

A Guide to Scheduling Good Practice

This paper is designed to provide guidance on ‘generally accepted good practices’ for the development of an effective, dynamic project schedule. The contents of this paper are consistent with the *Guide to the Project Management Body of Knowledge 5th Edition*¹ (PMBOK® Guide) and the PMI *Practice Standard for Scheduling*².

This paper assumes one person accepts the role of a ‘project scheduler’ responsible for the technical aspects of developing a ‘good schedule’. This person’s responsibility lays in ensuring the technical competence of the project schedule ‘model’ as the project progresses from feasibility, to planning to execution and control. Whilst an effective scheduler brings significant skills and knowledge to the schedule development process^a, ownership of the schedule must remain with the project team and the project manager.

Purpose of the Project Schedule

The purpose of the project schedule is to provide a useful ‘road map’ that can be used by the project manager and the project team to assist them in completing the project successfully. The schedule becomes a dynamic tool developed by the project scheduler, with input from the project manager and project team that reflects their vision of how the project will be performed and reacts appropriately to changes in progress, scope, etc., as they are incorporated into the project schedule over the life of the project. A well developed project schedule model is a dynamic tool that can be used to predict when the project work that remains to be completed can reasonably be expected to be accomplished. Simultaneously, it allows the project team to look at the performance of the project to date, and use that data to make more accurate projections about their intended work and actions in the future. It is important to remember we cannot manage or ‘control’ time; there will only ever be 24 Hrs in a day and they run sequentially! What the schedule can do is help the project team manage the use of the available time in a coordinated way.

The project schedule describes what work is to be done, who will undertake the work (resources), and when it should be done. ‘How’ to do the work is defined by other documents in the overall project plan as defined by the PMBOK Guide. When developing the project schedule, it is critical to remember the schedule cannot ‘control’ the work of a project (and neither can the project management team), the people who ‘control’ the work are the workers, management can only influence the workers and the schedule should be the guide management uses to determine what influence to bring to bear on the workflow.

Establishing a realistic and achievable project schedule is one of the critical initial actions in setting up a project. Equally important is the regular statusing and updating of the project schedule to support the on-going monitoring and controlling of progress as the project work is executed.

Planning & Scheduling

Project planning and scheduling are allied disciplines, but are not the same. Project planning is a team operation involving the management team, cost control team, design team and project

^a For more on the skills and attributes of a scheduler see: *The Roles and Attributes of a Scheduler* at http://www.mosaicprojects.com.au/PDF/Attributes_of_a_Scheduler.pdf



A Guide to Scheduling Good Practice

planner in creating the project development strategy. Whereas scheduling is a mixture of art and science, involving the interpretation of the results of project planning by using appropriate software tools and techniques to ascertain, amongst other things, the start and finish dates of activities and their sequence. During the three phases outlined above, virtually all of the effort in commitment planning is devoted to the planning function; whilst execution scheduling completes the planning process and starts the scheduling process which continues during performance control. It is not good practice to plan the work while attempting to schedule it.

Project planning^b involves the scheduler working with the project leadership to make decisions concerning:

- determining the policies, procedures and documentation required for planning, developing, managing and controlling the project schedule;
- developing the overall strategy of how the work process is to be broken down for control;
- how the control is to be managed;
- what methods are to be used for design, procurement and execution;
- the strategy for subcontracting and procurement;
- the interface between the various participants;
- the zones of operation and their interface;
- maximising efficiency of the project strategy with respect to cost and time;
- risk and opportunity management.

After planning, the scheduler should work with the people responsible for executing the work to determine:

- the duration of the activities;
- the party who will perform the activities;
- the resources to be applied to the activities;
- the method of sequencing one or more activities in relation to other activities, and
- communication and reporting formats, timing, etc.

Designing the Project Schedule

Based on the decisions made during project planning, the project schedule requires specific planning and design in the same way every project deliverable is planned and designed. To create a useful tool for controlling the progress of the project and communicating information regarding the planned work and progress the project team needs to consider a number of factors and seek to optimise the outcome. Some of the key questions to consider are:

- Determine if 'one schedule' is adequate or if a number of 'Levels' are needed to facilitate the schedule development and management processes^c. Then for each schedule determine:
- What is an appropriate level of detail to use for the activities? Too much detail produces a confusing and overly large project schedule that is difficult and expensive to manage; too

^b For more on **planning** see: http://www.mosaicprojects.com.au/WhitePapers/WP1039_Project_Planning.pdf

^c For more on **Schedule Levels** see: http://www.mosaicprojects.com.au/PDF/Schedule_Levels.pdf



A Guide to Scheduling Good Practice

little detail means there is insufficient information for the on-going management of the project.

