



30 Project Scheduling Techniques

WikiFree

PROJECT MANAGEMENT COLLECTION

30 Project Scheduling Techniques

PLAN
—
SCHEDULE
—
COORDINATE
—
DELIVER



Created by **WikiFree.com**

1



ACTIVITY DEFINITION



Identify the actual work required to produce each approved deliverable.

1 PURPOSE & VALUE

Break each work package into the actual activities needed to produce the approved deliverables. Activities show what will be done, by whom and when, making the schedule clear and actionable.



CLARITY:

Turns deliverables into specific, manageable work.



CONTROL:

Improves estimates, resource planning and progress tracking.



ACCOUNTABILITY:

Assigns a clear owner for each activity.



ALIGNMENT:

Links activities to the scope, acceptance criteria and WBS.

2 HOW TO APPLY

1



REVIEW SCOPE

Confirm the approved deliverable and acceptance requirements.

2



DECOMPOSE WORK

Break the work package into logical activities.

3



DEFINE EACH ACTIVITY

State a clear start and finish condition, owner and link to the scope.

4



CHECK DEPENDENCIES

Confirm the logical order and link to predecessor and successor activities.

5



REVIEW AND REFINE

Ensure activities are complete, understandable and ready for estimating and sequencing.

3 INPUTS & EVIDENCE



WORK BREAKDOWN STRUCTURE (WBS):

Approved work packages and deliverables.



SCOPE DOCUMENTATION:

Detailed scope, acceptance criteria and assumptions.



STAKEHOLDER INPUT:

Subject matter experts, users and project team members.



CONSTRAINTS & REQUIREMENTS:

Regulatory, technical and business rules to consider.



OWNERS:

Project manager facilitates; activity owners are team members with the required skills and authority.

4 PRACTICAL EXAMPLE

Illustrative: A reporting deliverable.



1. DEFINE FIELDS

Agree data fields, business rules and acceptance criteria.



2. BUILD EXTRACTION

Develop the data extraction and prepare output files.



3. RECONCILE TOTALS

Check totals against the source system and resolve differences.



4. OBTAIN APPROVAL

Provide the report for review and obtain formal approval.

5 PITFALLS & LIMITS



VAGUE TASKS:

High-level or unclear activities hide progress and make ownership difficult.



EXCESSIVE DETAIL:

Too many small activities make the schedule expensive to maintain.



MISSING SCOPE:

Activities not linked to the WBS or acceptance criteria can create gaps.



UNDEFINED BOUNDARIES:

No clear start or finish conditions lead to rework and disputes.



REMEMBER:

The right level of detail balances visibility with cost. Tailor to the project context.

6 ACTIONS & OUTPUTS



REVIEW COMPLETENESS:

Check that all activities are included for each work package and deliverable.



VERIFY LINKAGE:

Confirm each activity links to the WBS and acceptance requirements.



ASSIGN OWNERS:

Nominate a responsible owner for each activity.



PREPARE OUTPUTS:

A complete activity list with clear descriptions, start and finish conditions, owners and scope links.



NEXT STEP:

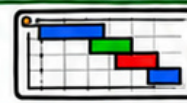
Use the activity list for estimating, sequencing and developing the detailed schedule.



Created by WikiFree.com

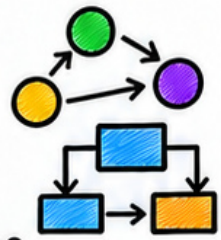
CARD 1 OF 30

30 Project Scheduling Techniques





ACTIVITY SEQUENCING



Arrange activities according to real work logic and interfaces.

① PURPOSE & VALUE

Arranges activities in the order they must be performed, based on real work logic and interfaces.



SHOWS LOGIC:

Defines which activities must come before others.



MANAGES INTERFACES:

Reveals handoffs between teams, work packages and external parties.



SUPPORTS PLANNING:

Provides a realistic sequence for estimating, scheduling and resource allocation.



IMPROVES RELIABILITY:

Reduces rework and waiting time by clarifying dependencies early.

② HOW TO APPLY

1



List all activities in scope with clear descriptions.

2



Identify immediate predecessors and successors for each activity.

3



Record mandatory relationships (must finish before, must start before).

4



Include discretionary relationships where logical (e.g. preferred sequence).

5



Validate external dependencies with relevant stakeholders.



CONTROL:

Do not force dates by inventing logic. Keep relationships realistic.

③ INPUTS & EVIDENCE



SCOPE BASELINE:

Approved scope, work breakdown structure and activity list.



REQUIREMENTS & INTERFACES:

Technical requirements, design interfaces and handoff points.



STAKEHOLDER INPUT:

Delivery owners, specialists and external parties to confirm logic and dependencies.



CONSTRAINTS & ASSUMPTIONS:

Known constraints, assumptions and imposed dependencies (e.g. regulatory approvals).



SUPPORTING DOCUMENTS:

Design documents, procurement plans, test plans and interface agreements.

④ PRACTICAL EXAMPLE

Illustrative project sequence:

Design Approved

Configuration follows approved design.

Configure System

Testing follows a stable configured build.

Prepare Test Data

Test data must be available before testing.

Test System

Testing starts after configuration and test data are ready.

⑤ PITFALLS & LIMITS



INVENTED LOGIC:

Do not create artificial relationships just to force dates or achieve a preferred finish.



OPEN ENDS:

Avoid leaving activities without predecessors or successors. Explain any intentional start or finish.



MISSING DEPENDENCIES:

Check for overlooked interfaces, external approvals or shared resources.



OVERLY COMPLEX:

Do not add unnecessary relationships. Keep the network clear and practical.

⑥ ACTIONS & OUTPUTS



CONFIRM NETWORK:

Review and confirm the activity sequence with delivery owners and key stakeholders.



MAINTAIN CLEAR START AND FINISH:

Ensure a single, logical project start and finish, with all activities connected.



UPDATE AS NEEDED:

Revise the sequence when scope, requirements or external dependencies change.



OUTPUTS:

Validated activity network with predecessors and successors, ready for duration estimating and schedule development.



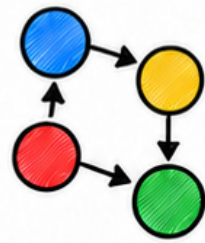
Created by WikiFree.com

CARD 2 OF 30

30 Project Scheduling Techniques



PRECEDENCE DIAGRAMMING



Model activities as nodes connected by dependency relationships.

1 PURPOSE & VALUE

Model project work as activities linked by logical dependencies to show the order in which work can start and finish.



CLARITY:

Shows how work is connected and the sequence of delivery.



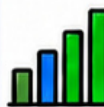
BETTER PLANNING:

Identifies predecessors and successors for each activity.



IMPROVED COORDINATION:

Supports communication across teams and suppliers.

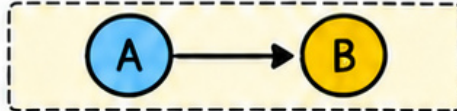


REALISTIC SCHEDULE:

Builds a sound basis for estimating, resource planning and risk analysis.

2 HOW TO APPLY

Create activity nodes and link them using dependency relationships that reflect the real work logic.



Finish-to-Start (FS):

Successor cannot start until predecessor finishes.



Start-to-Start (SS):

Successor cannot start before predecessor starts (zero lag).



Finish-to-Finish (FF):

Successor cannot finish before predecessor finishes (zero lag).



Start-to-Finish (SF):

Successor cannot finish before predecessor starts (zero lag).



CONTROL:

Use a relationship only when the work logic supports it.

3 INPUTS & EVIDENCE



ACTIVITY LIST:

Approved scope with clear activity definitions and IDs.



WORK LOGIC EVIDENCE:

Design information, technical specifications, constructability input and method statements.



STAKEHOLDER INPUT:

Project team, specialists, contractors and suppliers who understand the sequence.



CALENDARS:

Working days, working hours, holidays and any constraints.

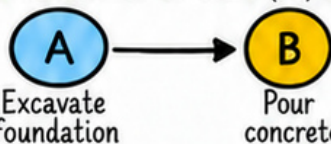


OWNERSHIP:

Schedule planner leads, with input from subject matter experts and activity owners.

4 PRACTICAL EXAMPLE

Illustrative finish-to-start (FS) link:



Activity B cannot start until activity A finishes, subject to any defined lag and the project calendars (working days or calendar dates).



LAG:

A lag can be added when a waiting period is required by the work logic, for example curing.



CALENDARS:

Start and finish dates are calculated using the project calendars, not assumed calendar dates.

5 PITFALLS & LIMITS



NOT RESOURCES:

A relationship describes work logic, not resource availability.



WRONG RELATIONSHIP:

Using an inappropriate relationship can create an unrealistic schedule.



UNCOMMON LINKS:

Start-to-start, finish-to-finish or start-to-finish should only be used when clearly justified and explained.



MISSING CONTEXT:

Links without documented reasoning are hard to validate and maintain.



LIMITATION:

Precedence logic does not consider cost, resource leveling or risk by itself; it is one input to the overall schedule.

6 ACTIONS & OUTPUTS



CHECK EACH LINK:

Confirm the intent, direction (predecessor → successor) and that it reflects real work logic.



REVIEW IMPACT:

Assess how each relationship affects the calculated schedule and the critical path.



ENGAGE OWNERS:

Discuss and validate links with activity owners and specialists.



DOCUMENT REASONING:

Record the basis for each relationship, including any lag, constraints and calendar used.



OUTPUTS:

A precedence diagram network, updated activity attributes and a schedule calculation that reflects the agreed logic.



Created by WikiFree.com

CARD 3 OF 30

30 Project Scheduling Techniques



The All-in-One PM Templates Bundle

940+ Premium Templates • One Payment • Lifetime Access

\$49

SAVE 94%

Worth \$3,300+



EXCEL & GOOGLE SHEETS



PROFESSIONAL & RELIABLE



AUTOMATED DASHBOARDS



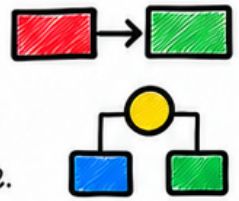
SAVE TIME & STAY ON TRACK



INSTANT ACCESS DOWNLOAD

GET LIFETIME ACCESS

FINISH-TO-START LOGIC



Link activities so the successor can start only after the predecessor is complete.

1 PURPOSE & VALUE

Use a finish-to-start (FS) dependency when a successor activity can begin only after the predecessor activity is fully complete.



REALISTIC ORDER:

Reflects technical, safety, quality or contractual needs.



CLEAR SEQUENCING:

Prevents work starting too early.



REDUCES RISK:

Helps avoid rework, safety incidents and wasted effort.



BETTER PLANNING:

Provides a reliable basis for resources, procurement and site access.

2 HOW TO APPLY

Apply a finish-to-start link with any justified waiting time.

1



Confirm the predecessor's completion evidence.

2



Create an FS dependency from predecessor to successor.

3



Apply any required lag (waiting time) and record the reason.

4



Check that the successor cannot start before the predecessor finishes.

5



Review for redundant links and parallel opportunities.



CONTROL:

Use FS only when a genuine reason exists to wait.

3 INPUTS & EVIDENCE



COMPLETION EVIDENCE:

Predecessor deliverables finished, inspected and formally accepted as complete.



OWNERS:

Activity owner, quality inspector and project manager confirm completion.



REQUIRED INPUTS:

Approved scope, method statements, permits, inspection records and acceptance documents.



SCHEDULE INFORMATION:

Activity IDs, planned durations (workings days), logic relationships and any required lag with rationale.



DEPENDENCIES:

Confirm no other outstanding predecessors are required before the successor can start.

4 PRACTICAL EXAMPLE

Illustrative project with an FS dependency:



Activity B can start only after activity A is complete and accepted.



STEP 1:

Equipment is delivered to site.



STEP 2:

Receipt inspection is completed and accepted.



STEP 3:

Installation starts after the evidence of completion from Step 2.



RESULT:

The FS link ensures installation does not start until delivery and inspection are finished and accepted.

5 PITFALLS & LIMITS



TOO MANY LINKS:

Linking every activity in a long chain can create false constraints and hide parallel opportunities.



UNNECESSARY WAITING:

Adding FS links without a real reason can delay the schedule and increase costs.



MISSED PARALLEL WORK:

Independent activities may be able to run at the same time.



OVER-CONSERVATIVE LOGIC:

Can lead to longer durations, lower resource productivity and reduced flexibility.



DATE CONFUSION:

A finish-to-start link controls sequence, but working days and calendar dates may differ.

6 ACTIONS & OUTPUTS



CHALLENGE THE NEED:

Review each proposed FS link. Remove unnecessary links while keeping genuine dependencies.



