Chapter Six

Developing a Project Plan

Developing the Project Network

- The Project Network
  - A flow chart that graphically depicts the logical sequences, interdependencies, and start and finish times of the project activities along with the longest path(s) through the network—the ***critical path***
    - Provides the basis for scheduling labor and equipment.
    - Enhances communication among project participants.
    - Provides an estimate of the project's duration.
    - Provides a basis for budgeting cash flow.
    - Identifies activities that are critical.
    - Highlights activities that are "critical" and should not be delayed.
    - Help managers get and stay on plan.

From WBS/Work Package to Network

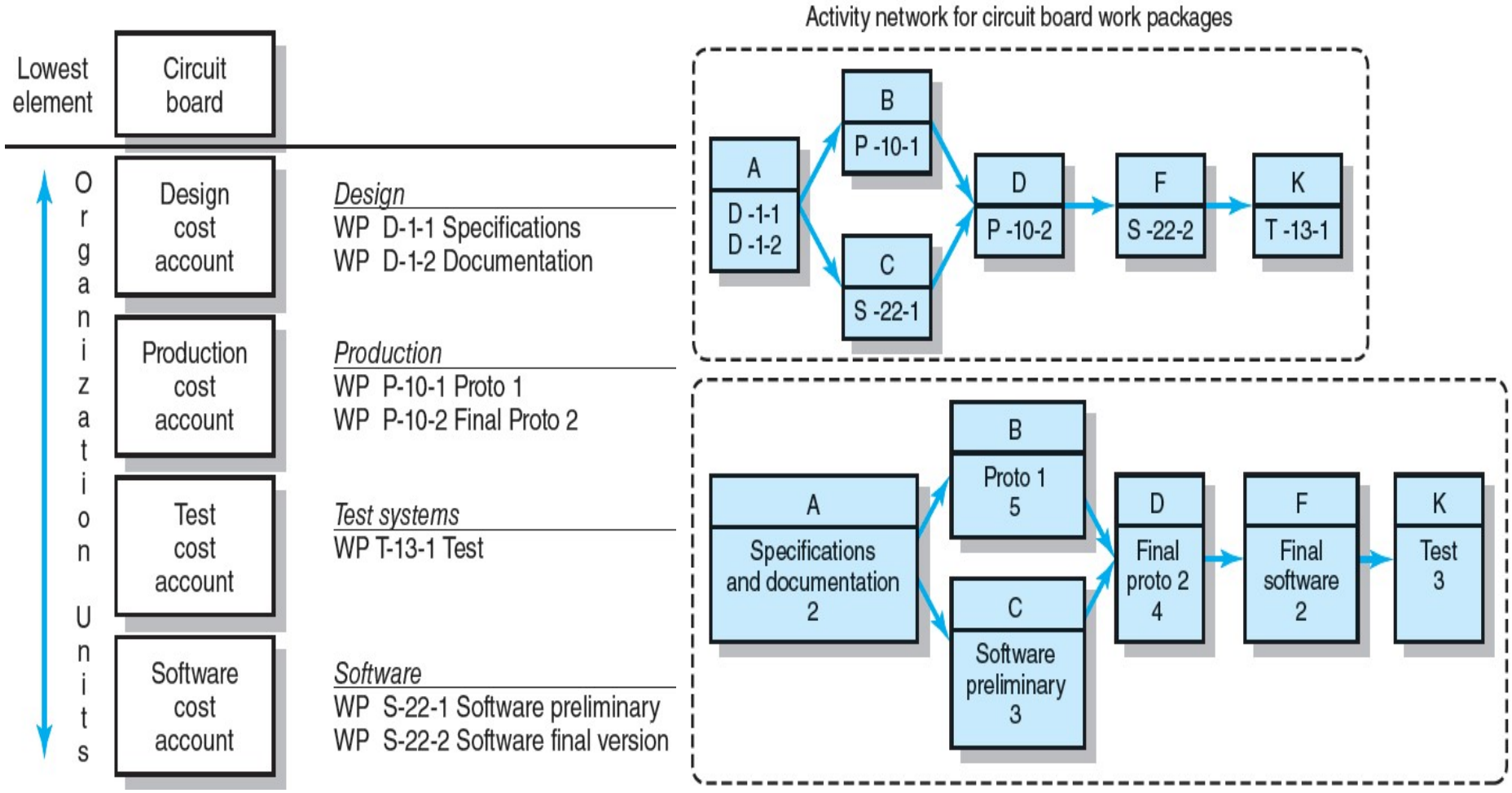
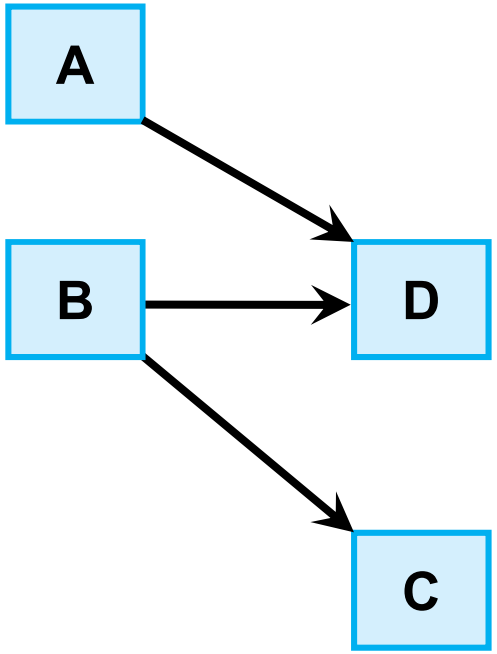


FIGURE 6.1

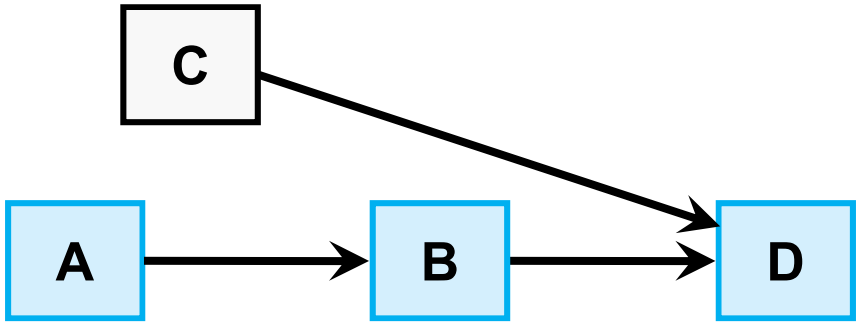
Constructing a Project Network

- Terminology
  - **Activity:** an element of the project that requires time but may not require resources
  - **Merge Activity:** an activity that has two or more preceding activities on which it depends (more than one dependency arrow flowing into it)
  - **Parallel Activities:** Activities that can occur independently and, if desired, not at the same time



Constructing a Project Network (cont'd)