- What is the preferred scheduling process to use; deterministic or probabilistic^d?
- What is an appropriate cycle for statusing and updating the Project Schedule Model? The period between updates needs to be long enough for the information from the last update to have been issued to the project team and for the team to have had a chance to act on the new information prior to the next status.

The choice of cycle time is influenced by the rate of change in the project. For relatively stable, low risk projects a monthly or fortnightly status cycle may be appropriate. For volatile, high risk projects updates may be required at every change of shift, or even hourly.

- What 'time scale' should be used for durations: minutes, hours, days, weeks or even months? The optimum answer depends on the frequency of the control processes and the level of detail needed in the activities.
- What reporting requirements will the schedule need to fulfil? Understanding the types of reports needed from the project schedule at each update (or reporting period) provides guidance on the optimum coding structures that need to be built into the Project Schedule.

There is no right answer to these questions, only the best answer for the particular project and the best answer will be influenced by the current phase of the project (feasibility, planning, execution^e), the culture and business norms of the project, its performing organisation and most importantly, the stakeholders the schedule will need to inform or influence^f.

The key steps in designing the schedule are:

- Intelligence gathering, know what information is available and what is needed to answer the questions posed above.
- A design summit where management and the scheduler agree the major compromises/parameters needed from the schedule to facilitate effective management of the project.
- Creation and adoption of a specification for the schedule that will guide its development and on-going maintenance (including change control processes).

^d The options are:

Deterministic = CPM = Critical Path Method. (Also called CPA – Critical Path Analysis). The two variants of CPM are ADM and PDM.

ADM = Arrow Diagramming Method. Also called AOA (Activity on Arrow). This style of network diagramming has largely faded from use.

PDM = Precedence Diagramming Method. Also called AON (Activity on Node). PDM is the standard form of CPM networking used today.

Probabilistic = PERT = Programme Evaluation Review Technique; or Monte Carlo. Both use range estimates.

For more information see:

<https://mosaicprojects.wordpress.com/2015/10/30/scheduling-acronyms-use-the-correct-terms/>

^e For more on **The Roles and Attributes of a Scheduler** see:

http://www.mosaicprojects.com.au/PDF/Attributes_of_a_Scheduler.pdf

^f For more on **managing schedule stakeholders** see 'Stakeholder Centric Scheduling' by Dr. L. Bourne:

www.mosaicprojects.com.au/Resources_Papers_049.html



A Guide to Scheduling Good Practice

- Followed by the consistent application of sound general practices as described below and defined in *The Practice Standard For Scheduling*^g published by PMI.

The overall sequence of work to develop the schedule should be:

- Select the scheduling method.
- Select the scheduling software and design the scheduling model based on the questions outlined above.
(NB: these decisions are frequently determined by project specifications and/or organisational policies).
- Enter project specific data into the scheduling model.
- The populated scheduling model becomes the project schedule model.
- The agreed project schedule model becomes the project baseline.
- Printed or stored versions of the 'project schedule model' are 'instances of the project schedule'.
- The project schedule model is maintained by statusing, updating and incorporating authorised changes as work progresses, creating new 'instances' of the schedule information as needed and in accordance with the overall schedule design and specification.
- On completion of the project, archive the project schedule model for future reference.

Key terms:

- **Scheduling Method:** the technique selected for use in developing the schedule model; CPM (ADM or PDM)^h, Critical Chain, etc (this is closely aligned with the selection of appropriate scheduling software).
- **Schedule Model:** the full set of data used to develop the schedule with its inherent logic, durations, resources, calendars, etc. This is closely integrated with the characteristics of the scheduling software selected for the project. The schedule model will be developed and maintained in accordance with the agreed schedule design and specification as the project progresses.
- **Instances of the Project Schedule:** A printed or stored version of the schedule model 'as at' a point in time or stage of development. 'Instances' do not change and should be uniquely named whereas the 'Schedule Model' is expected to be developed, maintained, statused, updated and revised as the project progresses (in accord with the project's change management policies).
- The **Baseline** is a particular 'instance' of the schedule used for comparing the current status with the approved schedule objectives. The baseline should only be changed for variations in scope and then only to the extent necessary to properly adjust the baseline for the scope change.

^g See <http://www.mosaicprojects.com.au/Books.html> for details of *The Practice Standard For Scheduling*.

