Working Against the Clock: Timescale-Driven Development

Ray Edgson, Chris Hammond, and Brian Moon

Traditional product development methods take more time than is generally acceptable today. They also treat time as an important but secondary consideration, beginning the planning process with a set of assumptions about the steps involved and only then asking, „How long will it take?“ We have developed an approach that we call timescale-driven development, which sets traditional methods on their heads. Time-scale-driven development allows new products to be developed much faster – even when they involve high technical risks or regulatory requirements. With timescale-driven development, we know how long we have – the question is what must we do and how must we do it before the deadline. Having a defined end date shapes the planning process, dictating how much time can be apportioned to certain fundamental activities, such as soak tests and tooling manufacture. The process even determines to some degree the specification – the number of prototypes that will be necessary, the resources that will be required, and whether resources must be found externally.

We have applied timescale-driven development to a number of projects, with remarkable results. For example, we recently used this approach to develop a revolutionary kidney dialysis monitor. This breakthrough product, which measures the removal of blood waste products in real time during dialysis, should be of major benefit to thousands of dialysis patients worldwide. Our client came to us with a laboratory demonstration of the measurement technique. From this beginning to the commencement of clinical trials in three countries, using preproduction prototype equipment built to FDA standards, took just 10 months. This project required bringing together many activities, including software, industrial design, user interface design, chemistry, molding design, electronics, and mechanical engineering. Technical challenges included electronics and software involving no fewer than seven microprocessors, as well as development of a new conductivity cell technology with unprecedented performance. The extraordinary speed of the development is particularly notable as it involved the first successful use of this technology in the field. This is in contrast with most fast-track concurrent development projects, which are undertaken for incremental advances in product evolution, rather than for revolutionary new products.

While we are not at liberty to reveal the technical details of the above project, or of other timescale-driven development projects we have undertaken on behalf of various clients, we can offer some insight into how timescale-driven development works.

New Rules

Concentrating on *how* rather than on *how long* changes many of the rules. Reevaluating normal practice is vital. Conventionally, for example, the bill of materials follows the assembly drawing, which follows the parts drawings. Unfortunately, this means that the parts themselves are ordered late in the process and arrive even later. Yet there is no reason why parts cannot be ordered in advance. At this stage in a product's life, the cost of components is generally a very small proportion of the total cost; if some parts are ordered but not used because of design changes, the loss is minimal compared to the cost of delays.

Concurrent, Modular Development. Traditional development methods are linear. Each activity or development module is undertaken in turn, with the output from one feeding into the next. Failure at any point involves an iteration loop and may even mean having to start the process again from the beginning. Although this linear approach reduces risk, it also takes a long time. Late market entry means shorter product life and therefore reduced returns. Faster progress is possible through the use of concurrent product development, which shortens the development cycle by telescoping sequential tasks where possible. As soon as information is available, it is shared, rather than being sorted for delivery when the consignment is complete.

A Focus on Integration. If modularizing the project in this way is a key to fast development, the integration of these modules must be treated as an activity in its own right and developed alongside the other modules, just as if it were a module itself. Developers must establish and test the interfaces before integration starts to avoid problems late in the program.

Simulation of the User Interface. It is also important to remember the final user in this process. Simulating the user interface early in the development ensures that the product will have the features and functions that the manufacturer and end user will find most useful – features that may not necessarily have appeared in the specifications. Developers can create this simulation in software long before any hardware is available. Early simulation minimizes the risk of encountering user problems late in the development program, when changes are difficult, time-consuming, and costly.

Managing Risk. Concurrent engineering is of course not new. Traditionally, while it improves information flow, its contribution to the speed of the project is limited, because the process presupposes that information must be available before dependent tasks can start. In timescale-driven development, however, we recognize that not all desirable information will be available, and we manage the resulting uncertainties to meet the given deadline.

Indeed, managing risk is a vital part of timescale-driven development. Developers must identify and control risk foci – aspects of the project in which issues may cause delay. Sensitivity to failure on these developments is high. Because just one broken link in the chain can cause crucial problems, the weaker links must be watched closely. Although risks can be identified by technical specialists, they must be assessed by those with responsibility for achieving the timescale. When assessing risk, it is important to look for evidence that similar problems have been solved before – whether by people working on the project or by others. Where the team identifies a high-risk focus, it must either remove it or avoid it. Certain questions are useful in these circumstances:

• Is the feature so important that we are prepared to let it threaten the delivery date? If not, can it be removed?

