

‘EARTHQUAKES’ IN THE PROJECT PORTFOLIO

How do you stop projects slipping? Graham Oakes (pictured) passes on some tips from his heavy-duty project management in the computer games industry.



Computer games are a serious business. A game for a current-generation console (say an Xbox or Playstation 3) can cost \$20 million or more to build. Even back in the mid 1990s, when my experience with the industry began, a game could easily cost \$2 million to develop. The company I worked for had about 60 such games under development at any one time.

My company, like most in the industry, had a problem. Projects slipped. They often slipped by months or even years. And this didn't do a lot to help our reputation with retailers, reviewers and customers. Perhaps even more critically, it made it impossible to predict cashflow. And so I became part of a team that was set up to bring predictability to our project delivery.

Each member of the team was responsible for providing an independent view of the status of about 10 projects in the development portfolio. Each week we looked at what our projects were producing and tracked this against the original milestone schedules.

We tracked the status of key risks. We read status reports. Above all, we talked with the project managers, discussing the issues they were dealing with and listening to their concerns. Sometimes we offered suggestions or put them in touch with other people who were dealing with similar issues, but often we just functioned as a sounding board to help them think through what was going on.

We also produced a weekly report for senior management – the development director and the chief financial officer. This consisted of a simple ordered listing of the projects under development, ranked by our assessment of their level of risk. We also wrote one or two sentences on each project, summarising our reasons for its ranking. This report was openly published to everyone in the company, which gave everyone plenty of chance to tell us where we'd got it wrong...

(Interestingly, project managers generally reckoned their project was riskier than we'd assessed it, while their line management generally thought projects were less risky than we'd assessed them. Either way, people started to actively track the positioning of their project, and to tell us how our ranking of its status could be improved. By publishing our report openly, we created a very useful channel for this information.)

After we'd been working with our projects for a while, we began to recognise a pattern. Projects would go through a couple of fairly formal investment-approval reviews when they were set up. They'd then run quietly for nine or twelve months. Then, about three months before the date they were due to be delivered into testing, they'd start to slip. Often they'd have a big slip initially, followed by a series of smaller slips as they got closer to the end date.

This pattern was remarkably consistent. Because we were working with a portfolio of 60 similar projects, we could draw graphs and start to see statistical trends. We found a strong correlation between the magnitude of each slip and the length of time left until the due date for delivery into testing. Projects took three-month slips when they were about three months from delivery, two-month slips when they were about two months from delivery, one-month slips when about a month from delivery, and so on.

Our company was developing racing games, adventure games, first person shooters, and so on. The development teams were based in the UK, in France and in America. But everywhere it was the same: our apparently stable projects suddenly started to slip in their end phases. And the pattern of the slips – a large slip followed by a succession of smaller ones – was the same for just about everyone.

To me, with my original training in geophysics, this pattern looks a lot like an earthquake. Stress gradually builds up as tectonic plates move. Finally the rocks break, give off a loud bang, and settle into a less strained position. Then a series of aftershocks release any residual stress.

So it was with our projects. For a long time people could ignore the small setbacks and minor slippages, perhaps hoping to make up the time later. Finally, as the end date loomed, they could no longer avoid reality. So they'd take a big slip to release all the built-up delay. Then the stress would build up again, and they'd take a series of smaller slips to release it.

We monitored this pattern as we continued our reviews. After a couple of years, we found that the pattern had shifted in time. Projects were still slipping. The general correlation was still pretty much the same – a large initial slip followed by progressively smaller ones. But the slips were happening about three months earlier in the project lifecycle.

There were several reasons for this movement – people were monitoring status more closely; project managers could use the review team to back their judgement as to when a slip was needed, so had the confidence to make the call earlier; we'd got better at defining clear milestones.

Overall, though, we were simply having a much more informed dialogue about our projects. This helped us to identify and relieve the stresses earlier. It also allowed us to give everyone three months' more notice about potential delays, meaning that the CFO could be a little more confident about setting analyst expectations, and about making payroll.

Of course, life's never as simple as the case studies make out. In order to operate effectively, the review team needed to overcome a number of challenges. For example:

- Development teams typically have very diverse skills.

In our case the teams included programmers, graphic artists, musicians and so on – and it can be difficult for a reviewer to understand the status of everything such specialists are working on.

By doing small, frequent reviews we could often get a good feel for overall trends, but sometimes we had to call on technical experts from other teams to help us understand what was going on.

- Even with this specialist support, it was often difficult for reviewers to get their heads around a project.

No individual can understand all the details of a substantial project in a few hours, or even a few days. Reviewers often wondered how they could add value when they were only skating across the surface of their projects. (Project teams often asked exactly the same question!)

We found that it helped to treat every review like a small project in its own right, with clearly defined objectives and well-bounded scope. That helped us to focus, and hence to maximise the value we were adding.

Likewise, we developed a well-defined process for running reviews, with a suite of checklists and other supporting assets, so that we could hit the ground running for each review. (And generic checklists don't work: we needed to tailor them for the specific types of project we were dealing with.)

- It could be very difficult to get people to act on the findings from reviews.

Project teams become very committed to their plans and designs, and project sponsors rarely want to hear bad news about their projects. We learned to ensure we had very clear evidence to back our judgements. This often meant delaying the delivery of our recommendations until we'd painstakingly built a strong case for them.

This could be very frustrating – sometimes we were very conscious of the money a project was burning as it went down a blind alley, but we had to let it proceed until we had a clear case for intervention.

If we released our recommendations prematurely, people would reject them and become even more entrenched in their current approach.

- Reviewers could become very isolated.

They floated across several teams rather than belonging to any one project. They reported outside the normal management

structures. Furthermore, they often had to back their own judgement and deliver bad news against the convictions of a project or line manager. So we needed to build our own internal mentoring and support structures.

- As we began to establish ourselves, we suddenly found that everyone wanted to be the first to see our reports.

Project managers naturally wanted a chance to fix any issues on their projects before we reported them more widely. The project managers' line management wanted to know what we were saying before their executive management heard it. And, of course, the executives we reported to wanted to hear of any issues as quickly as possible.

We had to evolve clear communication protocols to maintain everyone's buy-in.

Of course not all sectors are like games development. They have different drivers. Companies have different strategies and approaches. People differ in all sorts of ways. Games are very different to banking engines or to customer relationship management systems.

However, there are many similarities too. Many of the lessons we learned on that games portfolio can be generalised. In particular, three things stand out for me:

- People need a sounding board.

Most IT projects are complex, with many people involved and lots of moving parts. Project managers are rarely given the time to sit down and reflect on what's going on.

Simply by creating space for this reflection, we helped them identify and solve many of the problems on their projects.

- Openly published information creates a conduit for dialogue.

This dialogue helps improve the accuracy of the original information, and it also provides an opportunity to gather more information. Project, line and executive managers all need such accurate and complete information if they are to make effective decisions.

- Independent reviews can help validate the information provided by teams and project managers.

They can also help make the above reporting and reflection processes more robust. I believe project reviews can add value in virtually all circumstances, but you need to tailor them to each situation.

Overall, projects can only succeed when they deal with reality – and reflection, dialogue and independent reviews are the key tools for helping our projects keep in touch with reality.

- *Graham Oakes is principal of independent consultancy Graham Oakes Ltd, specialising in helping companies untangle complex technology, relationships, processes and governance. His book 'Project Reviews, Assurance and Governance' was published by Gower in October 2008. Email: graham@grahamoakes.co.uk. Website: www.grahamoakes.co.uk.*

- *If you would like more information about this article or any of the products or companies mentioned in the article, please contact us at info@evaluationcentre.com.*