^h CPM = Critical Path Method, this includes ADM and PDM but not PERT
ADM = Arrow Diagramming Method or 'Activity-on-Arrow',
PDM = Precedence Diagramming Method or 'Activity-on-Node'



Elements of developing a ‘good schedule’

This section offers a general overview of the essential elements that must be considered by the project scheduler and project team when developing a good project schedule.

Developing the scheduling framework

Determine how the Project Schedule Model will be developed

At the outset the project manager, in conjunction with the project team, should determine a development plan for the project schedule model. The key considerations are:

- determining if schedule densityⁱ will be used, or if rolling wave planning^j will be required, or if the project schedule model can be developed in its entirety, and
- determining the stakeholders whose input will be required as part of the schedule development process.

Understanding the full scope of the project

The scheduler should review and understand the project’s scope documents with particular emphasis given to the WBS³. These documents provide the background, information and understanding needed to develop the project schedule. The goal of this process is to ensure that all aspects of the project scope have been adequately defined and included in the project schedule.

Activities in the project schedule model represent the work that produces the deliverables identified in the WBS thus all work elements in the WBS should be directly traceable to a schedule activity or group of activities. Conversely, each activity should only roll up into one WBS element.

Assign Identifiers to the project and schedule

Every project schedule should have a unique name and identification number to identify the project and each version of the project schedule should have a unique version number or ID. This is essential to allow the proper archiving of project documents and audit processes.

Establish project calendars and work periods

The scheduler should determine, in conjunction with the project team, what work periods will be selected for the project. This includes determining for each portion of the project (and potentially each resource):

- The number of working days in a week.
- The number of shifts to be worked each day.
- The number of hours to be worked each shift or day.
- Any periods of scheduled ‘overtime’ work or non-working time (eg holidays)

All of these elements play a major role in determining the number and structure of the project calendars required for the schedule. The use of multiple calendars introduces significant complexity to the calculation of float and the critical path. However, while scheduling is

ⁱ For more on **schedule density** see:

http://www.mosaicprojects.com.au/WhitePapers/WP1016_Schedule_Density.pdf

^j For more on **rolling wave planning** see:

http://www.mosaicprojects.com.au/WhitePapers/WP1060_Rolling_Wave.pdf



A Guide to Scheduling Good Practice

simplified by the use of a single calendar, one calendar may be inadequate for managing the project.

Generally accepted practice is to use a project calendar which is adequate and reasonable to perform the work, based on normal working times. The project calendar then may be used as the primary or default calendar for the project. A limited number of special calendars may then be used for areas of the project (or resources) needing different working times.

Establish the optimum project update cycle

The project management team and the scheduler should determine the appropriate frequency for performing updates and statusing the schedule. The update cycle should account for how management intends to utilise the data generated from the project schedule model, including the timing of review meetings and other management reporting requirements. The key is to select a cycle time that provides management with an optimum level of control information without being overly burdensome on the people doing the reporting and analysing. The optimum update cycle will vary with industry and project intent; between hourly updates for outages to weekly or monthly updates. The chosen update cycle has a direct relationship or bearing on the activity durations contained within the schedule.

Determining the reporting requirements

The primary purpose of the schedule is to communicate information to various stakeholders^k about the planned flow of work, who is responsible for the work, and once the work has started, what is happening and what needs to change.

Selecting the right report formats with the right information for each of the key stakeholders requires an effective coding system (see below), appropriate summary activities and Milestones (designed into the schedule) and the appropriate use of other options such as the WBS, subprojects, schedule levels, etc. The best approach to designing the project reporting structure is:

1. Determine the information needs of the key stakeholders:
 - a. Summary / detailed / comparative / etc.
 - b. Timeframe - full project / short term?
2. Determine the best report type to communicate the information:
 - a. Tabulations / Charts / Network diagrams?
 - b. Summary Milestone or Bar Chart?
 - c. Detailed information for a section of the work / long or short term reports?
 - d. How should the information be sorted for ease of use?
3. Design the schedule data to allow the easy extraction of the reports.
4. Develop a standard 'batch process' for producing the required reports.

Once an effective code structure is in place, it is usually a simple process to reconfigure the selection and sorting processes to manage new requirements.

^k Effective reporting requires the right information to be provided to the right people at the right time. The value of the report is diminished if it contains extraneous information that is 'not needed' by the stakeholder - too much detail or too much information destroys effective communication. For more on **effective communication** see: http://www.mosaicprojects.com.au/PM-Knowledge_Index.html#PPM07



Designing an effective activity coding structure (to support schedule data usage)

A sensible code structure should be developed to facilitate the selecting and sorting of the schedule data to facilitate the development and maintenance of the project schedule, as well as meeting the project reporting requirements. A well designed code structure is also very helpful in analysing project performance data by facilitating aggregation selection and sorting to highlight trends and anomalies.

