

INFO3333

Semester 1, 2019

Module 6 Lecture:
Schedule Construction;
Stakeholder Management;
Project Communications

based on slides by Dr Rabiul Hasan and Prof Alan Fekete, using content in
“Information Technology Project Management (9th ed)” by K. Schwalbe, pub Cengage

Learning objectives

- Terminology
- estimate activity duration
- construct schedule using critical path analysis
- stakeholder management concepts
- communications management concepts

Recall: Activity list

- An activity is a task to be done (from WBS) with associated information (“activity attributes”) about expected duration, dependencies, constraints, cost, resource requirements, etc
- Activity list includes the activities, each with identifier and name as well as attributes

Recall: Project schedule

- Shows project activities and milestones with intended dates (intended start and finish, for each activity; date for each milestone to be reached)

Recall: Gantt Charts

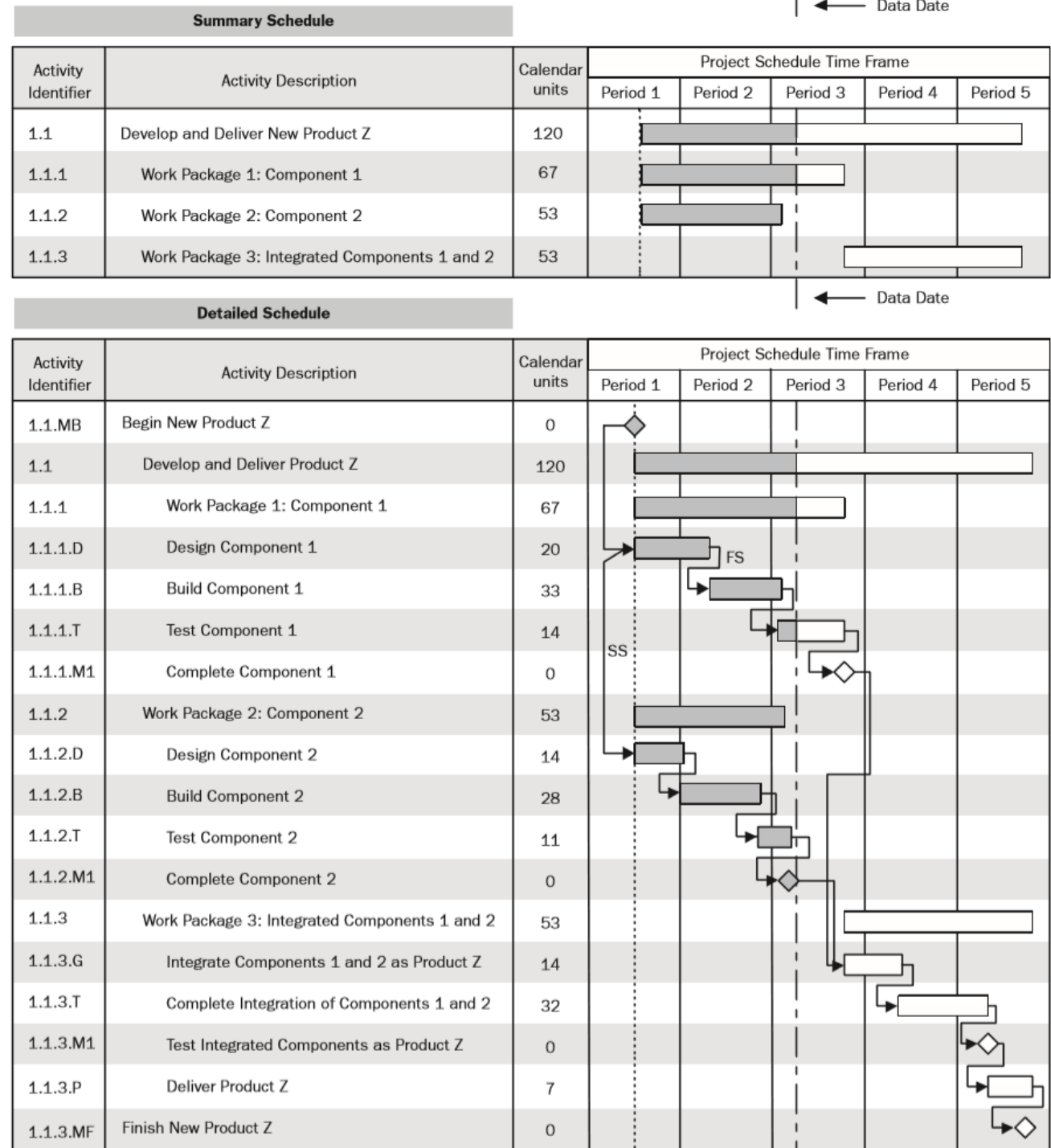
- **Gantt charts** provide a standard format for displaying project schedule information by listing project activities and their corresponding start and finish dates in a calendar format
 - plenty of tools to support this format (eg Microsoft Project)
- Symbols include:
 - A black diamond: a milestone
 - Thick black bars: summary tasks
 - Lighter horizontal bars: durations of tasks
 - Arrows: dependencies between tasks