• Can we do it a different way?

• Can/should we develop a different fallback solution in parallel?

• Can we identify a plan of action that allows the project to proceed in the event of failure at a risk focus? If we use this option, what needs to be started and when?

Some quite simple analyses can help risk management. A team member needs to be able to step back from the detail of a product or process simulation to make some basic assessments. Discovering that a process needs, for instance, to be a little faster – or a lot faster – will indicate the risk to be managed: defining the precise ratio is an unnecessary digression. There's no point defining whether a component will cost $16.26 or $16.34 if the project will not bear it costing more than $2. Unlike the case in incremental, evolutionary developments, where simulation tools may be useful in pinpointing reliability and accuracy, in revolutionary developments a feeling for the physics or economics may be faster and far more revealing.

Minding the „Gaps." Just as managing integration of development modules is vital, so too is managing and monitoring the risks inherent in the „gaps" between individuals and their specialties. As timescales are compressed, specialists tend to focus on their own tasks and on the concepts and tools with which they are most familiar. Everyone, naturally, prefers to work on areas where success is easiest. Pressure to achieve diverts attention away from tasks that are unfamiliar, frustrating, or slow. This natural selection process can allow problems to grow unnoticed until it is too late, spelling disaster for the project. Similarly, specialists tend to spot risks only in their own fields. Ask a paving slab expert about the weakest part of a pavement, and he or she is unlikely to tell you about the gaps between the slabs, and even less likely to consider the soil beneath. The areas without specialists must be identified and the risks assessed and managed in advance. Such gaps also appear in development teams between groups of different disciplines. Problems occur at the interfaces between the areas owned by each discipline – for example, the PCB designer and the case designer may both think it is the other person's job to specify the switch covers.

Managing timescale-driven developments demands a holistic approach. A specialist cannot readily recognize ways of solving the problem at hand that do not use his or her skill set: where a mechanical engineer, for example, might struggle to correct fluid oscillators that cause a flicker in a pressure display, a software engineer could correct the problem with one line of code. Individuals' abilities, perceptions, and misconceptions limit the „solution space" – the totality of possible solutions to a problem. Keeping the solution space as large as possible is important. That space must include avoiding the problem or choosing not to solve it by removing the requirement: for instance, developers may choose to abandon an apparently desirable feature or keep it for a midlife upgrade if its inclusion threatens the launch date.

Ensuring Headroom. Similarly, solutions must have headroom. If a solution seems to be just good enough on the basis of a feasibility assessment, it is not good enough for a timescale-driven development. Because of the limitations inherent in the prediction of performance at an early stage, a design must have scope for more performance before a fundamental change is required. In a recent product development audit, a development team proudly announced that it had optimized the design of a complex electromechanical product. The project was promptly stopped, as no more headroom existed and the likelihood of the remaining developments being successful was too remote to justify the investment.

As a further example, a biochemical monitor development required a disposable reaction column, the size of which was a balance between speed of instrument response and longevity of column. Trials on the longevity would take six weeks, but the design of the holder had to be started to meet a tooling deadline. The solution was to make the holder suitable for any length of disposable within the conceivable range and to make the tooling for the disposable such that the length could be modified quickly if necessary. In this way, developers could accommodate without significant delay any credible outcome from the longevity trials.

Conclusion

Timescale-driven development has many benefits. It is a tactical tool that companies can use to respond to competitive threats, to meet launch windows, or simply to maximize product return through early launch. The discipline imposed by the plans can be of great benefit to a project. For example, in terms of marketing benefits, marketers can assess and predict market requirements much closer to actual launch dates.

Companies don't need to undertake timescale-driven development for simple, incremental product improvements, when technical risks are low, or when development costs cannot be offset against the payback from an early launch. But when a product requires a significant technical breakthrough or involves high technical risk, and where early launch will more than justify development costs, timescale-driven development can enable quite remarkable achievements in the development of revolutionary new products.

Ray Edgson is the Technical Development Manager at Cambridge Consultants Limited, where he has been developing products for 10 years.

Chris Hammond leads the Product Engineering Group at Cambridge Consultants Limited! Cambridge, U.K., headquarters. This team specializes in the development of engineered products for clients in health care, consumer goods, and instrumentation.

Brian Moon leads Cambridge Consultants Limited`s Health Care Practice. Health care, which accounts for about 25 percent of CCL's business, is one of the five industries in which ADL coordinates its activity globally (the others being the automotive, chemical, energy, and, technology/information/media/electronics industries).