidently, more so for good staff than for average staff) but also have the further significant advantage of automatically defining some of the way of working, thus simplifying the definition of the process and ensuring conformance. Tools can encapsulate procedures and include tests against standards. Although all of this is true for workshop tools, it is vastly more true for software tools. Note that software tools are not only applicable to the development of the software, but also to many other parts of the system development process such as project management, administration, system requirements definition. hardware design, interface definition, integration testing, configuration management, quality management, etc.

This brings us to the subject of Integrated Project Support Environments (IPSEs). Note the emphasis on Project - the IPSE is intended to apply as widely as possible to the whole process for all techniques used on the project. Having determined the process and the organisation for the project, thus leading to the determination of the methods of working and techniques to be used, this in turn leads to determination of the tools that are required and appropriate to the tasks within the structure of the process. Note the order: process -> organisation -> methods -> tools. Not tools first: providing tools for a difficult project does not solve the difficulties of the project. To the man with a hammer, all problems look like nails, with the result that he is both dissapointed and very unpopular.

However, an IPSE is much more than simply the appropriate collection of tools for the project, even assuming that they are fully appropriate and provide all the suitable facilities. The tools have to be integrated so that they work together and perform further operations on transferred and shared data so that automation achieves much more than merely the sum of the parts. They also need a common human communication style so that staff are comfortable that they are working in a coherent environment. Even providing all this is not an IPSE, only an integrated toolset. All sorts of integrated toolsets are needed for all sorts of different development applications.

A true IPSE also supports the project process. In other words, the IPSE itself is a tool supporting the method of overall working of the project, as well as integrating all the tools used on the tasks in the project into a unified toolset. Note that while the IPSE support for the process is a tool of project management, there will be other project management tools specifically for the tasks of estimating, planning, work assignment, progress collection, analysis and reporting, etc. These are regarded as part of the set of tools to be selected to be members of the toolset in the IPSE so that it is a fully operational environment to support all the activities of the project.

Some IPSEs define a particular process, which makes them suitable candidates for use on a project for which their in-built process is applicable and sufficient. But note the expressed danger of choosing the tool first and then having to force the method of working to conform.

A more recently developed IPSE like GENOS provides the facilities to be tailored to any required process. This includes facilities to extend and modify the project process as the project development progresses. The facilities to integrate the tools into a unified toolset within the defined process are naturally extended to include the incorporation of any existing or separately supplied third party tools, so that, if required, different integrated toolsets can be provided for different projects.

This allows the project manager full freedom to:

- determine the process suitable for the project
- determine the organisation, methods and tools required
- instantiate the process by defining it in the IPSE
- integrate all the chosen tools into a unified toolset
- efficiently establish the project process early and completely
- support the inter-staff communications with IPSE facilities
- modify the defined process as required as the project develops
- gain the efficiency of the tools and support the defined process by the combination of the tools and IPSE itself
- easily obtain the required real feedback which is the foundation for control of the project

Installing and running such a powerful project support system involves expertise. Project Management is a professional skill like all the other technologies and disciplines involved, and, in the matter of expertise in how to make the best use of the tools of the profession, is no different from the others.

The establishment of expertise in processes for development projects will benefit from IKBS research and tools just as these ideas are being applied to the constituent techniques within the overall process, and the future will see the incorporation of such IKBS tools into IPSE technology.

The whole approach of defining a process for the project and establishing the project environment with IPSE support is object oriented - in other words, the production of items which in turn lead to the project goal of successful development of the product. The ultimate aim is the achievement of the Integrated Systems Factory (ISF) (or Systems Development Factory (SDF) as the concept is also known). This is envisioned as combining the processes of all the work of the project (hardware, integrated circuits, software), supported by a unified system with common style terminal facilities for the expert designers in the various technologies, and the automation of all the underlying production processes on the same timescale for fast and reliable production of the complete system as a quality product.

Reference

1. P.E.Rook "Controlling Software Projects",
IEE Software Engineering Journal, Jan 1986, pp 7 - 16