PRESERVE ESSENTIAL LINKS:

Keep FS links for safety, quality, technical, contractual or regulatory requirements.



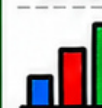
CONFIRM EVIDENCE:

Ensure the predecessor has been completed, inspected and accepted before the successor starts.



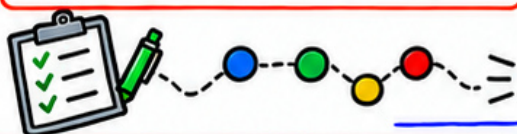
UPDATE THE SCHEDULE:

Add or adjust the FS link and any justified lag, with clear rationale.



OUTPUTS:

A realistic network showing FS dependencies, with documented evidence, owners and reasons for any waiting time.



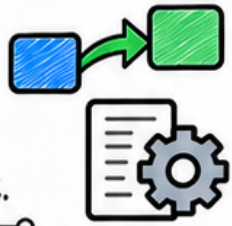
Created by WikiFree.com

CARD 4 OF 30

30 Project Scheduling Techniques



START-TO-START LOGIC



Allow overlapping work when the successor start depends on the predecessor start.

1 PURPOSE & VALUE

Start-to-start (SS) logic allows a successor activity to begin after the predecessor starts, with a required separation.



OVERLAP WORK:

Enables early collaboration and parallel effort.



TIME SEPARATION:

Use a minimum time or progress lag between starts.



MAINTAINS DEPENDENCIES:

Keeps logical sequence while reducing idle time.



USE WITH CARE:

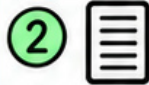
Check completion dependencies separately to ensure realistic finishes.

2 HOW TO APPLY

Define the start-to-start relationship and the required separation, then confirm other dependencies.



1 Identify the predecessor and successor activities and agree the overlap rule.



2 Set the minimum progress or time separation between starts (SS with a lag).



3 Check that both activities can work at the same time and share required resources.



4 Confirm finish relationships separately (FF or FS) if the successor cannot finish early.



5 Document the logic and lag rationale in the schedule.



CONTROL:

A start-to-start link does not control the successor finish. Review completion logic and work continuity.

3 INPUTS & EVIDENCE



ACTIVITY DETAILS:

Clear descriptions of predecessor and successor activities, including scope and expected deliverables.



OVERLAP JUSTIFICATION:

Technical, contractual or operational reason why early start is needed.



EVIDENCE OF READINESS:

Information, designs, approvals or resources available at predecessor start to allow the successor to begin.



KEY OWNERS:

Activity owners, technical experts and resource managers who approve the overlap and monitor progress.



SUPPORTING RECORDS:

Agreed lag, meeting notes, risk assessments and any constraints or assumptions.

4 PRACTICAL EXAMPLE

Illustrative project using a start-to-start link with a documented overlap rule:



A. Research starts (gather initial evidence).



B. Drafting may begin after research starts and initial evidence is available.



Overlap rule (illustrative):

Drafting starts a short time after research starts, once sufficient initial evidence is provided, as agreed by both owners.



Result:

Research and drafting work in parallel, reducing idle time while maintaining the required sequence. Final review and approval are managed separately.

5 PITFALLS & LIMITS



UNREALISTIC COMPLETION:

A start-to-start link alone may allow the successor to finish too early. Add a finish relationship if needed.



INSUFFICIENT READINESS:

Starting the successor without the required information or resources can cause rework.



WORK CONTINUITY:

Overlapping work may not be practical if the successor cannot proceed continuously after the start point.



RESOURCE CONFLICTS:

Both activities may compete for the same people or equipment.



LIMITATION:

Use start-to-start only when the overlap is logical, agreed and supported by evidence.

6 ACTIONS & OUTPUTS



VALIDATE THE OVERLAP:

Agree the need for start-to-start logic, the lag or progress condition, and the responsible owners.



UPDATE AS PROGRESS CHANGES:

Review the separation and readiness evidence regularly and adjust if conditions change.



COORDINATE OWNERS:

Ensure activity owners, technical experts and resource managers understand and support the logic.



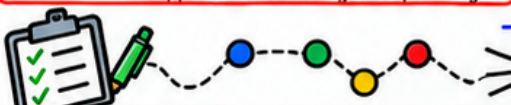
RECORD AND COMMUNICATE:

Document the start-to-start link, lag rationale and any related finish links in the schedule.



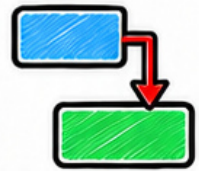
USEFUL OUTPUTS:

Updated schedule with SS link and lag, rationale log, and a clear record of required evidence, owners and review dates.





FINISH-TO-FINISH LOGIC



Coordinate completion where one activity cannot finish before another finishes.

1 PURPOSE & VALUE

Finish-to-finish (FF) logic links the completion of two activities. The successor cannot finish before the predecessor finishes.



COMPLETION DEPENDENCY:

Used when both activities must finish in sequence.



RIGHT PURPOSE:

Coordinates finalisation, close-out and handovers.



VALUE:

Prevents premature completion and supports realistic overlap of activity work.



USE WITH CARE:

Ensure start dates are also logically supported.

2 HOW TO APPLY

FF = Finish-to-Finish
Add a lag only if justified.



1 Identify the predecessor and successor activities and their deliverables.



2 Apply a finish-to-finish relationship (FF).



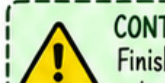
3 Add a lag only if there is a justified waiting period between finishes.



4 Check that each activity's start dates are also logically supported.



5 Validate with the project team and keep a clear rationale.



CONTROL:

Finish-to-finish logic does not automatically control start dates.

3 INPUTS & EVIDENCE



ACTIVITY DETAILS:

Scope, deliverables, acceptance criteria and definitions of 'finish' for both activities.



SUPPORTING EVIDENCE:

Procedures, technical requirements, handover plans and quality checklists.



OWNERS & STAKEHOLDERS:

Activity owners, subject-matter experts and the planning team confirm the dependency and any lag.



VERIFICATION RECORDS:

Inspection reports, test results or sign-off records showing completion is required.



SCHEDULE INPUTS:

Approved activity list, existing dependencies and project calendar (working days or calendar dates).



AUTHORISATION:

Keep the agreed logic, lag (if any) and rationale in the schedule change log.

4 PRACTICAL EXAMPLE

Illustrative project close-out activities:

Predecessor: Verify configuration information (Activity A).

Successor: Final documentation (Activity B). Activity B cannot finish before Activity A finishes.



SCENARIO:

Documentation must include the last verified configuration details, so it can only finish after verification finishes.



ILLUSTRATIVE RESULT:

Activity B is allowed to work in parallel but cannot finish before Activity A finishes.

5 PITFALLS & LIMITS



DOES NOT CONTROL STARTS:

FF logic does not determine when the successor starts.



RISK OF INSUFFICIENT TIME:

The successor may start very late and not have enough working time to complete.



UNJUSTIFIED LAG:

Adding a lag without a valid reason can delay completion unnecessarily.



WRONG DEPENDENCY TYPE:

Do not use FF when a finish-to-start or start-to-finish relationship is more appropriate.



CHECK THE WHOLE NETWORK:

Review other logic to ensure a consistent and realistic sequence.

6 ACTIONS & OUTPUTS



CHECK BOTH START AND FINISH:

Confirm the FF logic and verify that start dates are also reasonably supported.



AGREE AND DOCUMENT:

Record the dependency, any lag and the rationale with relevant stakeholders.



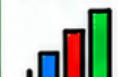
MAINTAIN TRACEABILITY:

Link the dependency to deliverable acceptance criteria, inspection records or sign-off evidence.



UPDATE THE SCHEDULE:

Implement the FF logic, revise dates as needed and check for unintended impacts on the critical path and resources.



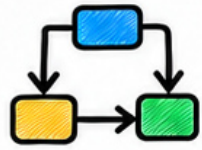
OUTPUTS:

Updated schedule, dependency rationale, change log entry and supporting evidence filed for audit.





LEAD AND LAG ANALYSIS



Make overlaps and waiting periods explicit in scheduling logic.

1 PURPOSE & VALUE

Lead and lag analysis makes overlaps and waiting periods explicit in scheduling logic so that timing is realistic and traceable.



LEAD: Intentional overlap between activities when justified.



LAG: Waiting period between activities when required.



VALUE: Improves realism, highlights dependencies and reduces surprises.



CLARITY: Shows why activities overlap or wait.



BETTER CONTROL: Supports risk management and resource planning.

2 HOW TO APPLY

Use leads for justified overlap and lags for waiting. Document the basis and calendar treatment. Prefer explicit activities for managed work.



1 Identify the predecessor and successor activities and the relationship type (e.g. Finish-to-Start).



2 Decide on a lead (negative offset) for overlap or a lag (positive offset) for waiting.



3 State the calendar used (working days or calendar dates) and apply it consistently.



4 Record the rationale, approval and references.



CONTROL:

Use explicit activities instead of large unmanaged leads or lags where work needs to be controlled.

3 INPUTS & EVIDENCE



SCHEDULE DATA:

Activity list, current network logic, existing leads and lags.



TECHNICAL BASIS:

Design information, method statements, construction methods or other technical requirements.



STAKEHOLDER INPUT:

Input from designers, construction team, procurement and operations.



CONSTRAINTS:

Regulatory or contractual requirements, site access, seasonal limits and calendar constraints.



CALENDAR INFORMATION:

Working days or calendar dates to be used for each offset.



OWNERS:

Activity owners and subject-matter experts to confirm and sign off each lead or lag.

4 PRACTICAL EXAMPLE

Illustrative example showing both a lag and a lead in a simple sequence:



EXAMPLE 1 – LAG (WAITING):

After pouring concrete, a specified curing period is required before starting the next activity. Model this as a Finish-to-Start with a lag equal to the curing period.



EXAMPLE 2 – LEAD (OVERLAP):

Start drafting early using approved portions of the design. Model this as a Finish-to-Start with a lead, allowing the draft to start before full design completion, where justified.



RESULT:

Overlap reduces idle time and the waiting period is visible and controlled in the schedule.

5 PITFALLS & LIMITS



HIDDEN LAGS:

Long or unexplained lags can obscure the real work and hide ownership.



NEGATIVE LAGS:

Can complicate interpretation, risk analysis and reporting if not well justified.



OVERUSE:

Excessive leads or lags can make the schedule hard to understand and maintain.



UNCERTAIN BASIS:

Using leads or lags without clear evidence increases risk and may give a false sense of progress.



LIMITATION:

Leads and lags do not replace detailed planning. They must be justified, documented and reviewed as the project evolves.

6 ACTIONS & OUTPUTS



REVIEW OFFSETS:

Check each lead or lag for necessity, justification and correct calendar treatment.



REPLACE WITH ACTIVITIES:

Where work needs to be managed, replace opaque offsets with explicit activities.



CONFIRM OWNERSHIP:

Agree owners for the predecessor, successor and any overlapping or waiting work.



UPDATE AND MONITOR:

Adjust leads and lags as design and construction information becomes available.



OUTPUTS:

Updated schedule with justified leads and lags, documented rationale and approvals, and a clear record of calendar treatment.



CALENDAR PLANNING



Apply realistic working and nonworking time to activities and resources.



1 PURPOSE & VALUE

Calendar planning applies realistic working and nonworking time to activities and resources so schedule dates reflect how work actually happens.



CONVERT DURATION:

Turn working duration into realistic start and finish dates.



SHOW AVAILABILITY:

Respect working weeks, holidays, shifts and exceptions.



IMPROVE COORDINATION:

Align project, resource and activity calendars.



REVEAL REAL CONSTRAINTS:

Highlight delays caused by nonworking time.



BETTER PLANS:

Produce more achievable and credible schedules.

2 HOW TO APPLY

1



Define the project calendar:
Set working week, holidays, shifts and working hours.

2



Set resource calendars:
Include different patterns and availability.

3



Assign calendars to activities and resources:
Ensure they are aligned.

4



Add exceptions:
Include individual holidays, part-time work or shutdowns.

5



Check the schedule:
Confirm start and finish dates reflect the correct working time.



CONTROL:

Keep project, resource and activity calendars consistent.
Use the same time zone and rules.

3 INPUTS & EVIDENCE



PROJECT CALENDAR:

Working weeks, holidays, shutdown periods, shifts and working hours.
Owner: Project scheduler.



RESOURCE CALENDARS:

Availability, part-time patterns, individual holidays and exceptions.
Owner: Resource manager.



ACTIVITY DETAILS:

Activity durations and any required calendar type. Owner: Project planner.