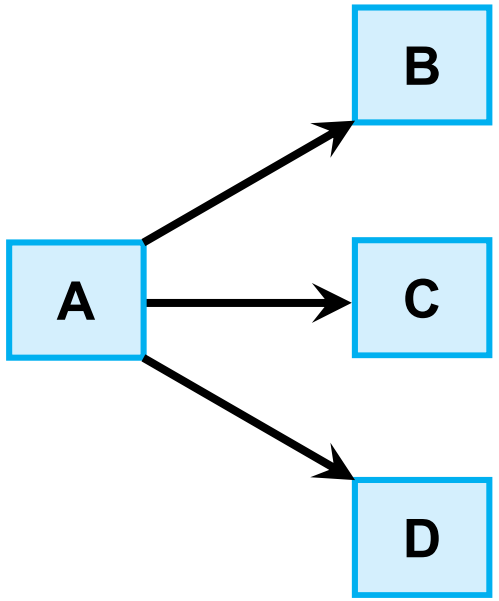
- Terminology
  - **Path:** a sequence of connected, dependent activities
  - **Critical Path:**
    - The longest path through the activity network that allows for the completion of all project-related activities
    - The shortest expected time in which the entire project can be completed.
    - Delays on the critical path will delay completion of the entire project.



min A + B < min C in length of times to complete activities

Constructing a Project Network (cont'd)

- Terminology
  - **Burst Activity:** an activity that has more than one activity immediately following it (more than one dependency arrow flowing from it)



- Two Approaches
  - Activity-on-Node (AON)
    - Uses a node to depict an activity.
  - Activity-on-Arrow (AOA)
    - Uses an arrow to depict an activity.

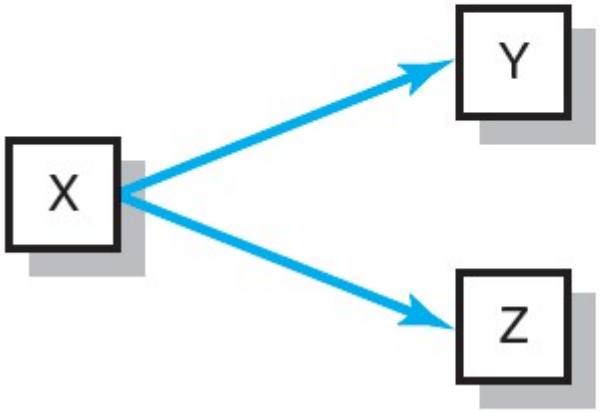
Basic Rules to Follow in Developing Project Networks

1. Networks typically flow from left to right.
2. An activity cannot begin until all preceding connected activities are complete.
3. Arrows indicate precedence and flow and can cross over each other.
4. Each activity must have a unique identify number.
5. An activity identification number must be greater than that of any predecessor activities.
6. Looping is not allowed.
7. Conditional statements are not allowed.
8. Use common start and stop nodes.

Activity-on-Node Fundamentals

A is preceded by nothing
B is preceded by A
C is preceded by B

(A)



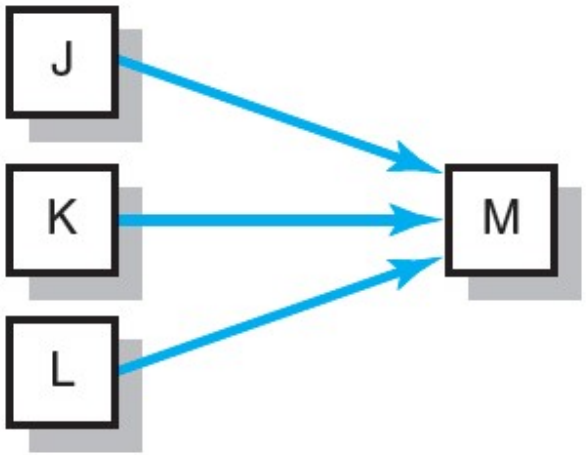
Y and Z are preceded by X

Y and Z can begin at the same time, if you wish

(B) X is a burst activity

FIGURE 6.2

Activity-on-Node Fundamentals (cont'd)

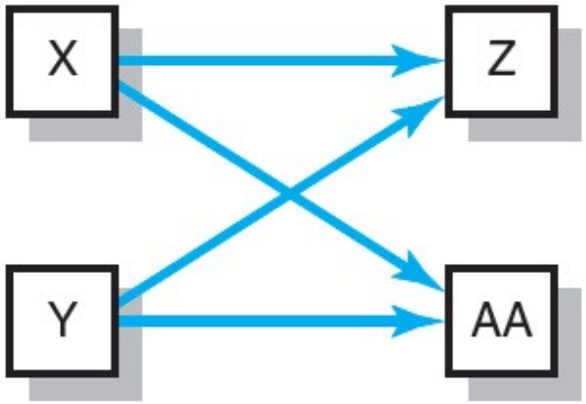


J, K, & L can all begin at the same time, if you wish (they need not occur simultaneously)

but

All (J, K, L) must be completed before M can begin

(C) M is a merge activity



Z is preceded by X and Y

AA is preceded by X and Y

(D)

FIGURE 6.2 (cont'd)

Network Information

AUTOMATED WAREHOUSE
Order Picking System

Activity	Description	Preceding Activity
A	Define Requirements	None
B	Assign Team	A
C	Design Hardware	A
D	Code Software	B
E	Build and Test Hardware	C
F	Develop Patent Request	C
G	Test Software	D
H	Integrate Systems	E, F, G

TABLE 6.1

Automated Warehouse—Partial Network

AUTOMATED WAREHOUSE
Order Picking System

Activity	Description	Preceding Activity
A	Define Requirements	None
B	Assign Team	A
C	Design Hardware	A
D	Code Software	B
E	Build and Test Hardware	C
F	Develop Patent Request	C
G	Test Software	D
H	Integrate Systems	E, F, G

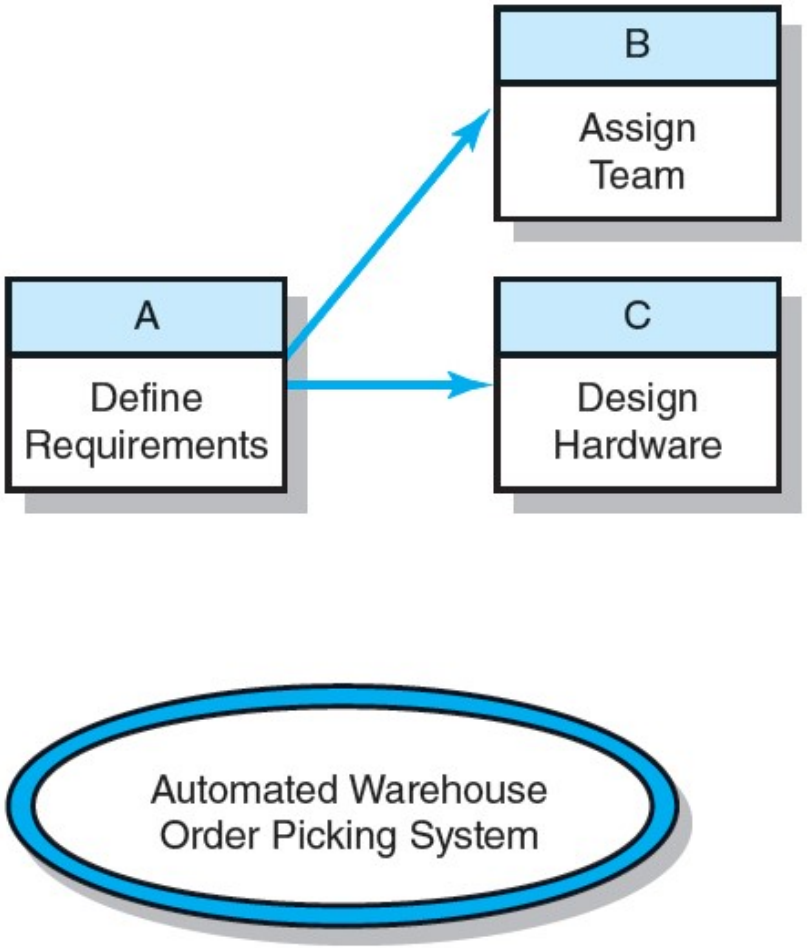


FIGURE 6.3

Automated Warehouse—Complete Network

AUTOMATED WAREHOUSE
Order Picking System

Activity	Description	Preceding Activity
A	Define Requirements	None
B	Assign Team	A
C	Design Hardware	A
D	Code Software	B
E	Build and Test Hardware	C
F	Develop Patent Request	C
G	Test Software	D
H	Integrate Systems	E, F, G