Summary Schedule

&

Detailed Schedule

Source: A Guide to the Project Management Body of Knowledge, Fifth Edition (PMBOK® Guide) © 2013 Project Management Institute



Recall: Project Time Management processes

- **Plan schedule management:** determining the policies, procedures, and documentation that will be used for planning, executing, and controlling the project schedule
- **Define activities:** identifying the specific activities that the project team members and stakeholders must perform to produce the project deliverables
- **Sequence activities:** identifying and documenting the relationships between project activities
- **Estimate activity durations:** estimating the number of work periods that are needed to complete individual activities
- **Develop the schedule:** analyzing activity sequences, activity resource estimates, and activity duration estimates to create the project schedule
- **Control the schedule:** controlling and managing changes to the project schedule

Estimate durations

- Duration is elapsed time needed to complete an activity (measured in hours or weeks etc)
 - Note: different from effort, which is amount of work (measured eg in person-days); conversion based on number of suitable people available, adjust for interaction and possible internal dependencies!
 - Project activities are usually not divisible into too many pieces
- Estimation is hard to do well (especially for unfamiliar activities)
 - Good PMs get better at it with practice/experience

PERT estimate

- Start with three estimates of duration:
 - most likely (things happen as normal)
 - optimistic (things go better than normal)
 - pessimistic (everything goes wrong, but not major disaster)
- Use $[\text{optimistic} + 4 * \text{likely} + \text{pessimistic}] / 6$ as estimate of duration

Velocity

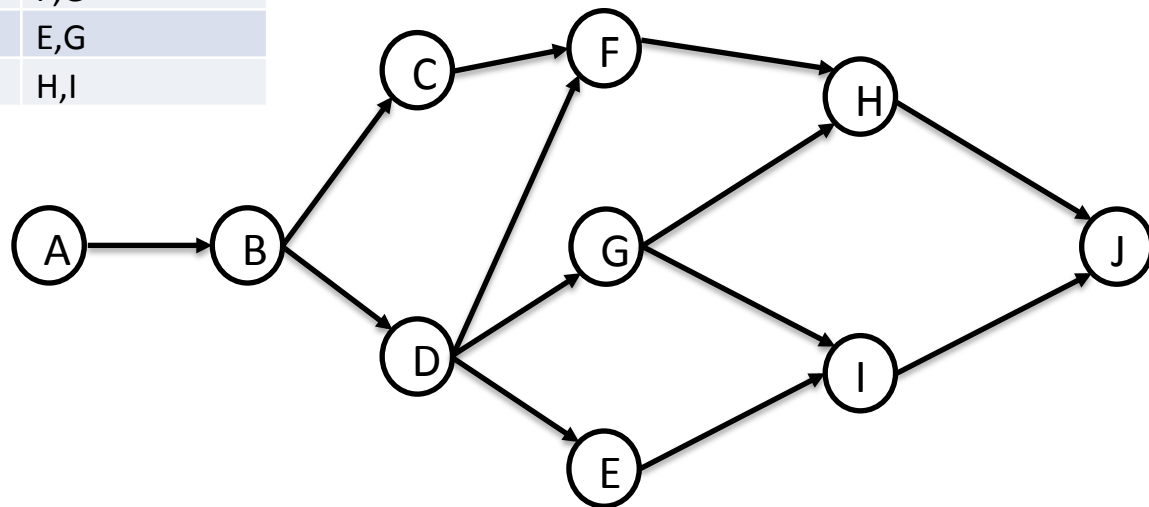
- Use in agile projects to estimate effort needed
- Keep records of how much effort is used to successfully deliver each user-story (relate to complexity of the user-story)
- Expect next story to take effort which is similar to what was needed for recent similar stories

