IDEF3

Process Modeling

What is a Process Model?

"Simply put, the Process Model is the way that work is divided in a value delivery system."

— James B. Swartz

- A representation of a process and its related components presented in a time-dependent fashion.
- The decision logic that may exist within the process.

Benefits of Process Modeling

- Document current processes for standardization.
- Provide guidelines for new process members to reduce the learning curve.
- Capture and analyze AS-IS processes.
- Design / redesign process for TO-BE scenarios.
- Test the design of a new process before embarking on an expensive development project.

What is IDEF3?

- The Process Description Capture Method.
- The Object State Transition Description Method.
- Supports descriptions at any desired level of detail through Decompositions.
- Employs the concepts of Scenarios to simplify the structure of complex process flow descriptions.
- Supports the capture of multiple viewpoints.

Flow Charting vs Process Modeling

Flow Charting	Process Modeling
Conveys process logic in an ambiguous manner	Conveys process logic with unambiguous syntax
Varying levels of abstraction cannot be captured	Can capture varying levels of abstraction
Does not provide information about the objects in a process	Embellishes the process with objects and simulation data

A Generic Process Modeling Tool ...

- Automates the IDEF3 method
- Adheres to the method standard.
- Provides background quality checking and advisory support.
- Utilizes SmartDraw capability.

IDEF3 Overview

- Section 1: Basic Elements of the Process Diagram
- Section 2: Documenting the Process Flow
- Section 3: Enhancing the Process Description

Basic Elements of the Process Diagram

Processes

Links

Junctions

Elements of a Diagram

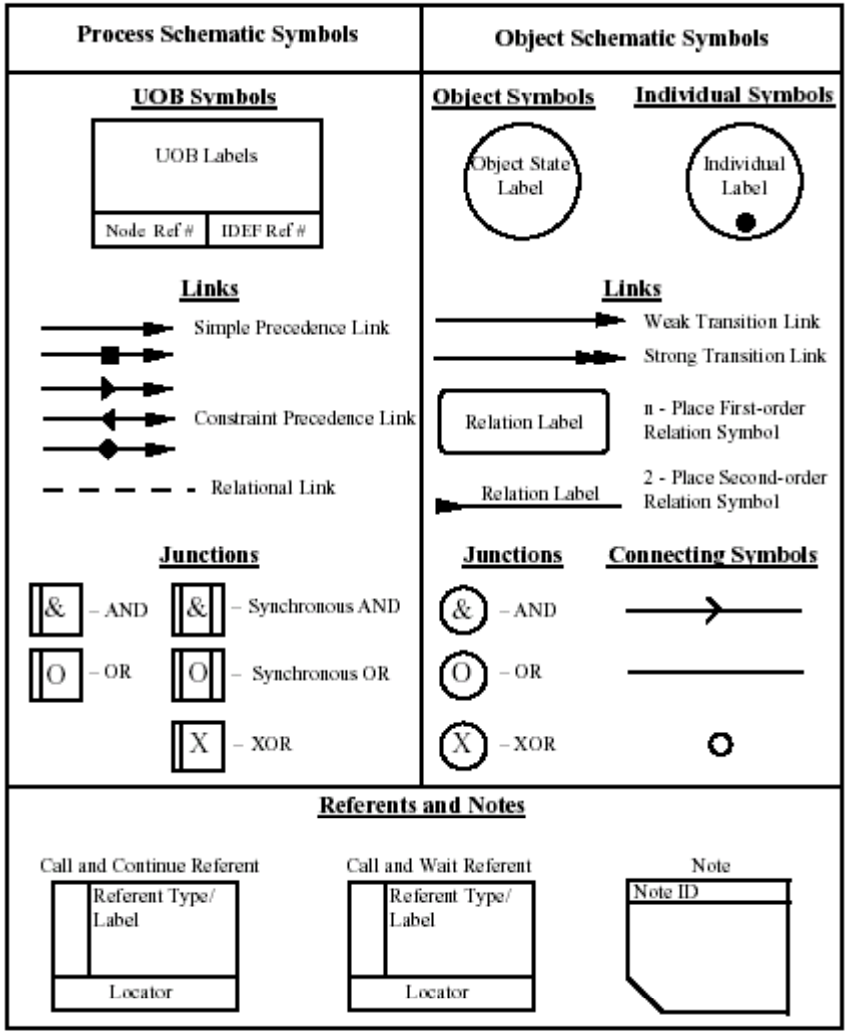
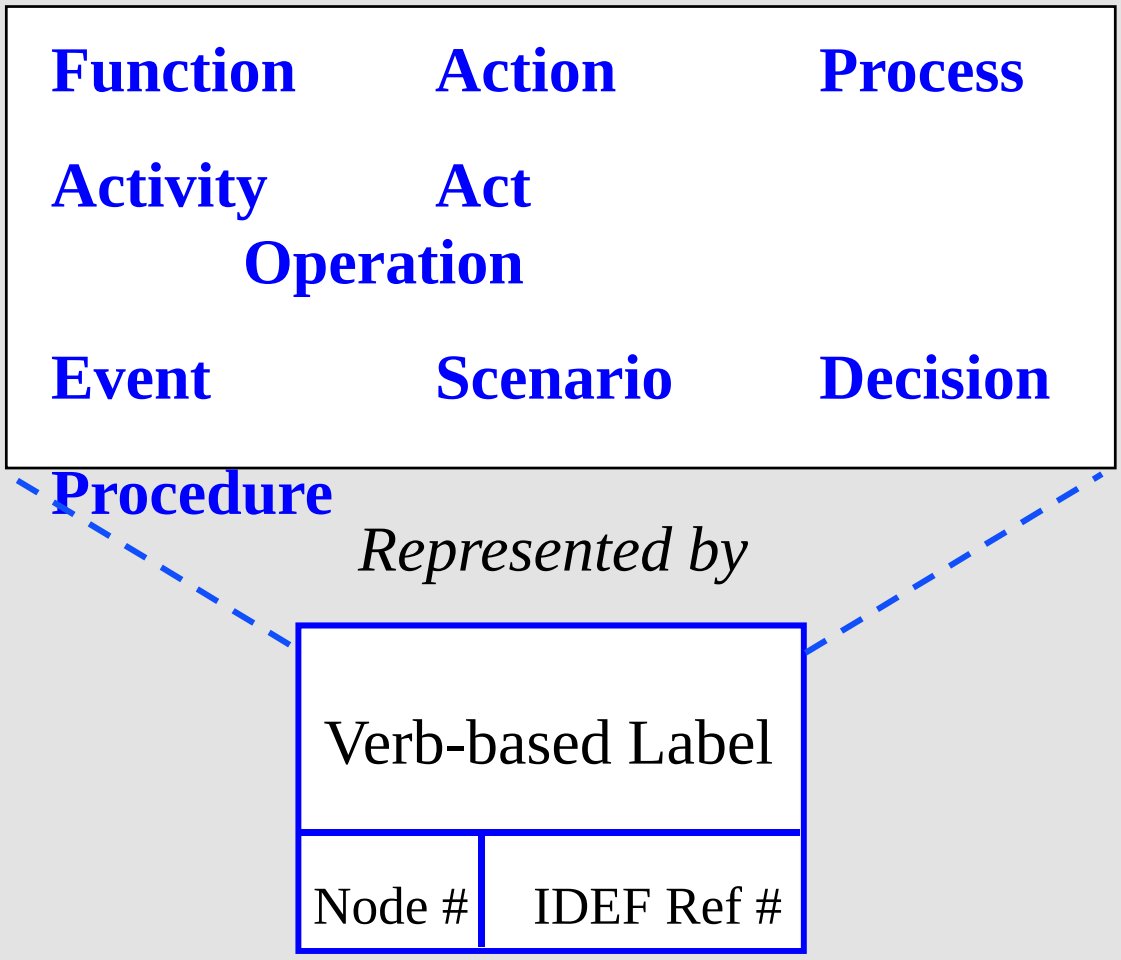


Figure 3-1a
Symbols Used for IDEF3 Process Description Schematics

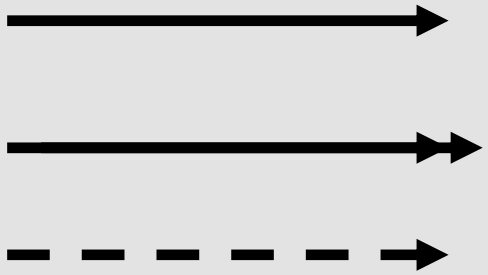
Processes



Unit of Behavior (UOB) boxes

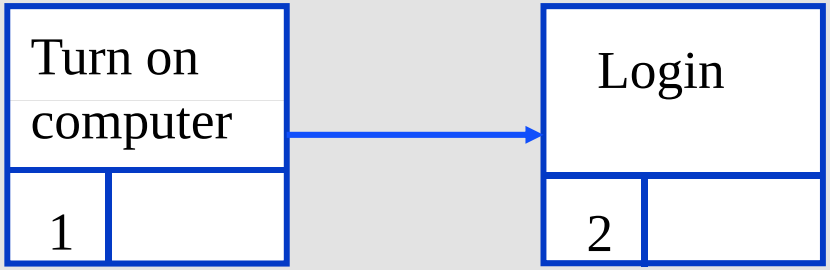
Links

- Purpose
  - Describe temporal, logical, conventional, or natural constraints between processes
- Types of Links
  - Simple Precedence
  - Object Flow
  - Relation



Precedence Link

- Express simple temporal precedence between instances of one process type and another.
- Each instance of the source process will complete before the paired instance of the destination process can begin.



You have to turn on the computer before you can login.

Links

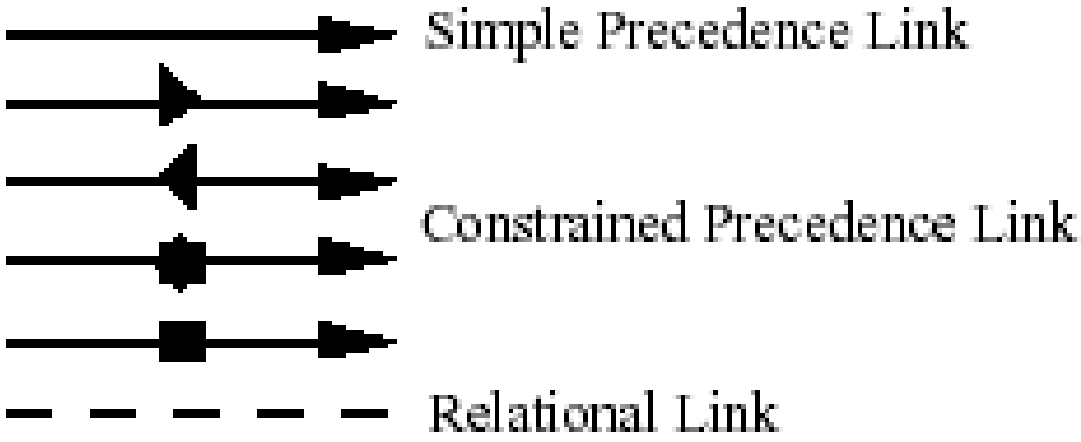


Figure 3-2
IDEF3 Link Types

Precedence Link

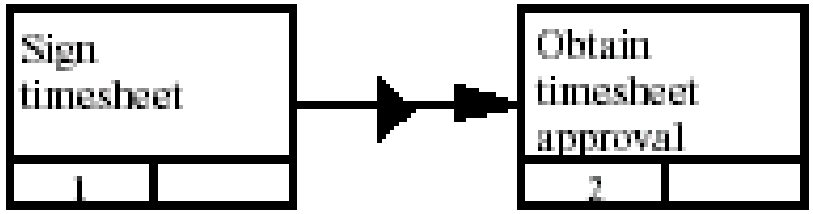


Figure 3-5a
Example of a Schematic Involving a Constrained Precedence Link

The first of the constrained precedence links indicates that any instance of the source UOB *must* be followed by an instance of the destination UOB.