POLICIES & AGREEMENTS:

Organizational working time rules, labor agreements and site calendars.
Owner: PMO or HR.



EXTERNAL DATES:

Public holidays, local site calendars and supplier shutdowns.
Owner: Procurement or site team.



HISTORICAL RECORDS:

Previous schedules and actual dates for reference. Owner: Project controls.

4 PRACTICAL EXAMPLE

Illustrative example with a five-day task using a five-day calendar (Monday-Friday).



CASE 1 - NO HOLIDAYS:

Start: Monday
Duration: 5 working days
Finish: Friday (same week).



CASE 2 - MIDWEEK HOLIDAY:

Start: Monday
Duration: 5 working days
Holiday: Wednesday (nonworking)
Finish: Next Monday.



INTERPRETATION:

The task still needs 5 working days, but the holiday extends the finish date because no work is done on Wednesday.

5 PITFALLS & LIMITS



WORKING VS ELAPSED TIME:

Activity duration is in working days, while elapsed time includes nonworking days such as weekends and holidays.



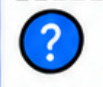
INCONSISTENT CALENDARS:

Different calendars for project, resources and activities can create surprising float or delays.



MISSING EXCEPTIONS:

Forgetting individual holidays, part-time work or site shutdowns gives unrealistic dates.



INCORRECT ASSIGNMENT:

Using the wrong calendar type for an activity or resource.



LIMIT:

Calendar planning depends on accurate working time information. It cannot fix incorrect durations or logic links.

6 ACTIONS & OUTPUTS



DOCUMENT ASSUMPTIONS:

Record working weeks, holidays, shifts, exceptions and time zone. Keep this with the schedule.



VERIFY KEY DATES:

Check start and finish dates around holidays, shutdowns and handoffs with the project team.



ALIGN CALENDARS:

Ensure project, resource and activity calendars are consistent across the schedule.



REVIEW CHANGES:

Update calendars when new holidays, shutdowns or resource availability change, and recheck affected dates.



USEFUL OUTPUTS:

Documented calendar assumptions, updated activity dates, and a schedule that reflects realistic working and nonworking time.



DURATION ESTIMATION



Estimate the working time needed under stated scope, resources and conditions.

1 PURPOSE & VALUE

Duration estimation estimates the working time needed to complete an activity under the agreed scope, resources and conditions.



PLAN REALISTICALLY:

Supports feasible schedules and resource plans.



SET EXPECTATIONS:

Helps identify time needed including review or waiting time.



IMPROVE DECISIONS:

Highlights long tasks and dependencies early.



BRIDGE SCOPE AND TIME:

Links work content, available resources and delivery dates.



WORKING DAYS:

Estimate duration in working days, not calendar dates.

2 HOW TO APPLY

Use evidence, expert input and ranges. Separate effort from duration and account for review or waiting time.

1



Understand the work scope, quality requirements and deliverables.

2



Gather estimates from experienced people and relevant records.

3



Consider a range of outcomes (e.g. optimistic, most likely, pessimistic) and agree a value.

4



Separate effort from duration and include review, approvals or waiting time.

5



Use the agreed value to update the schedule and check for dependencies.



CONTROL:

Use a consistent method and document the basis of the estimate.

3 INPUTS & EVIDENCE



SCOPE & REQUIREMENTS:

Agreed activity scope, acceptance criteria and quality standards.



HISTORICAL DATA:

Actual durations from similar work, lessons learned and organisational records.



EXPERT JUDGEMENT:

Input from people with relevant experience and technical knowledge.



RESOURCE INFORMATION:

Availability, skills, workload and working hours for the required specialists or teams.



CONSTRAINTS & CONDITIONS:

Dependencies, approvals, external suppliers, legal or regulatory requirements, and any expected delays or waiting time.



OWNERS:

Activity owner, subject matter experts, resource manager and scheduler.

4 PRACTICAL EXAMPLE

Illustrative example of effort versus duration for a single specialist:

Task requires 40 effort-hours.
Specialist is only 50% allocated (half time). Assume a working day of 8 hours.

Effort-hours: 40
Allocation: 50% (0.5)
Hours per working day: 8

Duration = $40 / (0.5 \times 8) = 10$ days



INTERPRETATION:

Although the effort is 40 hours, the duration is 10 working days because the specialist is only half allocated.

5 PITFALLS & LIMITS



ADDING PEOPLE:

More people does not always reduce duration due to coordination, communication and handovers.



EFFORT VS DURATION:

Effort measures total work; duration depends on resource availability and waiting time.



IGNORED DEPENDENCIES:

Upstream work, approvals or external suppliers can extend duration.



TOO OPTIMISTIC:

Relying on a single optimistic estimate increases the risk of late delivery.



LIMIT:

Estimates are uncertain and can change when scope, resources or conditions change.

6 ACTIONS & OUTPUTS



RECORD THE BASIS:

Document the estimate, method used, assumptions, inputs and people consulted.



UPDATE THE SCHEDULE:

Enter the duration, check dependencies and confirm that working days are used.



REVIEW AND REVISE:

Update remaining duration as new evidence becomes available or conditions change.



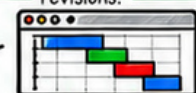
PRODUCE OUTPUTS:

Activity duration estimate, range (if used) and supporting notes for the schedule and risk discussions.



KEEP IT CURRENT:

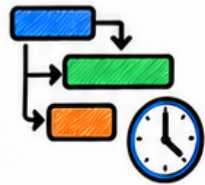
Revisit estimates when scope, resources or constraints change and record the rationale for any revisions.



THREE-POINT DURATION ESTIMATION



Represent duration uncertainty with optimistic, most likely and pessimistic estimates.



1 PURPOSE & VALUE

Incorporates uncertainty by using optimistic, most likely and pessimistic duration estimates for each activity, rather than a single fixed value.



CAPTURE UNCERTAINTY:

Reflects possible variation in activity duration.



BETTER PLANS:

Produces a more realistic schedule than a single estimate.



INFORM RISK:

Supports schedule risk analysis and contingency planning.



WIDER USE:

Useful for activities with uncertain scope, complex work or limited historical data.

2 HOW TO APPLY

PERT EXPECTED DURATION

$$E = (O + 4M + P) / 6$$

1



Define the activity and choose a documented estimation model (e.g. PERT).

2



Obtain optimistic (O), most likely (M) and pessimistic (P) estimates.

3



Calculate the expected duration using the selected formula.

4



Record assumptions and the rationale for each value.

5



Use the estimates in the schedule and a schedule risk analysis.



CONTROL:

Use a consistent model and document all assumptions.

3 INPUTS & EVIDENCE



ACTIVITY INFORMATION:

Clear description, scope, dependencies and acceptance criteria.



HISTORICAL DATA:

Similar past activity durations, lessons learned and organizational databases.



SUBJECT MATTER INPUT:

Estimates from people with relevant experience and knowledge of the work.



RISK INFORMATION:

Known risks, constraints and assumptions that could affect duration.



OWNERS:

Activity owner or estimator responsible for providing and validating the three estimates.

4 PRACTICAL EXAMPLE

Illustrative activity: develop a report.

Optimistic (O): 3 days
Most likely (M): 6 days
Pessimistic (P): 15 days

Expected duration (PERT):

$$E = (O + 4M + P) / 6$$

$$E = (3 + 4 \times 6 + 15) / 6$$

$$E = (3 + 24 + 15) / 6$$

$$E = 42 / 6$$

RESULT: 7 days



INTERPRETATION:

The activity's estimated duration is 7 working days, based on the three estimates and the PERT formula.

5 PITFALLS & LIMITS



IT IS A MEAN:

The expected duration is a mean under the model assumptions, not a guaranteed deadline or a confidence date.



MODEL ASSUMPTIONS:

Assumes the three estimates are reasonable and reflect a plausible range for the activity.



QUALITY OF ESTIMATES:

Biased or unrealistic estimates reduce accuracy and can mislead the schedule.



NOT A SUBSTITUTE:

Does not remove the need for schedule risk analysis or management reserves.



CONTEXT MATTERS:

Use appropriate for uncertain activities; may be less useful for very short or well-defined work.

6 ACTIONS & OUTPUTS



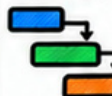
DEVELOP ESTIMATES:

Collect and agree O, M and P values with the activity owner and document the assumptions.



CALCULATE AND RECORD:

Compute the expected duration using the chosen model and record the method and rationale.



USE IN SCHEDULE:

Enter the expected duration in the schedule, using working days, not calendar dates.



ANALYZE RISK:

Use the three-point estimates in a schedule risk analysis to assess the impact of uncertainty.



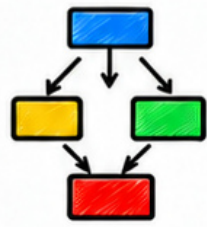
OUTPUTS:

Documented three-point estimates, expected duration, assumptions and updated schedule with supporting records.





CRITICAL PATH METHOD



Calculate the longest logical path that determines the modeled project finish.

1 PURPOSE & VALUE

Identifies the longest sequence of dependent activities (critical path) that determines the project finish. Shows which activities drive the duration and which have slack.



FOCUS:

Find the activities that control the finish date.



PRIORITIZE:

Highlights where delay will impact the project.



IMPROVE:

Supports better sequencing and resource decisions.



COMMUNICATE:

Provides a clear view of driving and near-critical work.

2 HOW TO APPLY



1

Build a valid activity list with logical dependencies and realistic durations.



2

Calculate early dates (ES, EF) and late dates (LS, LF) using forward and backward passes.



3

Identify the longest driving path. Zero total float marks it in a simple unconstrained network; imposed dates can change float.



4

Look for near-critical paths with small float.



5

Review after each update to see if the critical path has changed.



CONTROL:

Use working days as defined in the project calendar. Ensure logic is valid.

3 INPUTS & EVIDENCE



ACTIVITY LIST:

Approved scope of work with clear activity descriptions.



LOGICAL DEPENDENCIES:

Finish-to-Start, Start-to-Start, Finish-to-Finish or Start-to-Finish, with agreed rules.



DURATIONS:

Estimated working days for each activity, with basis and assumptions.



OWNERS:

Project planner prepares the model; activity owners provide durations and confirm dependencies.



SUPPORTING EVIDENCE:

WBS, design documents, contracts, resource plans and calendar details.

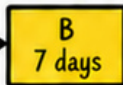


APPROVAL:

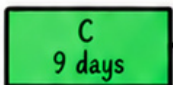
Validated schedule baseline with stakeholder sign-off.

4 PRACTICAL EXAMPLE

Illustrative network with two parallel paths (using working days and simple logic):



Path 1
total = 12 days



Path 2
total = 9 days

Path 1 (A → B) is the critical path because it is the longest path at 12 days.

Path 2 (C) has 3 days of slack under these simple assumptions.



RESULT:

Project duration = 12 working days (driven by Path 1).
Activity C has 3 days of total float.

5 PITFALLS & LIMITS



CHANGING CRITICALITY:

Constraints, calendars and resource limits can change which path is critical.



SHORT TASKS:

The shortest activities can still be on the critical path if there is no slack in the sequence.



RESOURCE EFFECTS:

Resource conflicts may delay an activity and create a new critical path.



CALENDAR DIFFERENCES:

Working days depend on the agreed calendar. Calendar dates can give a different finish date.



MODEL LIMITS:

The result is only as good as the activity logic, durations and assumptions used.

6 ACTIONS & OUTPUTS



INSPECT THE PATH:

After each status update, check the actual critical path and see if it has changed.



TRACK DRIVING ACTIVITIES:

Monitor progress, risks and issues on the critical path and near-critical paths.



TAKE ACTION:

Address delays with corrective actions such as resequencing, adding resources or reducing scope.



UPDATE & REFORECAST:

Revise the model as actuals change and maintain the logic and calendar.



KEY OUTPUTS:

Current critical path, list of driving activities, near-critical paths and revised finish date with explanation.



Created by WikiFree.com

CARD 11 OF 30

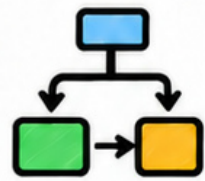
30 Project Scheduling Techniques



FORWARD PASS



Calculate earliest feasible starts and finishes through the dependency network.



1 PURPOSE & VALUE

Forward pass calculates the earliest feasible start (ES) and finish (EF) for each activity by moving through the network from left to right.



ESTABLISH EARLY DATES:

Finds the soonest each activity can start and finish based on dependencies and durations.



SUPPORTS PLANNING:

Identifies the earliest project completion date and highlights activities with no float later.



BUILDS THE BASELINE:

Provides early dates used in later calculations such as the backward pass and floats.



