Informatik-Projektentwicklung
– Lecture 4 –

Prof. Dr. Peter Müller

Software Component Technology

Wintersemester 05/06

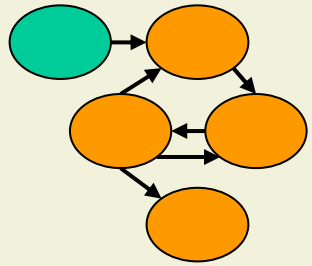
ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Project Initiation

A project charter is:

a. A formal, approved document used to guide both project execution and project control
b. A document issued by senior management that provides the project manager with the authority to apply organizational resources to project activities
c. A narrative description of products and services to be supplied
d. A document describing the organizational breakdown structure of the company

Initiation Process: Summary



- Purpose
  - To formally authorize a new project or that an existing project should continue into its next phase
  - Repeating the initiation process at the start of each phase helps to keep the project focused on the business need

Inputs	Tools & Techniques	Outputs
1. Product description 2. Strategic plan 3. Project selection criteria 4. Historical information	1. Project selection methods 2. Expert judgment	1. Project definition 2. Project charter 3. Project manager assigned

Project Initiation

Assumptions are:

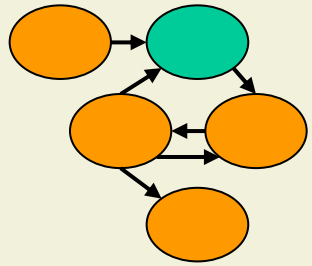
a. Factors that influence the change control system
b. Factors that limit the project management team's options
c. Factors that are considered to be true, real, or certain
d. Factors that influence the scope of the project

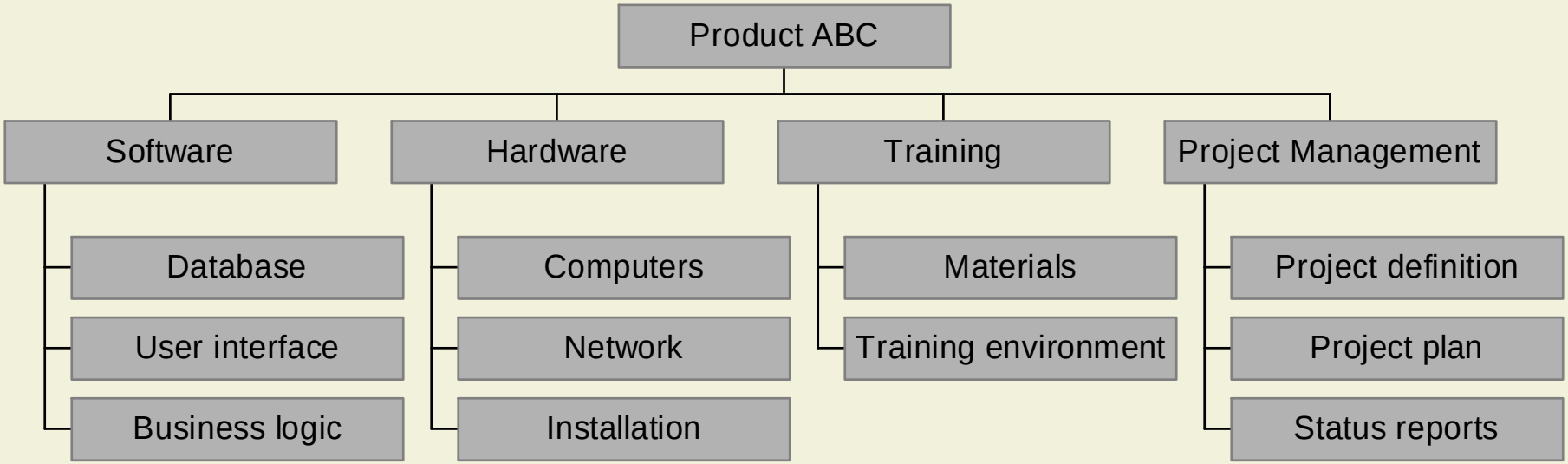
Project Planning

Which of the following statements is NOT true about the WBS?

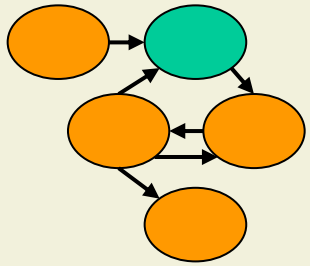
a. The WBS indicates when certain activities are to be done
b. The WBS is a hierarchical breakdown of the project deliverables
c. The WBS represents the entire scope of the project
d. The WBS shows both products and services

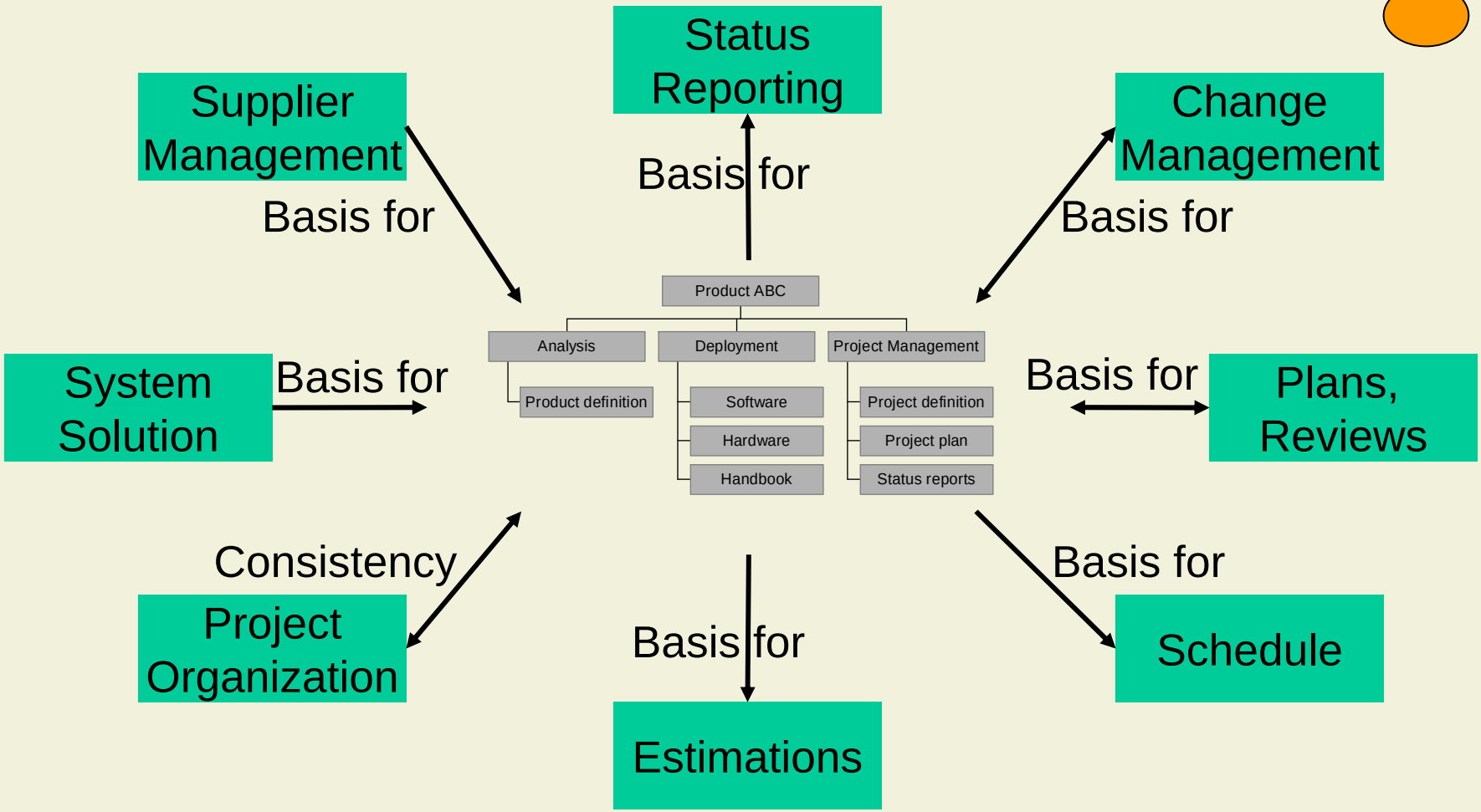
Decomposition Example 2





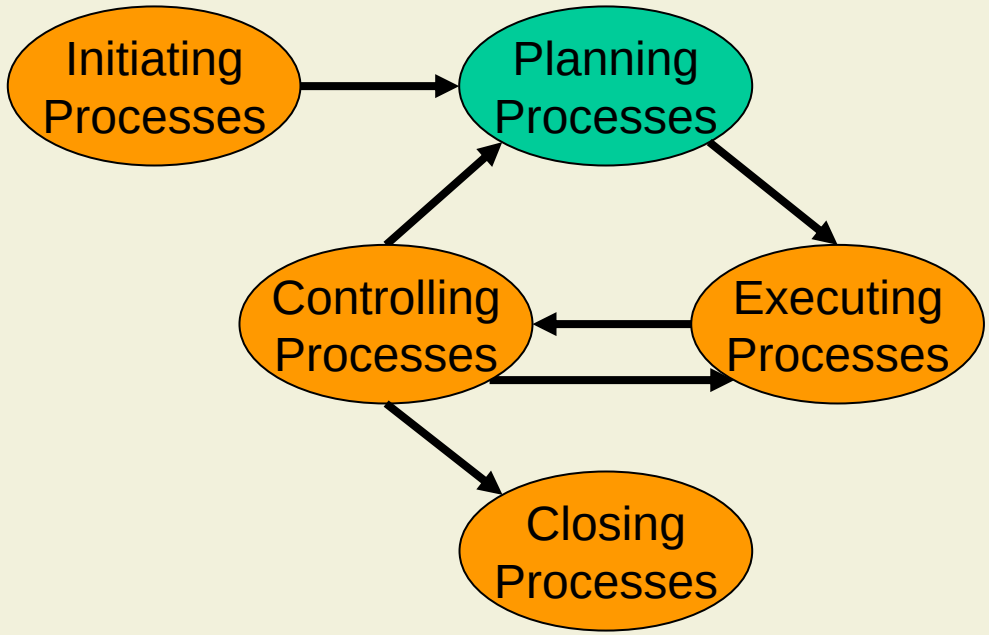
WBS Relationships





Agenda for Today

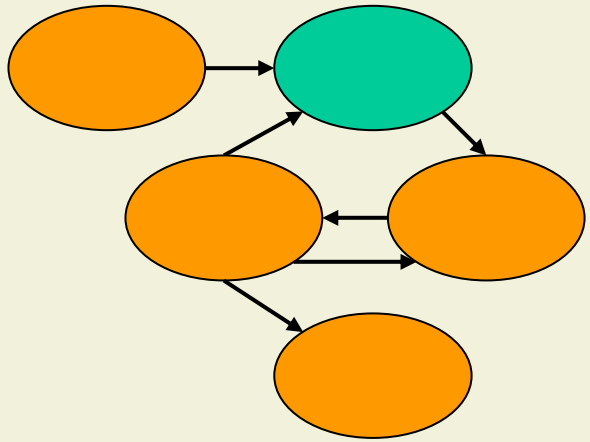
4. Scheduling
   - 4.1 Networks
   - 4.2 Schedule Analysis