Precedence Link

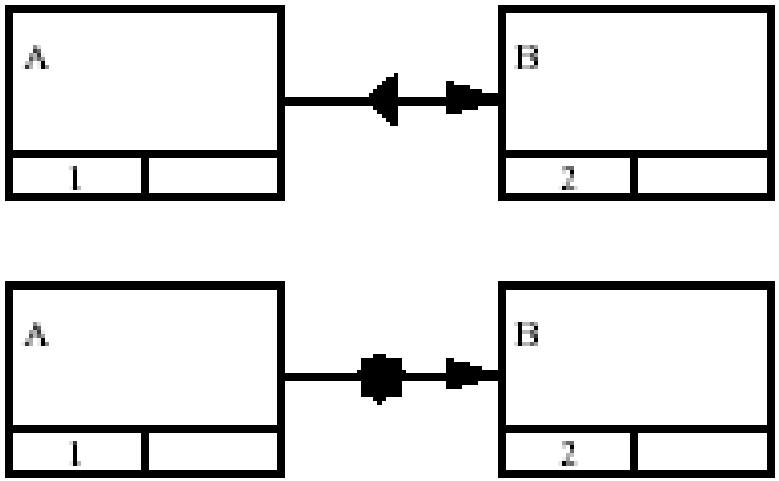


Figure 3-5b
Further Examples of Constrained Precedence Links

Precedence Link

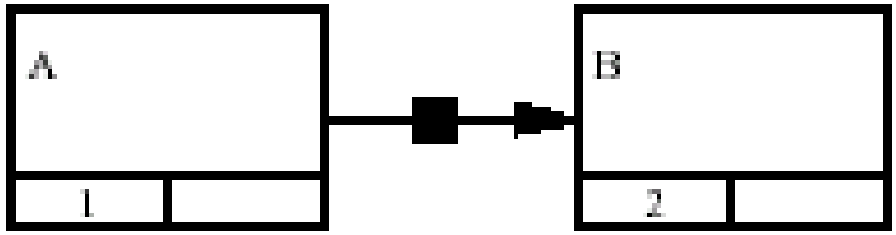
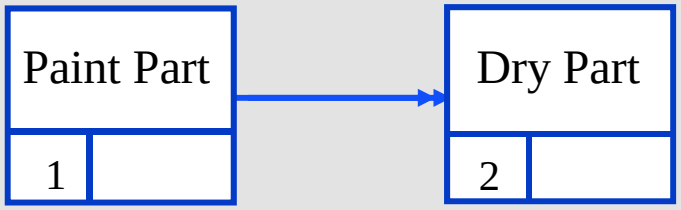


Figure 3-6
General Constrained Precedence Link

Object Flow Link

- Indicates the participation of an object in two process instances.
- Has the same temporal semantics as a precedence link.
- Lack of an Object Flow link does not preclude the existence of an object participation between two processes.

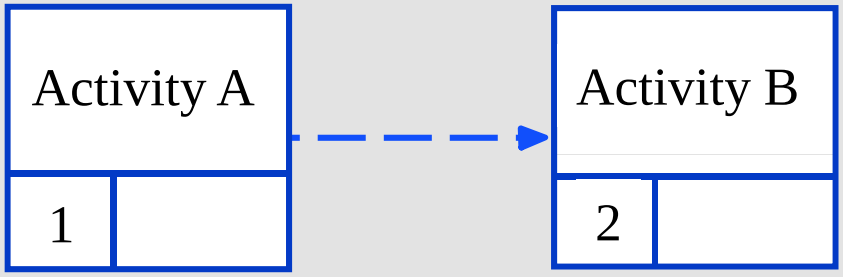


There is an object (Part) that is common to both processes.

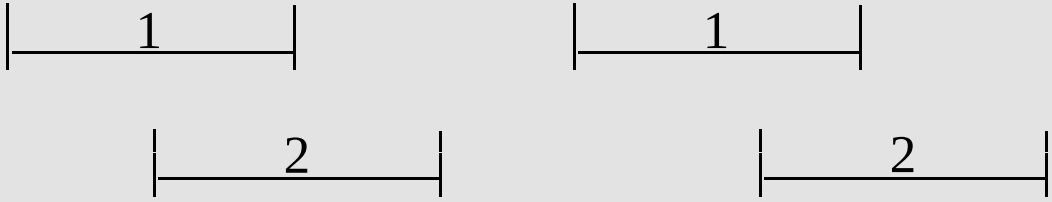
Relational Link

Commonly used relational (dashed) link relations:

Before	Meets	Starts	Triggers
During	Overlaps	Causes	
After	Finishes	Enables	



(a)



Nonbranching Process

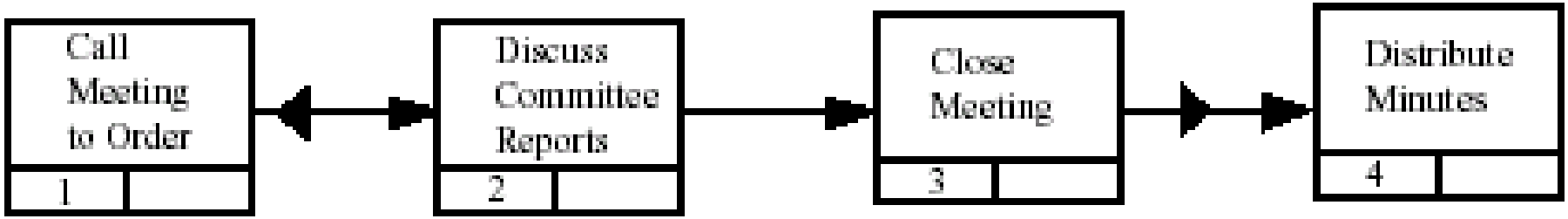


Figure 3-8
Nonbranching IDEF3 Schematic

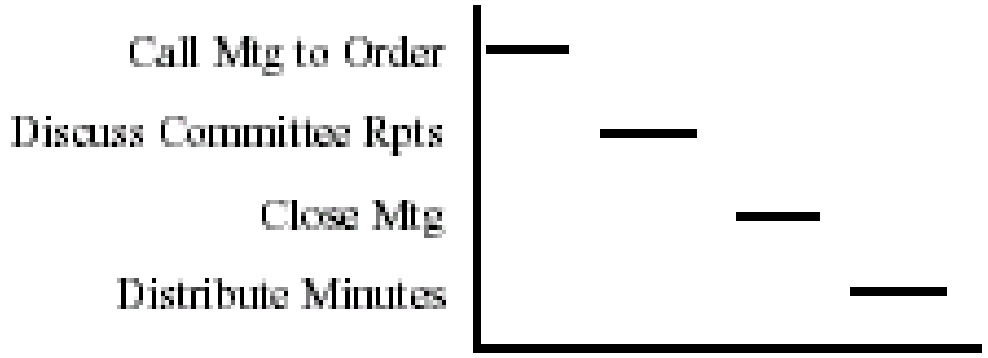
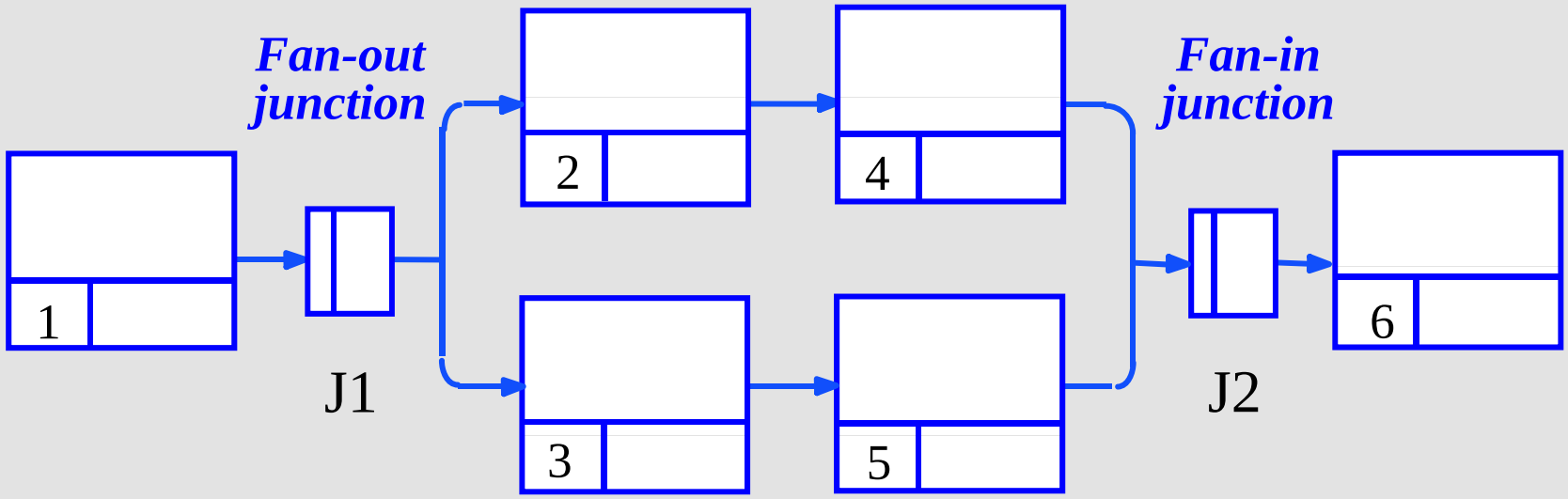


Figure 3-9
Activation Plot for Figure 3-8

Junctions

- IDEF3 junctions show convergence or divergence of multiple process flows and their timing.



Junctions

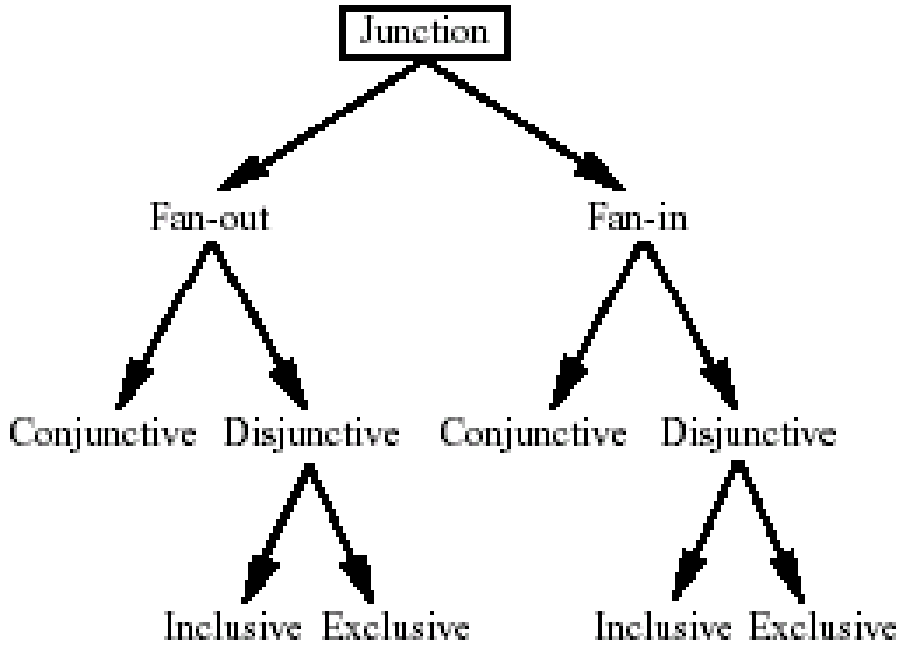
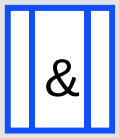


Figure 3-10
Classification of Junction Types

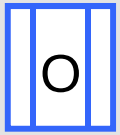
Junctions

Asynchronous And — All preceding (or following) actions must complete (or start).



Synchronous And — All preceding (or following) actions must complete (or start) simultaneously.

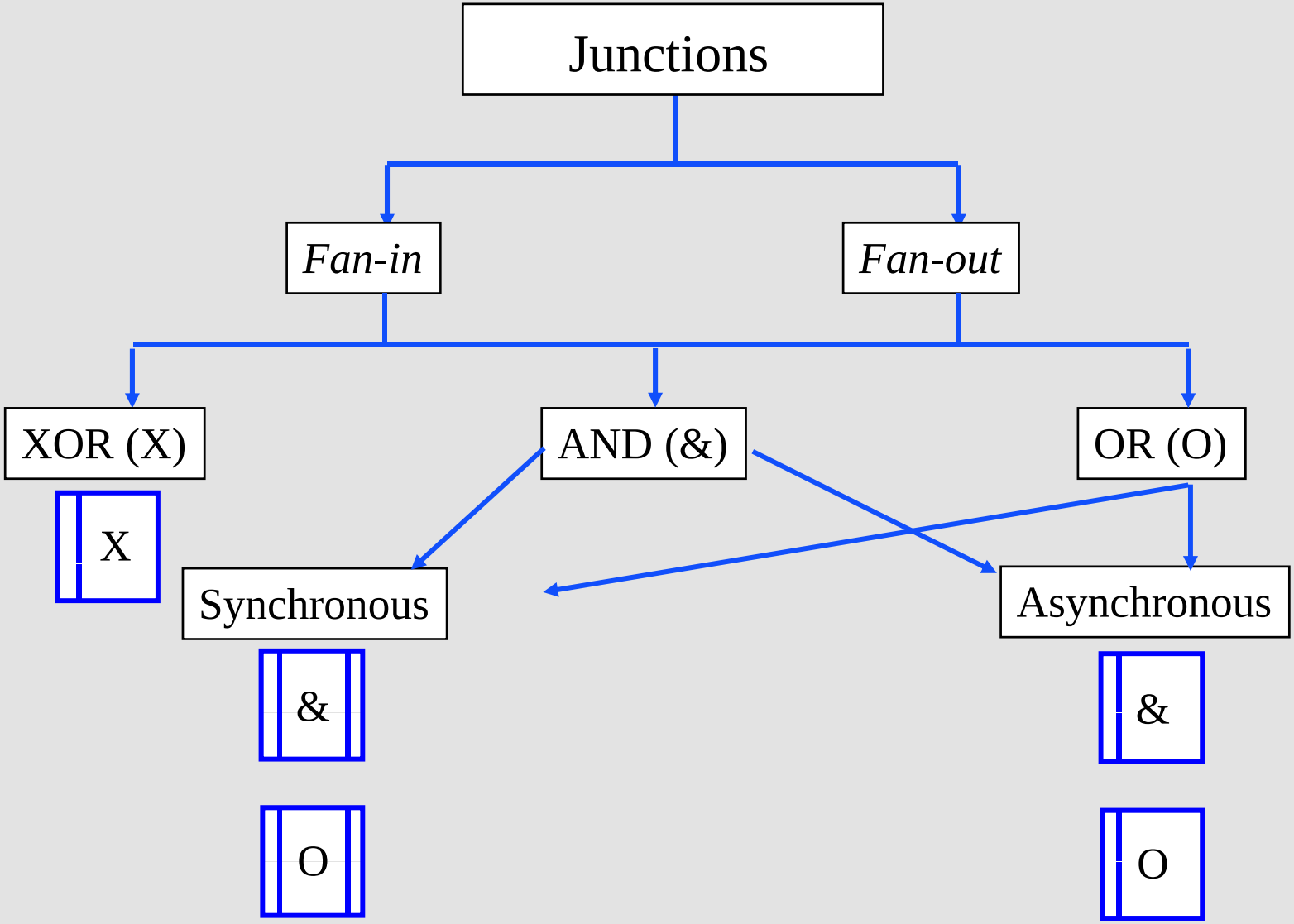
Asynchronous Or — One or more of the preceding (or following) will complete (or start).