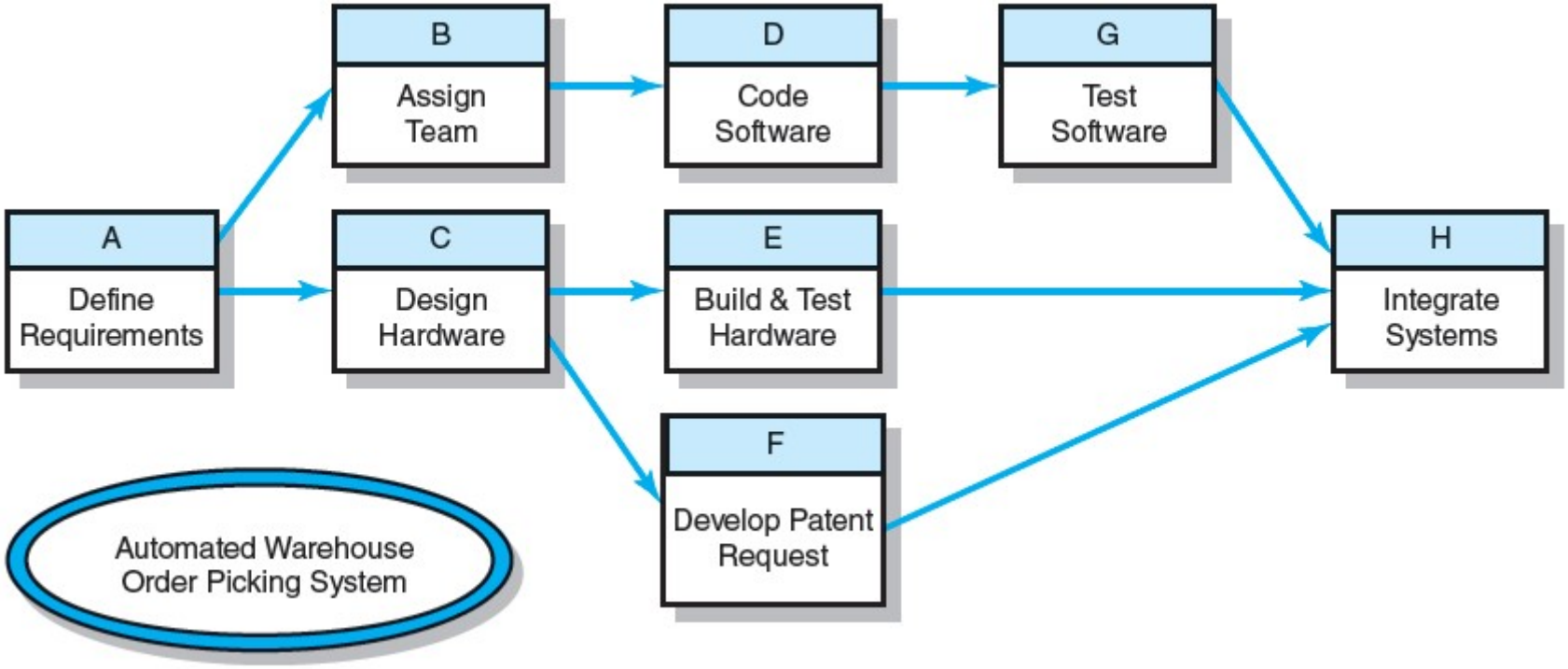


FIGURE 6.4

Network Computation Process

- Forward Pass—Earliest Times
  - How soon can the activity start? (early start—ES)
  - How soon can the activity finish? (early finish—EF)
  - How soon can the project finish? (expected time—TE)
- Backward Pass—Latest Times
  - How late can the activity start? (late start—LS)
  - How late can the activity finish? (late finish—LF)
  - Which activities represent the critical path?
  - How long can the activity be delayed? (slack or float—SL)

