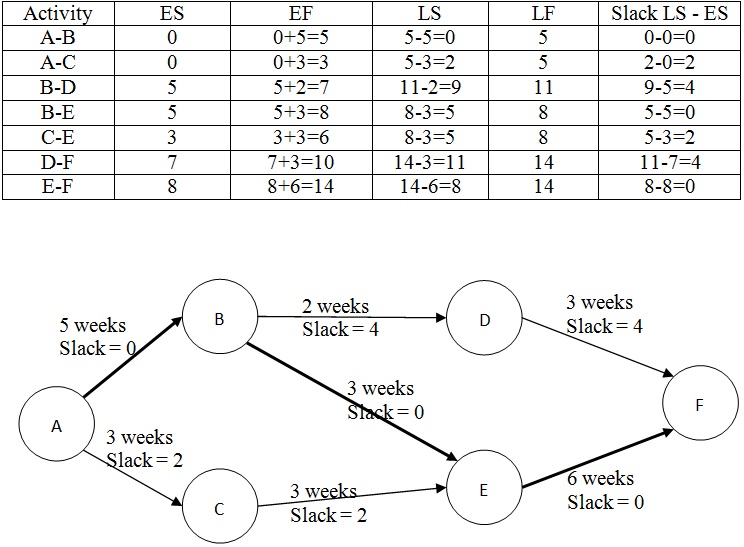
# Schedule Network Diagram



* + - 1. **PRECEDENCE DIAGRAMMING METHOD**

The precedence diagramming method (PDM) is a technique used for constructing a schedule model in which activities are represented by nodes and are graphically linked by one or more logical relationships to show the sequence in which the activities are to be performed.

PDM includes four types of dependencies or logical relationships. A predecessor activity is an activity that logically comes before a dependent activity in a schedule. A successor activity is a dependent activity that logically comes after another activity in a schedule. These relationships are defined below and are illustrated in Figure 6-9:

* + - * + **Finish-to-start (FS).** A logical relationship in which a successor activity cannot start until a predecessor activity has finished. For example, installing the operating system on a PC (successor) cannot start until the PC hardware is assembled (predecessor).
        + **Finish-to-finish (FF).** A logical relationship in which a successor activity cannot finish until a predecessor activity has finished. For example, writing a document (predecessor) is required to finish before editing the document (successor) can finish.
        + **Start-to-start (SS).** A logical relationship in which a successor activity cannot start until a predecessor activity has started. For example, level concrete (successor) cannot begin until pour foundation (predecessor) begins.
        + **Start-to-finish (SF).** A logical relationship in which a successor activity cannot finish until a predecessor activity has started. For example, a new accounts payable system (successor) has to start before the old accounts payable system can be shut down (predecessor).

In PDM, FS is the most commonly used type of precedence relationship. The SF relationship is very rarely used, but is included to present a complete list of the PDM relationship types.

Two activities can have two logical relationships at the same time (for example, SS and FF). Multiple relationships between the same activities are not recommended, so a decision has to be made to select the relationship with the highest impact. Closed loops are also not recommended in logical relationships.

istribu

**Activity A**

Finish to Start (FS)

Start to Start (SS)

Finish to Finish (FF)

Start to Finish (SF)

**Activity B**

**Activity A**

**Activity B**

**Activity B**

**Activity A**

**Activity A**

**Activity B**

**Figure 6-9. Precedence Diagramming Method (PDM) Relationship Types**

* + - 1. **DEPENDENCY DETERMINATION AND INTEGRATION**

Dependencies may be characterized by the following attributes: mandatory or discretionary, internal or external (as described below). Dependency has four attributes, but two can be applicable at the same time in the following ways: mandatory external dependencies, mandatory internal dependencies, discretionary external dependencies, or discretionary internal dependencies.

* + - * + **Mandatory dependencies.** Mandatory dependencies are those that are legally or contractually required or inherent in the nature of the work. Mandatory dependencies often involve physical limitations, such as on a construction project, where it is impossible to erect the superstructure until after the foundation has been built, or on an electronics project, where a prototype has to be built before it can be tested. Mandatory dependencies are sometimes referred to as hard logic or hard dependencies. Technical dependencies may not be mandatory. The project team determines which dependencies are mandatory during the process of sequencing the activities. Mandatory dependencies should not be confused with assigning schedule constraints in the scheduling tool.
        + **Discretionary dependencies.** Discretionary dependencies are sometimes referred to as preferred logic, preferential logic, or soft logic. Discretionary dependencies are established based on knowledge of best practices within a particular application area or some unusual aspect of the project where a specific sequence is desired, even though there may be other acceptable sequences. For example, generally accepted best practices recommend that during construction, the electrical work should start after finishing the plumbing work. This order is not mandatory and both activities may occur at the same time (in parallel), but performing the activities in sequential order reduces the overall project risk. Discretionary dependencies should be fully documented since they can create arbitrary total float values and can limit later scheduling options. When fast tracking techniques are employed, these discretionary dependencies should be reviewed and considered for modification or removal. The project team determines which dependencies are discretionary during the process of sequencing the activities.
        + **External dependencies.** External dependencies involve a relationship between project activities and non- project activities. These dependencies are usually outside of the project team’s control. For example, the testing activity in a software project may be dependent on the delivery of hardware from an external source, or governmental environmental hearings may need to be held before site preparation can begin on a construction project. The project management team determines which dependencies are external during the process of sequencing the activities.
        + **Internal dependencies.** Internal dependencies involve a precedence relationship between project activities and are generally inside the project team’s control. For example, if the team cannot test a machine until they assemble it, there is an internal mandatory dependency. The project management team determines which dependencies are internal during the process of sequencing the activities.
      1. **LEADS AND LAGS**

A lead is the amount of time a successor activity can be advanced with respect to a predecessor activity. For example, on a project to construct a new office building, the landscaping could be scheduled to start 2 weeks prior to the scheduled punch list completion. This would be shown as a finish-to-start with a 2-week lead as shown in Figure 6-10. Lead is often represented as a negative value for lag in scheduling software.

FS – 2 Weeks (Lead) SS – 15 Days (Lag)

**Edit Draft**

**Landscape Building Lot**

**Write Draft**

**Complete Punch List**

**Figure 6-10. Examples of Lead and Lag**

A lag is the amount of time a successor activity will be delayed with respect to a predecessor activity. For example, a technical writing team may begin editing the draft of a large document 15 days after they begin writing it. This can be shown as a start-to-start relationship with a 15-day lag as shown in Figure 6-10. Lag can also be represented in project schedule network diagrams as shown in Figure 6-11 in the relationship between activities *H* and *I* (as indicated by the nomenclature SS+10 (start-to-start plus 10 days lag) even though the offset is not shown relative to a timescale).

The project management team determines the dependencies that may require a lead or a lag to accurately define the logical relationship. The use of leads and lags should not replace schedule logic. Also, duration estimates do not include any leads or lags. Activities and their related assumptions should be documented.

**A**

SS

**C**

**D**

FS + 15

**Begin**

**H**

**F**

**G**

SS + 10

**I**

FF

**K**

**L**

**J**

**End**

**E**

**B**

**Figure 6-11. Project Schedule Network Diagram**

**Videos**

<https://www.youtube.com/watch?v=ij2Bmp6YMEE>

# MGMT627\_Lecture28

<https://www.youtube.com/watch?v=mo31Taqqhmw&list=PL11FB6AC4FFE54710&index=28>

MGMT627\_Lecture29

https://www.youtube.com/watch?v=2FGq2VBer1g&list=PL11FB6AC4FFE54710&index=29&pbjreload=10