Develop schedule

- Given a set of activities, with dependencies between them, and duration of each
- Determine a schedule that finishes as soon as possible
 - Note: there is usually some flexibility in the schedule, with the same early finish date
- This is done by tools, but we will cover manual technique so you understand what the tool is doing, and how to interpret the output of the tool

Activity On the Node (AON) Network Diagram

Activity	Estimated Duration	Predecessor
A	5	None
B	4	A
C	5	B
D	6	B
E	7	D
F	3	C,D
G	6	D
H	7	F,G
I	8	E,G
J	3	H,I



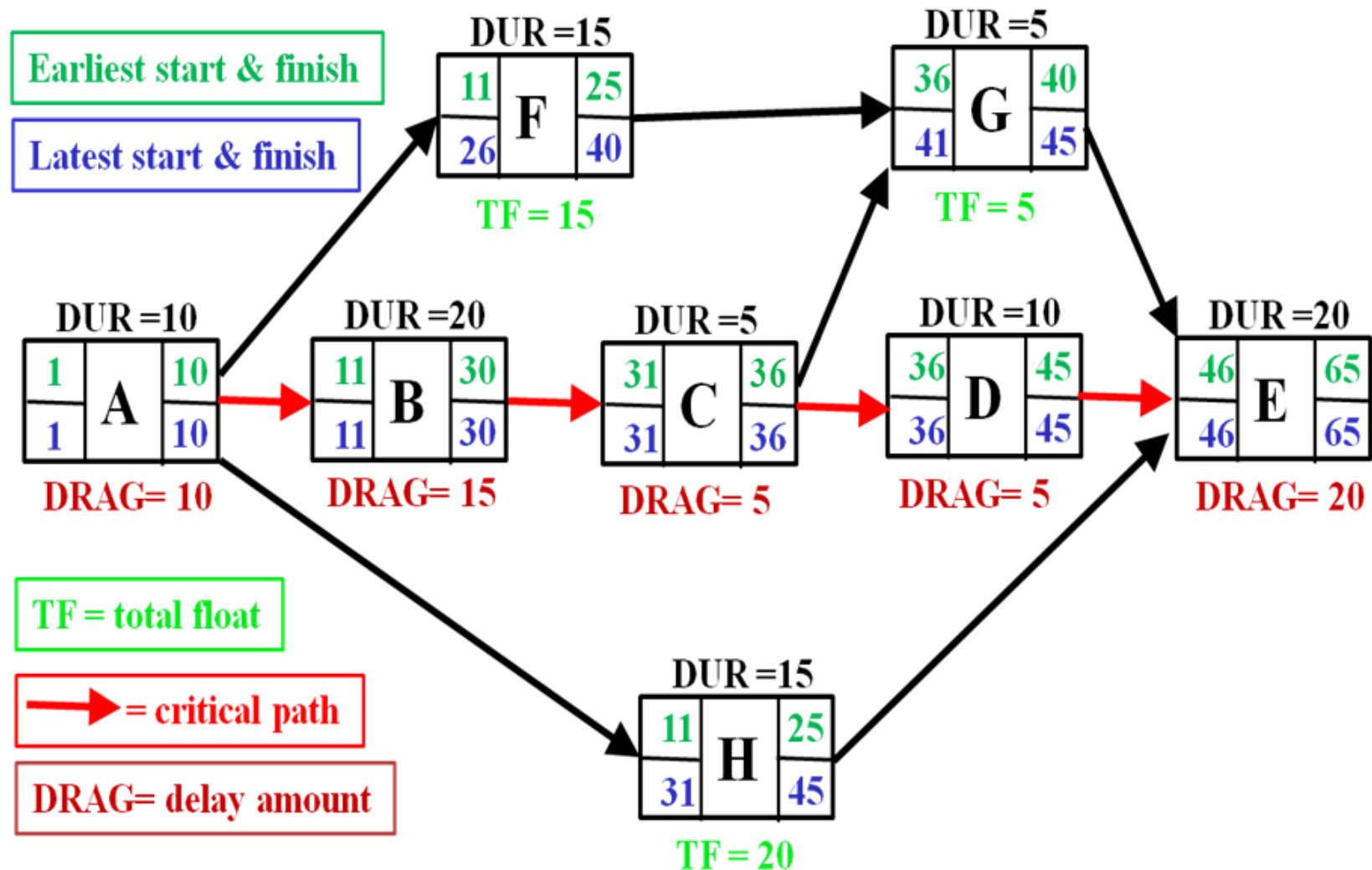
Critical Path Method (CPM)