Activity-on-Node Network

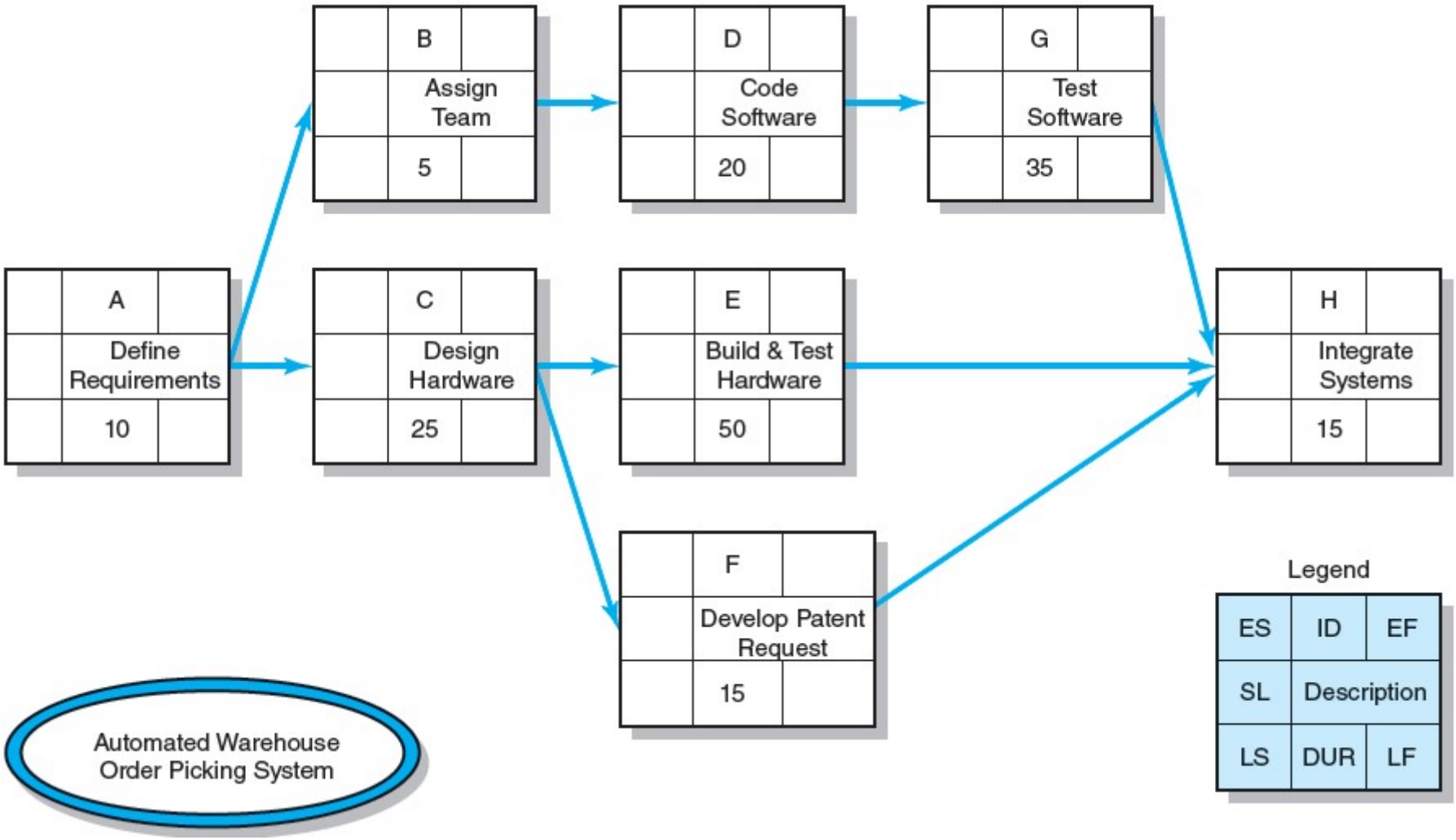


FIGURE 6.5

Forward Pass Computation

- Add activity times along each path in the network (ES + Duration = EF).
- Carry the early finish (EF) to the next activity where it becomes its early start (ES) ***unless…***
- The next succeeding activity is a *merge* activity, in which case the *largest* early finish (EF) number of all its immediate predecessor activities is selected.

Activity-on-Node Network Forward Pass

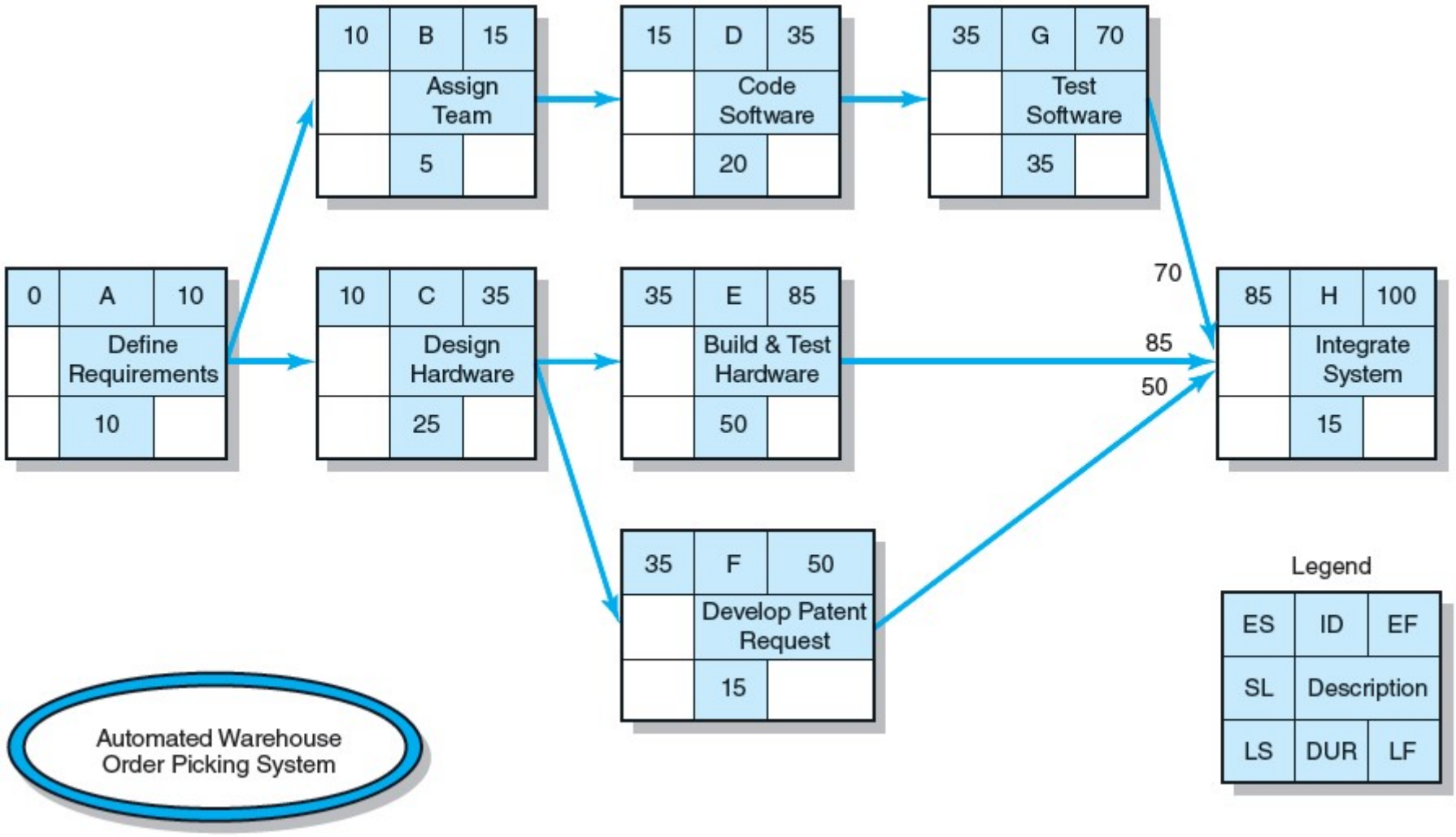


FIGURE 6.6

Backward Pass Computation

- Subtract activity times along each path starting with the project end activity (LF - Duration = LS).
- Carry the late start (LS) to the next preceding activity where it becomes its late finish (LF) ***unless…***
- The next succeeding activity is a *burst* activity, in which case the *smallest* late start (LS) number of all its immediate successor activities is selected.