Synchronous Or — One or more of the preceding (or following) will complete (or start) simultaneously.

Exclusive Or — Exactly one of the preceding (or following) will complete (or start).

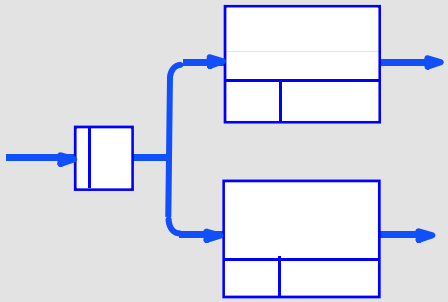
Junctions


Junctions
Fan-in
Fan-out
XOR (X)
AND (&)
OR (O)
X
Synchronous
Asynchronous
&
&
O
O

Junction Semantics

Fan-out (Divergence)

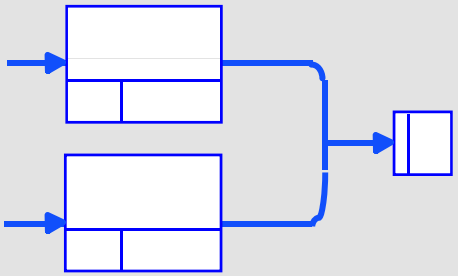
Junction Type	Meaning
& — Asynchronous “AND”	All succeeding process paths will eventually start, and all processes on each path will eventually happen.
& — Synchronous “AND”	All succeeding process paths will start together, and all processes on each path will eventually happen.
O — Asynchronous “OR”	One or more of the following process paths will eventually start, and all of the processes on these paths will happen.
O — Synchronous “OR”	There will be a synchronized initiation of one or more process paths.
X — “XOR”	Exactly one of the following process paths will be initiated, and only the processes on that path will happen.



Junction Semantics

Fan-in (Convergence)

Junction Type	Meaning
|&| — Asynchronous “AND”	All preceding processes must complete.
|&| — Synchronous “AND”	All preceding processes will complete simultaneously.
|O| — Asynchronous “OR”	One or more of the preceding processes will complete.
|O| — Synchronous “OR”	One or more of the preceding processes will complete simultaneously.
|X| — “XOR”	Exactly one of the preceding processes will complete.