WORKS WITH LOGIC:

Uses the agreed network logic, e.g. finish-to-start, with a consistent time convention.

2 HOW TO APPLY

Simple FS links, zero lag, one calendar.

The simplified formula does not cover other relationships or imposed date constraints.

$ES = \text{maximum predecessor EF}$

$EF = ES + \text{Duration}$



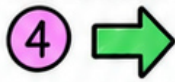
Start with activities that have no predecessors. Set ES to 0.



For each activity, set ES to the maximum EF of all predecessors.



Calculate EF by adding the activity duration to the ES, using one time convention throughout.



Move through the network from left to right, repeating for all activities.



CONTROL:

Use a consistent time convention (working days or calendar dates).

3 INPUTS & EVIDENCE



ACTIVITY LIST & DURATIONS:

Approved activities with estimated durations and time units (working days or calendar dates).



PRECEDENCE LOGIC:

Documented relationships (e.g. finish-to-start), including any lags or leads.



CALENDAR INFORMATION:

Project or resource calendars, working days, holidays and non-working periods.



CONSTRAINTS & ASSUMPTIONS:

Any fixed dates, start no earlier than, finish no later than, and time conventions used.

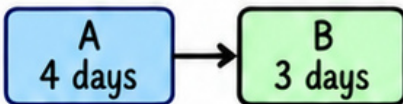


OWNERS & SOURCES:

Schedule planner (owner), activity owners, subject matter experts, and approved project documents.

4 PRACTICAL EXAMPLE

Illustrative network using finish-to-start logic and elapsed time (e.g. working days):



Activity A:

ES = 0 (no predecessors)

EF = 0 + 4 = 4

Activity B:

ES = 4 (EF of A)

EF = 4 + 3 = 7

RESULT: A EF = 4, B EF = 7



INTERPRETATION:

B cannot start before A finishes. The earliest finish for B is day 7.

5 PITFALLS & LIMITS



TIME CONVENTION MIX-UP:

Do not mix working days and calendar dates. Use one consistent convention.



INCLUSIVE DAY COUNTING:

Be clear whether the finish day is included or excluded, and apply the same rule throughout.



MERGE LOGIC ERRORS:

When an activity has multiple predecessors, use the maximum EF, not the earliest.



CONSTRAINTS AND LAGS:

Ignoring start/finish constraints or lags can lead to unrealistic early dates.



LIMITATION:

Forward pass gives only early dates. It does not confirm float or the critical path.

6 ACTIONS & OUTPUTS



CHECK NETWORK LOGIC:

Confirm all predecessors, successors, merges and lags before calculating early dates.



CONFIRM CALENDARS:

Ensure the correct project or resource calendars are applied and non-working days are handled consistently.



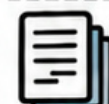
APPLY THE FORWARD PASS:

Calculate ES and EF for every activity using the agreed durations and one time convention.



REVIEW AND VALIDATE:

Check early dates at key merges, and test for constraints or unusual gaps in the sequence.



OUTPUTS:

Early start (ES) and early finish (EF) dates for all activities, used for the project schedule, further analysis and stakeholder review.



Created by WikiFree.com

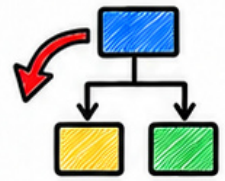
CARD 12 OF 30

30 Project Scheduling Techniques





BACKWARD PASS



Calculate latest starts and finishes consistent with a selected finish boundary.

1 PURPOSE & VALUE

Backward pass calculates the latest start (LS) and latest finish (LF) for each activity, working backwards from a selected finish boundary.



CONFIRMS FEASIBILITY:

Shows the latest dates that still meet the finish requirement.



SUPPORTS CONTROL:

Helps identify time flexibility and remaining float.



INFORMS DECISIONS:

Highlights activities that drive the finish date.



COMPLEMENTS FORWARD PASS:

Together they define the scheduling window for each activity.

2 HOW TO APPLY

Under simple finish-to-start logic:

$$LF = \text{minimum successor LS}$$

$$LS = LF - \text{duration}$$

1



Set the finish boundary for the project or phase and assign LF to the final activity.

2



Work backwards through each predecessor.

3



For each activity, set LF to the minimum successor LS.

4



Calculate LS by subtracting the activity duration from LF.



CONTROL:

Use the correct logic links and durations. Apply the same working day calendar used for the forward pass.

3 INPUTS & EVIDENCE



SCHEDULE NETWORK:

Approved activities, logic links and durations.



FINISH CONSTRAINT:

The selected finish date for the project or phase, and the working day calendar to be used.



ACTIVITY DETAILS:

Assumptions, calendars, and any activity constraints (for example must-finish-by).



OWNERS AND SOURCES:

Project scheduler prepares the analysis with input from activity owners and the project manager.

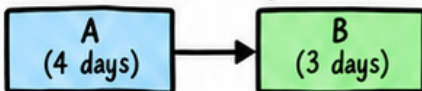


EVIDENCE TO RETAIN:

Schedule file, calculation notes and an activity date report showing LF and LS for each activity.

4 PRACTICAL EXAMPLE

Illustrative two-activity sequence with finish-to-start logic:



If activity B must finish at day 10:

$$B: \quad LF = 10$$

$$LS = 10 - 3 = 7$$

A must finish by day 7 to allow B to start:

$$A: \quad LF = 7$$

$$LS = 7 - 4 = 3$$



INTERPRETATION:

A must start by day 3 and finish by day 7 for B to finish by day 10. Dates are working days, using the project calendar.

5 PITFALLS & LIMITS



UNREALISTIC FINISH DATE:

An imposed finish earlier than the feasible forecast can create negative float.



INCORRECT LOGIC:

Missing or wrong links give misleading latest dates.



INCONSISTENT CALENDAR:

Using different working day calendars can produce wrong results.



CONSTRAINTS OVERRIDE:

Imposed dates may limit LS and LF, reducing or removing float.



LIMITATION:

Backward pass does not predict durations or resolve resource constraints. It only calculates latest dates.

6 ACTIONS & OUTPUTS



DOCUMENT RESULT:

Record LF and LS for all activities in the schedule file and share a date report.



INVESTIGATE ISSUES:

Check activities with negative float or very tight latest dates and review the finish constraint.



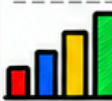
CONFIRM WITH OWNERS:

Discuss latest dates with activity owners and agree any necessary changes.



UPDATE AND ITERATE:

Adjust logic, constraints or durations as needed and rerun the backward pass.



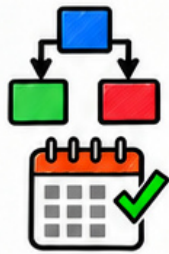
USEFUL OUTPUTS:

Activity report showing LF and LS, variance to targets, and activities needing attention. Share with the project team for action.





TOTAL FLOAT ANALYSIS



Measure schedule flexibility relative to the modeled completion constraint.

① PURPOSE & VALUE

Total float shows how much an activity can slip without delaying the modeled completion date, considering all logic, calendars and constraints.



SHOWS FLEXIBILITY:

Identifies non-critical activities with available time.



SUPPORTS DECISIONS:

Helps prioritize work, resource allocation and risk response.



HIGHLIGHTS RISK:

Low or zero float signals limited flexibility and higher risk of delaying the finish.



GUIDES TRADE-OFFS:

Use float to sequence work, resolve resource conflicts and optimize the schedule.

② HOW TO APPLY

$$\text{Total Float} = \text{LS} - \text{ES}$$

or

$$\text{Total Float} = \text{LF} - \text{EF}$$

1



Build a correct CPM network with logic relationships.

2



Confirm working calendars, holidays and data date.

3



Run the schedule to calculate early and late dates.

4



Check for constraints that may reduce float.

5



Review results and validate key activities.



CONTROL:

Float depends on correct logic, calendars and constraints.

③ INPUTS & EVIDENCE



SCHEDULE MODEL:

Approved CPM schedule with activities, logic, durations, calendars and constraints.



ACTIVITY DETAILS:

Activity IDs, descriptions, durations, calendars and dependencies.



OWNER INFORMATION:

Activity owners and responsible teams to coordinate float usage.



DATA DATE & STATUS:

Current status date, actual progress and remaining work.



CHANGE RECORDS:

Approved changes, assumptions and any imposed constraints (e.g., must-finish dates).



REPORTING RESULTS:

Calculated early/late dates and total float for all activities.

④ PRACTICAL EXAMPLE

Illustrative activity from a CPM model (using working days):

Activity A:

Early Start (ES): 4 days

Late Start (LS): 7 days

$$\text{Total Float} = \text{LS} - \text{ES}$$

$$= 7 - 4$$

RESULT: 3 days



INTERPRETATION:

Activity A can slip by 3 working days without delaying the modeled project completion date, assuming all logic, calendars and constraints remain unchanged.



NOTE:

This is an illustrative example. Actual float values depend on the specific project schedule.

⑤ PITFALLS & LIMITS



FLOAT IS SHARED:

Float is shared along the path and is not a reserve independently available to every activity.



INCOMPLETE MODEL:

Missing logic, incorrect calendars or unmodeled constraints can produce misleading float values.



CONSTRAINTS:

Imposed dates can reduce or eliminate float, even if logic suggests flexibility.



FOCUSED VIEW:

Total float does not show the cause of risk. An activity may have float but still be important due to resource or contractual constraints.



LIMITATION:

Float is relative to the current modeled completion constraint and can change after updates.

⑥ ACTIONS & OUTPUTS



MONITOR FLOAT:

Track float over time and watch for erosion due to delays, changes or emerging risks.



COORDINATE USAGE:

Discuss and agree how float will be used across affected activity owners and teams.



TAKE ACTION:

Re-sequence work, resolve resource conflicts or adjust plans to protect the completion date.



UPDATE SCHEDULE:

Incorporate agreed changes and recalculate total float.



KEY OUTPUTS:

Activity float report, list of low or zero float activities, actions agreed with owners, and an updated schedule with revised float values.



Created by WikiFree.com

CARD 14 OF 30

30 Project Scheduling Techniques



FREE FLOAT ANALYSIS



Measure the delay possible without moving the earliest start of an immediate successor.



① PURPOSE & VALUE

Free float shows how much an activity can be delayed without moving the earliest start of an immediate successor, under simple finish-to-start logic with no lag.



MEASURES:

Schedule flexibility at the activity level.



VALUE:

Helps identify non-critical activities with local delay room.



USE FOR:

Planning resequencing, resource leveling and managing short-term disruptions.



FOCUS:

Immediate successor only, not the whole network.

② HOW TO APPLY

Free Float =
minimum successor ES - activity EF.
Simple FS links, no lag, common calendar.



Confirm finish-to-start relationship with no lag between the activity and its immediate successor.



Read the activity's early finish (EF) from the forward pass.



Read ES for every immediate successor and choose the minimum.



Subtract activity EF from the minimum successor ES.



IMPORTANT:

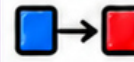
This formula applies to simple FS links with no lag.

③ INPUTS & EVIDENCE



SCHEDULE MODEL:

Activity list with logic relationships, activity durations and any lags.



LOGIC EVIDENCE:

Confirm the activity has an immediate successor and the relationship type and lag.



SCHEDULING DATA:

Early start (ES) and early finish (EF) dates from the current schedule (working days).



OWNERS:

Scheduler provides data, activity owners confirm logic and practical feasibility of delays.

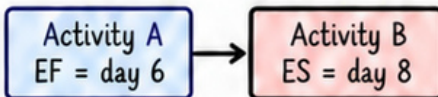


DEPENDENCIES:

Review upstream and downstream links to understand wider impacts.

④ PRACTICAL EXAMPLE

Illustrative example using working days:



$$\begin{aligned}\text{Free Float (FF)} &= \text{ES (B)} - \text{EF (A)} \\ &= 8 - 6 \\ &= 2 \text{ days}\end{aligned}$$

RESULT: 2 days



INTERPRETATION:

Activity A can be delayed by 2 working days without moving the earliest start of Activity B (assuming FS logic with no lag).

⑤ PITFALLS & LIMITS



NOT TOTAL FLOAT:

Free float is different from total float. It only considers the immediate successor.



RELATIONSHIP TYPES:

Formula applies to finish-to-start with no lag. Other relationship types or lags require adjusted calculations.



CALENDARS:

Use the schedule's working days and calendar. Calendar differences or non-working days can affect ES and EF.



WIDER IMPACTS:

Check all immediate successors, not just one: the most restrictive link governs free float.

⑥ ACTIONS & OUTPUTS



IDENTIFY OPPORTUNITIES:

Find activities with free float to evaluate local resequencing or resource leveling options.