Activity-on-Node Network Backward Pass

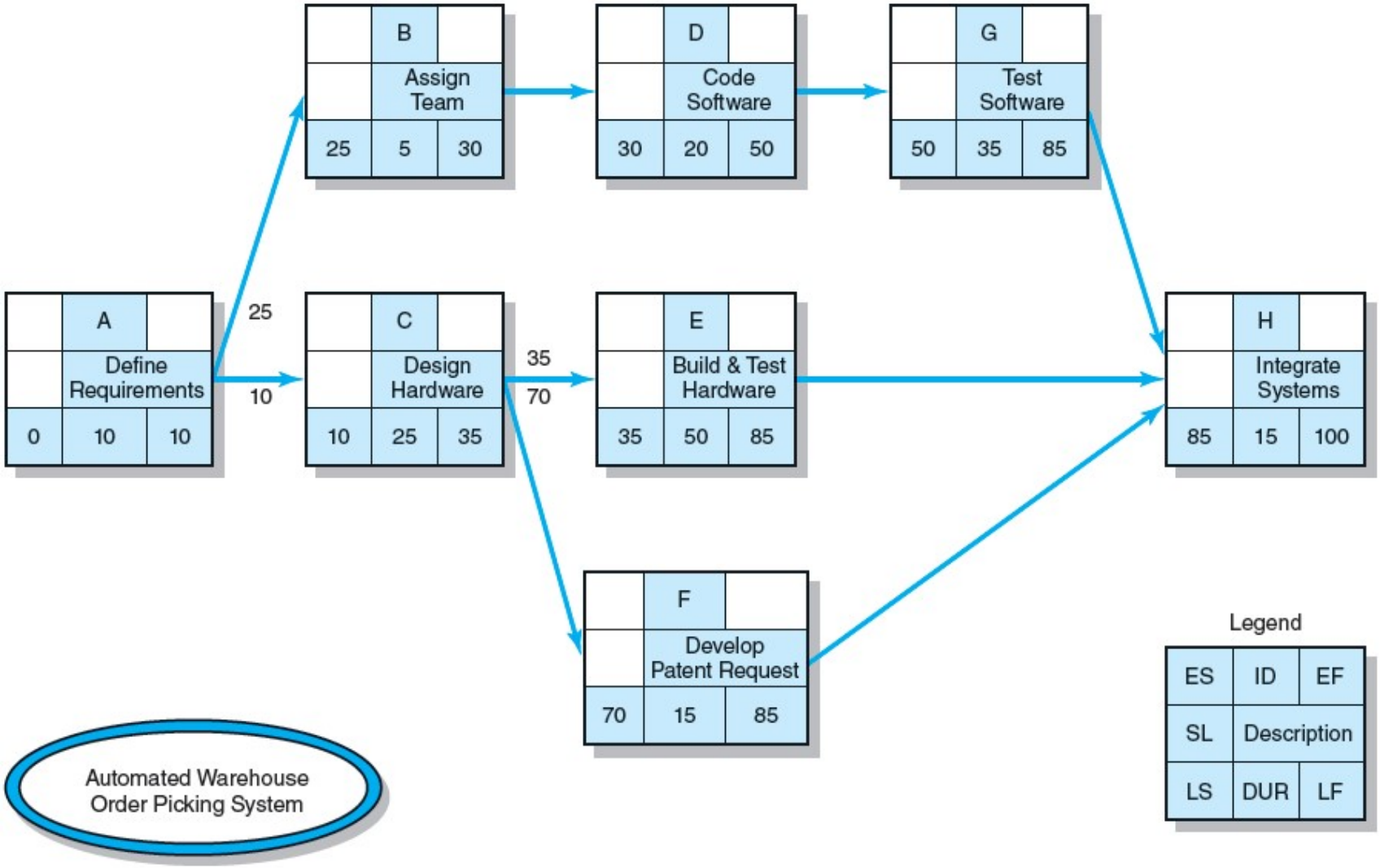


FIGURE 6.7

Determining Total Slack (TS)

- Total Slack (or Float)
  - Tells us the amount of time an activity can be delayed and not delayed the project.
  - Is how long an activity can exceed its early finish date without affecting the project end date or an imposed completion date.
  - Is simply the difference between the LS and ES (LS – ES = SL) or between LF and EF (LF – EF = SL).

Determining Free Slack (FS)

- Free Slack (or Float)
  - Is the amount of time an activity can be delayed after the start of a longer parallel activity or activities.
  - Is how long an activity can exceed its early finish date without affecting early start dates of any successor(s).
  - Allows flexibility in scheduling scarce resources.
  - Only activities that occur at the end of a chain of activities, where you have a merge activity, can have free slack.

Forward and Backward Passes Completed with Slack Times

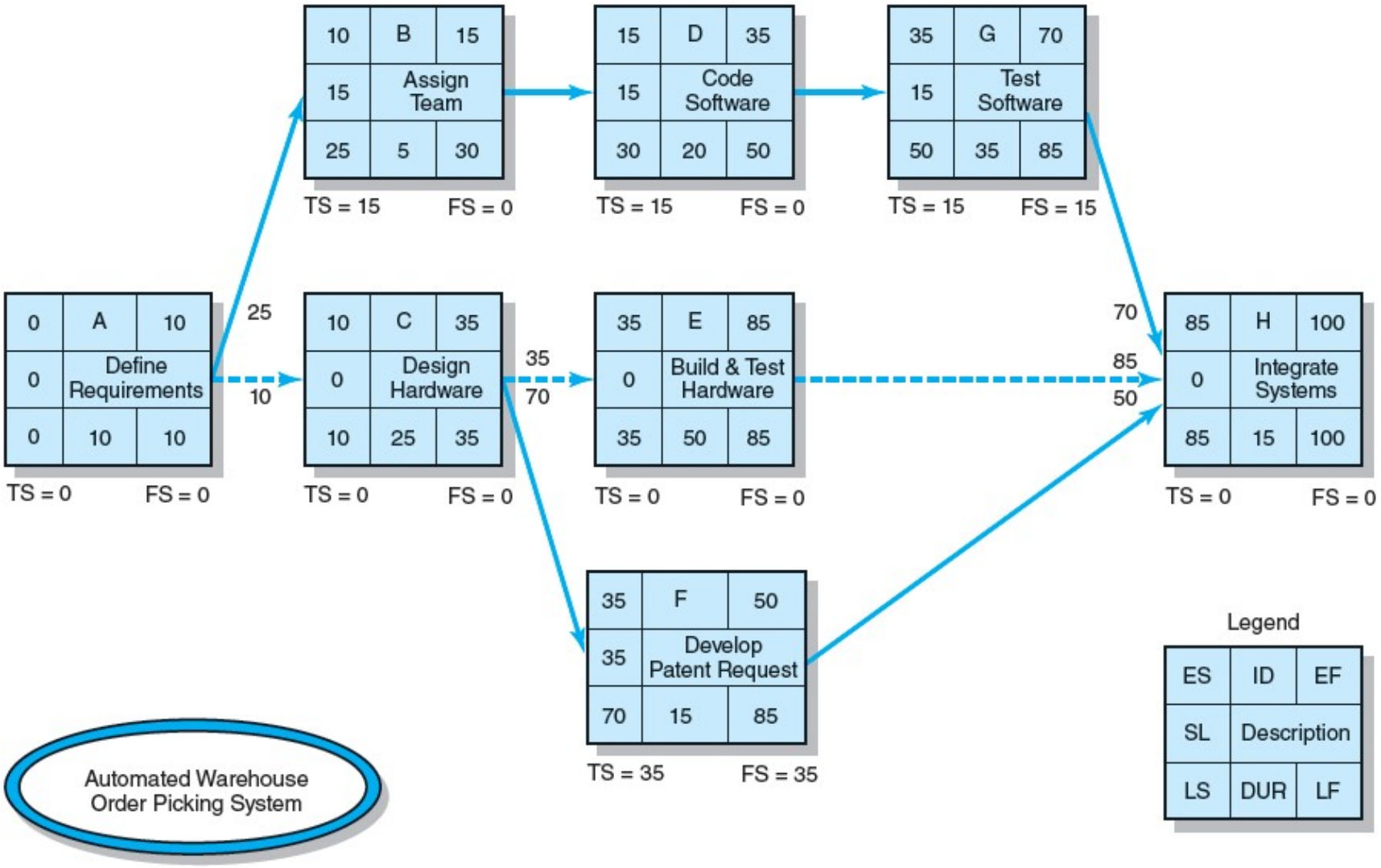


FIGURE 6.8

The Critical Path

- Is the network path(s) that has (have) the least slack in common.
- Is the longest path through the activity network.
- Is the shortest expected time in which the entire project can be completed.
- Is important because it impacts completion time.
- Is where you put best people on.
- Is where you pay extra extension when doing risk assessment.
- Is where you look when other managers asking to ‘borrow’ people or equipment.
- Is where you look when you don’t have time to monitor all activities.

