

Methods of managing madness

Project management methodologies have transformed throughout the ages. **Pedram Daneshmand**, **Roger Fance** and **Gabrielle Morrison** examine how. Introduction by **Adeline Teoh**

Since the birth of project management as a formal discipline in the 1950s, methods of managing projects have evolved from planning and scheduling tools to more comprehensive project methodologies that include benefits management, stakeholder management and communication strategies.

This article covers the Critical Path Method (CPM), a scheduling tool which began in the 1950s, follows through from Waterfall to Agile, brainchild of the IT sector, to arrive at the updated, widely accessible methodology PRINCE2.

Taking a Critical Path

CPM is the most dominant scheduling network analysis technique first developed by the company E I du Pont de Nemours (DuPont) through research in 1956, which led to the influential work of James Kelley and Morgan Walker in 1957, who used the term CPM “because of the central position that critical activities in a project play in the method”. One of CPM’s prime success stories is its use in the Apollo 11 Project, which landed two men on the moon on July 20, 1969. NASA used the CPM to help determine an efficient schedule for 2 million tasks that led to the moon landing.

To establish a meaningful Critical Path, it is necessary to develop a logic-based network of activities with empirically derived durations for execution in a realistic and practical manner. The project duration is computed

by applying the algorithm of Forward Pass and Backward Pass, which compute the ‘earliest’ and ‘latest’ start/finish times for every activity. The early start and finish times are the earliest time that each activity can be started and finished, given the technical precedence relationships in the network. The latest start and finish times are the latest an activity can be started and finished, without delaying the total time to complete the project. The difference between the early and late start (finish) time is the ‘Total Float’ or ‘Total Slack’ associated with an activity. The activities with zero total float cannot be delayed without delaying the entire project. These activities are on the

longest path through the network, which is called the Critical Path.

To form the schedule network, the work is divided into a set of interdependent activities grouped under a work breakdown structure. The relationships between individual activities are defined in the network by using the Arrow Diagramming Method, Activity on Arrow, Precedence Diagramming Method or Activity on Node.

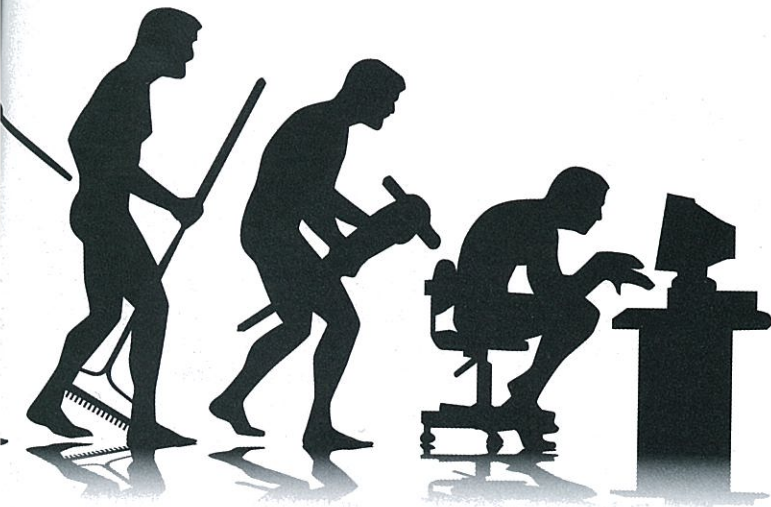
Scheduling using CPM has established its capacity to successfully manage time in complex projects. By 1962 it had demonstrated time reductions against manual methods of approximately 25 percent, as well as representing a significant cost saving.