- **CPM** is a technique used to schedule activities and predict total project duration, given duration of each activity and network diagram
 - Make sure there is a single start-project activity node, and a single end-project activity node
- Calculate for each activity when it can start and when it can finish
 - start and finish times each are in a range (earliest till latest)
- The critical path is the *path with greatest total duration* through the network diagram

Critical path example

By Nuggetkiwi - Own work, CC BY-SA 3.0,

<https://commons.wikimedia.org/w/index.php?curid=15424866>



Critical Path

- The critical path is *not* the one with all the most important activities; it only accounts for time
- There can be more than one critical path if the lengths of two or more paths are the same
- The critical path can change as the project progresses (when activities or durations are adjusted)

Understanding Float, Drag

- **Total float** (also called **total slack**) is the amount of time an activity may be delayed from its early start without delaying the planned project finish date
 - Also, sometimes see **Free slack** or **free float** used for the amount of time an activity can be delayed without delaying the early start of any immediately following activities
- **Drag** is amount existence of this critical path activity delays project completion
 - equal to lower of activity duration, total float of any parallel activity

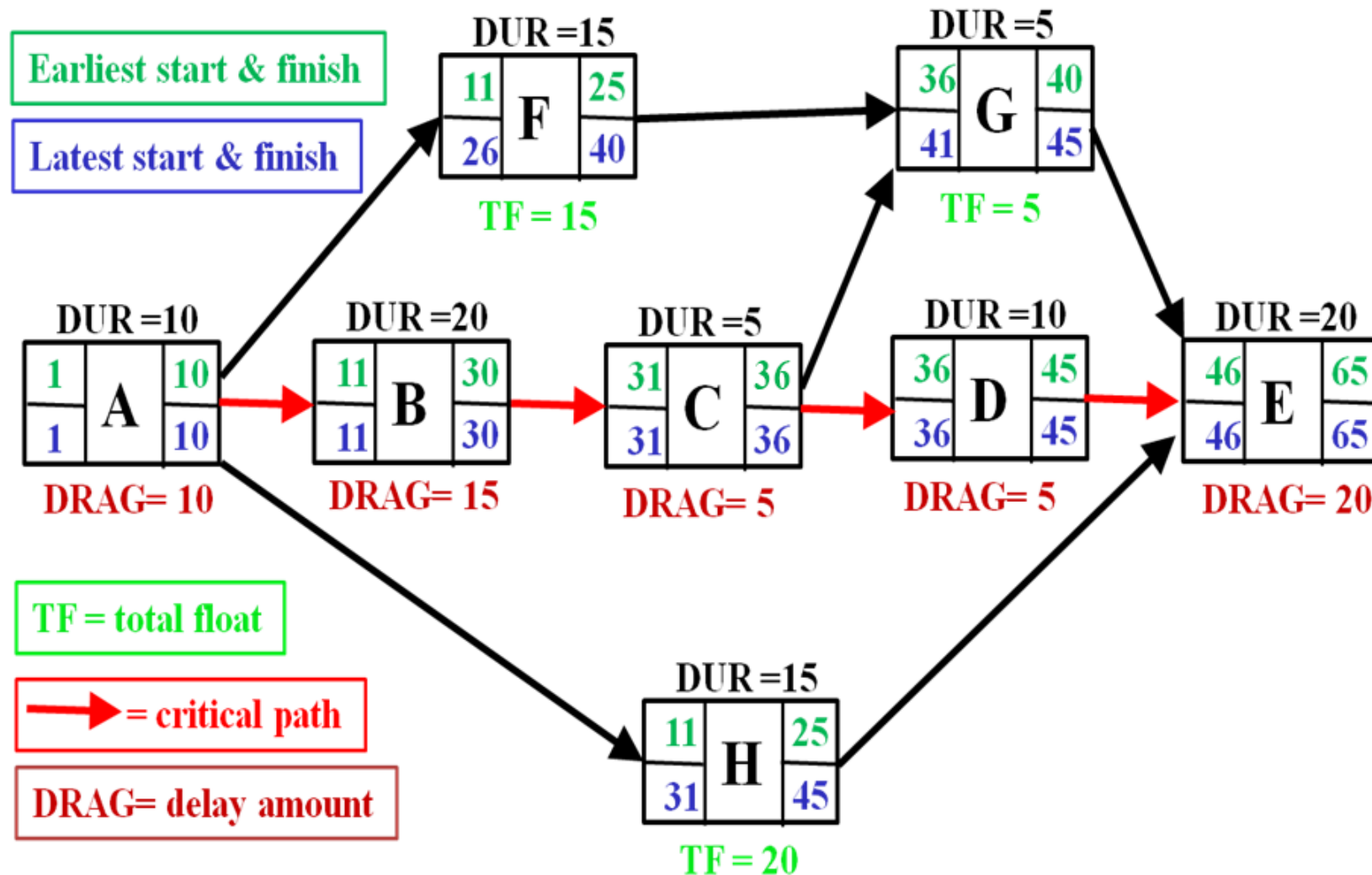
The calculations

- A **forward pass** (consider nodes after their predecessors) through the network diagram determines the early start (ES) and early finish (EF) dates
 - each activity has ES set equal to (greatest EF of any of its predecessors)+1
 - each activity has EF set equal to ES+duration-1
- A **backward pass** (consider nodes after their successors) determines the late start (LS) and late finish (LF) dates
 - for project's final activity, LF=EF, LS=ES
 - otherwise LF set to (least LS of any successor)-1
 - LS set equal to LF-duration+1
- Float and critical path then determined
 - total float is difference between earliest start and latest start
 - activities on critical path are those with zero total float
 - drag determined from duration and total float numbers