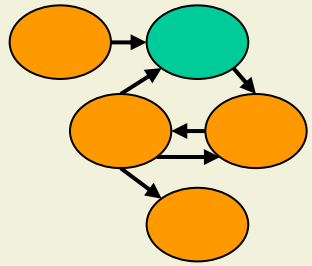
4. Scheduling

4.1 Networks

4.2 Schedule Analysis

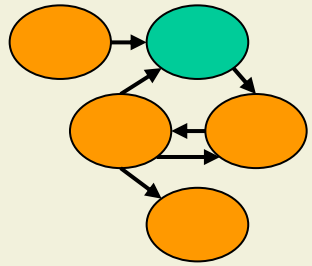


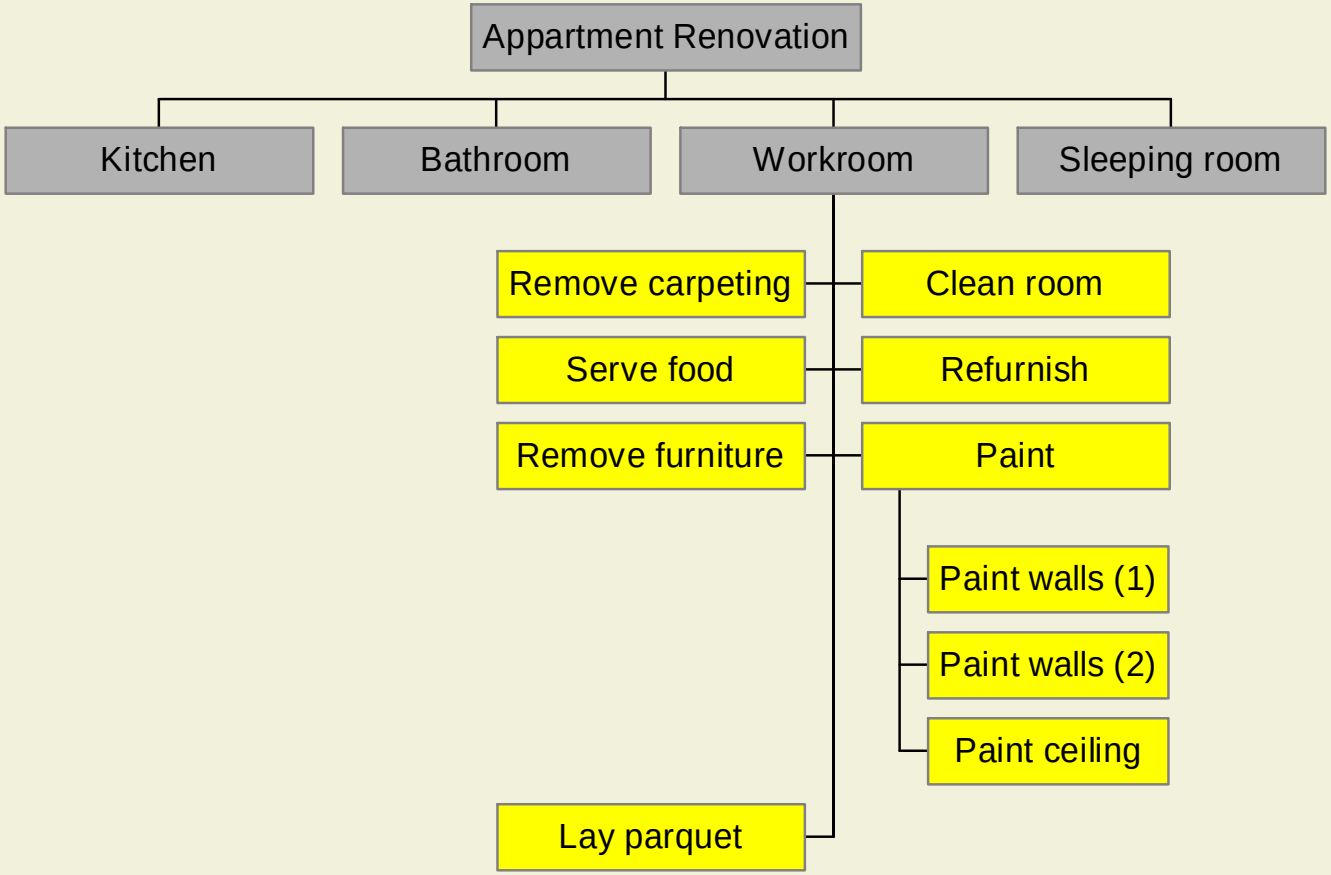
Purpose of Scheduling



- Track the progress of the project
- Determine how possible changes might affect the project
- Communication
  - Will the activities be completed in time?
  - When are which resources needed?
  - When will major milestones be reached?

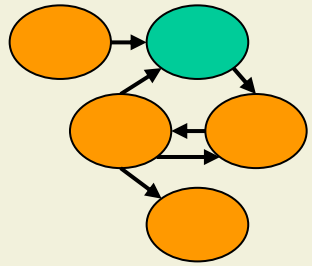
Activities





- Rule of thumb: 40 to 80 person hours per activity

Milestones

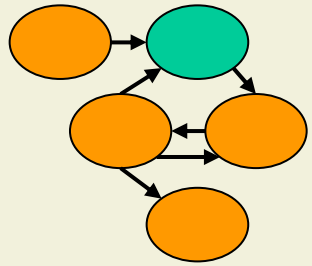


- Definition:
  A significant event in the project, usually completion of a major deliverable

- Milestones have no effort or duration
- Milestones do not have resources

- Example: Painting completed

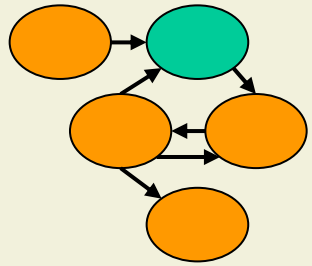
Activity Definition: Summary



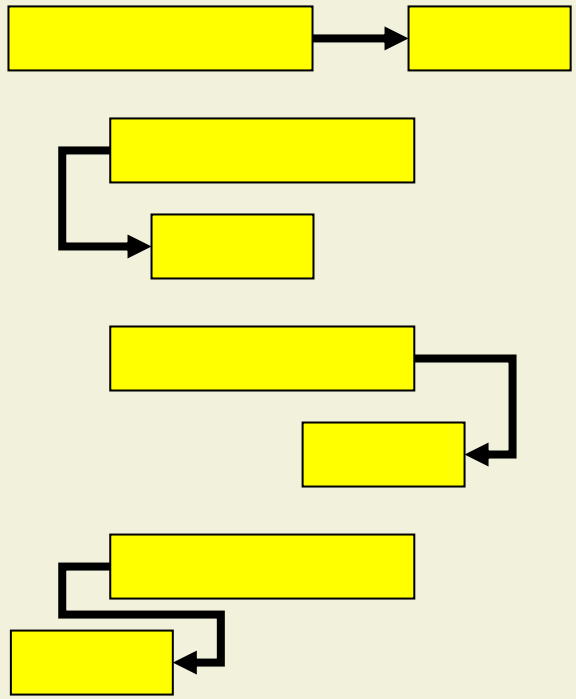
- Purpose
  - To identify and document the specific activities that must be performed to produce the deliverables documented in the WBS.

Inputs	Tools & Techniques	Outputs
1. WBS 2. Project definition	1. Decomposition	1. Activity list 2. WBS updates

Dependencies

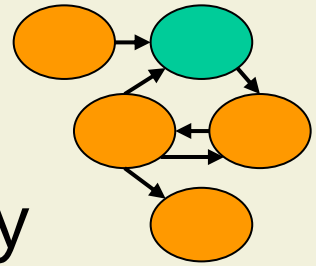


- Logical relationships among activities
  - Finish-to-Start (FS)
  - Start-to-Start (SS)
  - Finish-to-Finish (FF)
  - Start-to-Finish (SF)

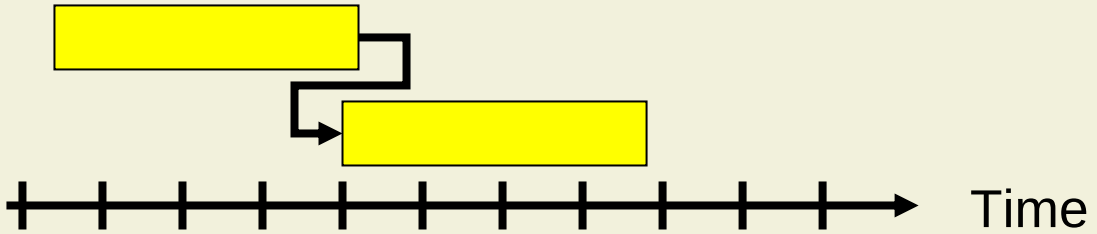


- Dependencies can be mandatory (hard logic) discretionary (soft logic), or external