Strengths

- Maximises efficiency in the use of time
- Makes dependencies between activities visible
- Provides a systematic approach for scheduling projects
- Calculates the total float for each activity
- Supports project compression by providing capability to analyse and crash the critical path
- Supports delay analysis and claim or dispute resolution
- Creates a platform for impact analysis of the identified risks and opportunities

Limitations

- Relies on the deterministic estimation of activity duration
- Accuracy is dependent on having reliable data
- A sophisticated software tool is required



- Only provides one set of the deterministic critical path
- Its efficiency may be limited in projects with repetitive processes
- Creates less focus on non-critical activities, which can ultimately cause risks and delays.

From Waterfall to Agile

All too often in life, people form a strong view on a subject, heavily invest energy and emotion, take up cudgels and fight to the death, sometimes literally, to defend their position or impose their opinion. Interestingly, we often find evangelistic fervour when it comes to supporting Waterfall or Agile software development methodologies and their associated project management techniques. For some, Waterfall is old-fashioned, inflexible or unrealistic. For others, Agile practices are a passing fad encouraging ill-disciplined cowboys to develop software in an uncontrolled and unaccountable environment.

Winston R Royce, a Lockheed aeronautical engineer, first described the Waterfall Model in a paper in 1970. His paper, 'Managing the Development of Large Software Systems', described a stepwise approach starting with system requirements and moving through the phases of software requirements, analysis, program design, coding, testing and operations. These phases have been reordered, renamed and refined over the years, although the basic concept remains the same. While believing in the concept, Royce recognised that the approach was risky and invited failure. He advocated techniques to reduce risk, including iterations (a feature of Agile!) between adjoining phases, high-level upfront design before detailed requirements were complete and very detailed and enforced documentation along all steps in the process.

Through misinterpretation this model, which came to be known as Waterfall for its 'step' illustrations in the paper, was taken up the US Department of Defense in its standard STD 2167 and later the Europeans.

Before long, it became the de facto software development standard. Project management frameworks, such as PMBoK and PRINCE2, usually implemented in a process-prescriptive and document-heavy way and implicitly or explicitly acknowledging a phased approach, closely support Waterfall.

Agile software development has a surprisingly long history too, dating back as far as 1957. A number of different 'agile' practices, initially inspired from a variety of sources—including notably, lean manufacturing at Toyota in the 1990s—coalesced into the Agile Manifesto in 2001 when a group of software developers came together to discuss lightweight development methods such as Scrum, Extreme Programming and Feature Driven Development, which had also evolved throughout the 1990s.

The Agile Manifesto describes four themes and 12 underlying principles for software development. Working software,



Project managers often take on the role of servant-leaders”

adaptability to change, personal interactions and collaboration are more highly valued than prescriptive processes, documentation, contracts, tools or plans. At the foundation of the Agile concept is incremental delivery of the highest business value first, or reduction of the highest risk as quickly as possible through iterative development of software.

In 2004, the Declaration of Interdependence supported the Agile Manifesto due to growing recognition that there was also a need for a broader statement addressing project management and general leadership. The statement, empathetic to the principles and practices behind Agile thinking, focused on value, customers, individuals, teams, uncertainty and context.

So, what are the benefits and shortcomings of each approach? Well, Waterfall is easy to understand and communicate to a range of stakeholders. There is natural appeal in its simple, linear, structured approach, which usually combines formal phase gating by key stakeholders, preventing the project progressing until satisfied the previous phase has been properly completed. Greater time spent in requirements and design is seen to reduce later coding and testing costs. Waterfall planning is predictive and can therefore inform the development of longer-term business plans and road maps, although longer timeframes also engender greater uncertainty. Waterfall appears more disciplined than Agile, but Agile enforces more discipline than is immediately obvious.

On the downside, Waterfall is less adaptable to changing business requirements, which occur when at the outset business people do not know exactly what they need, or a changing market compels a change of direction. There is also greater emphasis and effort devoted to following prescriptive process or transient project control documentation, which contributes little or no value to the required business goals.