Critical path calculations

By Nuggetkiwi - Own work, CC BY-SA 3.0,

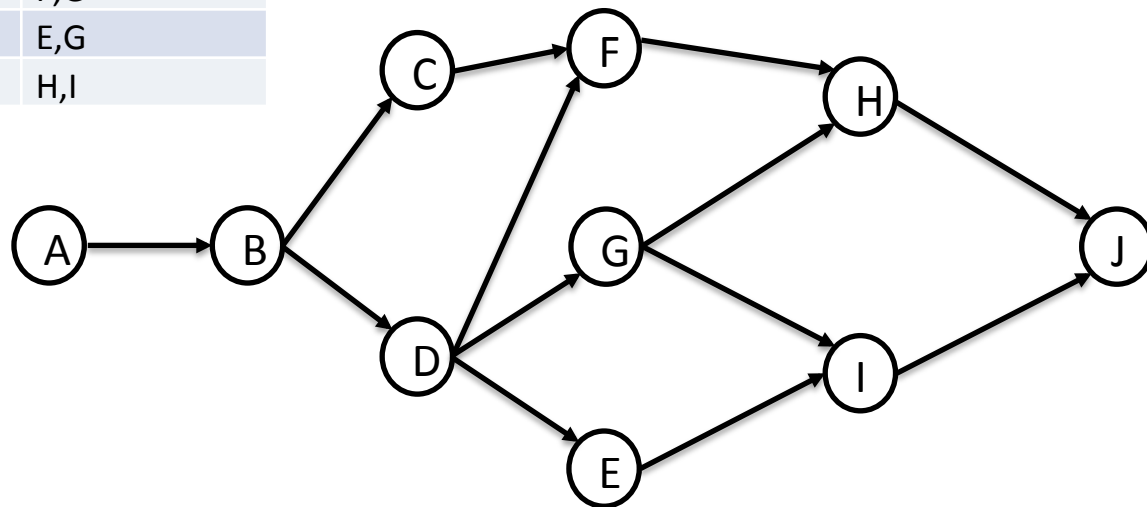
<https://commons.wikimedia.org/w/index.php?curid=15424866>



In-class exercise

Activity	Estimated Duration	Predecessor
A	5	None
B	4	A
C	5	B
D	6	B
E	7	D
F	3	C,D
G	6	D
H	7	F,G
I	8	E,G
J	3	H,I

Calculate ES, LS, EF, LF, TF, and determine critical path for this project!



Resource limits

- Schedule so far assumes that we can do many activities at same time, unless there is a logical dependency
- But in practice, resource issues may prevent parallel work
 - eg only 2 licences for a particular tool, means only 2 activities using that tool should be scheduled together
 - eg only 3 staff who are expert in some language, means only 3 activities doing complex programming in that language, should be scheduled together

Dealing with resource limits

- Introduce extra dependencies to prevent concurrent running
- Use tools that do resource allocation as well as scheduling
- Or, calculate ideal schedule, then modify it to respect constraints
 - called Resource Leveling “A technique in which start and finish dates are adjusted based on resource limitation with the goal of balancing demand for resources with the available supply” [PMBOK]

Shortening a Project Schedule

- What to do when customers or management request/demand that the project finish quicker (before the earliest finish calculated)?
- The only possibilities are to argue against it (usually futile) or to change the constraints
 - remove dependencies
 - and/or shorten task durations
 - called “crashing”
 - focus on the critical path
 - prioritise tradeoff between improvement in completion time versus increased cost

Ways to shorten task duration I

- Add more resources
 - But remember: you can't create a baby in 1 month with 9 women!
 - especially for software development: “mythical man-month” and “adding people to a late project makes it later” [Brookes]
 - new people need to learn about context, and learn to work together
 - existing people are diverted from productive work, to teach new people
 - overheads of interaction among a larger team
 - software isn't easy to divide into independent pieces

Ways to shorten task duration II

- Reduce scope
 - Hard to do without impact on other later activities
 - Needs close stakeholder coordination
- Reduce quality
 - eg do less testing, less bugfixing
 - eg do less refactoring
 - builds “technical debt” that will add work later (especially in maintenance phases)

Ways to remove dependencies

- Run work in parallel with other activities that it ought to use/know about
 - say we “fast track” an activity
- Typically this will increase the total work that needs to happen, and/or decrease quality

Project stakeholder management

- Stakeholder “an individual, group, or organization, who may affect, be affected by, or perceive itself to be affected by a decision, activity, or outcome of a project” [PMBOK]
- May be inside organization doing the project, or outside

PMBOK: Project stakeholder management processes

- Identify stakeholders
- Plan stakeholder engagement
- Manage stakeholder engagement
- Monitor stakeholder engagement

Examples of stakeholders

- project sponsor
- senior management
- project team leader
- project team members
- project customer
- resource managers
- line Managers
- Product user group
- Project testers
- labour union leaders
- regulators
- suppliers
- potential users
- subcontractors to the project
- consultants to the project

[expanded and adapted from https://en.wikipedia.org/wiki/Project_stakeholder]

Stakeholder register

- Document that lists stakeholders: name, position, role/relationship to project, contact information etc
- Also may have highly sensitive ideas on how their involvement will be managed!