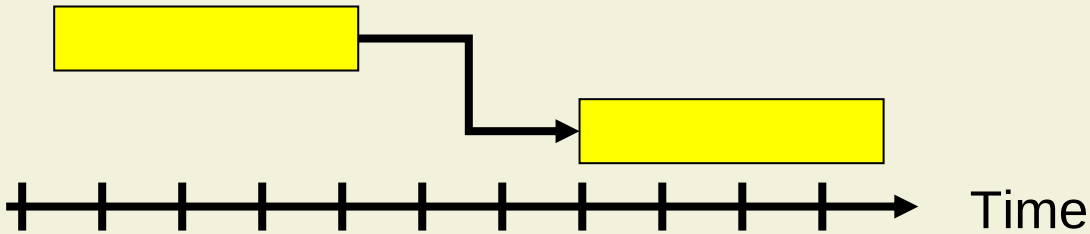
Lag and Lead



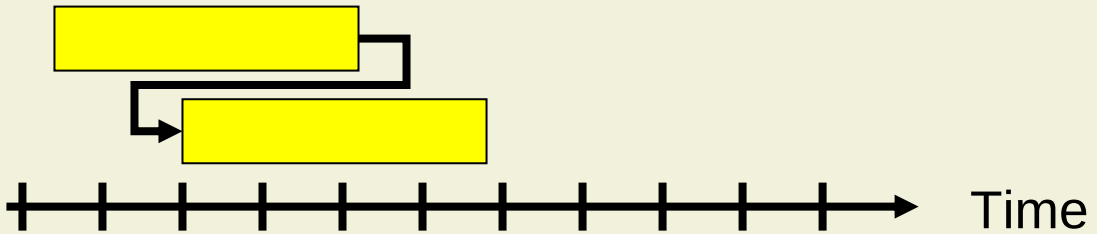
- Modify a logical relationship to direct a delay or acceleration of the successor task
- No modifier



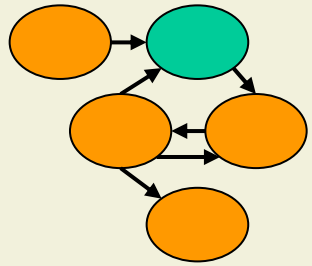
- Lag (+3 units)



- Lead (-2 units)

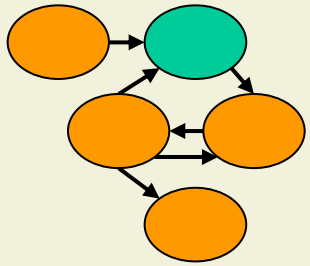


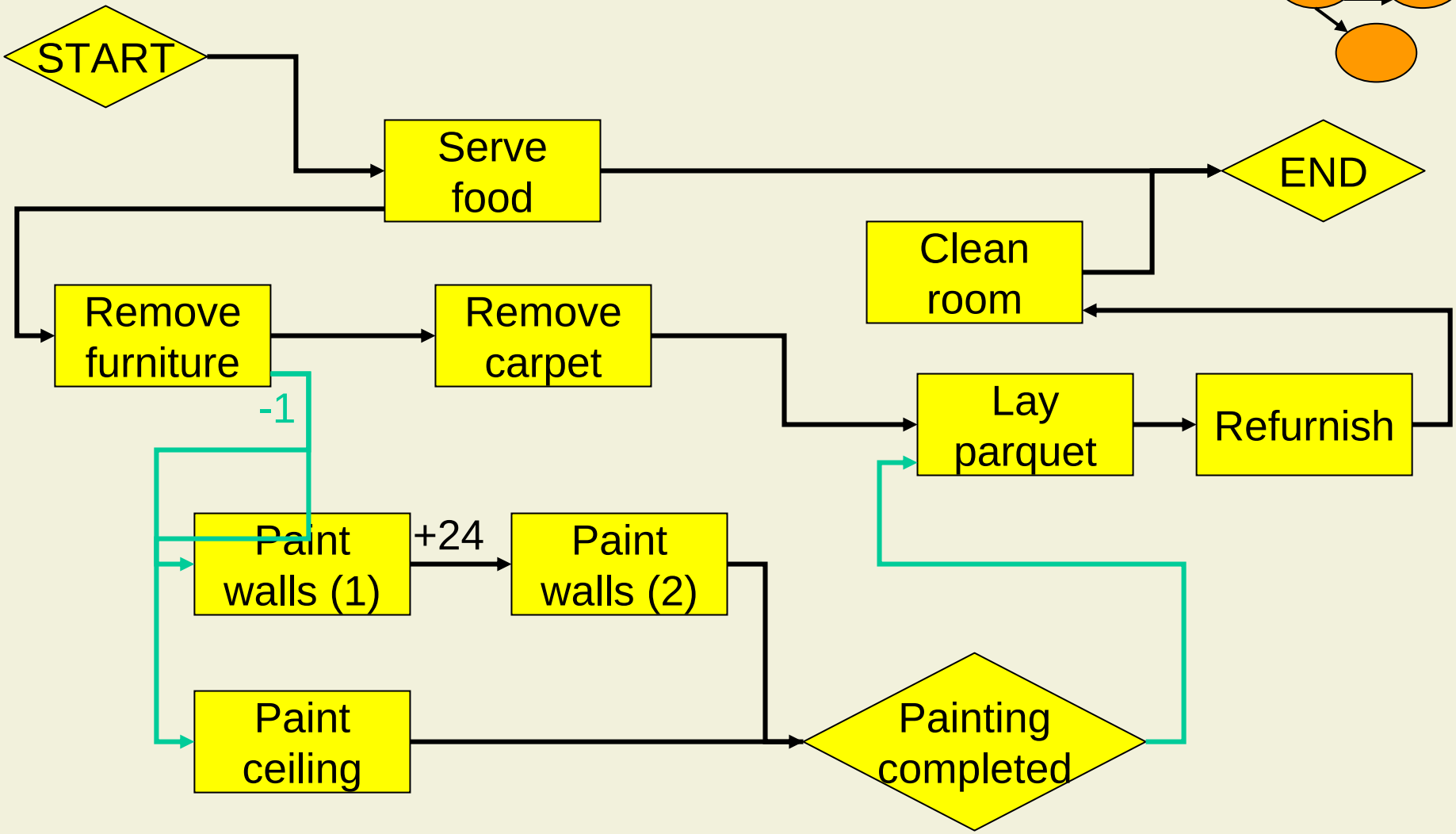
Network Diagrams



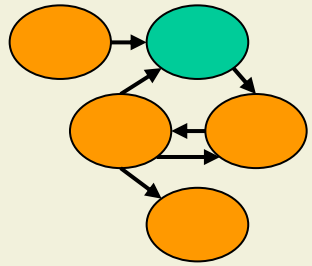
- Precedence Diagramming Method
  - Show all activities (depicted by boxes)
  - Show the logical flow (depicted by arrows)
  - Clearly illustrate dependencies
- Rules
  - Each activity has at least one predecessor and successor (start and end as milestones)
  - No loops, no dangling arrows
- Other network diagramming methods
  - Arrow diagramming method (activity-on-arrow)
  - Conditional diagramming methods

Network Example





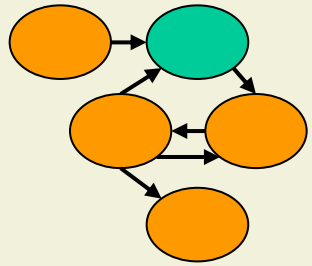
Activity Sequencing: Summary



- Purpose
  - To identify and document logical relationships among activities

Inputs	Tools & Techniques	Outputs
1. Activity list 2. Product description 3. Dependencies • Mandatory • Discretionary • External	1. Network diagramming	1. Network diagrams 2. Activity list updates

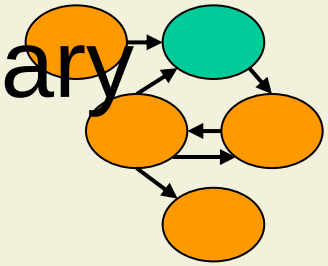
Resource Planning: Summary



- Purpose
  - To determine what physical resources (people, equipment, materials) and what quantities of each should be used and when they would be needed to perform project activities

Inputs	Tools & Techniques	Outputs
1. Activity list 2. Activity duration estimates 3. Resource pool description	1. Expert judgment	1. Resource requirements

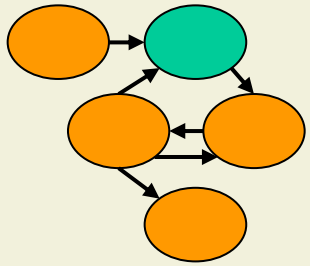
Activity Duration Estimating: Summary



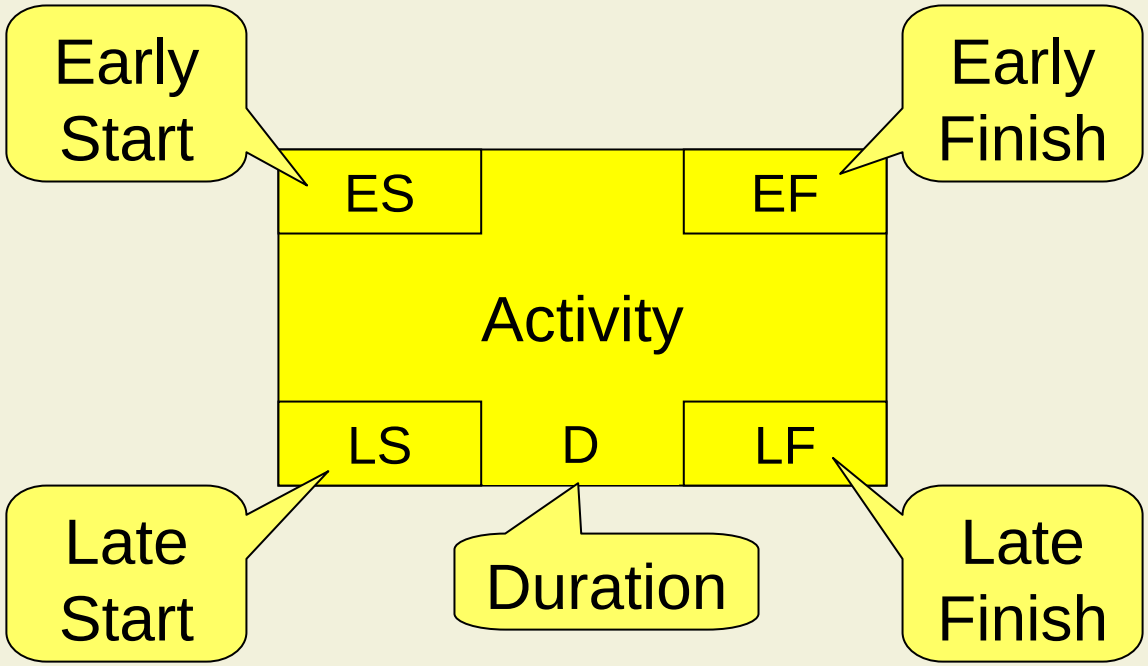
- Purpose
  - To estimate durations based on information on project scope and resources
- Duration = Effort / Resources