TEST CHANGES:

Consider delaying within the free float and check that the successor's earliest start does not move.



CHECK WIDER NETWORK:

Review effects on other successors, resource availability and project milestones.



OUTPUTS:

List of activities with free float, recommended sequencing or resource changes, and updated schedule for review.



Created by WikiFree.com

CARD 15 OF 30

30 Project Scheduling Techniques



NEAR-CRITICAL PATH REVIEW



Monitor paths with little remaining flexibility that may become driving.



① PURPOSE & VALUE

Near-critical path review monitors paths with little remaining flexibility that may become the driving path due to delays, risks or resource constraints.



EARLY WARNING:

Highlights paths that could overtake the current critical path.



FOCUSED ATTENTION:

Directs effort to the most vulnerable activities.



PROACTIVE MANAGEMENT:

Enables timely actions to prevent schedule slippage.



BETTER DECISIONS:

Supports prioritization of resources and risk responses across multiple paths.

② HOW TO APPLY



Set a context-specific float threshold to define near-critical paths.



Identify all paths with total float less than or equal to the threshold.



Examine remaining duration, own risks and shared resources on those paths.



Assess the likelihood of delays and the potential impact on the project end date.



Compare and update regularly as new information emerges.



CONTROL:

Use a float threshold suited to the project. Review and update regularly.

③ INPUTS & EVIDENCE



SCHEDULE DATA:

Activity list, durations, logic dependencies and current total float for each path.



RISK INFORMATION:

Known risks, assumptions and uncertainties affecting activities on the paths.



RESOURCE INFORMATION:

Resource assignments, potential resource conflicts and availability.



STATUS UPDATES:

Actual progress, forecast completions and any recent changes.



KEY OWNERS:

Scheduler (analysis), project manager (decision), activity owners (input) and resource manager (feasibility).

④ PRACTICAL EXAMPLE

Illustrative parallel path with one start, one finish, FS links, no lags and one working calendar.

Path A: 12 working days, zero total float, current critical path.

Path B: 10 working days, 2 days of total float, near-critical.



A supplier delay adds 3 working days to Path B.

RESULT:

Path B becomes 13 days and is now the critical path. Project duration becomes 13 days, one day later than before.



INTERPRETATION:

A three-day delay consumes two days of float and delays completion by one day.

⑤ PITFALLS & LIMITS



NO UNIVERSAL THRESHOLD:

There is no standard near-critical float value. Use a level appropriate to the project's size, risk and complexity.



SINGLE CRITICAL PATH VIEW:

Focusing only on the current critical path can hide emerging risk on other near-critical paths.



CHANGING CONDITIONS:

Path status can change quickly due to delays, re-sequencing or resource constraints.



MISLEADING CONFIDENCE:

A path with small float can still become critical even if current duration is longer.



LIMITATION:

This technique complements, not replaces, full schedule risk analysis. It does not predict outcomes.

⑥ ACTIONS & OUTPUTS



MONITOR PATHS:

Add near-critical paths to regular tracking with increased review frequency.



ASSIGN ACTIONS:

Allocate owners to monitor risks, dependency changes and resource constraints on each path.



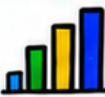
DEVELOP RESPONSES:

Prepare mitigation or acceleration options for high-risk activities on near-critical paths.



OUTPUTS:

List of near-critical paths, identified risks, assigned owners and agreed actions, included in the schedule update and risk log.



FOLLOW UP:

Track the effectiveness of actions and update the near-critical assessment at each data date.





RESOURCE LEVELING



Resolve resource overallocations by moving work within or beyond available float.

1 PURPOSE & VALUE

Resource leveling resolves resource overallocations by moving work within or beyond available float.



REMOVE CONFLICTS:

Avoid assigning more work than a resource can do.



KEEP SCHEDULE LOGIC:

Adjust start and finish dates without breaking dependencies.



USE FLOAT:

Move non-critical activities within available float first.



PROTECT PRIORITIES:

Apply agreed priorities and constraints.



RESULT:

A feasible schedule that respects resource limits.

2 HOW TO APPLY



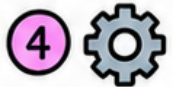
Set realistic resource availability for each role (e.g., working days per period).



Apply documented priorities, constraints and leveling rules.



Resolve overallocations by moving work within float where possible.



If no float is available, move work beyond float and accept the impact.



Review the resulting dates and check for changes to critical paths.



CONTROL:

Respect dependencies and verify that resource limits are met.

3 INPUTS & EVIDENCE



ACTIVITY DATA:

Approved schedule, activity durations, logic and float values.



RESOURCE INFORMATION:

Resource calendars, availability and skills for each role.



PRIORITIES & CONSTRAINTS:

Business priorities, fixed dates and resource-specific constraints.



OWNERS:

Resource owners and functional managers to confirm availability and approve changes.



EVIDENCE OF OVERALLOCATION:

Resource usage or allocation views showing periods where demand exceeds availability.



APPROVAL RECORDS:

Agreed leveling rules, decisions and updated schedule baseline.

4 PRACTICAL EXAMPLE

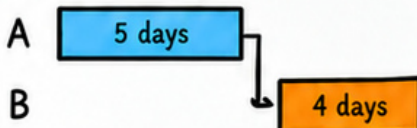
Illustrative example with one specialist available for one activity at a time:

Activity A: 5 working days

Activity B: 4 working days

Both planned to start on the same day using the same specialist.

To remove the overallocation, B is moved to start after A finishes.



Result: B now finishes at elapsed day 9 instead of day 4, a delay of 5 working days. If these are the only project activities, the project duration increases from 5 to 9 working days.

5 PITFALLS & LIMITS



CAN CHANGE CRITICAL PATH:

Moving work can create a new critical path and extend the project finish date.



SOFTWARE DEFAULTS:

Automatic leveling options may not reflect your business priorities or constraints.



MAY INCREASE DURATION:

If no float is available, the project finish date can move later.



QUALITY RISK:

Excessive leveling can create unrealistic handoffs and reduce overall schedule quality.



REVIEW REQUIRED:

Always review the leveled plan, resource usage and the critical path for unintended effects.

6 ACTIONS & OUTPUTS



RUN AND REVIEW:

Apply leveling with agreed priorities, then review dates, resource usage and critical paths.



ENGAGE STAKEHOLDERS:

Review the leveled plan with resource owners and functional managers to confirm feasibility.



COMPARE SCHEDULES:

Preserve the unleveled version and compare it with the leveled plan to understand the impact.



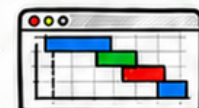
AGREE AND APPROVE:

Document decisions, any changes to priorities and the updated schedule baseline.



OUTPUTS:

Leveled schedule, resource usage profile, change log and records of approvals and assumptions.





RESOURCE SMOOTHING



Adjust work within available schedule flexibility to reduce resource peaks.

① PURPOSE & VALUE

Resource smoothing adjusts work within available schedule flexibility to reduce peaks in demand for key people, equipment or facilities, without changing the project completion constraint.



BALANCE DEMAND:

Use available float to move or split work where valid.



IMPROVE FEASIBILITY:

Reduce overloads and make better use of limited resources.



KEEP SCOPE & FINISH:

Do not change the project scope or required finish date.



MAINTAIN LOGIC:

Respect all dependencies and technical constraints.

② HOW TO APPLY



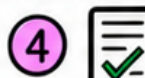
1 Identify resource peaks in the schedule.



2 Check total float for activities causing the peak.



3 Move activities within their float or split work into smaller parts where valid.



4 Keep all logic, constraints and the project completion date unchanged.



5 Recalculate the schedule and review the resource profile.



CONTROL:

Only adjust noncritical work within its available float.

③ INPUTS & EVIDENCE



SCHEDULE MODEL:

Activity list with durations, dependencies, constraints and total float.



RESOURCE INFORMATION:

Resource calendars, availability, skills and current assignments.



DEMAND EVIDENCE:

Resource histograms or reports showing periods of overload.



PROJECT REQUIREMENTS:

Fixed finish date, key milestones and any immovable constraints.



OWNER:

Scheduler with input from resource manager and project manager.



APPROVALS:

Change log with rationale for moved or split work.

④ PRACTICAL EXAMPLE

Illustrative example, one specialist with Monday-Friday working calendar.



Before smoothing:

Design occupies Monday-Friday of Week 2. Review needs three working days, originally Wednesday-Friday of Week 2. Review has two working days of total float.



Action:

Move Review two working days later, to Friday of Week 2 through Tuesday of Week 3.



Result:

The overlap falls from three days to one day. The resource peak is reduced but not eliminated. The project finish remains unchanged if this move fits the validated float and logic.

⑤ PITFALLS & LIMITS



CANNOT REMOVE ALL PEAKS:

Smoothing may reduce but not completely eliminate overloads.



PRODUCTIVITY RISK:

Splitting work into smaller parts may reduce efficiency due to setup time or context switching.



NOT FOR CRITICAL WORK:

Activities with zero total float cannot be moved without affecting the project finish date.



RESPECT CONSTRAINTS:

Do not violate dependencies, minimum lag times or fixed dates.



CHECK THE IMPACT:

Always recalculate and confirm that the project completion constraint is unchanged.

⑥ ACTIONS & OUTPUTS



ADJUST THE SCHEDULE:

Move or split noncritical activities within their float and update resource assignments.



VERIFY FEASIBILITY:

Check the revised resource profile to confirm demand is within available capacity.



CONFIRM NO CHANGES:

Ensure no dependency was broken and the project finish date and scope are unchanged.



ENGAGE STAKEHOLDERS:

Review changes with the resource manager and project manager and obtain approval.

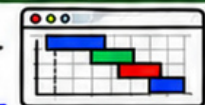


OUTPUTS:

Updated schedule, resource profile and a change log showing moved or split activities with rationale.



Created by WikiFree.com





CRITICAL CHAIN SCHEDULING



Plan around resource constraints and manage uncertainty with buffers.

1 PURPOSE & VALUE

Schedules work around both logical dependencies and resource constraints, and manage uncertainty with buffers.



FEASIBLE SCHEDULE:

Reflects real resource availability, not just logic.



GREATER RELIABILITY:

Buffers protect the completion commitment from variability.



BETTER FOCUS:

Reduces multitasking and promotes steady progress.



CLEARER PRIORITIES:

Highlights the activities that determine the finish date.

2 HOW TO APPLY

FIND THE RESOURCE-CONSTRAINED CHAIN AND USE BUFFERS



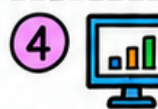
1 List activities, dependencies and resource requirements (e.g. specialist skills).



2 Identify the resource-constrained chain (critical chain).



3 Add a documented buffer approach (project and, where relevant, feeding buffers).



4 Execute and monitor buffer consumption against progress.



CONTROL:

Adjust actions based on buffer use, not individual task variances.

3 INPUTS & EVIDENCE



ACTIVITY INFORMATION:

Scope, logical dependencies, estimates and resource needs (working days, not calendar dates).



RESOURCE DATA:

Availability of key resources (specialist skills, equipment) and calendars.



BASIS AND ASSUMPTIONS:

How activity durations and buffer sizes were determined.



PROGRESS EVIDENCE:

Actual start/finish data, resource usage and buffer consumption records.



OWNERS:

Project manager (method and monitoring) and resource owners (provide availability and updates).

4 PRACTICAL EXAMPLE

Illustrative project where a shared specialist determines the chain:



Design (general team)



Specialist analysis (shared specialist)



Build



Test

Critical chain driven by the specialist resource.



Project buffer -->



RESULT:

The project buffer protects the completion commitment from variability in the chain.

5 PITFALLS & LIMITS



BUFFER SIZING:

Method-specific and project-dependent. Do not use a fixed percentage as a universal rule.



NOT HIDDEN PADDING:

Buffers are visible management tools, not a place to hide unrealistic task estimates.



MULTITASKING:

Splits key resources and increases variability, reducing the effectiveness of buffers.



MISLEADING INDICATORS:

Do not rely only on individual task variances. Focus on buffer consumption and overall flow.



LIMITATION:

Effectiveness depends on accurate logic, realistic resource data and disciplined execution.

6 ACTIONS & OUTPUTS



LIMIT MULTITASKING:

Keep key resources focused on the chain. Remove lower priority work.



MONITOR BUFFERS:

Track project (and feeding) buffer consumption against completed work.



TAKE ACTION:

Investigate meaningful buffer use, remove blockers and reallocate resources if needed.



UPDATE FORECAST:

Use current progress and buffer trends to revise the expected finish (working days).