Junction Example

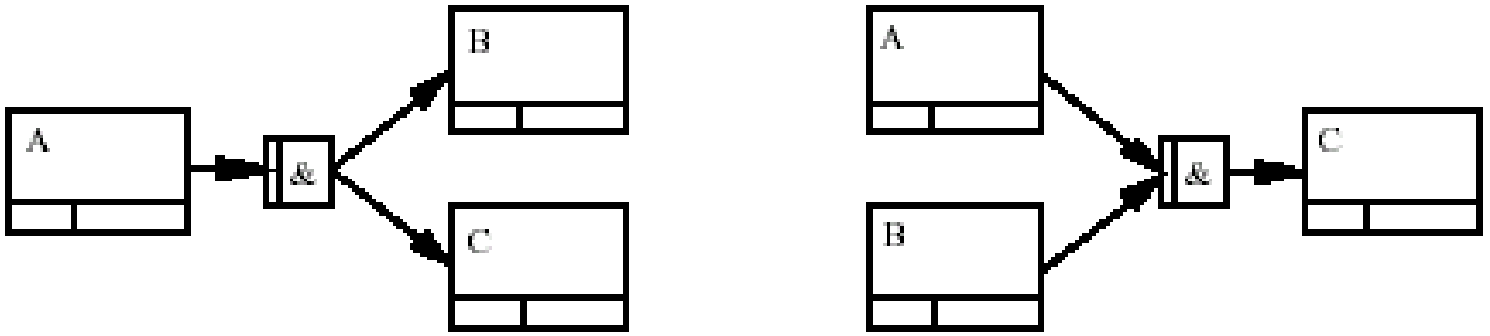


Figure 3-11
Diverging and Converging Parallel Subprocesses

Junction Example

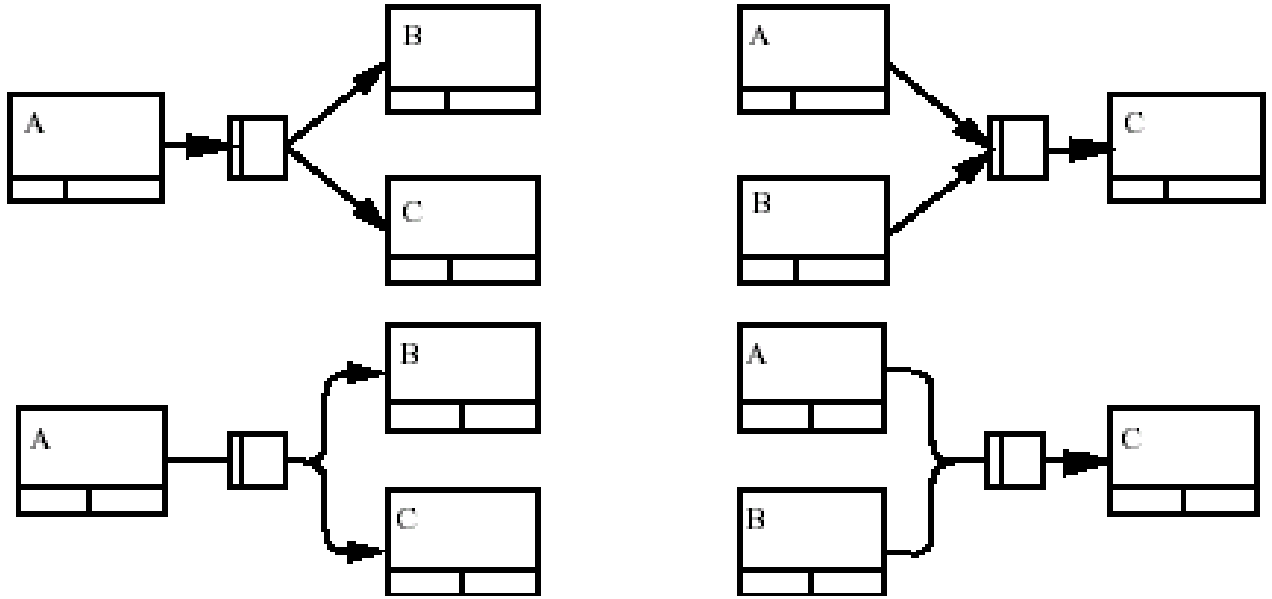


Figure 3-12
Graphical Conventions for Precedence Links Connecting to Junctions

Junction Example

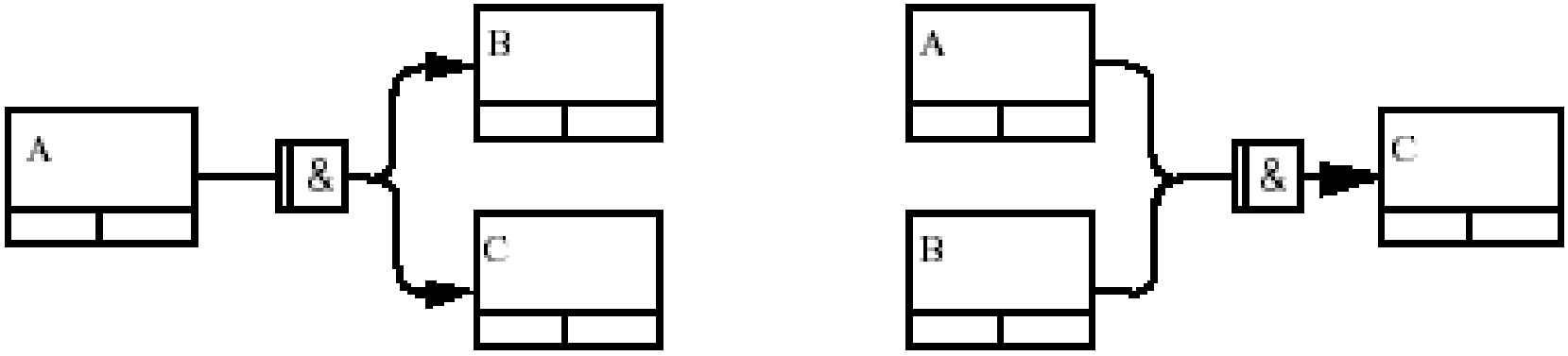


Figure 3-13
Sample Schematics to Illustrate Semantics of AND Junctions

Junction Example

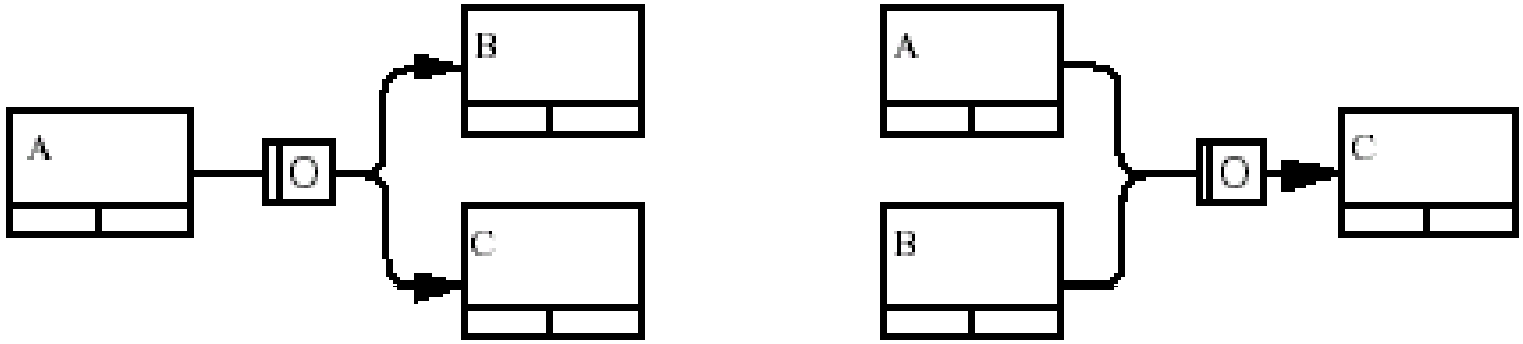


Figure 3-14
Sample Schematics to Illustrate Semantics of OR Junctions

Simple Process

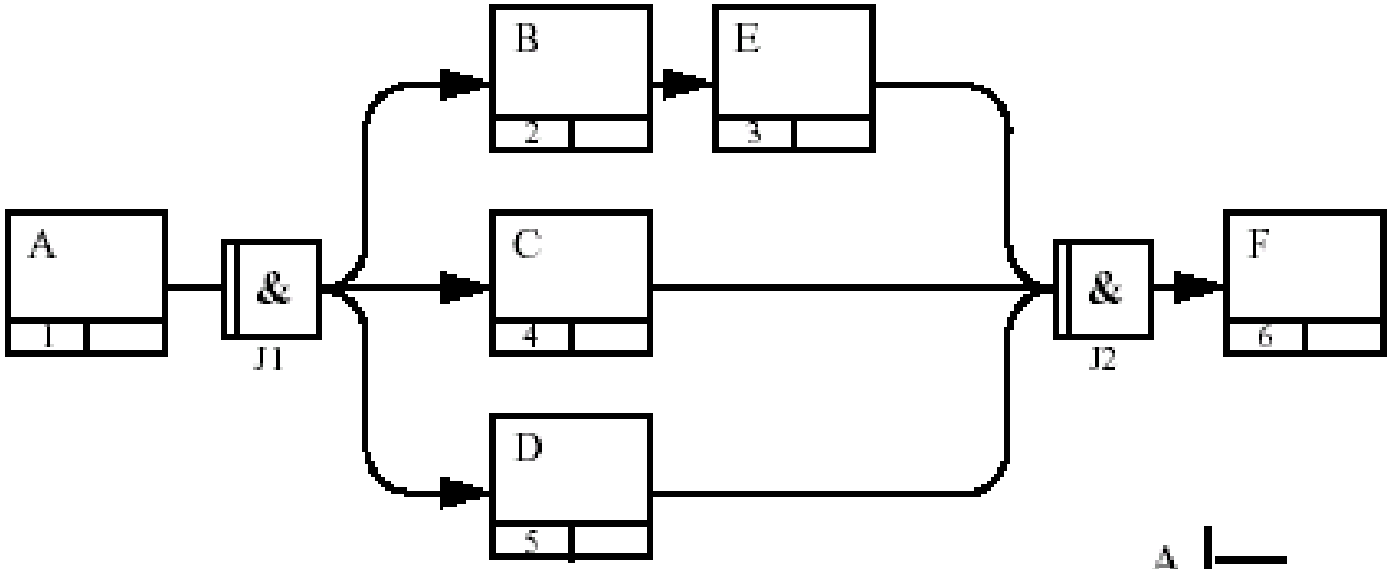


Figure 3-15
Schematic with Asynchronous AND Junctions

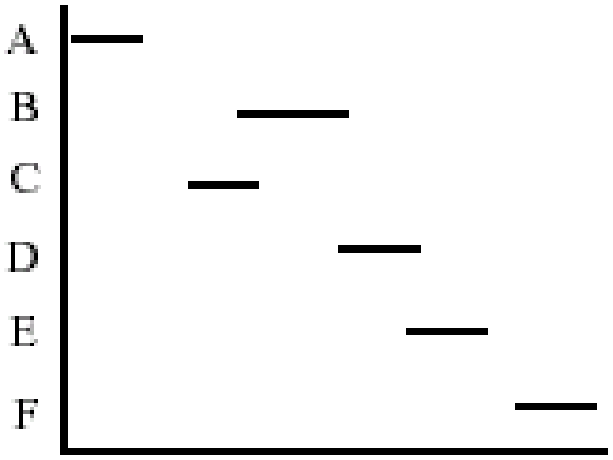


Figure 3-16
Activation Plot for Figure 3-15

Simple Process

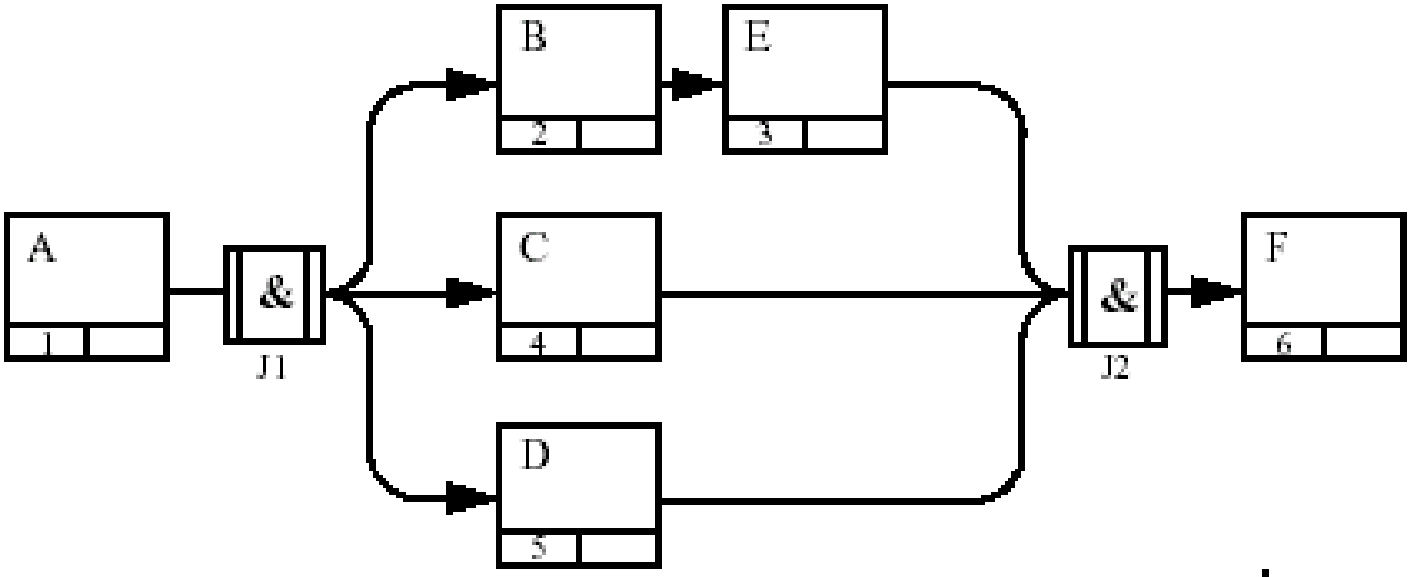


Figure 3-17
Synchronous AND Junctions

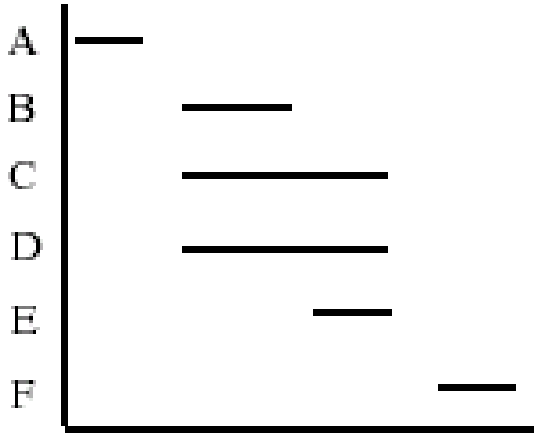


Figure 3-18
Activation Plot for Figure 3-17

Simple Process