Engagement options

- Rough categories of style of engagement
 - unaware
 - resistant
 - neutral
 - supportive
 - leading
- Important to distinguish what PM wants the engagement to be, from current engagement

Agile approach to engagement

- “Value customer collaboration over contract negotiation” [from Agile Manifesto]
- Agile methods expect close involvement by a representative of the client, throughout the project
 - especially in making the continual decisions about priority among features
 - based on business value of each feature
- Client representative should be embedded with project team
- Scrum term: “Product Owner”

Project communications management

- Crucial so that people know what they need to know, so they can do their jobs well
- But avoid overloading them with what they don't need
 - attention is precious

PMBOK: Project communications management processes

- Plan communications management: determine the information and communication needs of the stakeholders
- Manage communications: create, distribute, store, dispose of communications
- Monitor communications: check that stakeholders information needs are being met

Media choices

- Hard-copy report
 - Hard-copy letter
 - E-mail
 - Phone
 - SMS
 - Face-to-face
-
- Each has strengths and weaknesses, appropriate in different contexts (eg depends on seniority or target, whether this is report or request, etc)

Communication paradigms

- Interactive
 - best for achieving common understanding, but not easy to arrange
- Push
 - send it out to targets, hope they pay attention
- Pull
 - put it where target can find it if they want

Timing of communication

- When milestone achieved
 - or especially, when milestone is slipped, or slipping is expected
- At fixed times (eg monthly status update)

Clarity of message

- Vital to be clear whether this is fact (it has happened) vs expectation (we think this will happen) vs intention (we want this to happen)
- Vital to be clear between report and request
 - target should know what they need to do with the communication
 - common technique: label “action items”
- Vital to connect to organizational goals

Bad news

- A hard but crucial task for PM is to let stakeholders know when things go seriously wrong (they will go wrong!)
 - This is not about normal adjustments that are not impacting on agreed delivery dates, budget, scope etc
 - those are internal to project
 - Rather, about things that affect projects capacity to deliver what has been agreed
- Don't hide it
- Don't delay the notification
- Do offer concrete suggestions on mitigation or other responses

Agile communications

- Within development team: try to keep everyone aware of what is happening, and what they can be doing
 - Eg scrum: daily “standup” meeting

Standup

- Timebox the meeting (no more than 15 minutes)
 - standing helps keep it short
- Same time every day
- Everyone of the team should attend if possible, but meeting happens anyway
 - outsiders may observe but not disrupt
- Inspect work done since previous meeting; decide on work for the next day
- Raise issues, rather than necessarily resolve them if they get complex

Scrum standup questions

- “What did I do yesterday that helped the Development Team meet the Sprint Goal?”
- “What will I do today to help the Development Team meet the Sprint Goal?”
- “Do I see any impediment that prevents me or the Development Team from meeting the Sprint Goal?”
- from <https://www.scrumguides.org/scrum-guide.html#events-daily>

Scrum sprint review

- Held at the end of each sprint
 - last up to 4 hours
- Involves Development Team and stakeholders
- Product Owner explains what of the previous Backlog was “Done”, and what was not
- Team demos what was Done
- Reflect on how the sprint went
- Discuss what might be good to do in next sprint
 - consider changes in context (market, potential uses)
 - note: this meeting doesn’t make decisions; instead that is in Sprint Planning Meeting
- adapted from
<https://www.scrumguides.org/scrum-guide.html#events-review>

Key knowledge (quiz, exam!)

- terminology
- critical path calculations
- critical path implications
- stakeholder management (PMBOK, agile, compare)
- communications (PMBOK, agile, compare)