OUTPUTS:

Resource-constrained schedule, documented buffers, buffer monitoring log and recovery



Created by WikiFree.com

CARD 19 OF 30

30 Project Scheduling Techniques



SCHEDULE BUFFER MANAGEMENT



Monitor how planned protection against uncertainty is being consumed.

1 PURPOSE & VALUE

Schedule buffer management tracks how planned protection against uncertainty is being used, helping the team respond before it is exhausted.



PROTECT DELIVERY:

Absorbs delays from risk and variability.



EARLY WARNING:

Shows when more attention or corrective action is needed.



FOCUSED RESPONSE:

Highlights where to investigate and recover.



BETTER DECISIONS:

Supports informed trade-offs while there is still time.



SCOPE:

Applies to project buffers and feeding buffers within the agreed scheduling approach.

2 HOW TO APPLY

$$\text{CONSUMPTION \%} = \frac{(\text{ORIGINAL BUFFER} - \text{REMAINING BUFFER})}{\text{ORIGINAL BUFFER}} \times 100$$

1



Define buffer purpose, size, ownership and measurement method.

2



Track actual progress and remaining work against the plan.

3



Calculate buffer consumption using the agreed method.

4



Compare consumption with remaining work and current risk exposure.

5



Review results with relevant stakeholders and agree actions if needed.



CONTROL:

There is no universal threshold. Use project-specific limits.

3 INPUTS & EVIDENCE



SCHEDULE BASELINE:

Planned activities, critical path and defined buffers.



CHARTER & PLANS:

Buffer purpose, size, ownership and measurement method.



PROGRESS EVIDENCE:

Actual start/finish dates, current status and updated forecasts.



RISK INFORMATION:

Current risks, issue log and assumptions that may affect buffer use.



KEY OWNERS:

Project manager, scheduler, work package owners and risk owner.



TOOLS & RECORDS:

Schedule model, change log, status reports and recovery action records.

4 PRACTICAL EXAMPLE

Illustrative project buffer at a key review point:

Original buffer: 10 days
Consumed buffer: 5 days
Remaining buffer: 5 days

$$\begin{aligned} \text{CONSUMPTION \%} &= (10 - 5) / 10 \times 100 \\ &= 5 / 10 \times 100 \\ &= 50\% \end{aligned}$$

RESULT: 50% consumed



INTERPRETATION:

Five days of the buffer have been used while substantial critical work remains. This indicates a need for review and possible action.



NOTE:

This example is illustrative. Thresholds and responses are project-specific.

5 PITFALLS & LIMITS



CONSUMPTION ≠ FAILURE:

Buffer use alone does not prove the project will be late.



LACK OF CONTEXT:

Ignoring remaining work, risk exposure and the nature of delays can lead to wrong conclusions.



TOO LATE TO ACT:

Waiting until the buffer is nearly or fully consumed reduces recovery options.



WRONG FOCUS:

Looking only at the percentage without understanding the drivers can mislead decisions.



ONE SIZE DOES NOT FIT ALL:

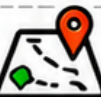
Buffer size, thresholds and responses depend on the project, its risks and the delivery approach.

6 ACTIONS & OUTPUTS



INVESTIGATE DRIVERS:

Identify the causes of buffer consumption, such as delays, productivity issues or emerging risks.



ASSESS IMPACT:

Review remaining critical work, current risk exposure and the likelihood of further consumption.



AGREE RECOVERY ACTIONS:

Decide on actions such as re-sequencing work, adding resources or addressing risks.



UPDATE PLANS:

Revise the schedule and risk register as needed and confirm new buffer outlook.



OUTPUTS:

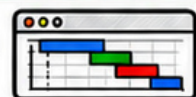
Updated schedule, agreed actions, change log entry and clear communication to stakeholders.



Created by WikiFree.com

CARD 20 OF 30

30 Project Scheduling Techniques



FAST TRACKING



Shorten project duration by overlapping activities originally planned in sequence.



1 PURPOSE & VALUE

Fast tracking starts a successor activity before its predecessor is fully complete, overlapping work to shorten the project duration.



REDUCE DURATION:

Overlap activities that were originally in sequence.



MEET PRIORITIES:

Helps when earlier delivery is needed for business value or external deadlines.



USE JUDGEMENT:

Apply only where technically feasible and the additional risk of rework is acceptable.



SAME SCOPE:

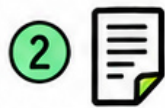
Does not reduce the total scope of work, it changes how the work is scheduled.

2 HOW TO APPLY



IDENTIFY CANDIDATES:

Find sequential activities that can be overlapped technically.



CONFIRM FEASIBILITY:

Decide what partial inputs are available and assess rework risk and coordination needs.



AGREE & AUTHORIZE:

Obtain approval from relevant stakeholders to change the schedule.



UPDATE THE PLAN:

Revise dependencies, reflect overlaps and communicate changes.



MONITOR:

Track progress and manage issues arising from the overlap.

3 INPUTS & EVIDENCE



SCHEDULE BASELINE:

Current activity sequence, dependencies and critical path to identify overlap options.



TECHNICAL INFORMATION:

Design documents, specifications and any partial inputs available early.



RISK ASSESSMENT:

Analysis of rework risk and coordination requirements.



STAKEHOLDER AGREEMENT:

Approval from the sponsor, project manager and key functional owners.



OWNERS/ROLES:

Scheduler analyses options, technical leads confirm feasibility, and the project manager authorizes the change.

4 PRACTICAL EXAMPLE

Illustrative example of overlapping work:



A training material drafting activity starts from the approved stable design sections.



The remaining design work continues in parallel.



This overlap shortens the overall duration compared with waiting for the full design to be completed.



INTERPRETATION:

Training can make progress earlier, but draft content may need to be revised if later design changes affect the materials.

5 PITFALLS & LIMITS



REWORK RISK:

Parallel work can create rework if earlier outputs change later.



NOT ALWAYS FASTER:

If the activities are not on the critical path, overlapping them may not shorten the overall project duration.



MORE COORDINATION:

Requires closer communication and integration between teams.



LIMIT:

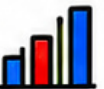
Only use fast tracking where the added risk, cost and complexity are acceptable for the expected time savings.

6 ACTIONS & OUTPUTS



UPDATE SCHEDULE:

Record the new dependencies and overlapping dates.



TRACK PROGRESS:

Monitor both activities to ensure the expected time savings are being realized.



CHECK IMPACTS:

Look for rework, quality issues or resource conflicts and take action early.



COMMUNICATE:

Keep stakeholders informed of changes and any emerging risks.



OUTPUTS:

Revised project schedule, updated risk log and change records, with clear documentation of the approved fast tracking.



Created by WikiFree.com

CARD 21 OF 30

30 Project Scheduling Techniques





SCHEDULE CRASHING



Shorten selected activity durations by adding cost or resources where feasible.

① PURPOSE & VALUE

Schedule crashing deliberately reduces the duration of selected activities, usually on the critical or driving path, by adding cost or resources where feasible.



MAIN PURPOSE:

Deliver the project sooner when time is critical.



ADDED VALUE:

Provides a structured way to buy time and evaluate the cost of each time saving.



TYPICAL USE:

Applied to activities on the current critical path.



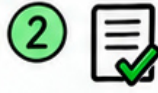
KEY POINT:

Only compress activities that are feasible and technically practical.

② HOW TO APPLY



1 Identify the current critical path and activities that can be compressed.



2 List feasible options to shorten each activity (e.g. extra resources, overtime, alternative method).



3 Estimate the incremental cost for each option and the expected time reduction.



4 Compare options and reassess the schedule, as the critical path may change after crashing.



5 Select the best combination that meets the time need and is authorized.



CONTROL:

Use realistic estimates and check impacts on the whole schedule.

③ INPUTS & EVIDENCE



SCHEDULE DATA:

Approved schedule with durations, predecessors and the current critical path.



COST AND RESOURCE DATA:

Resource rates, overtime costs and alternative methods for compression.



INPUTS FROM OWNERS:

Activity owners and functional managers provide feasibility, resource availability and cost estimates.



CONSTRAINTS & DEPENDENCIES:

Contractual dates, resource limits and technical constraints that affect what can be crashed.



REQUIRED EVIDENCE:

Documented options, estimated costs and time reductions, and updated schedule results.

④ PRACTICAL EXAMPLE

Illustrative option for an activity on the critical path:

Normal duration:	10 days
Option duration:	7 days
Time saved:	3 days
Additional cost:	\$6,000

RESULT: \$2,000 per day

Cost slope = $\$6,000 / 3 \text{ days}$
= \$2,000 per day of compression



INTERPRETATION:

This option shortens the activity by 3 days at an extra cost of \$6,000, giving \$2,000 per day of compression.

⑤ PITFALLS & LIMITS



NONLINEAR COSTS:

Cost slopes may not be linear. The next increment of time reduction can cost more.



COORDINATION LOSS:

Adding people can create communication and coordination problems, reducing productivity.



NEW CRITICAL PATH:

After crashing, a different path may become critical, so reassess the schedule.



TECHNICAL FEASIBILITY:

Not all activities can be crashed due to technical, contractual or quality constraints.

⑥ ACTIONS & OUTPUTS



CHOOSE OPTIONS:

Select authorized compression options with the best overall trade-off between time and cost.



UPDATE THE SCHEDULE:

Apply the agreed changes and recalculate the critical path.



MONITOR OUTCOMES:

Track actual duration, cost and impacts on other activities.



KEY OUTPUTS:

Updated schedule, revised critical path, documented cost and time changes, and records of decisions.



ONGOING REVIEW:

Continue to assess further crashing opportunities if more time reduction is required.



Created by WikiFree.com

CARD 22 OF 30

30 Project Scheduling Techniques





ROLLING WAVE SCHEDULING



Detail near-term work while retaining credible future planning packages.

① PURPOSE & VALUE

Rolling wave scheduling progressively details near-term work while keeping credible future planning packages for later refinement.



FOCUSED DETAIL:

Produces executable activities for the near term.



CREDIBLE FUTURE:

Keeps later work as linked planning packages.



CONTINUOUS ALIGNMENT:

Maintains interfaces and milestones across waves.



SUPPORTS UNCERTAINTY:

Handles evolving scope in complex or innovative projects.



IMPROVES CONFIDENCE:

Increases schedule reliability as detail becomes available.

② HOW TO APPLY

Follow a staged planning approach that elaborates work over time.

1



Set an elaboration horizon for detailed scheduling (e.g., next month).

2



Keep future work as planning packages with high-level scope.

3



Maintain interfaces, dependencies and key milestones across waves.

4



At each checkpoint, convert the next package into detailed activities.

5



Progressively refine durations and logic as more information emerges.



CONTROL:

Unplanned distant work is not the same as rolling wave planning.

③ INPUTS & EVIDENCE



PROJECT SCOPE:

Approved scope, objectives and constraints.



PLANNING PACKAGE LIST:

Future work packages with preliminary scope and assumptions.



DEPENDENCIES & MILESTONES:

Known interfaces, handoffs and milestone dates.



ESTIMATES & ASSUMPTIONS:

High-level estimates for future packages with documented assumptions.



OWNERS & ROLES:

Scheduler, work-package owners and subject-matter experts.



APPROVALS & CHANGE LOG:

Governance for updates and records of refinements.

④ PRACTICAL EXAMPLE

Illustrative project with a one-month detail horizon. Later work remains as planning packages.



NEAR-TERM DETAIL:

Activities for the next month are fully defined with durations, logic and resources.



FUTURE PACKAGES:

Later releases remain as linked planning packages pending discovery and further detail.



MAINTAIN LINKS:

Dependencies and milestones connect detailed work with future packages.



RESULT:

The schedule is executable now, while later work stays credible and aligned.

⑤ PITFALLS & LIMITS



NOT A BLANK SPACE:

Unplanned distant work is not the same as rolling wave planning.



OVERLY VAGUE PACKAGES:

Packages without clear scope or assumptions reduce credibility.



BROKEN INTERFACES:

Ignoring dependencies can create unrealistic schedules.



FALSE CERTAINTY:

Do not assign detailed dates too far in the future.



ONGOING EFFORT:

Requires regular checkpoints and disciplined refinement.

⑥ ACTIONS & OUTPUTS



SET HORIZON:

Agree the near-term elaboration period with the project team.



ASSIGN OWNERS:

Nominate a work-package owner for each future package.



PLAN CHECKPOINTS:

Set dates to convert each package into executable activities.



REFINE AND UPDATE:

Turn the next package into detailed activities, updating durations and logic.



OUTPUTS:

Updated schedule, revised planning package list, change log and visible milestones across waves.