Inputs	Tools & Techniques	Outputs
1. Activity list 2. Constraints 3. Assumptions 4. Resource requirements 5. Identified risks	1. Expert judgment 2. Estimating techniques 3. Reserve time	1. Activity duration estimations 2. Basis of estimates 3. Activity list updates

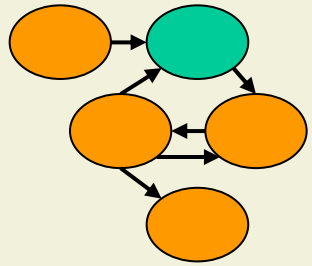
Computing a Schedule



- A schedule consists of the planned dates for all activities and milestones
- Notation



Forward Pass

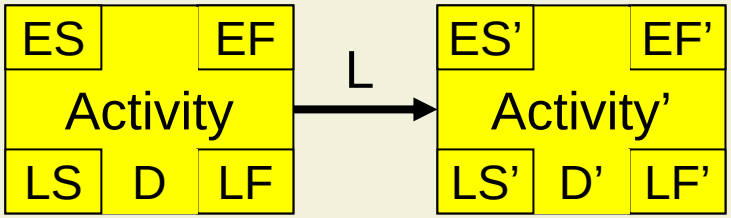


- Determines overall project duration
- First activity starts on time unit 0
- Calculation of the early start and early finish dates
- For Activity A:

  $ES(A) = MAX_{P \in predecessors(A)} ES_P(A)$

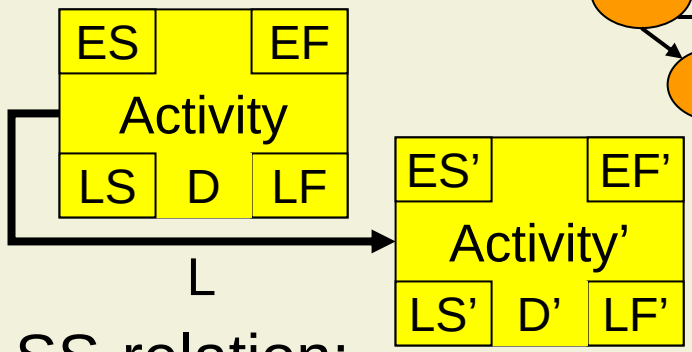
  $EF(A) = ES(A) + Duration(A)$

Calculating Early Start



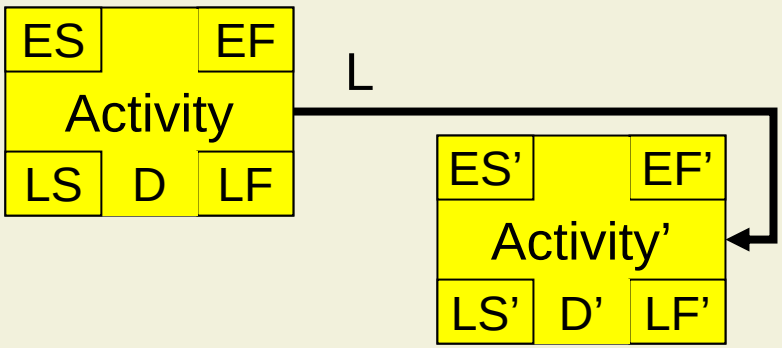
FS-relation:

ES' := EF + L



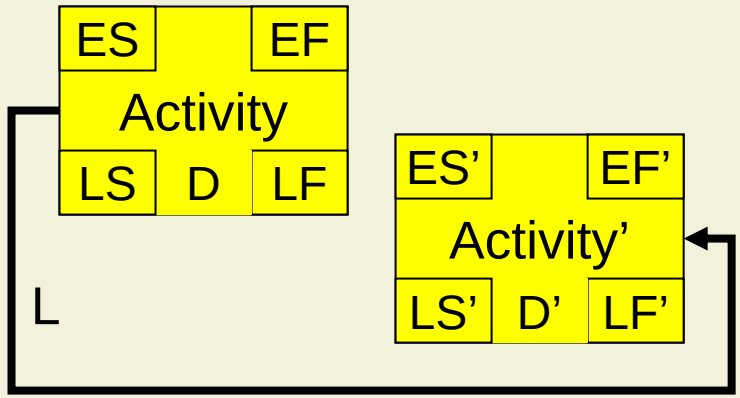
SS-relation:

ES' := ES + L



FF-relation:

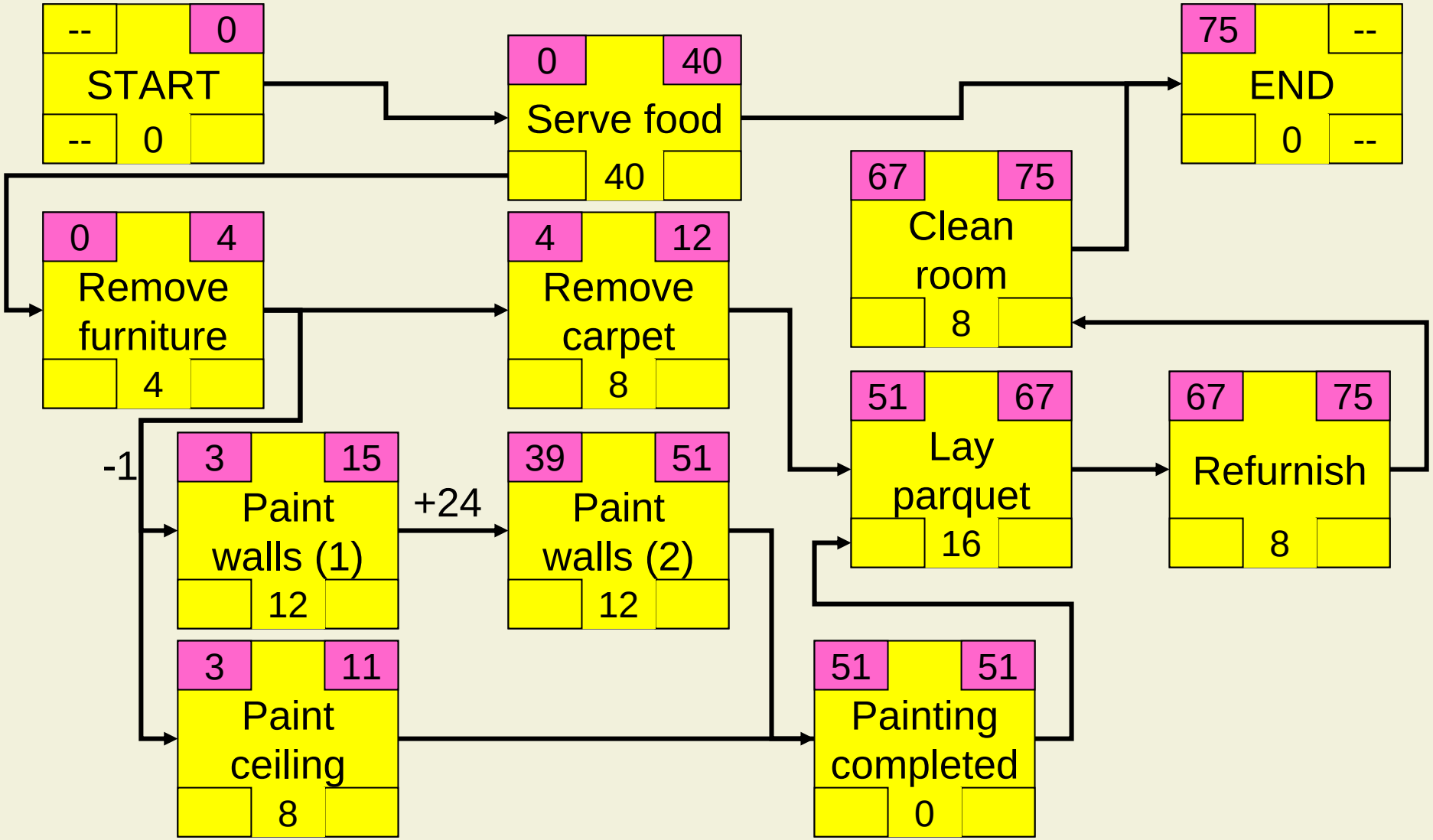
ES' := EF + L – D'



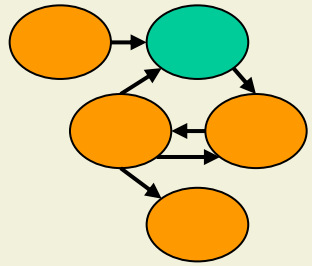
SF-relation:

ES' := ES + L – D'

Forward Pass Example



Backward Pass

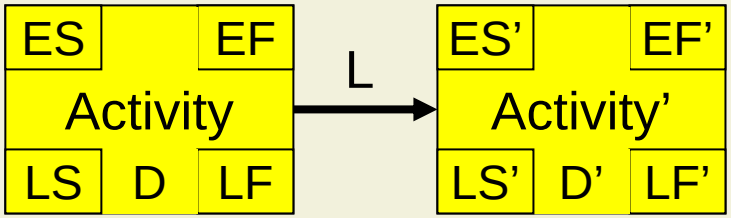


- Determines latest possible dates for each activity that do not delay the overall project
- Last activity ends at time unit of project duration
- Calculation of the late start and late finish dates
- For Activity A:

  $LF(A) = MIN_{P \in successors(A)} LF_P(A)$

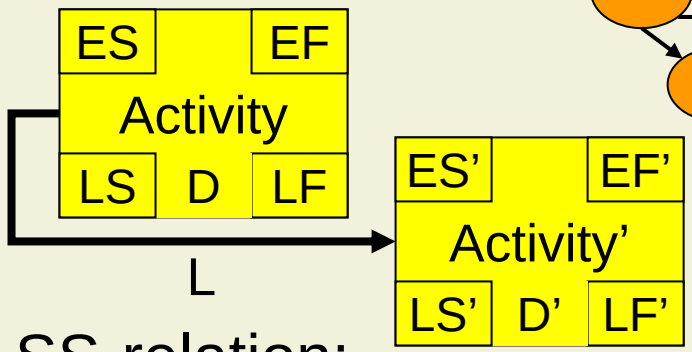
  $LS(A) = LF(A) - Duration(A)$
- The logic is “inverted”
  - early ↔ late, start ↔ finish, + ↔ -, primed ↔ unprimed

Calculating Late Finish



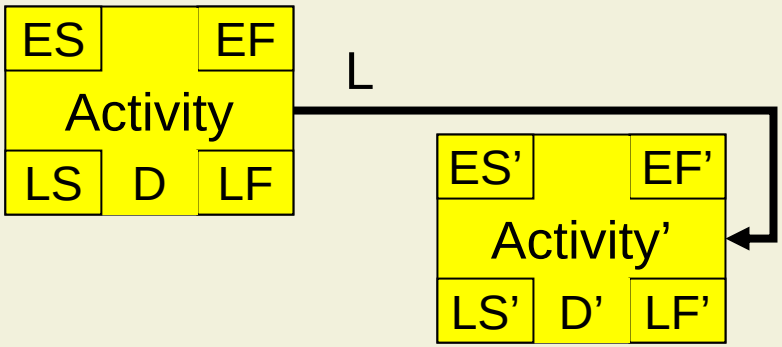
FS-relation:

LF := LS' – L



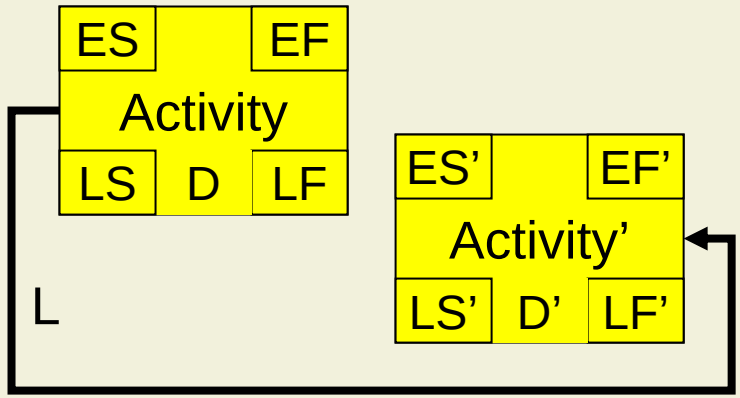
SS-relation:

LF := LS' – L + D



FF-relation:

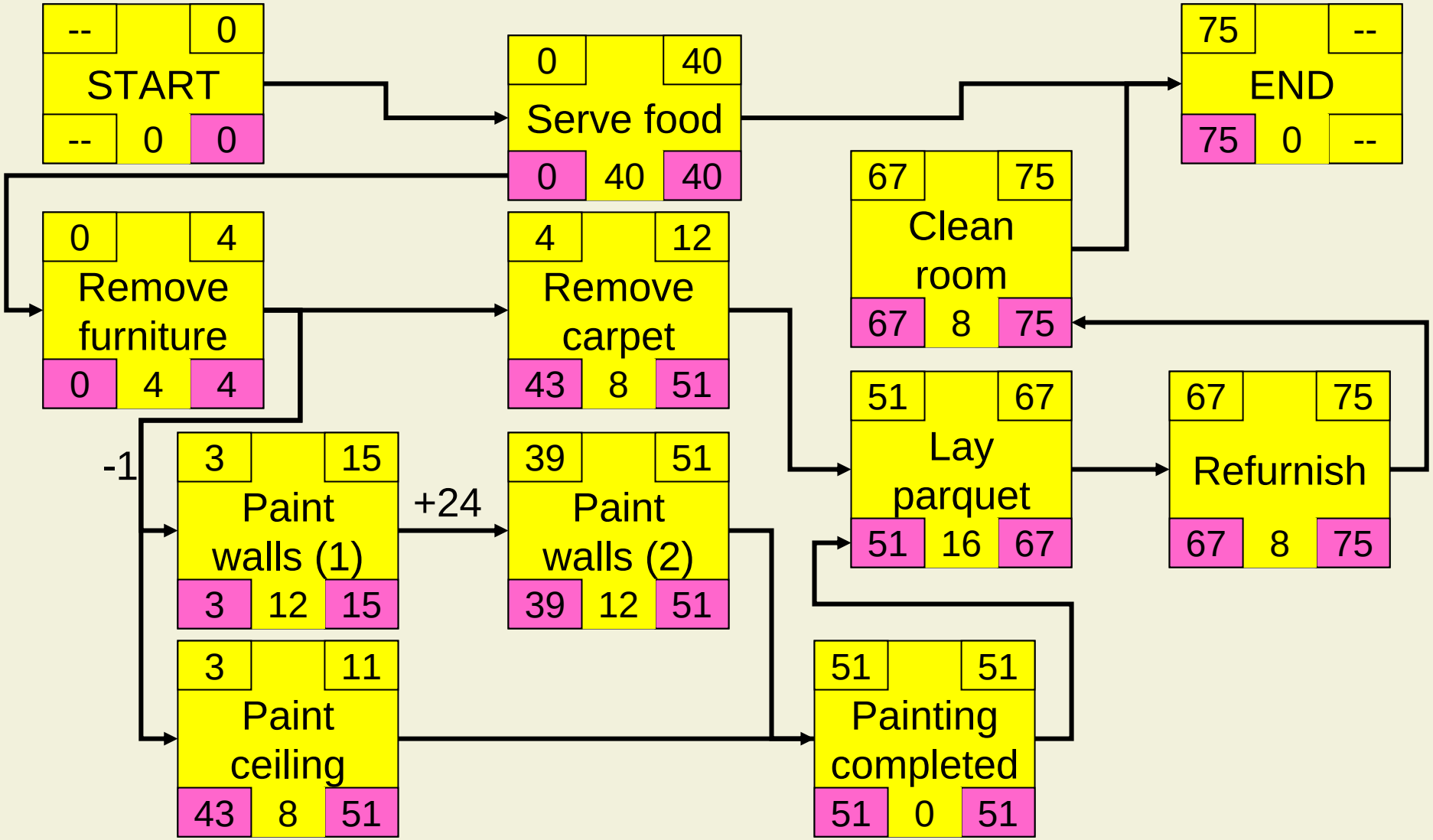
LF := LF' – L



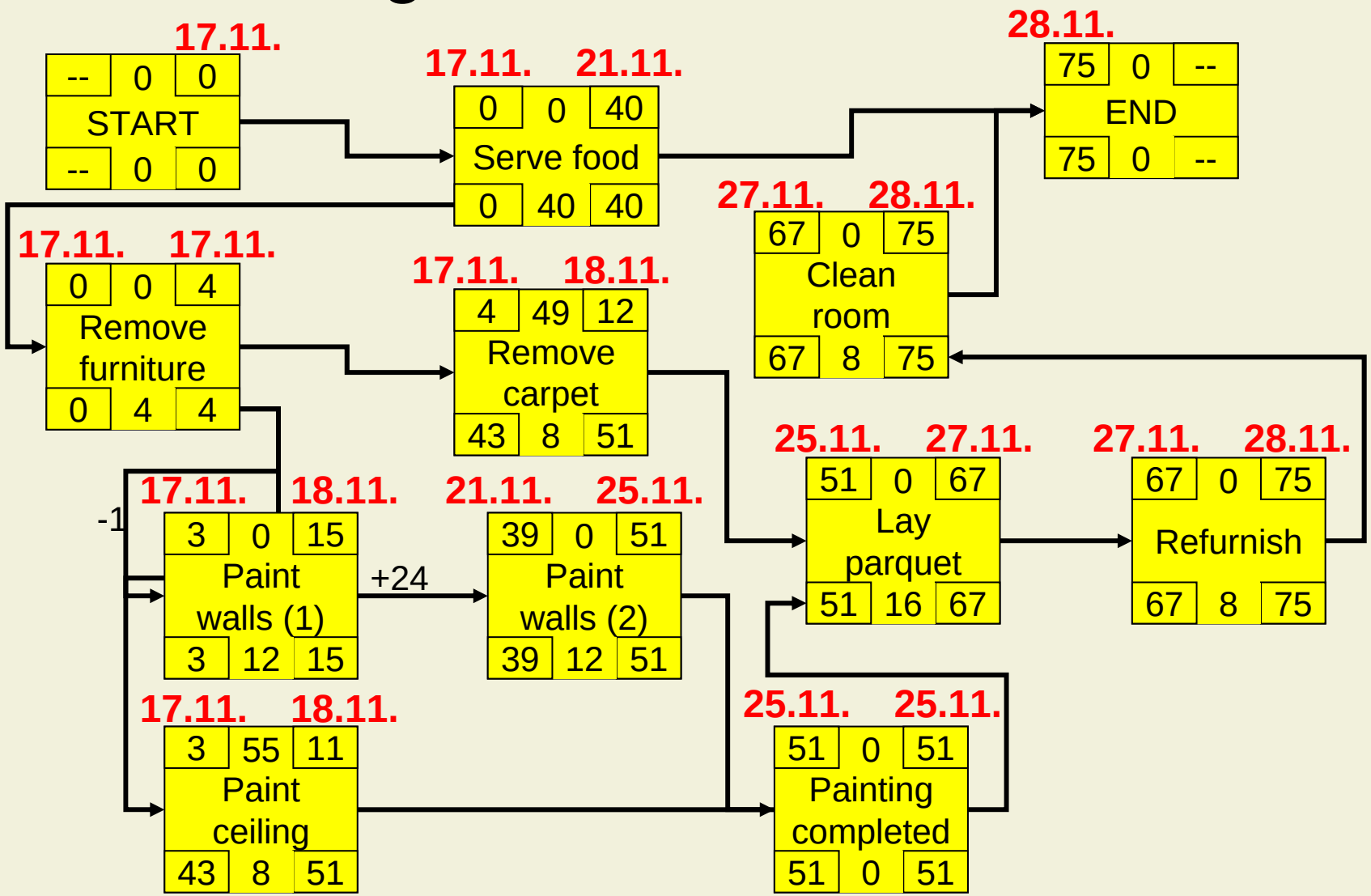
SF-relation:

LF := LF' – L + D

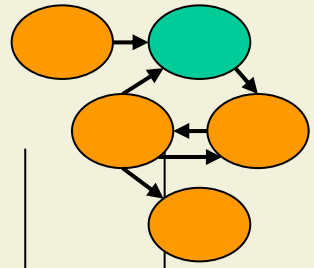
Backward Pass Example

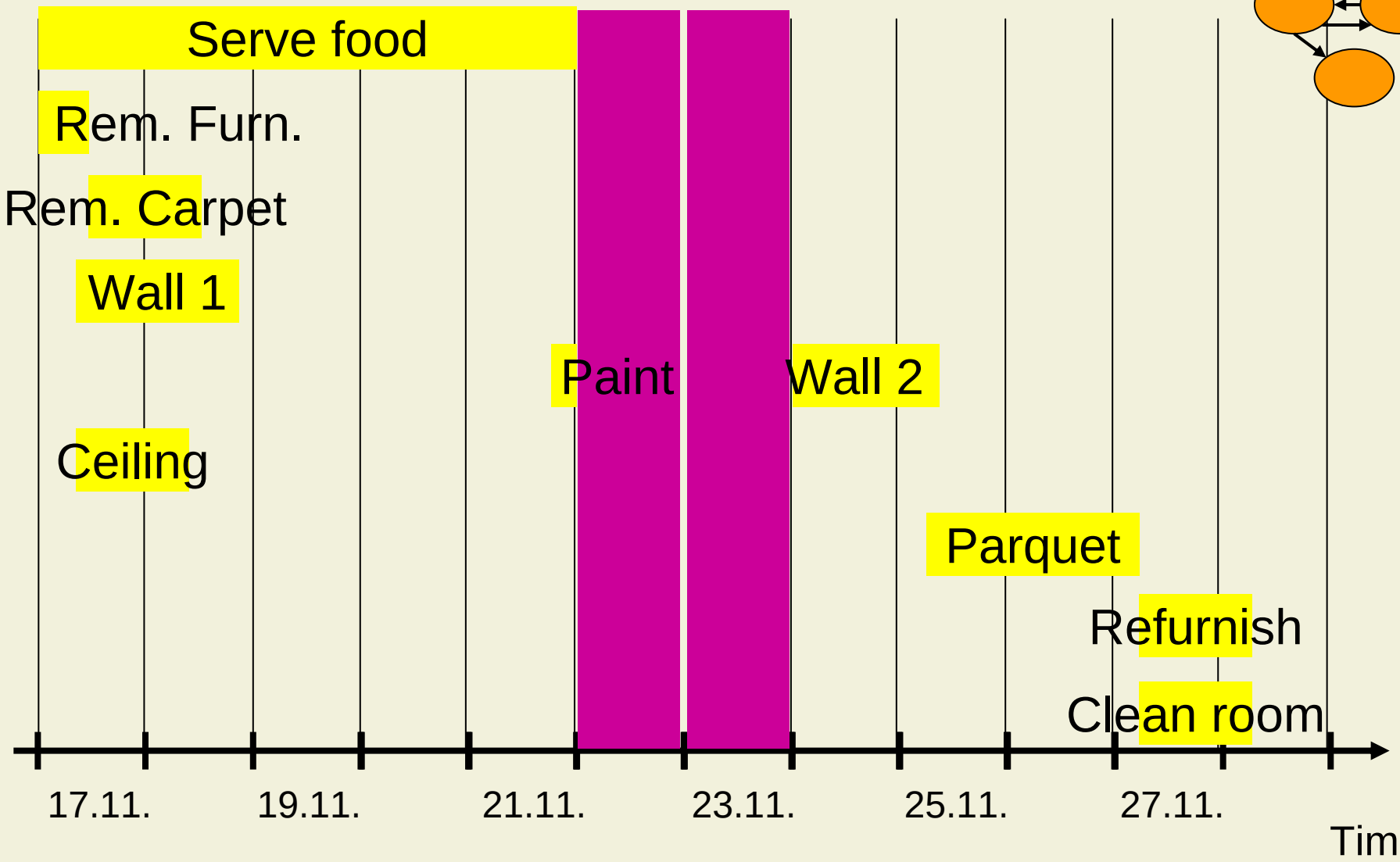


Network Diagrams with Dates

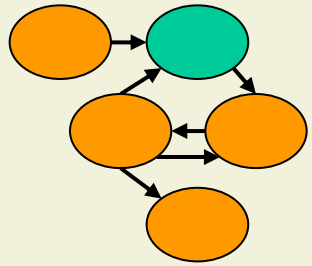


Bar (Gantt) Charts





Milestone Charts

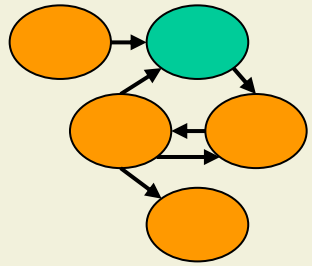


Current Date

Milestone	17.11.	18.11.	19.11.	20.11.	21.11.	22.11.	23.11.	24.11.	25.11.	26.11.	27.11.	28.11.
START ▼												
Painitng completed									△			
END												△

Planned △ Actual ▼

Diagramming Methods

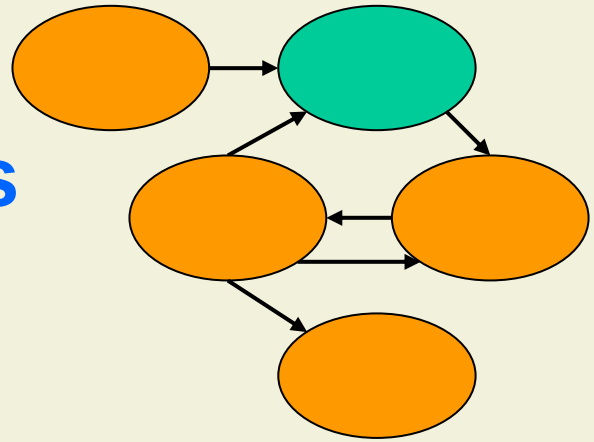


- Network diagrams
  - Show dependencies and workflow
  - Purpose: planning
- Gantt charts
  - Show dates and durations
  - Purpose: reporting and progress tracking
- Milestone charts
  - Show major events
  - Purpose: reporting to management and customer

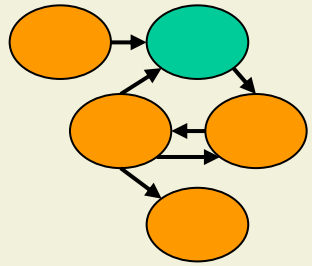
4. Scheduling

4.1 Networks

4.2 Schedule Analysis

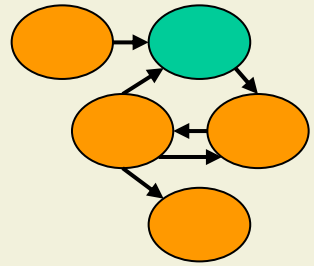


Analyzing a Schedule



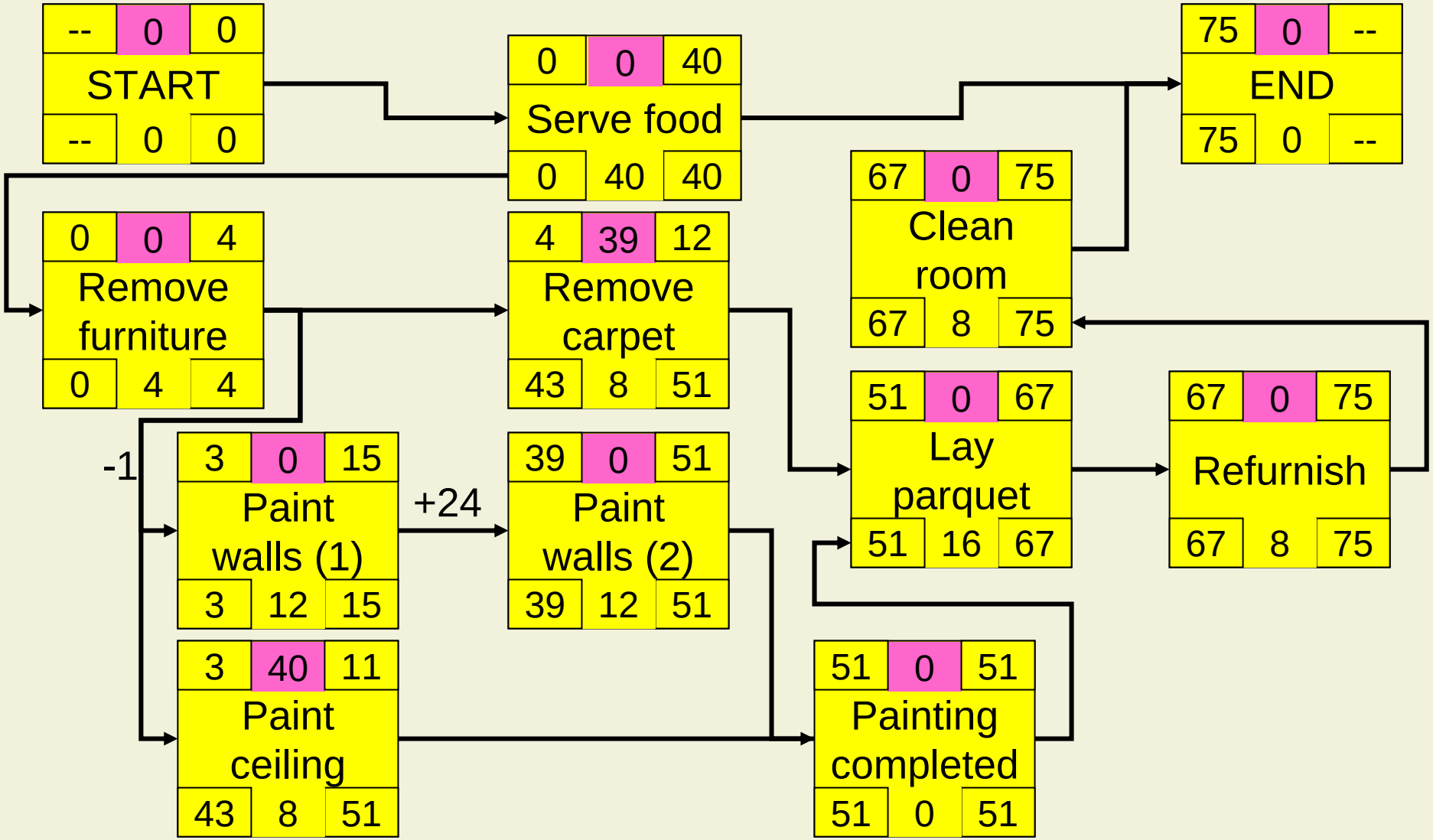
- Identify schedule risks
- Determine if deliverables will be made on time
- Check resource usage
- Find potentials for compressing the schedule
- Consistency