A structured activity ID/numbering scheme may form part of the overall coding design. Using a structured numbering system may allow the users of the schedule to have a better understanding of how a particular activity fits into the bigger project picture by grasping the significance of the activity number itself. At a minimum an activity number must be unique, and follows a scheme appropriate to the project.

Activities should be coded with more than one code for each activity, each code holding a separate value, thus allowing outputs to be customized for different purposes. For example, codes can be used to identify project phases, sub-phases, location of work, responsible person, the organisation doing the work, the type of work, the summary level, etc.

Determining resource planning requirements

If the schedule is to take resource availability into account, the resource pool available to the project needs to be determined together with any special resource calendars, skill sets and availabilities. Resources used for scheduling purposes may be the same or a subset of the resources used for cost estimating (PMBOK Section 6.4 and 7.2).

Just as activity codes can be used to classify and organise activities, resource codes can be assigned to resources to classify resources according to organization, skill level or type, reporting structure, etc. And, just as Activity IDs should be structured into a meaningful scheme, resource IDs should be similarly structured.

Determine Integration Requirements

The schedule should be integrated into the risk management system to ensure all planned risk mitigation tasks are properly integrated into the schedule (both at the planning stage and at subsequent risk reviews undertaken during the project).

Similarly, an appropriate level of integration with the Cost Management system may be required to facilitate transfer of data (eg, expected resource consumption {Hrs of effort}) however, the usefulness and integrity of the schedule should not be compromised to 'fit in' with a cost control system. A closely aligned requirement may be the need to integrate the schedule with an Earned Value Management system. If this is required the EV system will dictate part of the schedule coding system and the project WBS. The design of the schedule will need to honour these requirements without compromising the other design aspects outlined in this paper and the schedule's overall usefulness to the project team – an effective schedule and time management/performance on a project underpin to a significant extent the overall 'Earned Value' performance of the project¹.

¹ "Schedule is king" – Stephen Gumley, former CEO, Defence Material Organisation.

See *Standardising Quality in Project Scheduling*: www.mosaicprojects.com.au/Resources_Papers_071.html and *Scheduling in a Defence Environment*: www.mosaicprojects.com.au/Resources_Papers_069.html



Developing the Baseline schedule

Define project milestones

Once the scheduler has a feel for the overall scope of the project, he or she can begin to determine the project's major and minor milestones. Milestones should have zero duration and be used as bench marks to measure progress against. They can also reflect the start and finish points for various project events or conditions and/or identify external constraints or interim deliverables. As a minimum each project must have a start milestone and finish milestone.

Design the project's activities (PMBOK process 6.2 – Define Activities)

The scheduler, in conjunction with the project team members responsible to perform the work of the project, should create the list of activities that will need to be performed to complete the project.

The characteristics of a well designed activity include:

- The activity is a discrete element (or block) of work
- A single person should be responsible for performing the activity. This does not preclude the idea that multiple resources may be required to accomplish the activity, but it does require that a single entity is responsible for its performance. That person should be the same one who will report progress on the activity.
- Activities describe work that must be accomplished. For example, 'Pour the wall foundation from x to y', or 'Review Chapter 3 terminology'. Each activity description should be unique and leave no room for interpretation, ie, it can be identified without ambiguity.
- The work represented by an activity should, once started, be capable of proceeding to completion without interruption (except for naturally occurring non-work periods in the calendar). If the work of an activity must be interrupted, it is desirable for the activity to be split into two or more activities at the natural break points.
- The work contained in an activity should be scoped so that the activity's duration will be less than 2 times the update cycle (ideally never more than 3 times the update cycle). This allows the reporting of the start and finish of an activity within the one or two update cycles, allowing management to focus on performance and corrective action if needed.

Exceptions to this general rule are long duration activities that are naturally continuous and have no obvious 'break points' including:

- long, continuous construction activities, such as boring a 2-mile long tunnel or paving several miles of highway, and
- procurement activities; where a single work item (e.g. fabricating and shipping a component to a remote site) can take significantly longer than 3 update cycles, and

When complete, the activity list should describe 100% of the work required to complete the project, although not all activities necessarily need to be fully detailed if rolling wave planning is being used.