Created by WikiFree.com

CARD 23 OF 30

30 Project Scheduling Techniques





MILESTONE SCHEDULING



Use outcome-based checkpoints to communicate major progress and decisions.

1 PURPOSE & VALUE

Milestones are zero-duration events that mark the completion of key outcomes, approvals or decisions. They show progress, synchronize teams and highlight what must be achieved next.



CLEAR FOCUS:

Highlight major outcomes and decision points.



BETTER COMMUNICATION:

Provide simple, shared checkpoints for all stakeholders.



ALIGNMENT:

Link work packages to business objectives.



EARLY ISSUE DETECTION:

Reveal delays before they affect critical outcomes.

2 HOW TO APPLY

Define and schedule milestones using zero-duration events with clear acceptance evidence, owners and logical predecessors.

1



Identify key outcomes, approvals and decisions for the project.

2



Define each milestone as a zero-duration event in the schedule.

3



Set logical predecessors that must be complete first.

4



Define acceptance evidence and assign an owner for each milestone.

5



Add the milestones to the timeline and review with stakeholders.



CONTROL:

Milestones have zero duration. Do not add activity work to them.

3 INPUTS & EVIDENCE



SCOPE BASELINE:

List of milestone outcomes and decisions required.



ACCEPTANCE EVIDENCE:

Documents, test results, approvals or sign-offs that confirm completion.



OWNERS:

Responsible person or team for delivering the evidence and approving the milestone.



LOGICAL PREDECESSORS:

Work packages or earlier milestones that must be complete first.



RELATED PLANS:

Schedule, quality, procurement and transition plans that define milestone requirements.



STAKEHOLDER AGREEMENT:

Confirmed list of milestones, criteria and owners.

4 PRACTICAL EXAMPLE

Illustrative project where operational readiness depends on multiple milestones:



The operational readiness milestone occurs only after training, access, support and acceptance evidence are complete. It is a zero-duration event that confirms the outcome.

5 PITFALLS & LIMITS



NOT A DETAILED PLAN:

A milestone chart cannot replace the detailed network needed to assess feasibility.



MISSING EVIDENCE:

Undefined acceptance criteria lead to premature or disputed completion.



UNCLEAR OWNERSHIP:

No assigned owner can cause delays and confusion.



OVERUSE:

Too many milestones reduce clarity and make tracking harder.



DEPENDENCY RISKS:

Incorrect predecessor logic can show milestones too early or too late.

6 ACTIONS & OUTPUTS



DEFINE & VALIDATE:

Agree milestone list, acceptance evidence, owners and logical predecessors.



BASELINE DATES:

Set and communicate the planned dates for each milestone.



FORECAST DATES:

Regularly update with forecast dates based on current progress.



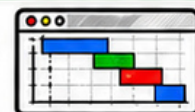
ANALYZE VARIANCE:

Compare forecast dates with baseline dates and explain any variances.



OUTPUTS:

Baseline and forecast milestone dates, variance explanations, and an updated milestone list shared with stakeholders.





LINE OF BALANCE



Coordinate repetitive work across locations or units using production rates.

1 PURPOSE & VALUE

Line of Balance (LOB) schedules repetitive work across locations (such as floors, blocks or units) using production rates. It shows when each crew starts and finishes in each unit to achieve a smooth flow.



COORDINATE FLOW:

Align crews and trades across repeated units.



REDUCE DELAYS:

Prevent crew interference and idle time.



IMPROVE VISIBILITY:

Show the sequence and overlap of activities.



SUPPORT DECISIONS:

Help adjust rates, crews and start dates.

2 HOW TO APPLY

1



DEFINE UNITS:

Choose the repeated locations (e.g. floors, blocks or units).

2



SET PRODUCTION RATES:

Determine crew output rate for each activity per unit.

3



SET HANDOFFS:

Define logical finish-to-start links between crews.

4



BUILD LOB CHART:

Plot planned production lines for each activity.

5



CHECK INTERFERENCE:

Ensure crews do not overtake each other and maintain required separation.



CONTROL:

Maintain realistic rates and handoffs. Revise the plan if lines cross or crews interfere.

3 INPUTS & EVIDENCE



SCOPE INFORMATION:

Drawings, typical unit layouts and list of repetitive activities.



PRODUCTION DATA:

Crew compositions, work methods and expected production rates for each activity.



SEQUENCE & DEPENDENCIES:

Required order, handoffs and minimum separation between crews.



OWNERS & RESPONSIBILITIES:

Construction manager, planners, site supervisors and trade contractors to provide data and agree the plan.



EVIDENCE OF PROGRESS:

Site records, completed units and actual start/finish dates to confirm rates and update the schedule.



APPROVALS:

Agreed rates, handoffs and any constraints or work fronts that affect sequencing.

4 PRACTICAL EXAMPLE

Illustrative project with two crews working in sequence on repeated floors:

Crew A (Activity 1):

Completes one floor every two days.

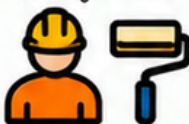
Crew B (Activity 2):

Requires three days per floor and follows Crew A.

Crew A
(2 days/floor)



Crew B
(3 days/floor)



RESULT:

Crew B is slower and falls progressively behind Crew A; the separation grows across successive floors. Check whether the slower crew threatens the required finish.

5 PITFALLS & LIMITS



ASSUMING IDENTICAL UNITS:

Repetitive units may still differ. Variation can hide complexity differences between units.



INVALID RATES:

Using unrealistic or unverified production rates gives misleading lines and can cause crew conflicts.



UNAVAILABLE WORK FRONTS:

Access, space constraints or dependent work may limit when crews can start in each unit.



A faster following crew can catch the preceding crew. Preserve safe separation and revise starts or production rates.



LIMITATION:

LOB is best for regular, repetitive work. It may not suit unique or highly variable activities.

6 ACTIONS & OUTPUTS



VALIDATE DATA:

Agree realistic production rates with site teams and confirm unit sequence and handoffs.



SET BUFFERS:

Include appropriate separation between crews to avoid interference at interfaces.



UPDATE WITH ACTUALS:

Compare planned and actual production, then revise the LOB chart and future unit start dates.



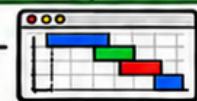
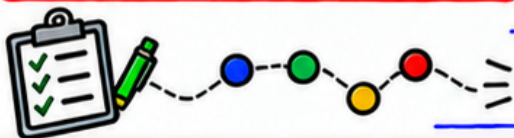
OUTPUTS:

Line of Balance chart showing planned production lines, crew sequence, start and finish dates for each unit.



USE RESULTS:

Coordinate crews on site, resolve conflicts early and communicate changes to all stakeholders.





TIMEBOXING

Limit a work period and prioritize a feasible outcome within it.



1 PURPOSE & VALUE

Timeboxing sets a fixed period for work and focuses the team on delivering a feasible outcome within that time.



SET FOCUS:

Creates a clear time boundary and a specific goal.



PRIORITIZE: Delivers the most valuable work within the time limit.



REDUCE RISK: Provides early evidence and learning.



IMPROVE DECISIONS:

Helps decide what to continue, change or stop based on real results.

2 HOW TO APPLY

1



Set the timebox duration and a clear, achievable goal.

2



Define entry conditions for starting the timebox.

3



Agree the review criteria and the type of evidence needed.

4



Plan and sequence the highest-priority work within the timebox.

5



Manage scope actively, using the agreed priorities.



CONTROL:

Keep scope within the agreed priorities. Do not add new work without re-planning.

3 INPUTS & EVIDENCE



GOAL AND SCOPE:

Clear, agreed outcome for the timebox and the initial scope with priorities.



KEY PEOPLE:

Product owner, delivery team and a decision-maker for the review.



ENTRY CONDITIONS:

Required information, access, dependencies and any approvals to start.



EVIDENCE TO COLLECT:

Working prototype or deliverable, test results, feedback and a brief summary of what was learned.



BASELINE REFERENCE:

Current plan, backlog items and relevant constraints to inform what can be attempted.

4 PRACTICAL EXAMPLE

Illustrative project at the monthly cutoff:

A two-week prototype timebox tests the highest-risk workflow and ends with evidence and a review decision.



TIMEBOX PLAN:

Duration: Two weeks

Goal: Working prototype of the highest-risk workflow

Entry conditions: Team available, test data prepared, required access granted

Review criteria: Working prototype demonstrated, test results and key feedback collected.



OUTCOME:

The team produces a prototype, shares the evidence and the decision-maker reviews whether to continue, change or stop.

5 PITFALLS & LIMITS



SCOPE OVERLOAD:

Not all desired work will fit within the timebox. Focus on the highest priorities.



QUALITY:

A timebox does not remove normal quality obligations. Ensure essential standards and testing are applied.



UNREALISTIC GOALS:

A goal that is too large or unclear reduces effectiveness.



DEPENDENCIES:

External dependencies can limit what can be achieved. Address key blockers early.



MISINTERPRETATION:

A timebox is not a license to rush work or ignore risks.

6 ACTIONS & OUTPUTS



REVIEW EVIDENCE:

Assess the delivered prototype, test results and feedback against the review criteria.



DECIDE NEXT STEPS:

Choose to continue, change or stop based on what was learned.



UPDATE PLANS:

Incorporate outcomes into the project plan and revisit priorities for subsequent timeboxes.



COMMUNICATE:

Share results and decisions with stakeholders and the wider team.



KEY OUTPUTS:

Working prototype or deliverable, summary of results and a clear decision with updated priorities.





LOOKAHEAD PLANNING



Translate the current schedule into actionable near-term commitments.

1 PURPOSE & VALUE

Lookahead planning reviews upcoming work in detail to confirm readiness and remove constraints before it starts.



TRANSLATE PLAN:

Turn scheduled work into clear, actionable commitments.



IMPROVE READINESS:

Ensure prerequisites, resources and handoffs are in place.



REDUCE DELAYS:

Find and remove blockers early, avoiding rework and idle time.



ADD VALUE:

Create a reliable near-term plan that the team can execute with confidence.

2 HOW TO APPLY

1



Review the next few weeks of scheduled work in detail.

2



Check constraints and readiness for each activity (access, permits, materials, resources and handoffs).

3



Identify and assign owners to remove each blocker before work starts.

4



Confirm interfaces and coordination with other trades and work packages.

5



Update the near-term plan with only work that is ready to start.



CONTROL:

Remove or mitigate constraints before the planned start date.

3 INPUTS & EVIDENCE



SCHEDULE INFORMATION:

Approved baseline schedule, recent updates and logic links for upcoming work.



READINESS EVIDENCE:

Permits, access approvals, drawings, materials on site and resource availability.



CONSTRAINT LOG:

Identified blockers, status, required actions and target removal dates.



KEY OWNERS:

Project manager, construction manager, discipline leads and relevant subcontractors.



COORDINATION RECORDS:

Interface meetings, handoff agreements and cross-trade dependencies.

4 PRACTICAL EXAMPLE

Illustrative three-week lookahead:



WEEK 1:

Review upcoming installation work and check readiness.



WEEK 2:

Identify a constraint: missing access approval for installation.



WEEK 3:

Assign the facilities manager to obtain the access approval before the planned start.



RESULT:

The access approval is received before the installation start date, and the work proceeds as planned.



ILLUSTRATIVE:

This example shows how a lookahead finds a blocker and assigns its resolution. Details will vary by project.

5 PITFALLS & LIMITS



DISCONNECTED PLAN:

A lookahead not linked to the main schedule can create conflicting priorities.



INCOMPLETE INFORMATION:

Missing or outdated data can hide constraints.



NO OWNER:

Identified blockers without a clear owner are unlikely to be resolved.



OVERLY OPTIMISTIC:

Including work that is not ready leads to new delays and loss of trust.



LIMITATION:

A lookahead focuses on the near term and does not replace the full project schedule.

6 ACTIONS & OUTPUTS



REMOVE CONSTRAINTS:

Complete required actions and confirm readiness before work starts.



UPDATE NEAR-TERM PLAN:

Include only activities that are ready, with confirmed resources and handoffs.



FEED INTEGRATED FORECAST:

Record confirmed changes in the main schedule and forecast.



COMMUNICATE:

Share the updated plan with the project team and affected trades.



OUTPUTS:

Ready-to-start activity list, updated constraint log, agreed actions with owners, and revised forecast dates.



Created by WikiFree.com

CARD 27 OF 30

30 Project Scheduling Techniques



SCHEDULE RISK SIMULATION



Assess completion uncertainty using repeated sampling of a valid schedule model.



1 PURPOSE & VALUE

Assess completion uncertainty by running many simulations of a valid schedule model with uncertain durations and risks.