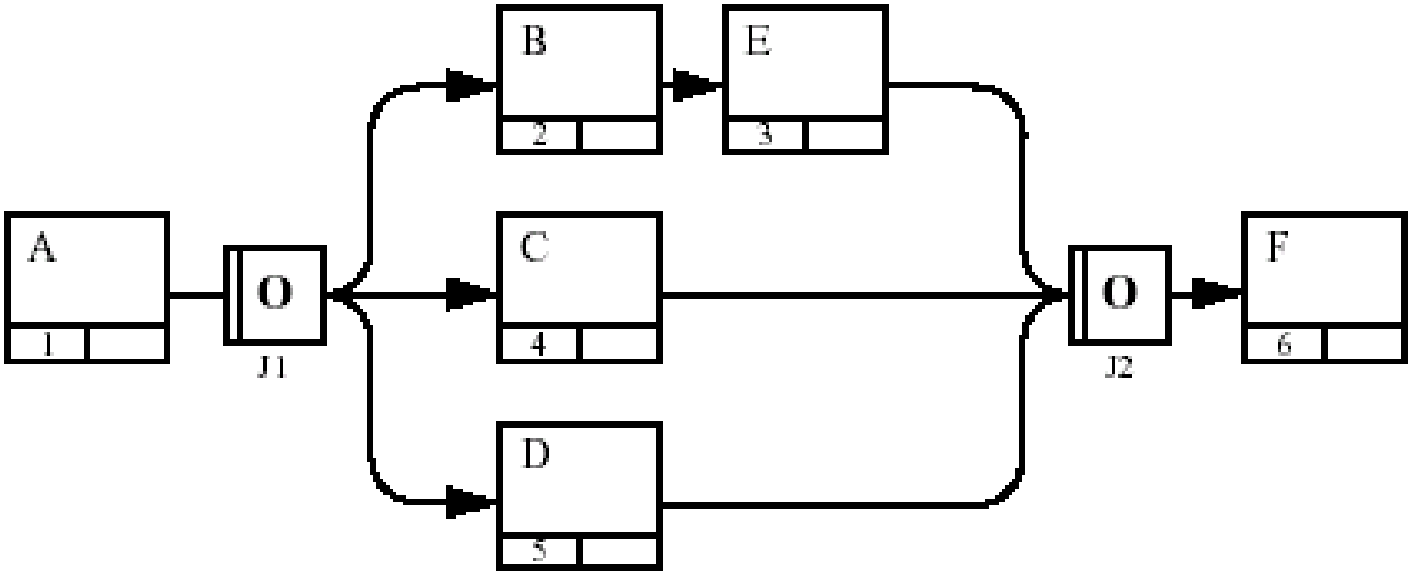


Figure 3-19
Asynchronous OR Junctions

Simple Process

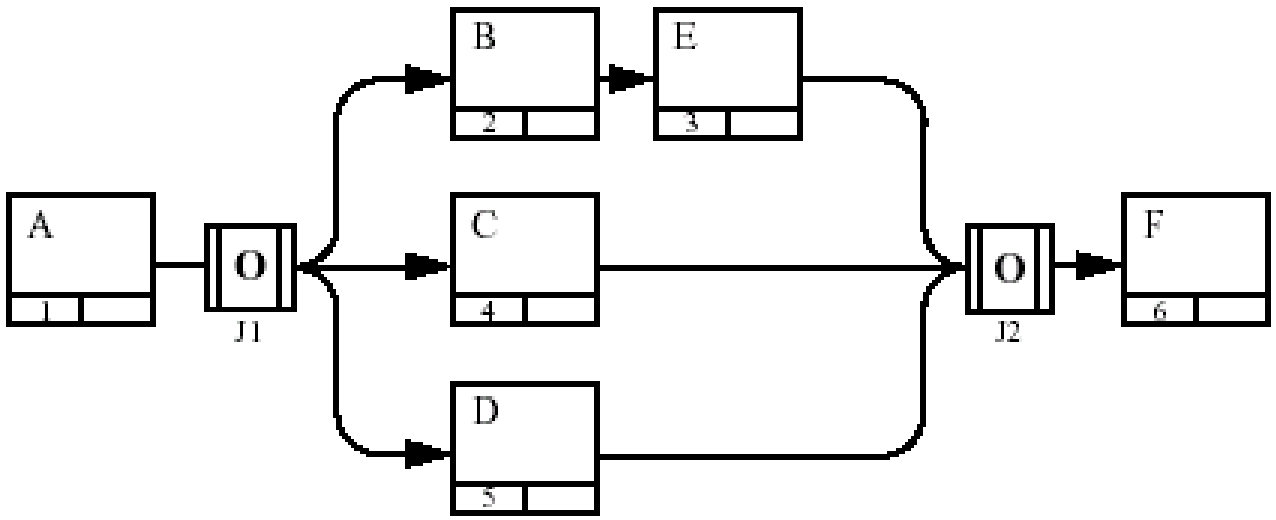


Figure 3-20
Synchronous OR Junctions

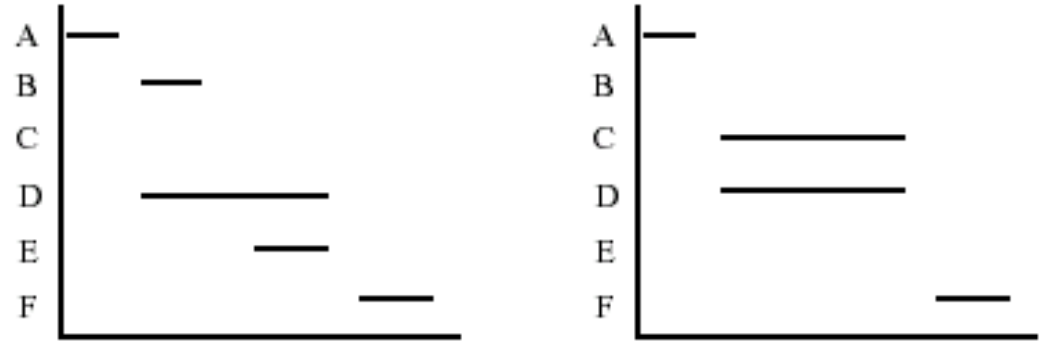


Figure 3-21
Activation Plots for Figure 3-20

Simple Process

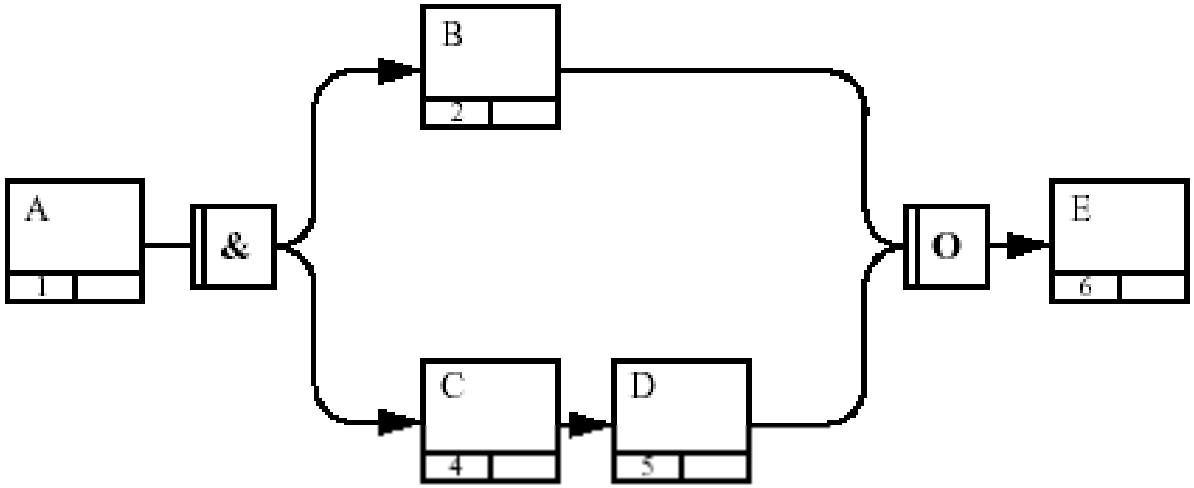


Figure 3-22
Fan-out AND Junction Followed by a Fan-in OR Junction

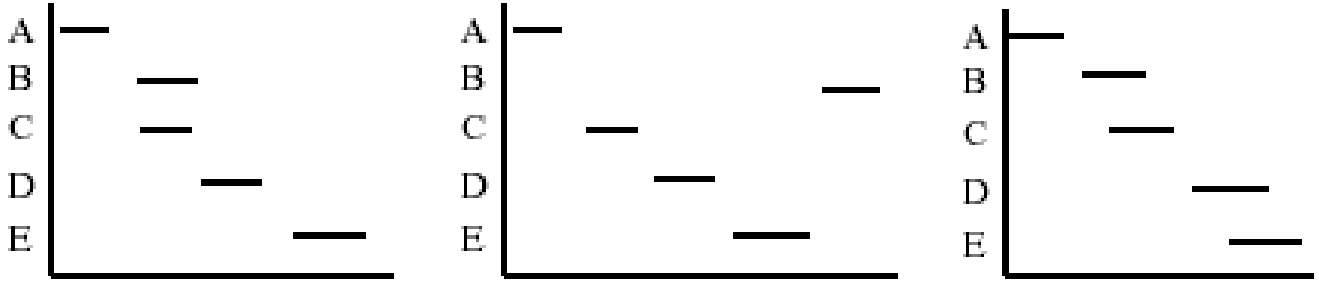


Figure 3-23
Activation Plots for Figure 3-22

Some Concrete Examples

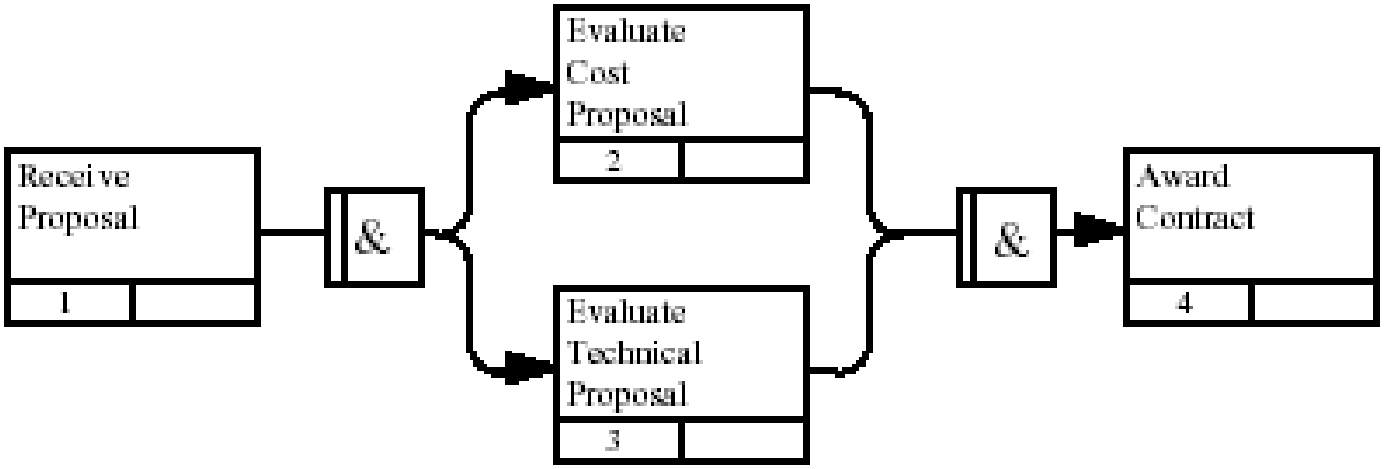


Figure 3-24
Asynchronous AND Junction Example

Some Concrete examples

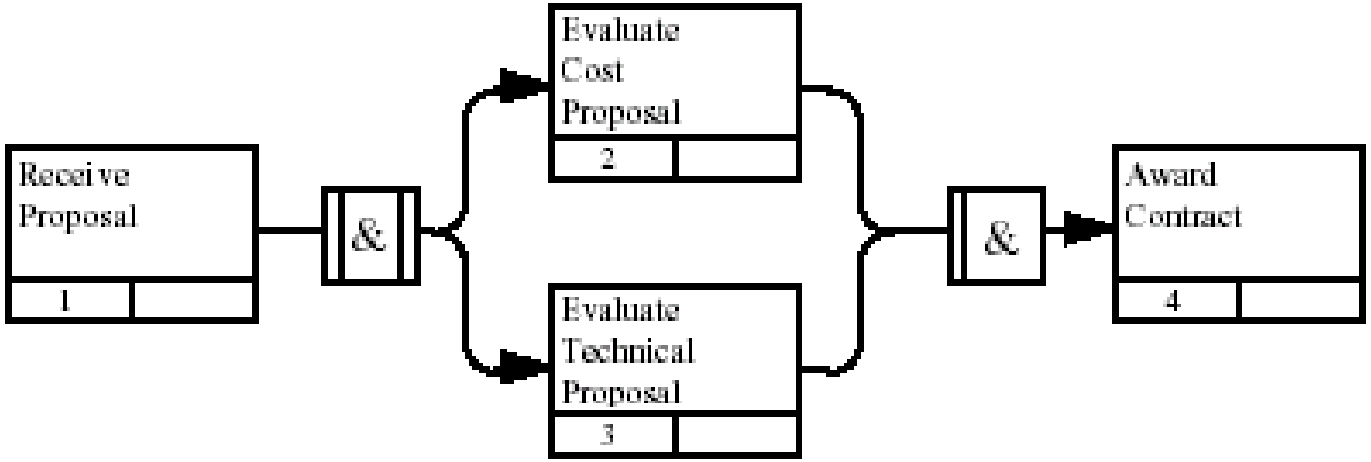


Figure 3-25
Synchronous AND Junction Example

Some Concrete examples

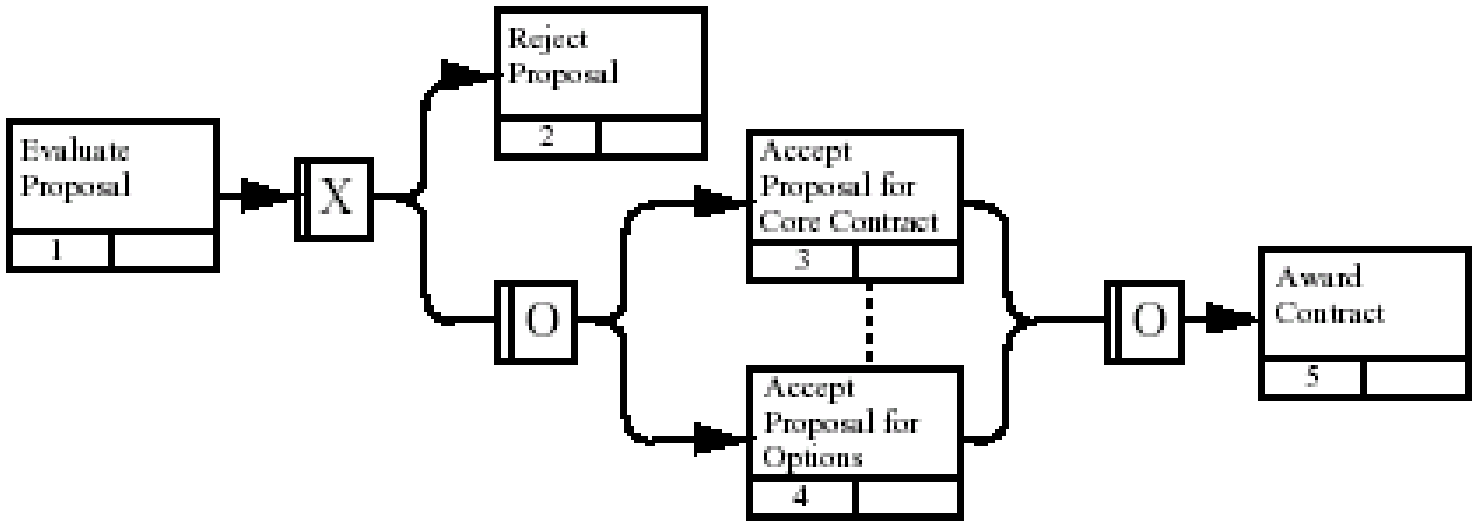


Figure 3-26
Asynchronous OR Junction Example

Invalid Examples

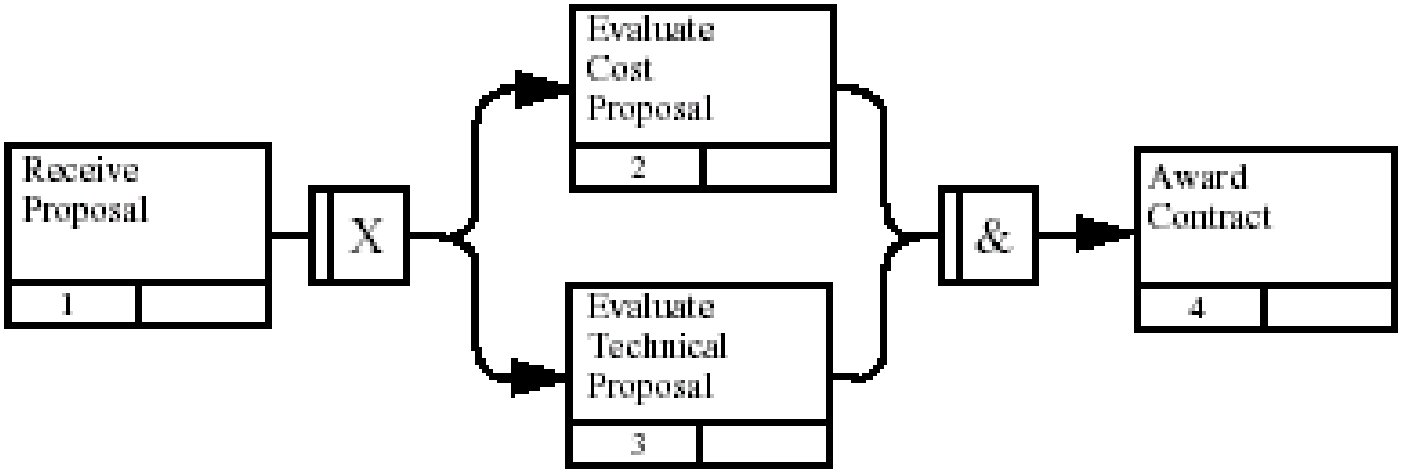
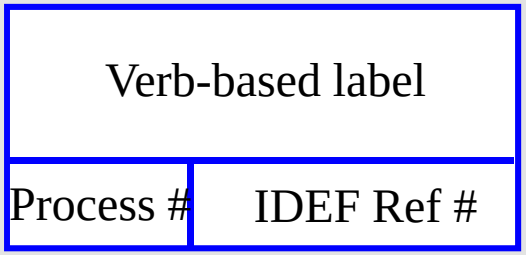