A Guide to Scheduling Good Practice

Rolling wave planning entails planning the near-term work in detail and presenting the future work at a summary level until the time for that work draws nearer^m.

Design the project's logic (PMBOK process 6.3 –Sequence Activities)

Connecting the activities and milestones together with ‘sensible’ logic is the bedrock of any dynamic project scheduleⁿ. The method of connection is defined as a relationship. Every activity and milestone except the first and last must be connected to at least one predecessor and one successor. With the exception of the start milestone, something must occur prior to an activity starting, and in turn that activity must be totally or partially completed to allow another activity to start. Ensuring compliance with this practice will prevent the schedule from containing open ends, where activities or milestones are missing predecessors or successors.

For most instances, each activity would finish prior to its successor activity (or activities) starting (known as a finish-to-start (FS) logic tie), but that is not always possible. If it is necessary to overlap activities, the scheduler may elect to use start-to-start (SS), finish-to-finish (FF) or start-to-finish (SF) links. However, whenever possible the FS logic relationship should be used. If it is necessary to use any of the others types of link, the scheduler should use them sparingly and fully understand how the links have been implemented in the scheduling software being used.

A common logic mistake is using a start-to-start link with a lag, when a finish-to-finish with a lag is more appropriate. If Task B can't finish until Task A is complete, then finish-to-finish is the correct relationship. Using a start-to-start means that Task B can start once A starts and the lag period is reached, and that Task B can go on to completion. However, it assumes Task A will be completed per the plan – which may not be valid. Where allowed by your scheduling tool, genuinely overlapping tasks are best described with both a start-to-start and a finish-to-finish so that the interdependency is fully described.

Ideally, all activities will be linked in such a way that the start of every activity is linked to a predecessor and the finish of every activity is linked to a successor.

In addition to logic ties, the scheduler may also assign duration lag(s) which are required times between activity start and finishes. For example if an activity has a FS lag of 5, it would delay the start of the successor activity until 5 days after the predecessor activity has finished. The practitioner is cautioned to use lags with care and understand their impacts^o.

^m **Note:** Effective rolling wave scheduling requires a ‘proper allowance’ to be made for future works based on a clear understanding of the project: this process tests the skill and knowledge of the scheduler and introduces an element of risk; however, this ‘risk’ is not reduced by adding unfounded detail. By way of example, if a software deliverable has not been designed, the accuracy of the schedule allowance for testing is not improved by adding multiple tasks describing multiple tests that may, or may not be required for multiple modules that may, or may not be in the final design of the software. But there is an absolute certainty testing will be required and there is statistically reliable data available for both the time this will take (based on the nature of the software being developed) and the likely level of defects and rework. The role of the scheduler is to know this information and ensure adequate allowances are maintained in the ‘summary tasks’ to allow realistic detail to be added once the actual design is known and the actual testing regime has been decided. Fictitious detail does not improve schedule accuracy; the time to add an appropriate level of detail to the schedule is shortly after adequate information becomes available to allow the project team and the scheduler to determine what work has to be done. There may be one ‘step’ from a summary task to the detailed schedule; alternatively, there may be progressive improvements in the information facilitating two or more steps between the original ‘summary’ schedule and the final working schedule.

ⁿ For more on the benefits of **‘dynamic scheduling’** see: *Dynamic Scheduling*:
http://www.mosaicprojects.com.au/Planning.html#Core_Papers

^o For an in-depth discussion on **Links, Lags & Ladders** see:
http://www.mosaicprojects.com.au/Planning.html#Core_Papers



A Guide to Scheduling Good Practice

Lags should not be used to represent a period of time when work is actually occurring, such as review of document before the next phase proceeds. This type of work should be shown as activities in the project schedule model and where such activities are included; they could be coded to show that these are activities for which another party, for example the client, is responsible.

It is also possible to assign constraints^p to activities and milestones which require the activity or milestone to start or finish at a specific point in time. The scheduler is strongly encouraged to study the various types of constraints that might be used and understand the effect and nuance their use has upon the schedule. The only legitimate use of a constraint is to mirror an impost on the schedule, typical examples include:

- Access to a work area will not be available until the 14th August. A ‘start no earlier’ constraint on the Milestone ‘Take possession of work area’ (or the activity) of Aug. 14th is sensible and reflects reality. The Milestone or activity should still be connected to the preceding work in the schedule because whilst access cannot be obtained prior to the contracted date, it may be delayed until later by the normal flow of work within the project.
- The contract calls for the handover of a deliverable on the 10th November. A ‘finish no later’ constraint on the Milestone ‘Deliver deliverable’ of Nov. 10th is sensible and reflects reality. The Milestone should still be connected to succeeding work (or the project completion) because unless this is the absolute end of the project, the project cannot complete until the milestone is achieved and all of the other project work (eg archiving ‘lessons learned’).