QUANTIFIES UNCERTAINTY:

Shows a range of possible completion dates, not a single estimate.



INFORMS DECISIONS:

Supports realistic commitments, reserves and risk responses.



INTEGRATES RISKS:

Includes activity duration ranges, discrete risks and dependencies.



IMPROVES CONFIDENCE:

Highlights key drivers and the likelihood of meeting dates under modeled assumptions.

2 HOW TO APPLY



Build a valid, logic-linked schedule model with durations and dependencies.



Assign duration ranges to activities and define discrete risks with probabilities and impacts.



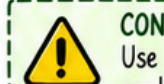
Run many simulations using random sampling (e.g., Monte Carlo).



Inspect the distribution of finish dates and the sensitivity of key drivers.



Derive confidence dates (e.g., P80) and review assumptions and results.



CONTROL:

Use a valid schedule model with credible ranges, risks and logic.

3 INPUTS & EVIDENCE



SCHEDULE MODEL:

Activity logic, calendars, durations and constraints. Owner: Scheduler.



DURATION RANGES:

Estimates with minimum, most likely and maximum. Owner: Subject matter experts.



RISK REGISTER:

Discrete risks with probabilities and time impacts. Owner: Risk manager and project team.



DEPENDENCIES:

Logical relationships and calendars (including resource and external). Owner: Scheduler.



HISTORICAL DATA:

Similar project performance and expert judgment. Owner: Project controls and SMEs.



ASSUMPTIONS LOG:

Modeling choices, distributions, correlations and exclusions. Owner: Project team.

4 PRACTICAL EXAMPLE

Illustrative project at the monthly cutoff:



A risk simulation is run using the schedule model, with activity duration ranges, discrete risks and dependencies. The results show a distribution of possible finish dates.

A **P80 forecast** is the date at or before which 80 percent of simulated finishes occur under modeled assumptions.



INTERPRETATION:

There is an 80 percent chance the project will finish on or before the P80 date, based on the model and its assumptions.

5 PITFALLS & LIMITS



NOT A GUARANTEE:

A P80 date is not an 80 percent guarantee of finish in the real world.



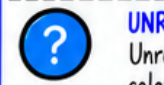
INVALID LOGIC:

Missing or incorrect dependencies distort the results.



OMITTED CORRELATION:

Ignoring correlated risks or duration links can understate or overstate uncertainty.



UNREALISTIC INPUTS:

Unreliable ranges, risks or calendars lead to misleading outcomes.



MODEL LIMITATIONS:

Results depend on modeling assumptions and may not capture all real-world factors.

6 ACTIONS & OUTPUTS



REVIEW RESULTS:

Use finish date distribution, sensitivity and key drivers to understand the risk profile.



INFORM COMMITMENTS:

Use confidence dates (e.g., P80) to support realistic stakeholder discussions.



PLAN RESERVES:

Set schedule contingencies based on the risk profile and project context.



PRIORITIZE RISK RESPONSES:

Focus on high-impact drivers identified by the simulation.



UPDATE AND RE-RUN:

Incorporate new information, mitigation actions and changes to assumptions, and rerun as needed.



Created by [WikiFree.com](https://www.wikifree.com)

CARD 28 OF 30

30 Project Scheduling Techniques





BASELINE AND FORECAST COMPARISON



Compare the approved plan with current expected dates while preserving history.

1 PURPOSE & VALUE

Compares the approved baseline with current expected dates to show schedule movement while keeping a clear record of history.



VISIBILITY:

Highlights slippages or improvements against the plan.



ACCOUNTABILITY:

Keeps the original baseline intact for fair comparison.



BETTER DECISIONS:

Shows where action is needed and supports realistic recovery planning.



TRANSPARENCY:

Separates execution performance from approved scope changes.

2 HOW TO APPLY

1



Keep the approved baseline dates fixed unless changed through formal change control.

2



Update actual start and finish dates as work progresses.

3



Update remaining durations using current information.

4



Calculate current forecast dates for key activities and milestones.

5



Compare forecast dates with baseline dates and analyse differences.



CONTROL:

Do not change the baseline to reflect execution slippages. Use the change control process.

3 INPUTS & EVIDENCE



BASELINE SCHEDULE:

Approved activity and milestone dates, with scope and assumptions.



CURRENT STATUS DATA:

Actual start and finish dates, percent complete and remaining durations from the team.



FORECAST UPDATES:

Revised expected dates based on current information and credible estimates.



CHANGE RECORDS:

Approved scope changes with effective dates and rationale.



OWNERS & STAKEHOLDERS:

Scheduler, project manager, work package owners and the change control board.



SUPPORTING EVIDENCE:

Progress reports, meeting notes and any relevant risk or issue records.

4 PRACTICAL EXAMPLE

Illustrative milestone comparison:

Baseline date: 1 June

Forecast date: 8 June

Movement = 8 June - 1 June

RESULT: 7 calendar days late



CAUSE:

Delay in receiving a key supplier deliverable, which pushed downstream work.



INTERPRETATION:

The forecast is seven calendar days later than the approved plan. The baseline remains unchanged unless an authorized scope change is approved.

5 PITFALLS & LIMITS



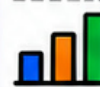
CHANGING THE BASELINE:

Updating the baseline to hide delays destroys comparability and masks true performance.



INCOMPLETE DATA:

Outdated actuals or unreliable remaining durations lead to misleading forecasts.



IGNORING ROOT CAUSES:

Reporting the movement without analysing the cause prevents effective corrective action.



MIXING CHANGES WITH PERFORMANCE:

Include only authorized scope changes in the baseline revision. Execution slippage should be handled as performance, not a baseline change.

6 ACTIONS & OUTPUTS



REVIEW VARIANCES:

Check forecast dates against the baseline for key milestones and critical path activities.



MAINTAIN CHANGE RECORDS:

Document and approve any scope changes through the change control process.



SEPARATE ANALYSIS:

Distinguish schedule movement caused by execution performance from that caused by approved scope changes.



COMMUNICATE FINDINGS:

Update the schedule, report variance reasons and agree actions with stakeholders.



KEY OUTPUTS:

Updated schedule with current forecasts, variance narrative and a change log showing authorized scope changes.



SCHEDULE UPDATE AND RECOVERY



Use a single status date to assess performance and test actions when completion is threatened.



1 PURPOSE & VALUE

Keep the schedule model accurate and evaluate specific actions when completion is threatened, using a single status date.



ACCURATE MODEL:

Reflects actual progress, logic and realistic remaining durations.



EARLY IDENTIFICATION:

Reveals emerging slippage and new driving paths.



INFORMED DECISIONS:

Tests recovery options to re-establish control.



CLEAR PRIORITIES:

Focuses on actions that address the true drivers of delay.



TRANSPARENT COMMUNICATION:

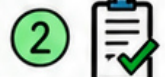
Provides a common view for the team and stakeholders.

2 HOW TO APPLY

Update the schedule to one status date and evaluate recovery options.



Set a single status date for the update.



Collect actual starts, finishes and remaining durations.



Resolve out-of-sequence logic and check dependencies.



Recalculate the schedule and identify the current driving path.



Test recovery options (e.g., resequencing, added resources) and compare results.



CONTROL:

Use one status date, realistic durations and valid logic. Do not insert dates to force a target.

3 INPUTS & EVIDENCE



BASELINE SCHEDULE:

Approved logic, durations, milestones and scope.



PROGRESS RECORDS:

Actual start and finish dates, % complete and open work.



RESOURCE INFORMATION:

Availability, current assignments and any confirmed changes.



ISSUES & DEPENDENCIES:

Delays, constraints and out-of-sequence activities.



STAKEHOLDER INPUT:

Requirements, change requests and key constraints.



OWNERS:

Scheduler, project manager, work-package owners and functional leads provide and confirm the data.

4 PRACTICAL EXAMPLE

Illustrative review delay at the status date.



CURRENT POSITION:

A technical review is still unfinished. Its remaining duration pushes the current project forecast beyond the target.



OPTIONS:

Test adding qualified reviewers to the remaining review work. Also test overlapping downstream preparation that does not depend on final approval.



COMPARE:

Recalculate completion dates, resource demand, cost and rework risk for each option.



DECIDE:

Authorize a feasible recovery plan, assign owners and update the forecast.



CONTROL:

Preserve actual dates. Recovery changes future work; it cannot erase past delay.

5 PITFALLS & LIMITS



PERCENT COMPLETE ONLY:

Do not use % complete alone to calculate remaining duration. It can be misleading.



FORCED DATES:

Do not insert impossible dates to restore a target. This masks issues and reduces credibility.



INVALID LOGIC:

Out-of-sequence activities or missing dependencies give unreliable results.



UNREALISTIC ASSUMPTIONS:

Recovery options must be practical and resourced.



LIMITATION:

Recovery may not be possible if constraints (e.g., scope, resources) cannot be changed.

6 ACTIONS & OUTPUTS



PUBLISH FORECAST:

Updated schedule with recalculated dates, current driving path and completion forecast.



ASSIGN RECOVERY ACTIONS:

Specific actions, owners and deadlines based on the tested options.



DOCUMENT DECISIONS:

Record the options evaluated and the rationale for the selected approach.



FOLLOW UP:

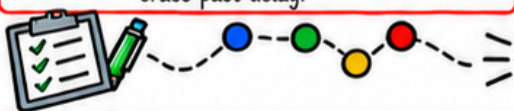
Track action progress and update the schedule at the next status date.



MAINTAIN CONTROL:

Repeat regularly using the same process and data quality standards.

Created by WikiFree.com





HOW TO USE THIS GUIDE

Get the most value from the 30 Project Scheduling Techniques cards.



1 START WITH A NEED

Select a card that matches a real project decision. Confirm the scope, project lifecycle, stakeholders and constraints before you begin.



CONFIRM SCOPE:

Define what is in and out of scope for the decision.



CHECK CONTEXT:

Understand the lifecycle phase, stakeholders and key constraints.



USE CONSISTENT TERMS:

Apply the card definitions consistently across your team.



READ THE EXAMPLE:

Use the worked example as an illustration, not a promise or benchmark.



GATHER EVIDENCE:

Select the data and evidence needed before acting.

2 APPLY & REVIEW

Follow a clear process, document your work, and review the result against your project needs.



ASSIGN AN OWNER:

Give a clear owner for applying the technique and reviewing the result.



AGREE INPUTS AND OUTPUT:

Confirm the data required and the expected output before you start.



RECORD ASSUMPTIONS:

Note key assumptions, scope, constraints and data sources.



APPLY THE CARD:

Use the guidance, calculations and checks in the card.



REVIEW THE RESULT:

Interpret the output, adapt frequency and thresholds to the project.



IMPORTANT CONTROLS:

For a denominator of zero use N/A, not zero percent. Protect sensitive information and maintain a version history.

3 ABOUT THIS EDITION

This guide is Book 5 of the WikiFree Project Management Collection, covering 30 original practical learning cards.



PURPOSE:

Provide clear, practical techniques to support project scheduling decisions.



SCOPE:

The terminology and techniques are a practical synthesis, not an exhaustive standard or formal certification.



CUSTOM METRICS:

Define working metrics locally to suit your project context.



EXAMPLES:

All examples are illustrative, using the same approved scope and reporting date.



FURTHER READING:

Selected sources for deeper learning are provided in panels 4-6.

4 SOURCE 1

GAO - Schedule Assessment Guide

The U.S. Government Accountability Office (GAO) Schedule Assessment Guide provides practical guidance on assessing and improving project schedules in the public sector.



LATEST GUIDANCE:

Covers good practices for schedule development, assessment and risk management.



USEFUL TOPICS:

Explains characteristics of a reliable schedule and common schedule risks.



NOTE:

This is a foundation for further reading and is not an endorsement of this publication.



Click this panel to open the source.

5 SOURCE 2

PMI - Rolling Wave Planning

The Project Management Institute (PMI) provides guidance on rolling wave planning as a progressive planning technique used in projects with evolving details.



KEY IDEA:

Plan near-term work in detail and keep later work at a higher level until more information is available.



WHEN TO USE:

Useful in iterative and adaptive environments where uncertainties exist.



NOTE:

This is a foundation for further reading and is not an endorsement of this publication.



Click this panel to open the source.

6 SOURCE 3

PMI - Work Breakdown Structure

The Project Management Institute (PMI) provides guidance on the work breakdown structure (WBS) as a key tool for defining and organizing the total scope of work.



KEY IDEA:

Break the project into smaller, manageable components to support planning, estimating and control.



BENEFITS:

Improves clarity, helps assign responsibilities and supports schedule development.



NOTE:

This is a foundation for further reading and is not an endorsement of this publication.



Click this panel to open the source.