Agile acknowledges requirements' uncertainty or rapidly changing market conditions by working in short iterations to deliver incremental business value. This is facilitated by small, cross-functional teams that combine »

» the complementary skills and experience of business and IT people. These teams are self-disciplined and self-organised although they take their business priority direction from the key business stakeholder. Project managers often take on the role of servant-leaders, facilitating information exchange, protecting the team from interference and removing progress blockers rather than directing the team.

Agile has its weaknesses too. There is a heavy reliance on close business involvement, ideally on a daily basis. This places a great burden on business people who must also juggle daily operational responsibilities. Teams are more efficient if they are small, co-located and have the right physical working environment, which is often hard to achieve in large, geographically separated organisations. It can be a lot harder to scale up and control Agile teams for larger software development efforts or to integrate them with other areas of a business that do not operate using these techniques. Agile is not very effective in integration projects or particularly valuable where requirements are well understood and stable.

It is important to be agile and not just do Agile. Company culture, individual personalities, the nature of the project

and a range of other factors should be taken into account when choosing a development and management method. In recent years, work has been done to combine the best of predictive and phased approaches with adaptive and iterative approaches forming hybrid methodologies such as the Rational Unified Process or the lightweight Agile Unified Process. Applying these or adapting in-house development methodologies and project management practices to the business circumstances are the third way. The answer is not binary!

The PRINCE of project management methods

The history of PRINCE2 reveals its IT and government roots, but its development reflects the continuous input from users and industry to its present form, suitable for any project. The efficacy of PRINCE2 relies upon project managers' active engagement with the methodology. Project managers in turn rely upon their organisations to support them as they deliver change and benefits.



PRINCE2's development reflects the continuous input from users and industry to its present form, suitable for any project"

In the 1970s the UK Government saw the need for an IT project management standard and Central Computer and Telecommunications Agency (CCTA) adopted PROMPT. In 1989, PRINCE (Projects IN Controlled Environments) developed from this method; it became PRINCE2 in 1996. The development of PRINCE2 from PRINCE was facilitated by users, project management specialists and a review panel of 150 organisations both public and private.

PRINCE2 is a structured project management method, with a focus on the business case, the promotion of consistency in projects, the cataloguing of lessons learnt, common vocabulary and the representation by stakeholders in the project planning. Although the roots of PRINCE2 were in IT, it is now a

generic best practice tool, flexible enough to be tailored to any organisation and project.

PRINCE2 is owned by, and is a registered trade mark of, the Office of Government Commerce (OGC), which is now part of the Efficiency and Reform Group within the UK Government's Cabinet Office. The OGC also has methodologies around program and value management, risk, PMO-type offices and portfolio management. PRINCE2 foundation and practitioner qualifications are available Australia-wide through a number of accredited training organisations.

In 2010, the Queensland University of Technology undertook an empirical study into the impact of PRINCE2 on project performance. In parallel, the study also researched the project performance of other unspecified (non-PRINCE2) contemporary frameworks. The research participants were either PRINCE2 practitioners or project managers using another framework, with two or more years of recent project management experience. Participants, mostly in the ICT, construction and transport industries, were from Australia, Europe, the UK and USA.

In summarising the findings, the researchers found that PRINCE2 is perceived as a robust, comprehensive and pragmatic project management framework. Problems and issues that impede success of projects using PRINCE2 are organisational not methodological. The dominant issue identified by participants was poor project sponsor/board performance and a reflection that organisations either do not know how, or do not possess the commitment, to properly implement PRINCE2.

This study, with a range of other relevant findings, provides a welcome point of reference for future development of the methodology. Visit www.prince-officialsite.com to view the paper 'Creating Value in Project Management Using PRINCE2'. ♦

Pedram Daneshmand (Critical Path Method) is the associate director of international consultancy blueVisions. Roger Fance (Waterfall, Agile) is a principal consultant at UXC Consulting and heads the UXC Consulting Agile Centre of Excellence. Gabrielle Morrison (PRINCE2) is the external relations manager of PRINCE2 training accreditation organisation APMG-Australasia.