Constraints should be used very sparingly, every ‘fixed date’ in the schedule limits the ability of the overall model to respond dynamically to other changes in logic or to changes in actual progress during status updates; every constraint changes (distorts) the float information calculated by the schedule model.

Nether constraints nor lags should not be used to replace logic.

Determining the duration for each activity (PMBOK process 6.5 – Estimate Activity Durations and 6.4 – Estimate Activity Resources)

The duration is an estimate of how long it will take to accomplish the work involved in the activity. In many cases, considering the anticipated number of resources that are expected to be available to accomplish an activity and the quantity of work, may determine the activity’s duration. An increase or decrease to a driving resource allocated to the activity will have a direct effect on the duration (but this is not a simple ‘straight line’ relationship^q). Other factors influencing the duration are the type or skill level of the resources available to undertake the work, resource calendars and the intrinsic nature of the work. Some activities (e.g. a 24 hour stress test) will take a set amount of time to complete regardless of the resource allocation.

^p A personal preference of the author of this paper is to place all date constraints on Milestones. This is not mandated, but it makes finding and managing constrained dates as the project progresses much easier. The practice also makes the identification of ‘errors’ easier – imposed dates on normal tasks are obviously in the wrong place. Constrained dates impact the calculation of ‘Float’. The only way negative float can be generated in a schedule is by the presence of a fixed date of some form.

^q For more on **duration estimating** see:

http://www.mosaicprojects.com.au/WhitePapers/WP1052_Time_Estimating.pdf
and ‘The Cost of Time - or who’s duration is it anyway?’
www.mosaicprojects.com.au/Resources_Papers_009.html



A Guide to Scheduling Good Practice

While it is feasible to determine a duration for an activity at any time, generally accepted good practice recommends defining the activity first, then tying it logically into the overall schedule sequence and then focusing on how long it will take to accomplish the work. At this time the relationship between the activity and other work in the schedule will be more easily appreciated and resource flows, team sizes and the like can begin to be determined.

Analysing the schedule output (PMBOK process 6.6 – Develop Schedule)

Once complete, the project schedule model will contain a number of unique activities, of varying durations, logically linked together. The schedule is now analysed to calculate the dates and other values of the project schedule. Despite the speed of many computer programs, the scheduling function always requires three distinct processes for ‘Time Analysis’, and a fourth if resource smoothing or levelling is being used.

The discrete steps are:

- A start date is assigned to the start milestone (or the first activity if a milestone has not been used) and then moving throughout the network from activity to activity (from left to right) and in the sequence defined by the logical relationships start and finish dates are calculated for each activity and milestone as determined by their defined durations. This is called the forward pass. The start and finish dates on each activity are called the early dates and when the analysis reaches the end of the network it establishes the earliest possible finish date for the project.
- Next, a finish date is assigned to the end milestone (or last activity). This could be the same date as the one calculated by the forward pass or a different date applied as a constraint. The analysis process then works back through the network, from right to left, until it arrives back at the start milestone and another set of start and finish dates have been calculated for each activity. This is called the backward pass and establishes the late dates for each activity and milestone.
- Float values are calculated by comparing the early and late dates; typically Total Float is calculated by subtracting the early start date from the late finish date and then deducting the duration ($[LFT - EST] - Dur$). Free float is calculated by subtracting the early finish date of the activity from the earliest start date of any of its successors. Free float is never negative^r.
- Once the float values have been calculated resource smoothing and/or levelling may be carried out to minimise resource over allocations or reduce the fluctuations in resource demand. If this process is to be done automatically, the scheduler needs to determine the processes and algorithms to be used. Most project management software packages have multiple options and settings that can have a significant impact on the resulting ‘resource levelled’ schedule. Some practitioners may be tempted to do the resource levelling manually by adjusting the logic or adding constraints to delay the start of certain activities. However, this is not a good practice as it distorts the normal scheduling calculation and can ‘build up’ problems as artificial constraints and logic are added over time.