Figure 3-27
Invalid XOR/AND Structure Example

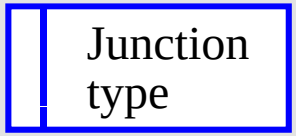
Review

Process



Function Process
Activity Operation
Action Event

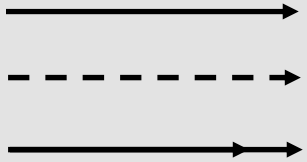
Junctions



Asynchronous

Synchronous

Links



Precedence Link
Relational Link
Object Flow Link

Documenting the Process Flow

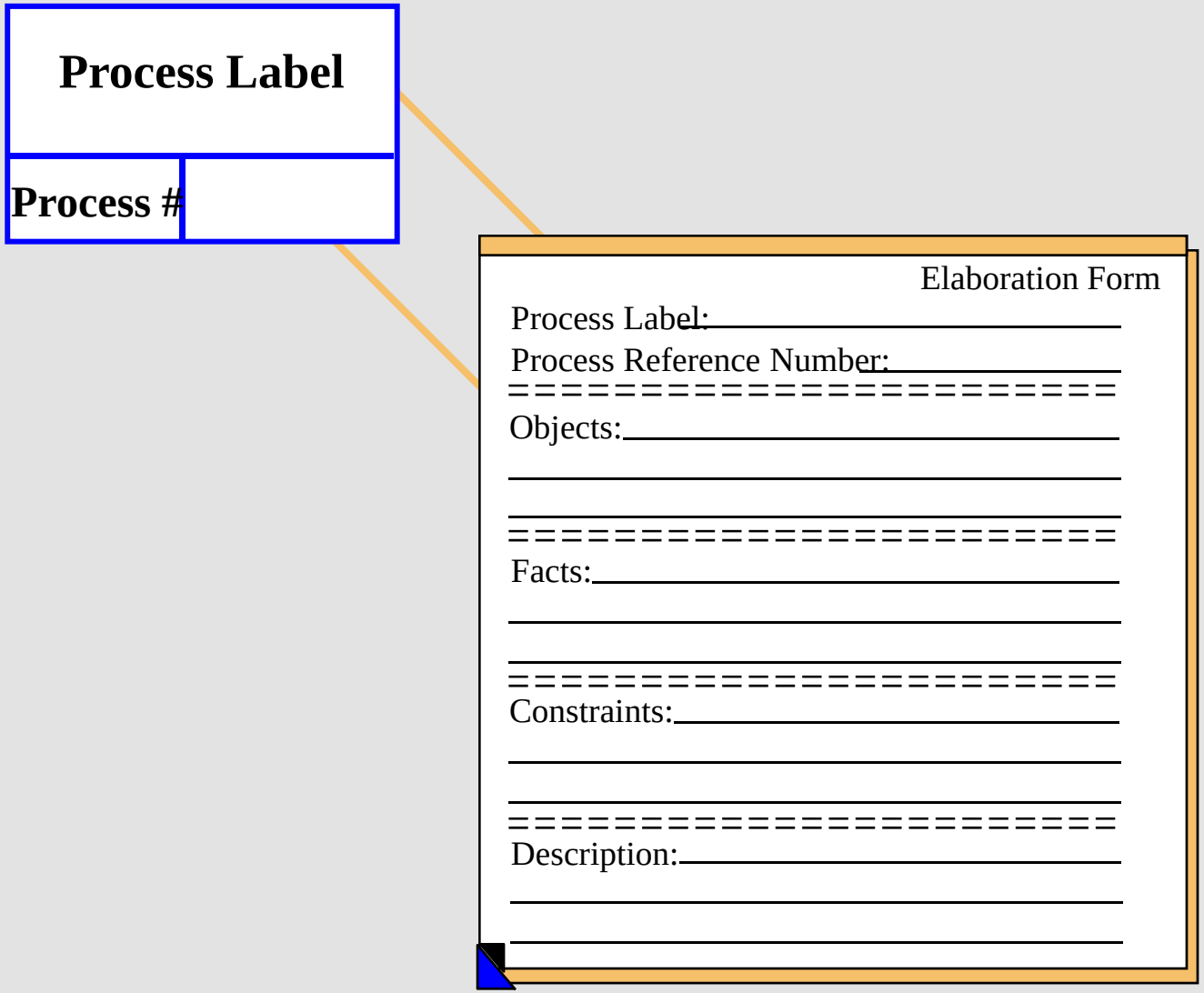
Process Elaboration

Objects

Referents

Other Documentation

Process Elaboration



Objects Linked to a Process

Object Types

- Entity
- Location
- Resource
- Queue
- Transport

Instances of Object Types

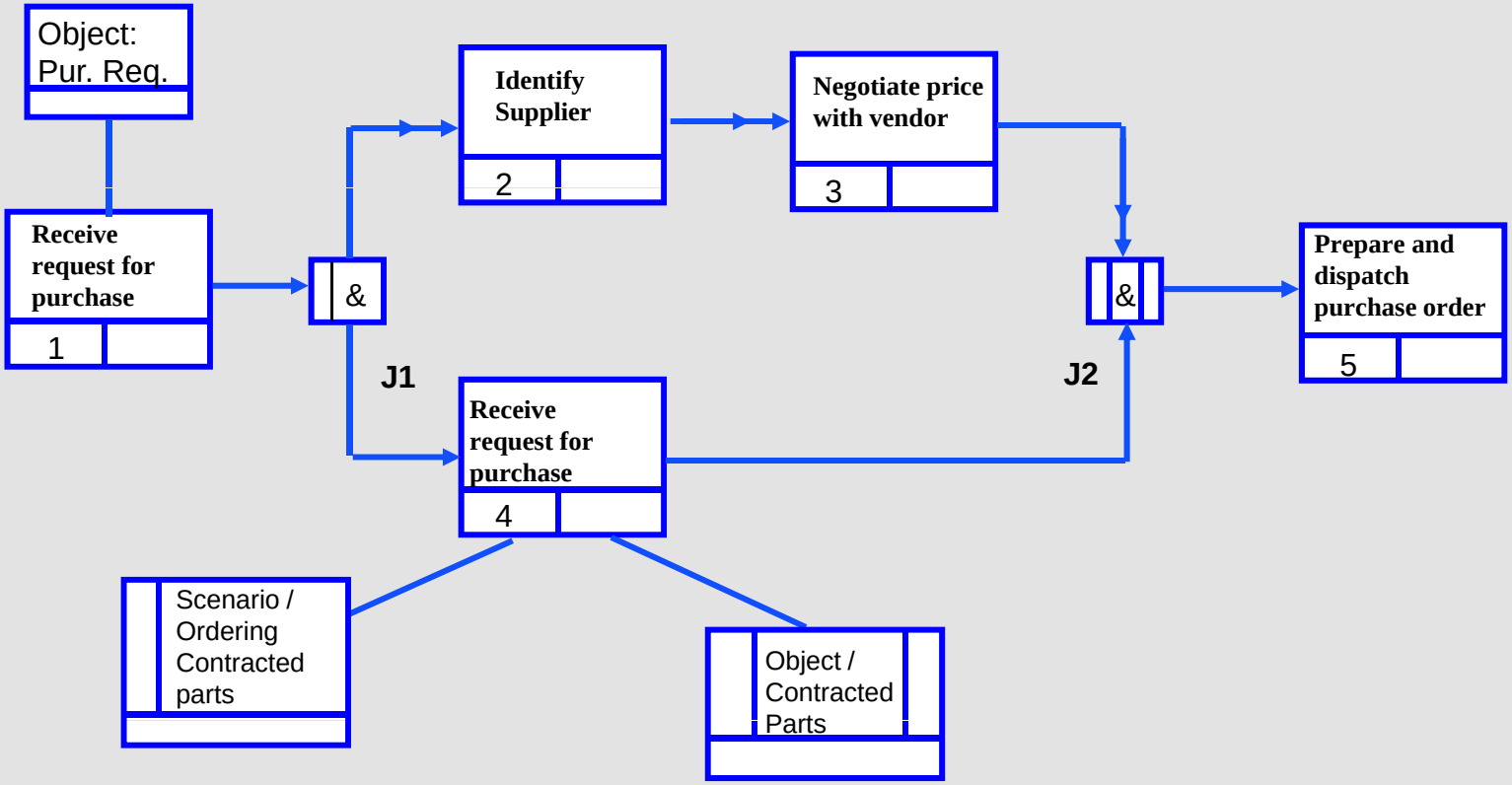
- Paint/Part
- Paint Booth
- Operator
- Part Queue
- Conveyor

Referents

- Referents draw the reader's attention to an important point or note.
- Referents are often used to:
  - Point to other model elements without showing an explicit process flow.
  - Indicate a "Go-To" location in complex process flows.
  - Specify constraints on junctions.
  - Provide links to Object State Transition Networks.

Referents

. . . simply point the reader to some other aspect of the model that needs to be considered.



Other Documentation

Glossary

- Textual descriptions of the process elements.

Sources

- Source material used in the construction of the process description.

Notes

- Annotations resulting from the model review process.

Enhancing the Process Descriptions

Scenario

Scenario Objectives

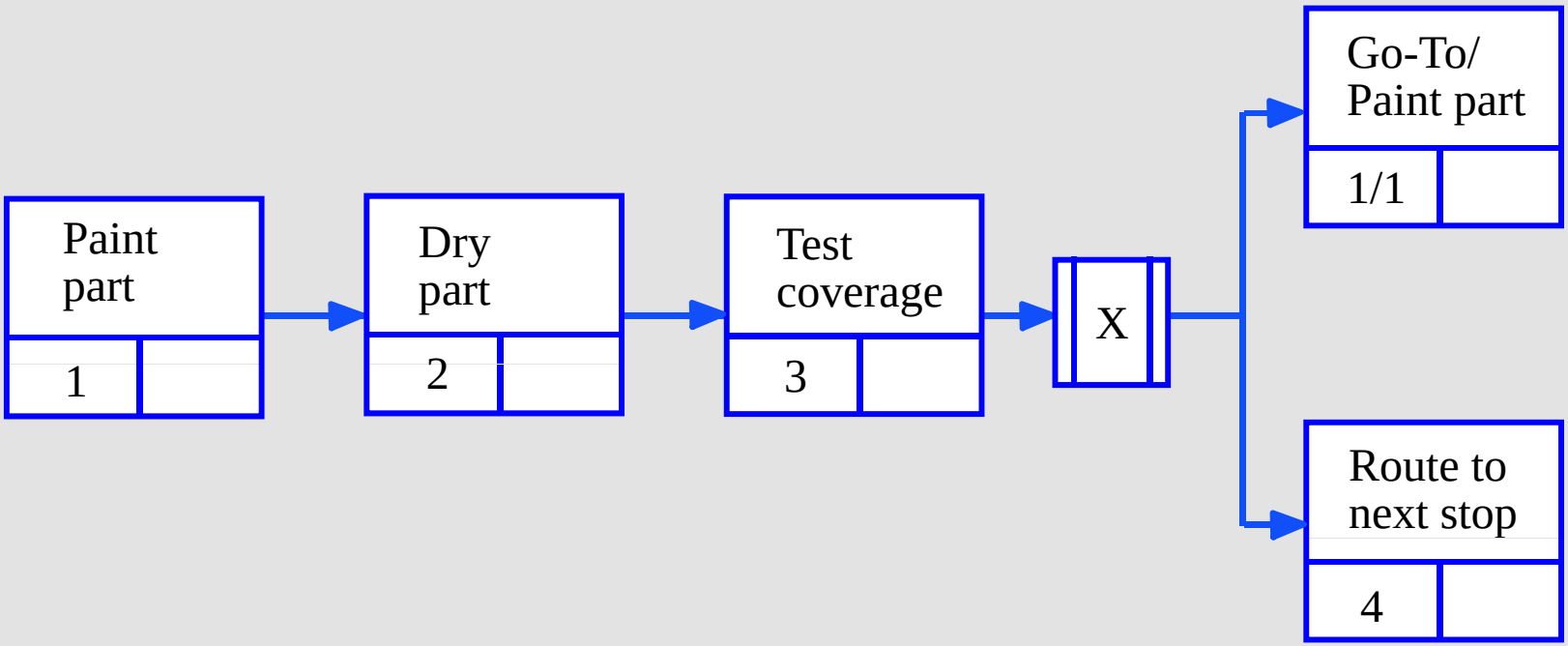
Decompositions

Object State Transmission Networks

A Scenario

- Scenarios are the organizing structure for IDEF3 descriptions.
  - A scenario represents a commonly occurring situation.
- Different views can be different scenarios.
- A base scenario is always needed.

Paint Shop Example



"Painting a part in the company paint shop."

Scenario Objectives

- Viewpoint
  - Determines what can be seen and from what perspective.
- Purpose
  - Establishes the goal of the communication intended by the description.
  - Defines why the description is being developed, and specifies how it will be used.
- Context
  - Establishes the subject of a description.
  - Establishes the subject as a part of a larger whole.
  - Creates a boundary within the environment.