Float

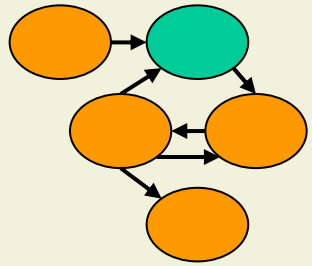


- Definition:
  The amount of time that an activity may be delayed from its early start without delaying the project finish date
- Float = LF – EF = LS – ES
- Interpretation
  - Float > 0: Time is available
  - Float = 0: Situation is critical
  - Float < 0: Project is behind
- Sometimes called *Total Float*, *Slack*, or *Total Slack*

Float Example

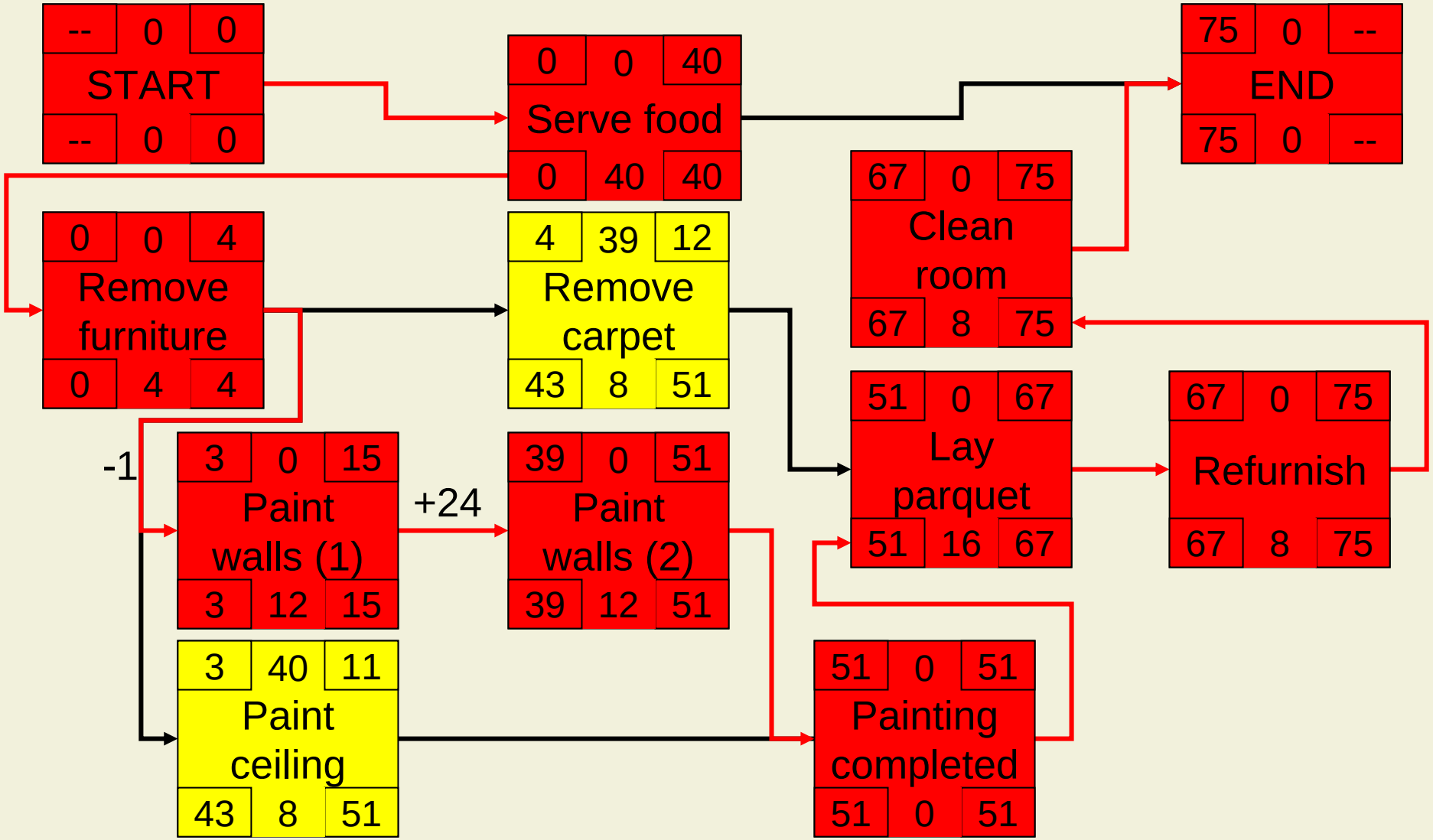


Critical Path

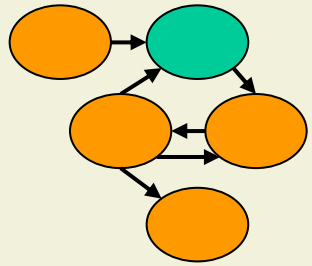


- Definition:
  The series of activities that determines the duration of the project (the longest path through the network)
- Sum of float on critical path is zero (or negative)
- Critical path is important
  - To shorten project duration
  - To focus progress control
  - To identify schedule risks
- There can be several critical paths in a project

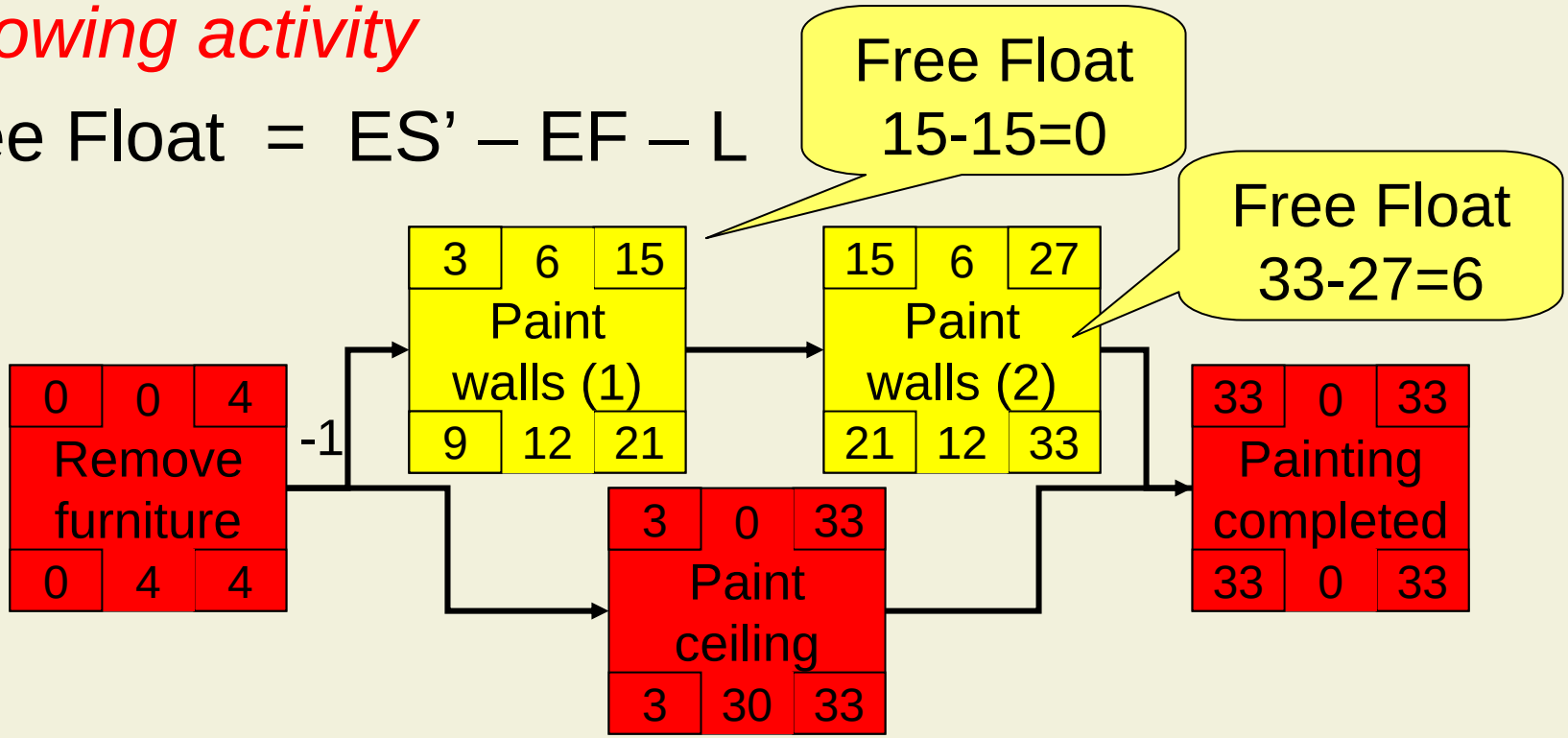
Critical Path Example



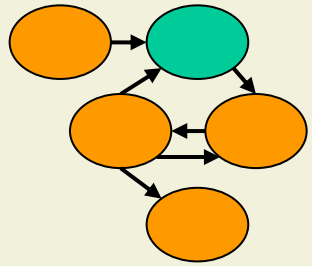
Free Float



- Definition:
  The amount of time that an activity can be delayed without delaying the early start of any immediately following activity
- Free Float = ES' – EF – L