Approving the schedule

The project team and key stakeholders should be actively involved in reviewing the results of this initial scheduling process. The review should consider the analysed project end date, milestone

^r For more on **Time Analysis calculations** see: http://www.mosaicprojects.com.au/PDF/Schedule_Calculations.pdf



completion dates and resource requirements (compared to resource availability) to determine the acceptability of the schedule. Where alterations are required, variations are made to the schedule logic, resource allocations and/or durations and then the schedule is reanalysed. The most often pursued alteration involves actions to reduce the overall duration of the schedule, the key techniques used to compress the schedule are ‘crashing’ and ‘fast tracking’ (PMBOK 6.6.2.7)

These iterations continue until an acceptable project schedule is developed, one that all of the relevant project stakeholders can agree with.

Baselining the schedule (PMBOK 6.6.3.1)

Once agreed, the final version of the schedule should be captured or copied for future reference. This is called the Project Baseline Schedule. This Baseline becomes the benchmark against which project performance is measured. It is a generally accepted practice that every project should have a Baseline Schedule in place before the execution of the project work commences. Once the Baseline has been agreed, reports are distributed in accordance with the project’s communication plan.

Maintaining the schedule

Status, Update and Change Control (PMBOK process 6.7 – Control Schedule)

Change is inevitable; every project will experience it! The last major component needed to ensure successful project execution is to maintain effective schedule change control. The key is to determine how the project will approve and track schedule change as it occurs throughout the project’s life cycle. Change can occur simply by work progressing more quickly or slowly than planned, as well as when changes in other elements of the project occur (e.g. scope changes) and/or the project team decides to modify their approach to the project work.

The status/update process occurs on a regular basis determined during the project planning process^s. The updated schedule must reliably reflect the actual situation of the project. To achieve this the following seven principles apply:

1. Seek accuracy not precision – excessive detail is less important than a clear overall picture of progress and its real consequences.
2. Updating is a bottom-up process – collect progress information based on the short-term detailed schedule.
3. Face reality – you need to understand the true situation before you can implement effective management actions.
4. Manage information delay and bias^t – people are innately optimistic and loss averse which means they are inclined to want to hide bad news (particularly future predictions that have not occurred at this point in time) and genuinely expect things to get better; they generally do not! It is impossible to eliminate bias but you can minimise its effect.
5. Updating must be comprehensive – a partial update is useless.
6. Manage information bottlenecks – timely information is essential for effective management.
7. Implement the progress measurement best practices outlined below.

^s For more on the **power of regular updates** see **Managing for Success:**
http://www.mosaicprojects.com.au/PDF_Papers/P002_MFS_Full.pdf

^t For more on the **effects of bias** see: http://www.mosaicprojects.com.au/WhitePapers/WP1069_Bias.pdf



The steps involved in maintaining the schedule at each status/update are:

- Collect and record the actual status of the work at a pre-determined date/time for the project. The information collected should include the actual start dates for all activities that have commenced and actual finish dates for all activities that have been completed during the reporting cycle. Where an activity is in progress, the amount of work accomplished, and the time needed to complete the remaining work, should be determined. Other information gathered at this time may include data on resource utilisation and costs incurred. The data is collected ‘as of’ a nominated date/time. This date/time is called the ‘Data Date’ or ‘Time Now’.
- Investigate and minimise the 99% complete syndrome.
 - Ensure progress is reported accurately and quantified. For most activities the ‘time remaining’ is critical information.
 - Do not report ‘complete’ unless the activity is completely finished.
 - Ensure any approvals are obtained before the activity is considered complete.
 - Encourage the completion of ‘almost finished’ activities ahead of starting new ones - finishing matters!
- Enter the status information into the project schedule model and re-analyse the remaining work to determine the project status. All incomplete work will be rescheduled to a date/time after the Data Date. Store the statused schedule.
- Update the schedule with any minor changes needed to reflect the reality of actual work on site. These minor changes only affect work currently in progress or about to start and may include altering logic ties, durations, resource allocations, calendars, etc to align the schedule with reality and to keep it relevant to the management of the work.
- Compare the newly updated project schedule model outputs with the stored baseline and, where necessary, employ actions to lock in gains and/or recover losses (manage schedule variances) these actions will normally involve using the project’s change management processes.
- In parallel, review the overall ‘momentum’ of the project to ensure adequate work is being accomplished overall. Building and maintaining an adequate ‘performance intensity’ is critical to achieving the overall completion of the project (almost regardless of what’s happening on the current critical activities); all non-critical work will eventually become critical if it is delayed beyond its available float. Two options for measuring overall performance are ‘Momentology’^u and ‘Earned Schedule’^v. It is as important to maintain focus on short term objectives and keeping production going ‘today’ as it is on the overall objectives.
- Revise the schedule to incorporate any agreed changes resulting from the overall change control process to ensure the project schedule represents 100% of the current work scope of the project (scope changes may require corresponding alterations to the baseline schedule – see below).