Automated Warehouse Order Picking System Network

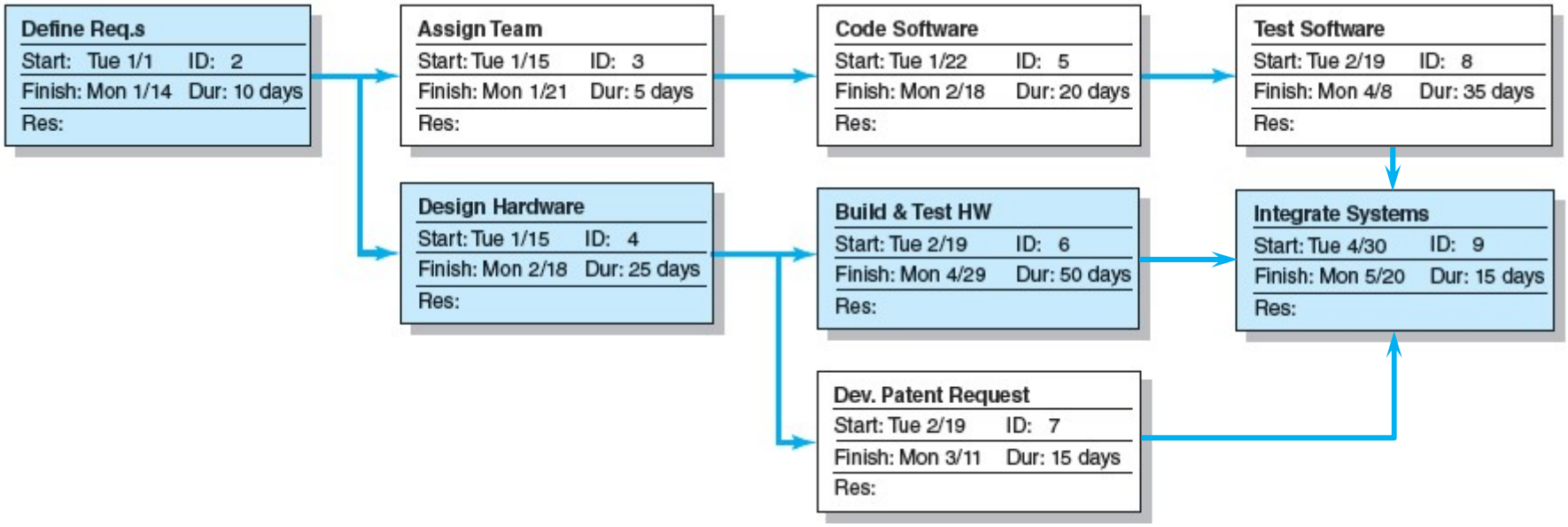


FIGURE 6.10

Automated Order Warehouse Picking System Bar Chart

Automated Warehouse Order Picking System Schedule Table

ID	Duration	Task Name	Start	Finish	Late Start	Late Finish	Free Slack	Total Slack
2	10	Define Req.s	Tue 1/1	Mon 1/14	Tue 1/1	Tue 1/15	0 days	0 days
3	5	Assign Team	Tue 1/15	Mon 1/21	Tue 2/5	Mon 2/11	0 days	15 days
4	25	Design Hardware	Tue 1/15	Mon 2/18	Tue 1/15	Mon 2/18	0 days	0 days
5	20	Code Software	Tue 1/22	Mon 2/18	Tue 2/12	Mon 3/11	0 days	15 days
6	50	Build and Test HW	Tue 2/19	Mon 4/29	Tue 2/19	Mon 4/29	0 days	0 days
7	15	Dev. Patent Request	Tue 2/19	Mon 3/11	Tue 4/9	Mon 4/29	35 days	35 days
8	35	Test Software	Tue 2/19	Mon 4/8	Tue 3/12	Mon 4/29	15 days	15 days
9	15	Integrate Systems	Tue 4/30	Mon 5/20	Tue 4/30	Mon 5/20	0 days	0 days

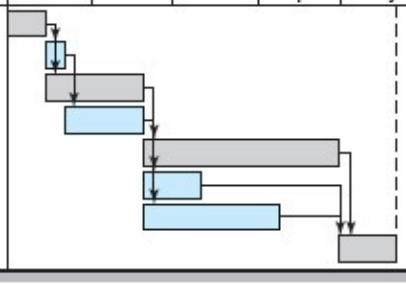


FIGURE 6.11

Extended Network Techniques to Come Close to Reality

- Laddering
  - Activities are broken into segments so the following activity can begin sooner and not delay the work.
- Lags
  - The minimum amount of time a dependent activity must be delayed to begin or end.
    - Lengthy activities are broken down to reduce the delay in the start of successor activities.
    - Lags can be used to constrain finish-to-start, start-to-start, finish-to-finish, start-to-finish, or combination relationships.