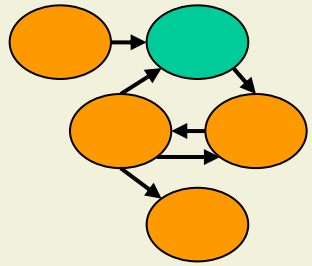


Schedule Compression



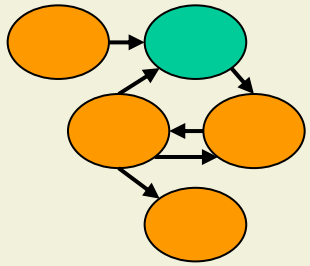
- Fast tracking to shorten critical path
  - Do activities in parallel instead of in sequence
  - Problem: increases risk
- Crashing the network
  - Add resources to the critical path (e.g., from non-critical activities)
  - Problem: Law of diminishing returns
- Increasing productivity by different technology
- Extended hours and weekends should not be considered during planning
  - You will need them during project execution anyway

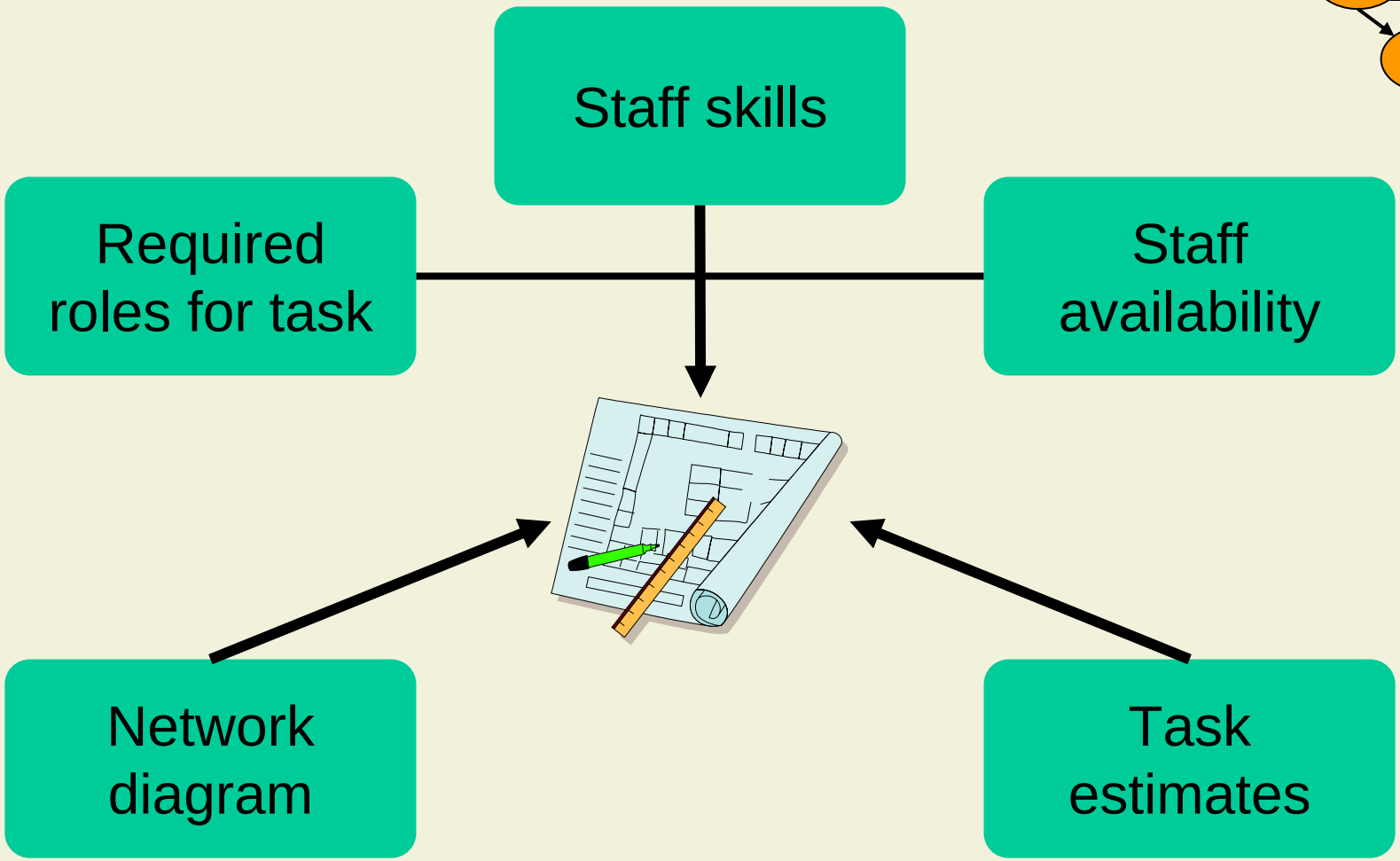
Resource Leveling



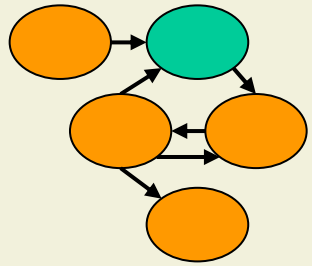
- Common results of critical path method
  - More resources required than available
  - Changes of resource levels are not manageable
- Analysis: Resource histograms
- Heuristic: Resource-based method
  - Allocate scarce resources to critical path first
- Resource leveling usually leads to longer project duration

Consistency





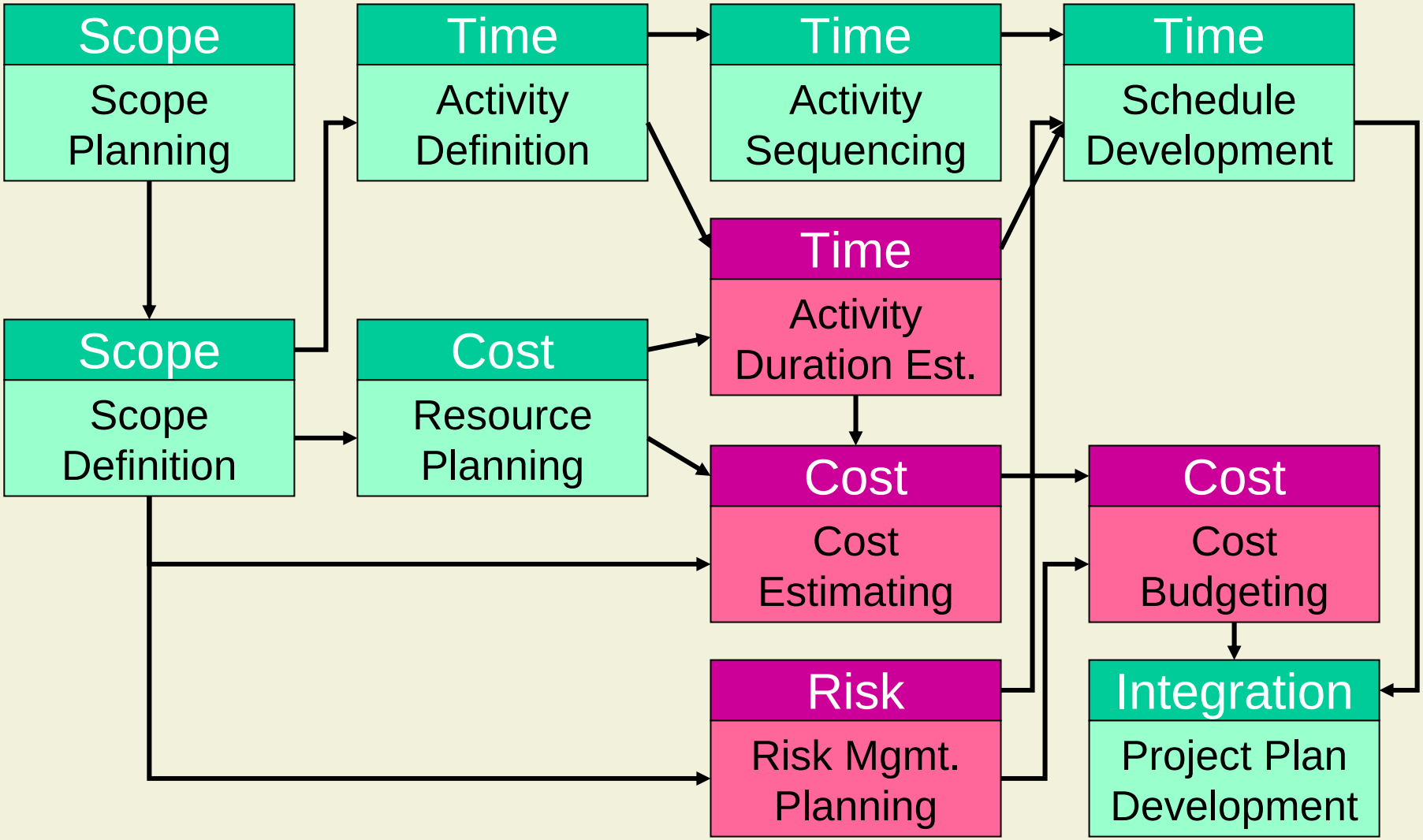
Schedule Development: Summary



- Purpose
  - To determine start and finish dates for project activities
- Schedule development is often iterated, not only during planning

Inputs	Tools & Techniques	Outputs
1. Network diagrams 2. Activity duration estimations 3. Resource requirements 4. Calendars	1. Mathematical analysis (CPM) 2. Duration compression 3. Resource leveling heuristics 4. PM Software	1. Project schedule 2. Schedule management plan 3. Resource requirement updates

Main Planning Processes



Systematics of Processes

	Initiating	Planning	Executing	Controlling	Closing
Integration		Project Plan Dev.	Project Plan Execution	Integr. Change Control	
Scope	Initiation	Scope Planning Scope Definition			
Time		Act. Definition, Act. Sequencing, Schedule Dev.			
Cost		Resource Planning			
Quality					
HR					
Comm.					
Risk					
Procurement					

BACKUP