^u For more on **momentology** see: http://www.mosaicprojects.com.au/WhitePapers/WP1036_Momentology.pdf

^v For more on **Earned Schedule** see: <http://www.earnedschedule.com/>



- **The updating and adjustment processes may need a number of iterations to achieve an agreed project schedule that remains realistic and achievable^w.**
- Distribute reports in accordance with the project's communication plan once the updated schedule has been confirmed to be accurate. Each 'saved' version of the schedule model (either in the form of a printed output or an electronic file) is an 'instance' of the schedule model. Each 'instance' should be uniquely named and a record kept for future reference.
- Update the baseline if authorised scope changes have been incorporated into the updated project schedule model – baseline changes should only be to the extent needed to incorporate the changed scope.
- Maintain records that explain all changes in activity durations or logic as the alterations are being made in the schedule. Activity log notes are often used for this purpose. These records will provide valuable data if it becomes necessary to reconstruct what happened and why.

The status/update process is the key element in making the schedule an effective project management tool. It progressively corrects the errors (or incorrect assumptions) embedded in every schedule^x and proactively influences the perceptions and attitudes of the project team^y.

Conclusion

Scheduling is a relatively new management discipline^z that uses computer systems to combine 'art' and 'science' in an attempt to predict the future. The tools and processes (science) used are mathematical models of the intended work needed to complete the project. However, the components of this model are subjective and at best, imprecise estimates of what might happen in the future.

The 'art' of planning is to make this data as useful as possible to help the project team navigate their project through to a useful conclusion. The skill of the scheduler lays in being able to make effective use of the 'science' of scheduling to help the project team develop a shared view of the project 'future' and then to help them make that future a reality whilst recognising the inherent uncertainty implicit in any attempt to accurately predict the future.

^w **Note:** Because of errors in estimating durations and the simplistic nature of schedule relationships (See Footnote 'p' for more on these issues), the 'real critical path' for the project may be on a path that is showing 'some float' in the schedule. These 'near critical paths' should be carefully reviewed as well as the actual critical path when making adjustments. A heuristic suggested by Murray Woolf is to consider all paths with a total float value less than 2.5% of the original project duration as 'near critical' and give them a similar level of consideration as the actual critical path.

^x For more on the factors causing uncertainty around the calculation of the 'critical path' and the overall schedule see '*Float - Is It Real?*' by P. Weaver:
www.mosaicprojects.com.au/Resources_Papers_043.html

^y For more on **updating**, see '*Managing for Success - The power of regular updates*' by P. Weaver:
www.mosaicprojects.com.au/Resources_Papers_002.html

^z See '**A Brief History of Scheduling - Back to the Future**' by P. Weaver:
www.mosaicprojects.com.au/Resources_Papers_042.html



A Guide to Scheduling Good Practice

This paper is a core reference in our PMI-SP and PTMC credential courses.

For more information on these courses see:

http://www.mosaicprojects.com.au/Training-Planning_One-on-One.html

First published 13th Dec. 2007 – extensively augmented and updated.

References

- ¹ A Guide to the Project Management Body of Knowledge (PMBOK® Guide) - Fifth Edition. Project Management Institute.
- ² The Practice Standard For Scheduling. Project Management Institute.
- ³ WBS – Work Breakdown Structure, see Practice Standard for Work Breakdown Structures, ISBN: 1880410818. Project Management Institute.

The papers in this series:

- A Guide to Scheduling Good Practice: http://www.mosaicprojects.com.au/PDF/Good_Scheduling_Practice.pdf
- Attributes of a Scheduler: http://www.mosaicprojects.com.au/PDF/Attributes_of_a_Scheduler.pdf
- Dynamic Scheduling: http://www.mosaicprojects.com.au/PDF/Dynamic_Scheduling.pdf
- Links, Lags & Ladders: http://www.mosaicprojects.com.au/PDF/Links_Lags_Ladders.pdf
- Schedule Float: http://www.mosaicprojects.com.au/PDF/Schedule_Float.pdf
- Schedule Levels: http://www.mosaicprojects.com.au/PDF/Schedule_Levels.pdf
- Schedule Calculations: http://www.mosaicprojects.com.au/PDF/Schedule_Calculations.pdf

Additional information; see Mosaic's Scheduling Home page at: <http://www.mosaicprojects.com.au/Planning.html>