Example of Laddering Using Finish-to-Start Relationship

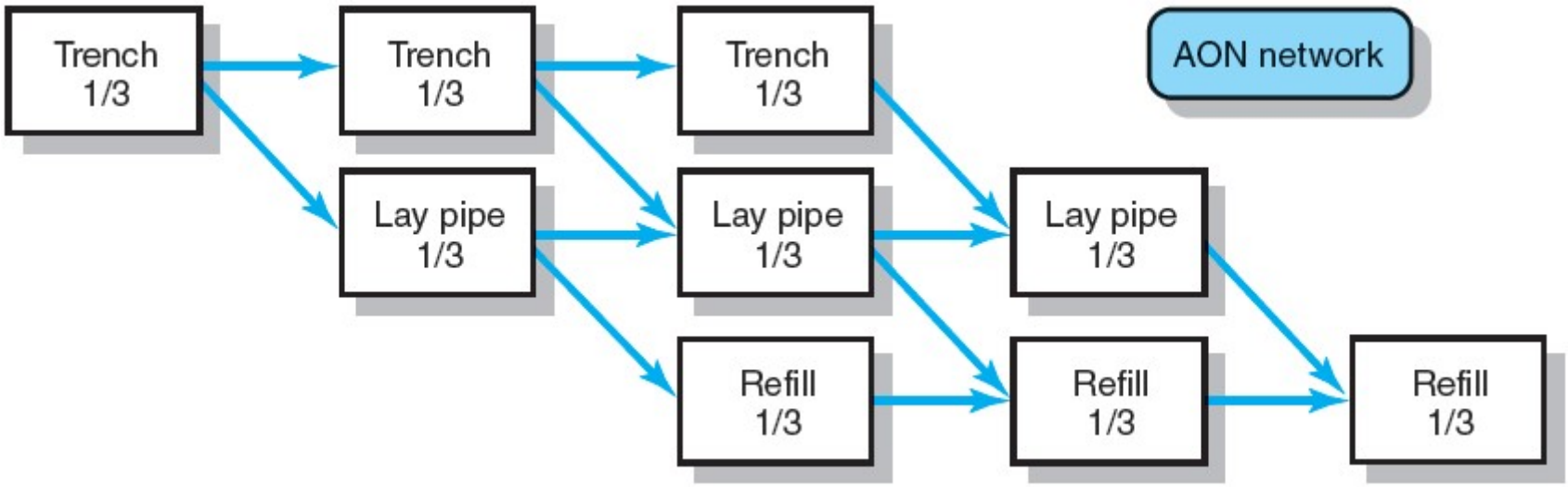


FIGURE 6.12

Use of Lags

Finish-to-Start Relationship

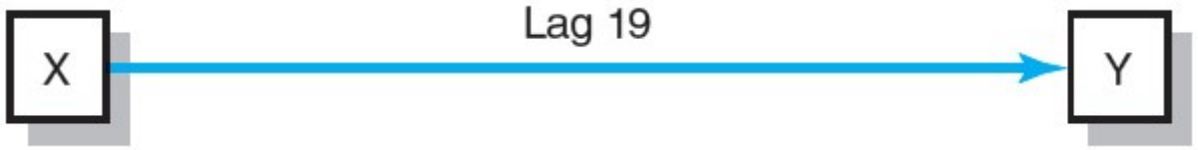


FIGURE 6.13

Start-to-Start Relationship

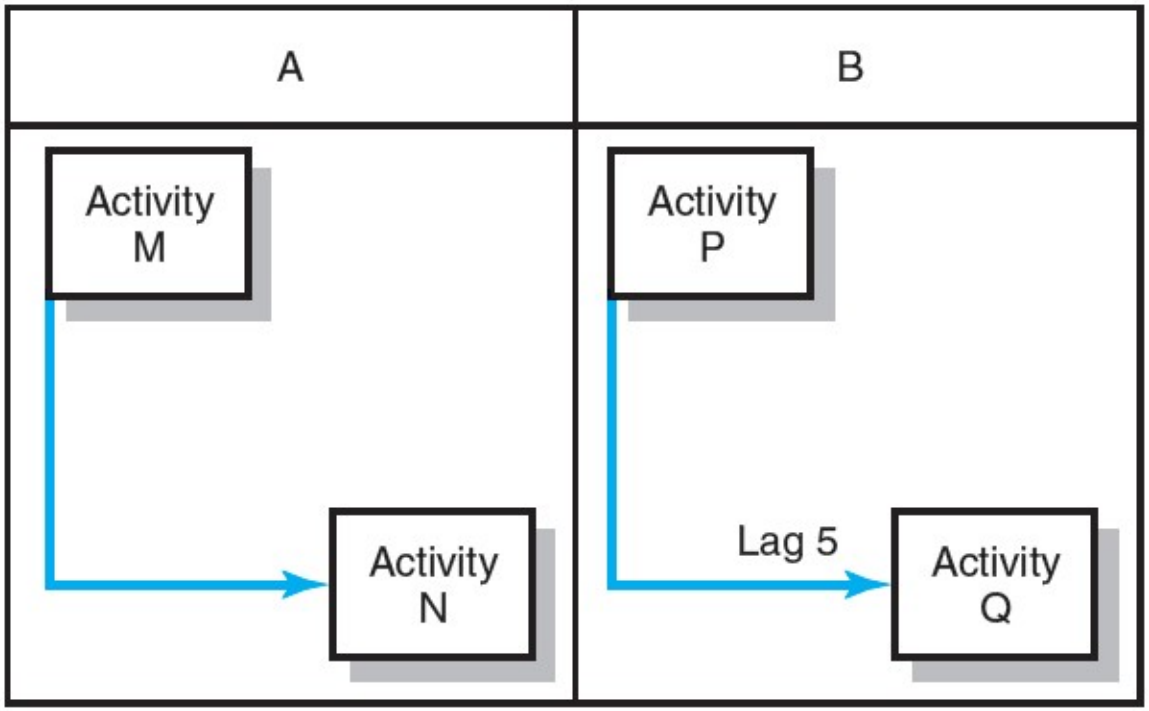
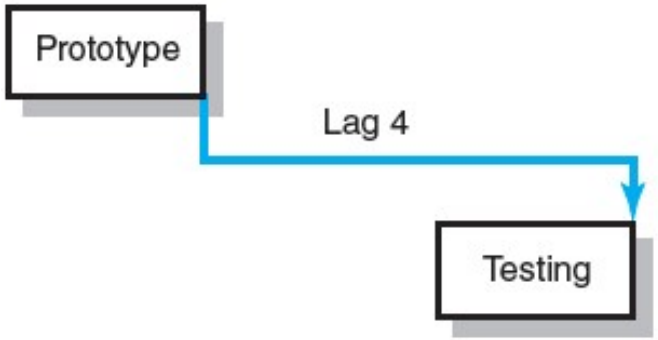


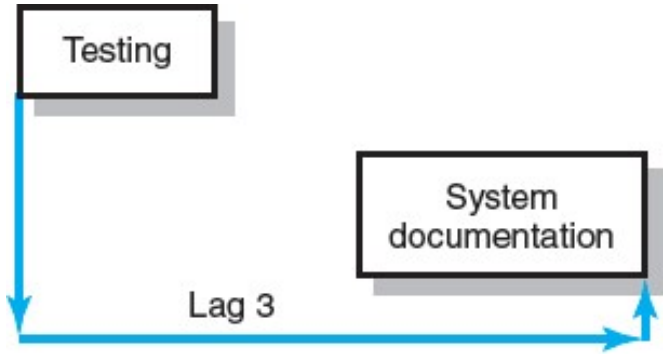
FIGURE 6.14

Use of Lags (cont'd)



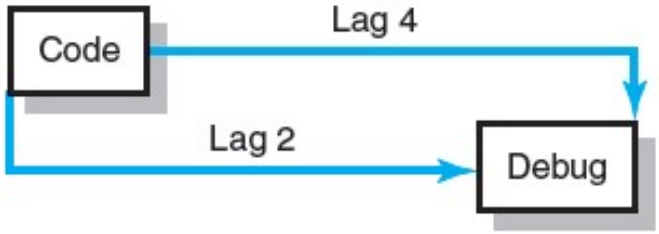
Finish-to-Finish Relationship

FIGURE 6.17



Start-to-Finish Relationship

FIGURE 6.18



Combination Relationships

FIGURE 6.19

Network Using Lags

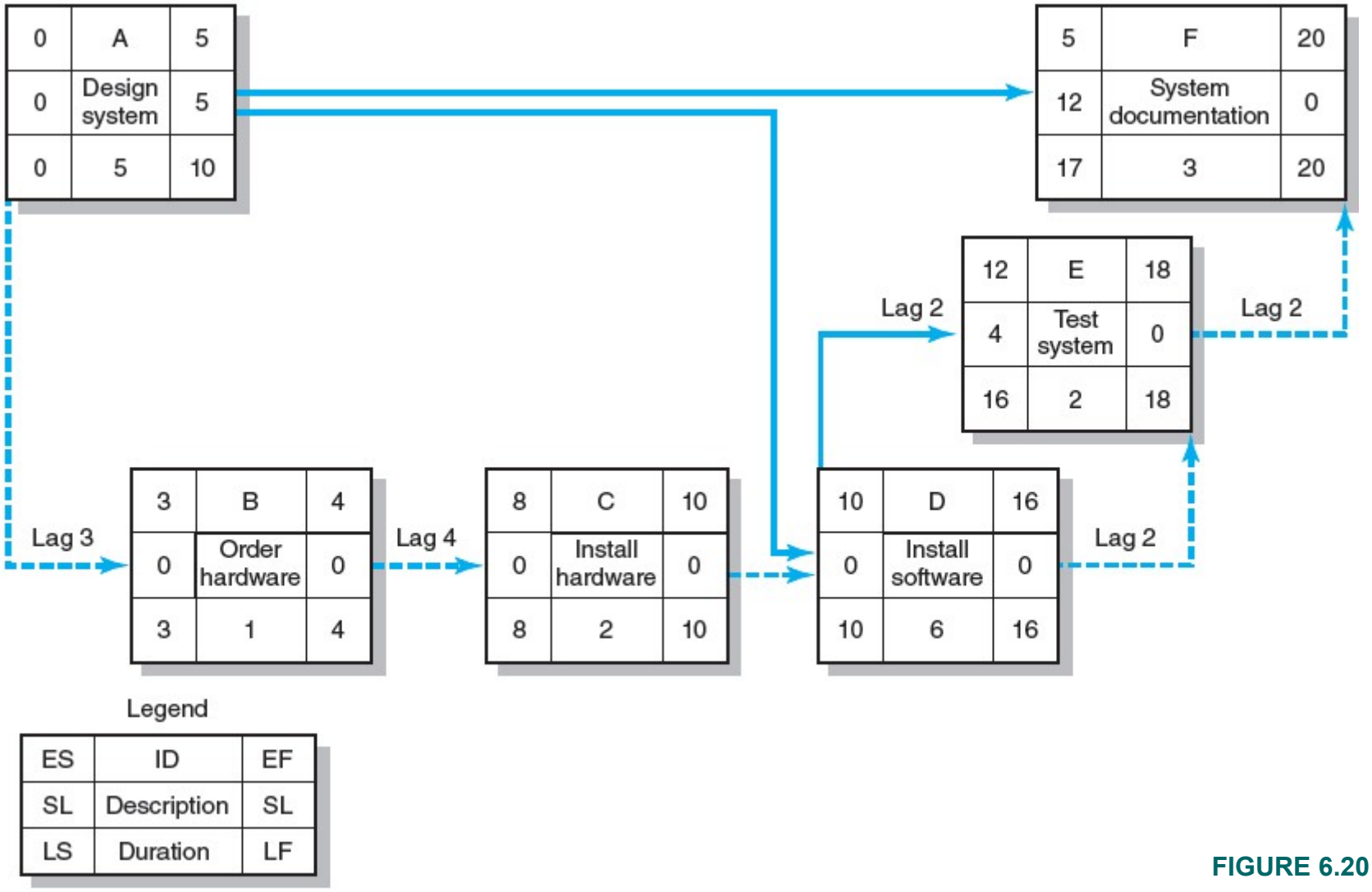


FIGURE 6.20

Hammock Activities

- Hammock Activity
  - Spans over a segment of a project.
  - Has a duration that is determined after the network plan is drawn.
  - Is very useful in assigning and controlling indirect project costs.
  - Is used to aggregate sections of the project to facilitate getting the right level of detail for specific sections of a project.

Hammock Activity Example

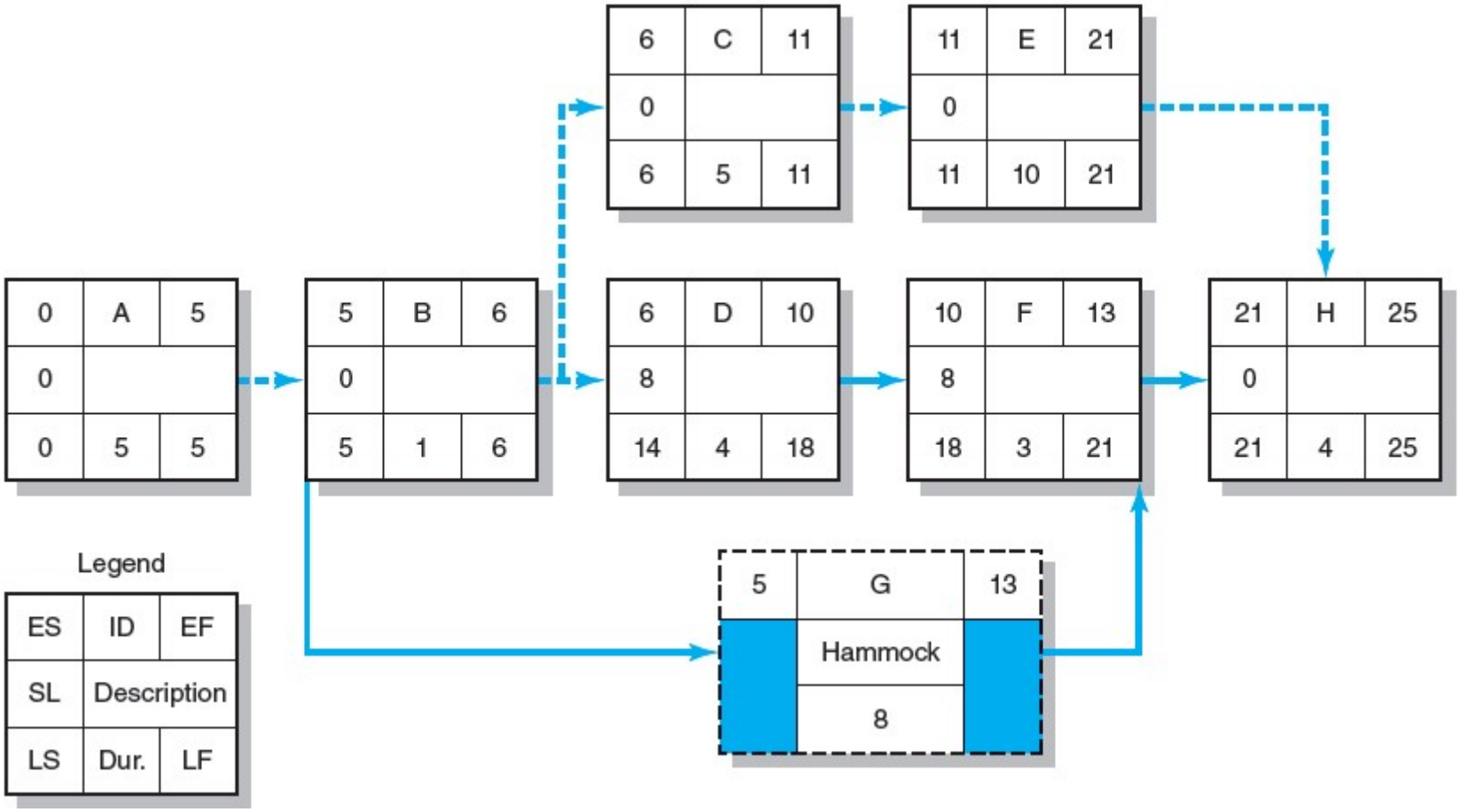


FIGURE 6.21