Decomposition

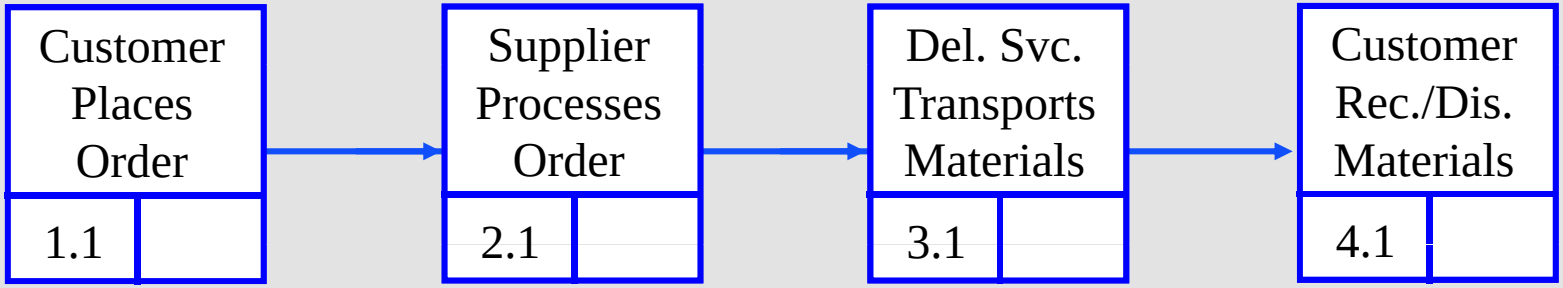
- Purpose
  - Decreases complexity of a diagram.
  - Enables the capture of descriptions at varying levels of abstraction.
  - Provides the ability to model the same process from different knowledge sources or different points of view.

Syntactically, a decomposition is just another IDEF3 process flow diagram.

Decomposition

- Types
  - **Objective view**: Multiple view decompositions may be consolidated into an objective view--the view perceived by a neutral observer. There can be only one objective view.
  - **Role view**: The view of a process as understood by, or from the perspective of, one individual, role type, or functional organization. There may be more than one role view of a process.

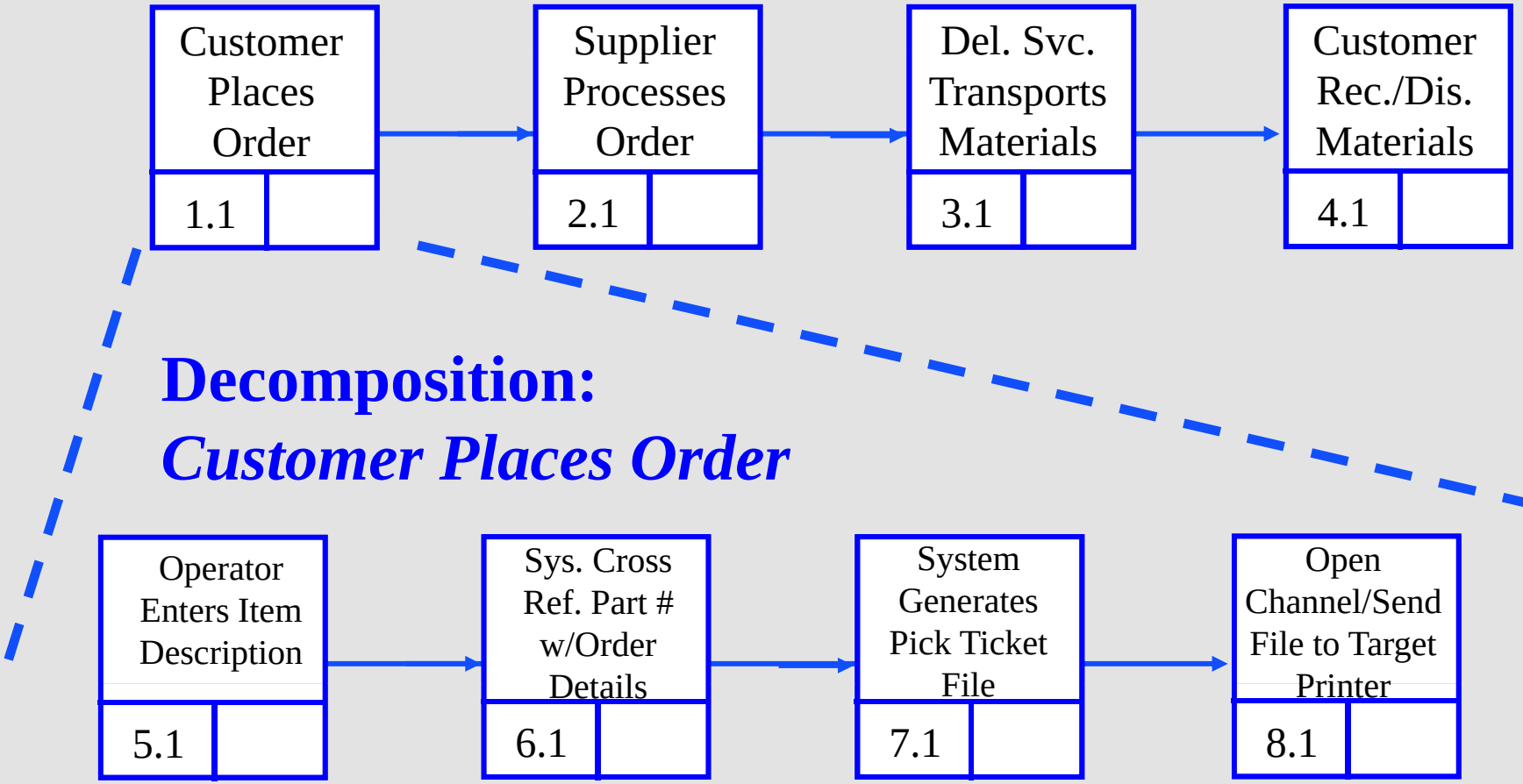
Purchase Order Example



Top-level Scenario:

AS-IS Order Process

Purchase Order Example



Decomposition

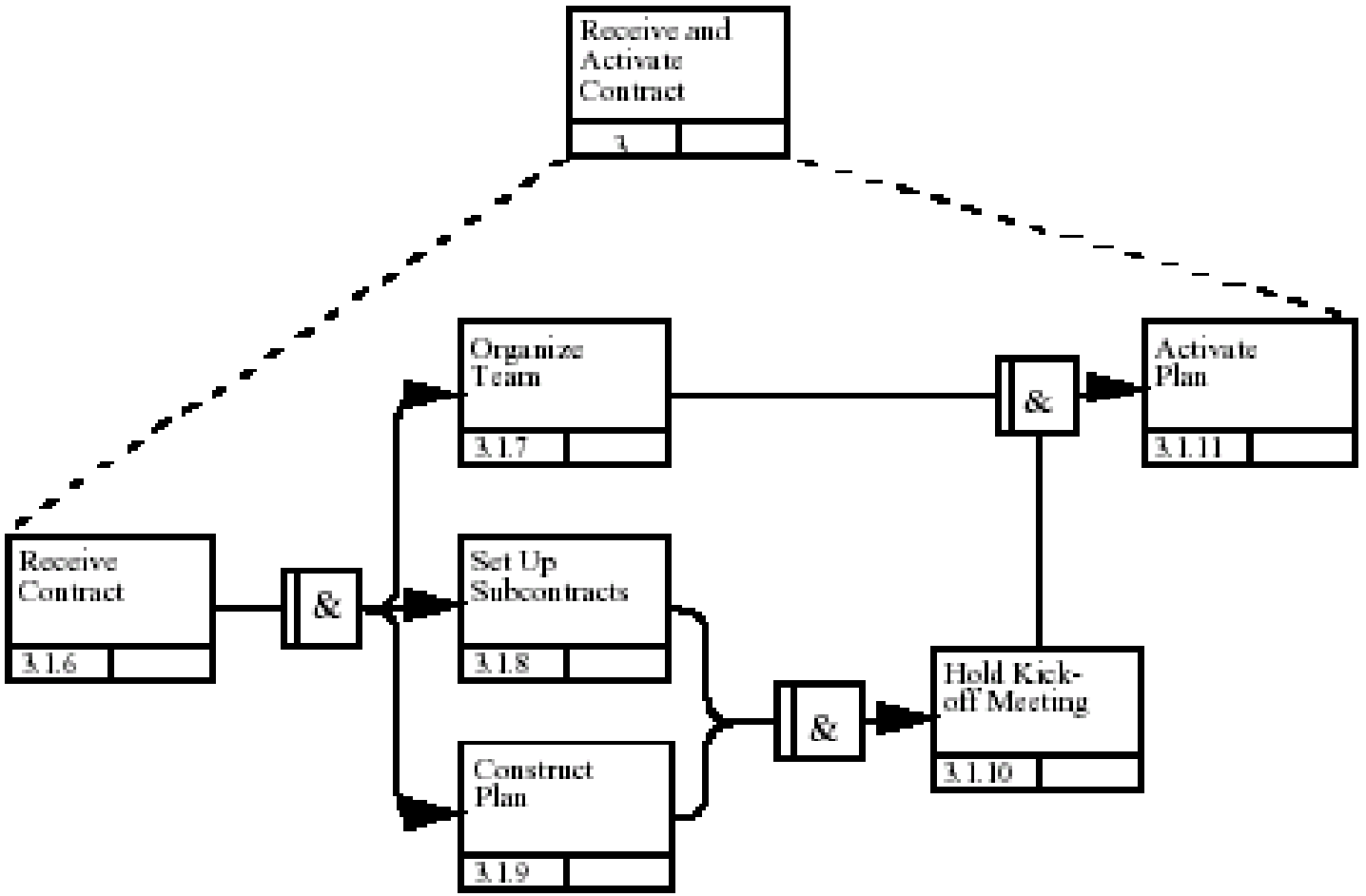


Figure 3-28
Decomposition 3.1 of the UOB *Receive and Activate Contract*

Decomposition

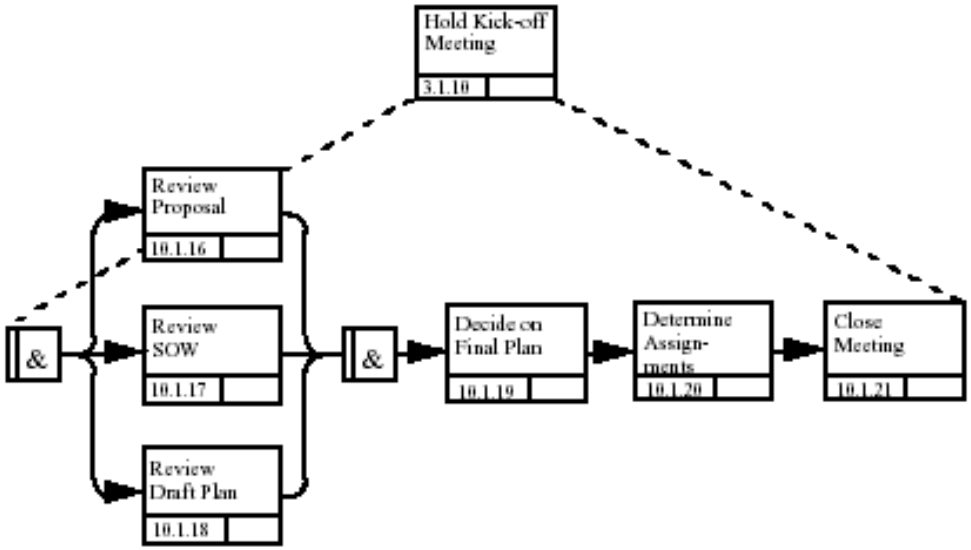


Figure 3-29
Decomposition 10.1 of *Hold Kick-off Meeting* UOB

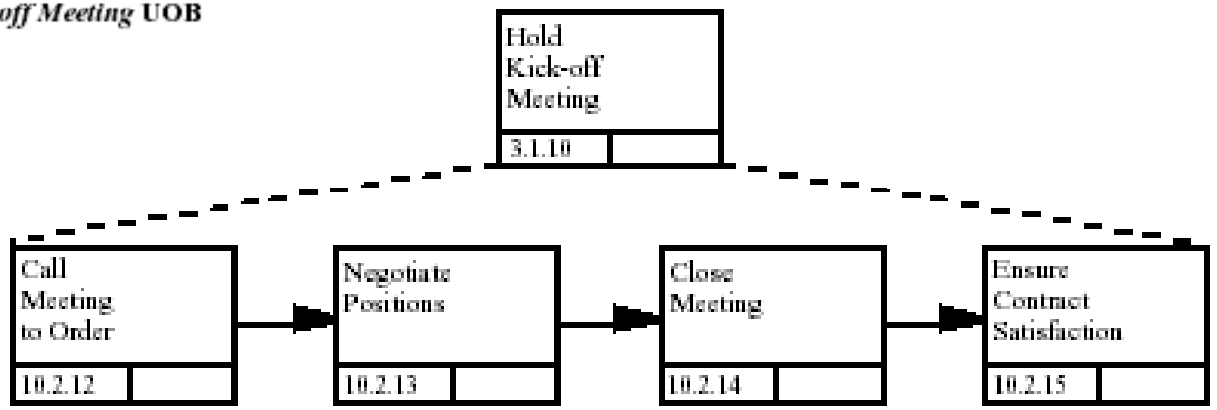


Figure 3-30
The Project Manger's View Decomposition

Decomposition

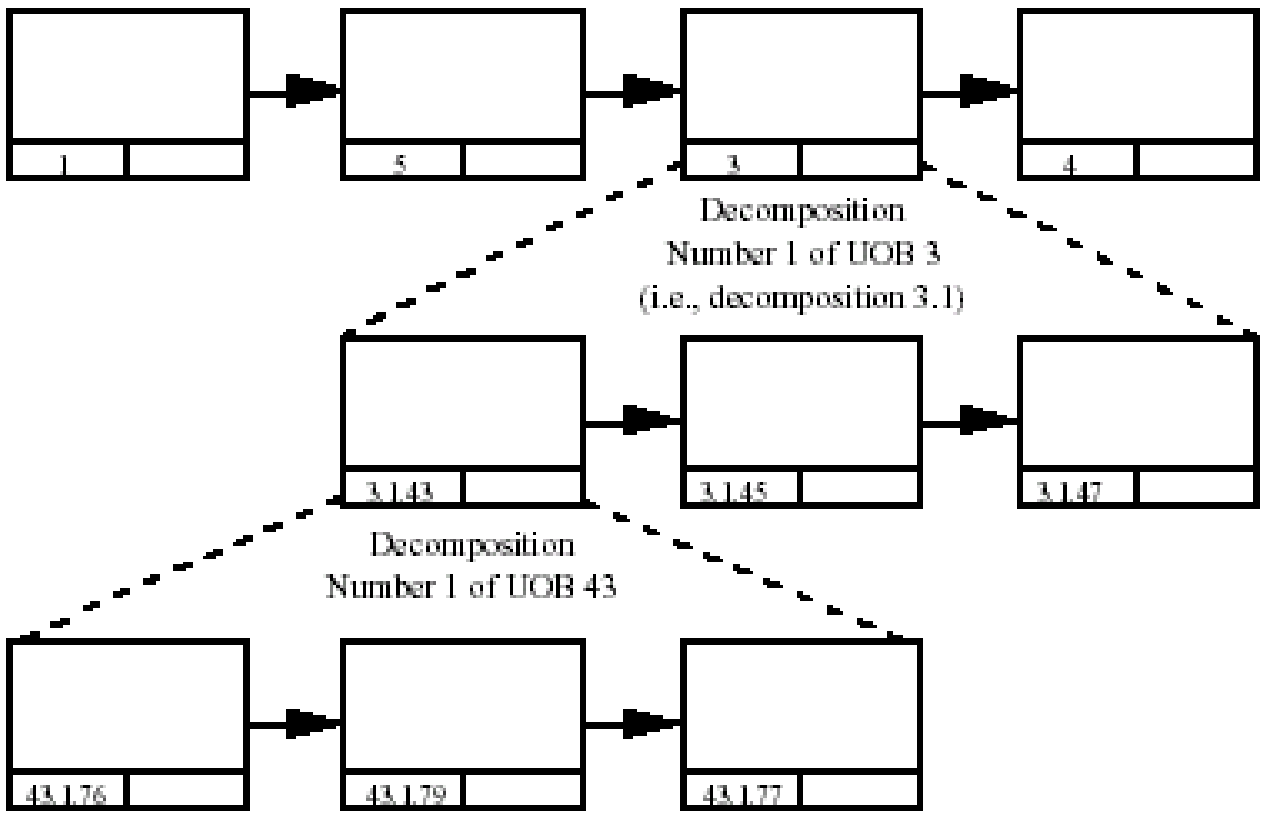


Figure 3-31
Unit of Behavior Numbering Scheme

Decomposition

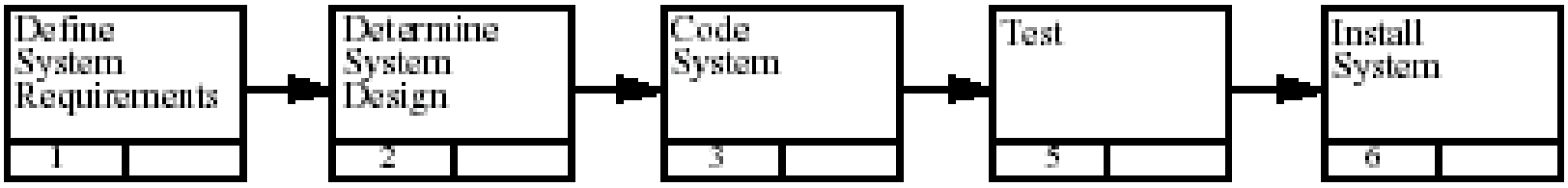


Figure 3-32
Disconnected UOB Example

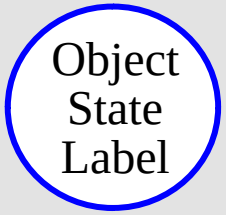
Uncertainty of the domain expert's knowledge

Analyzing Objects & Object States

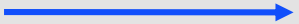
- Objects and their related processes can be studied in an object-centered view by using the Object State Transition Network (OSTN).

The IDEF3 OSTN Language

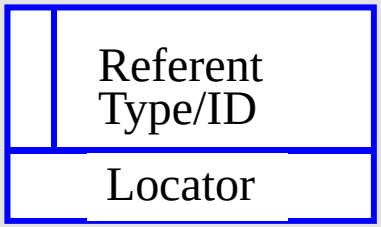
Object State



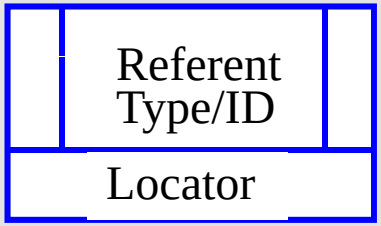
Transition Arc



Referents

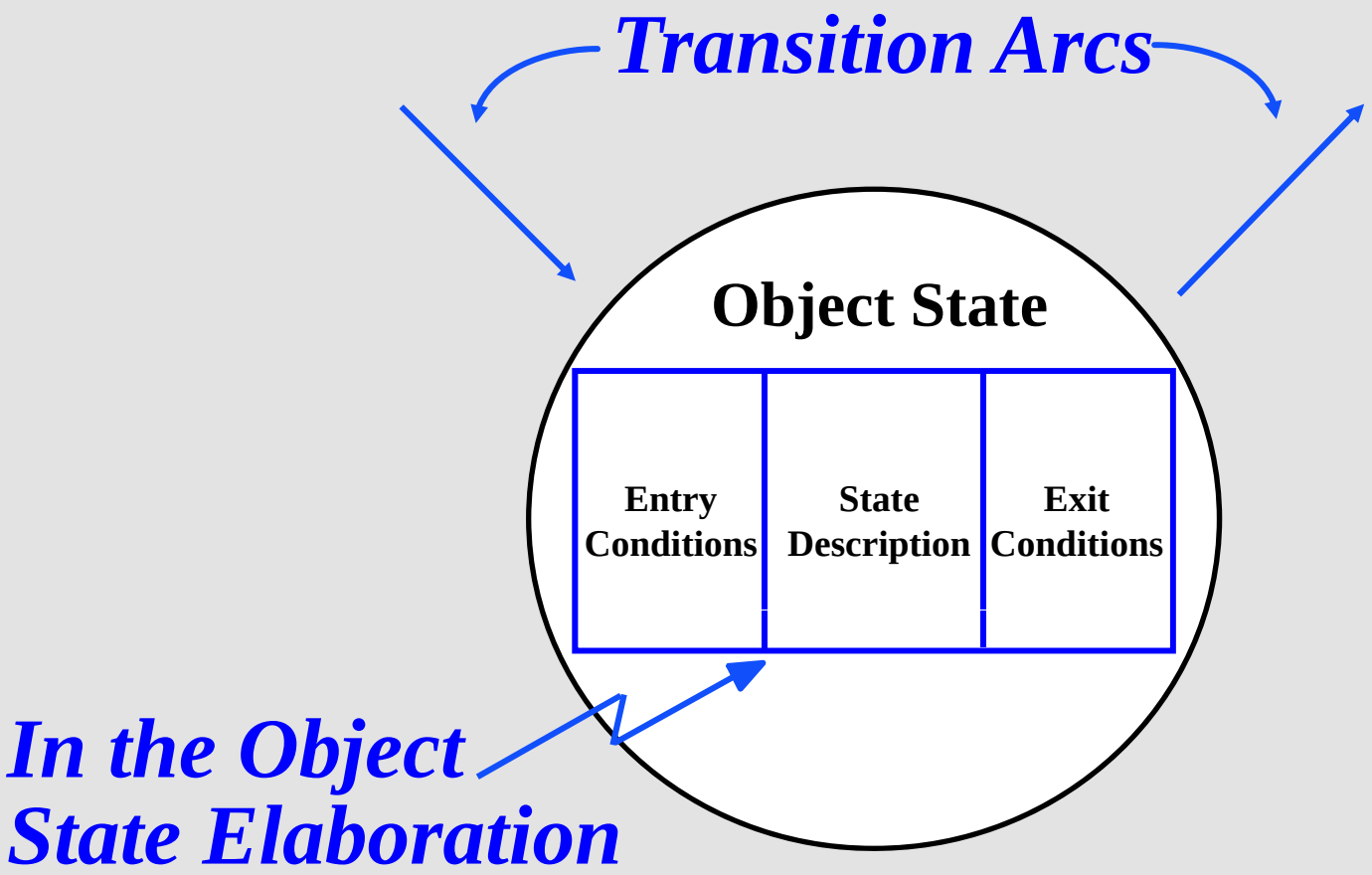


Asynchronous
Referent



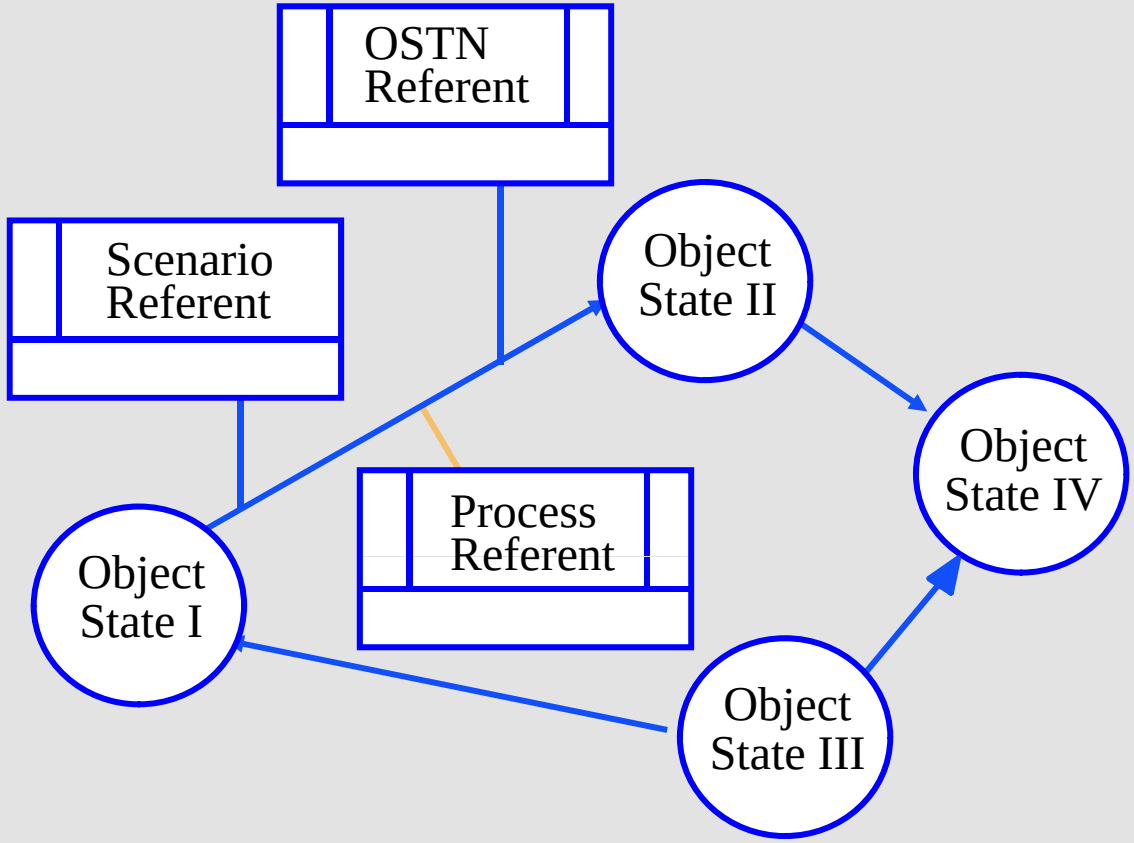
Synchronous
Referent

The IDEF3 OSTN Language

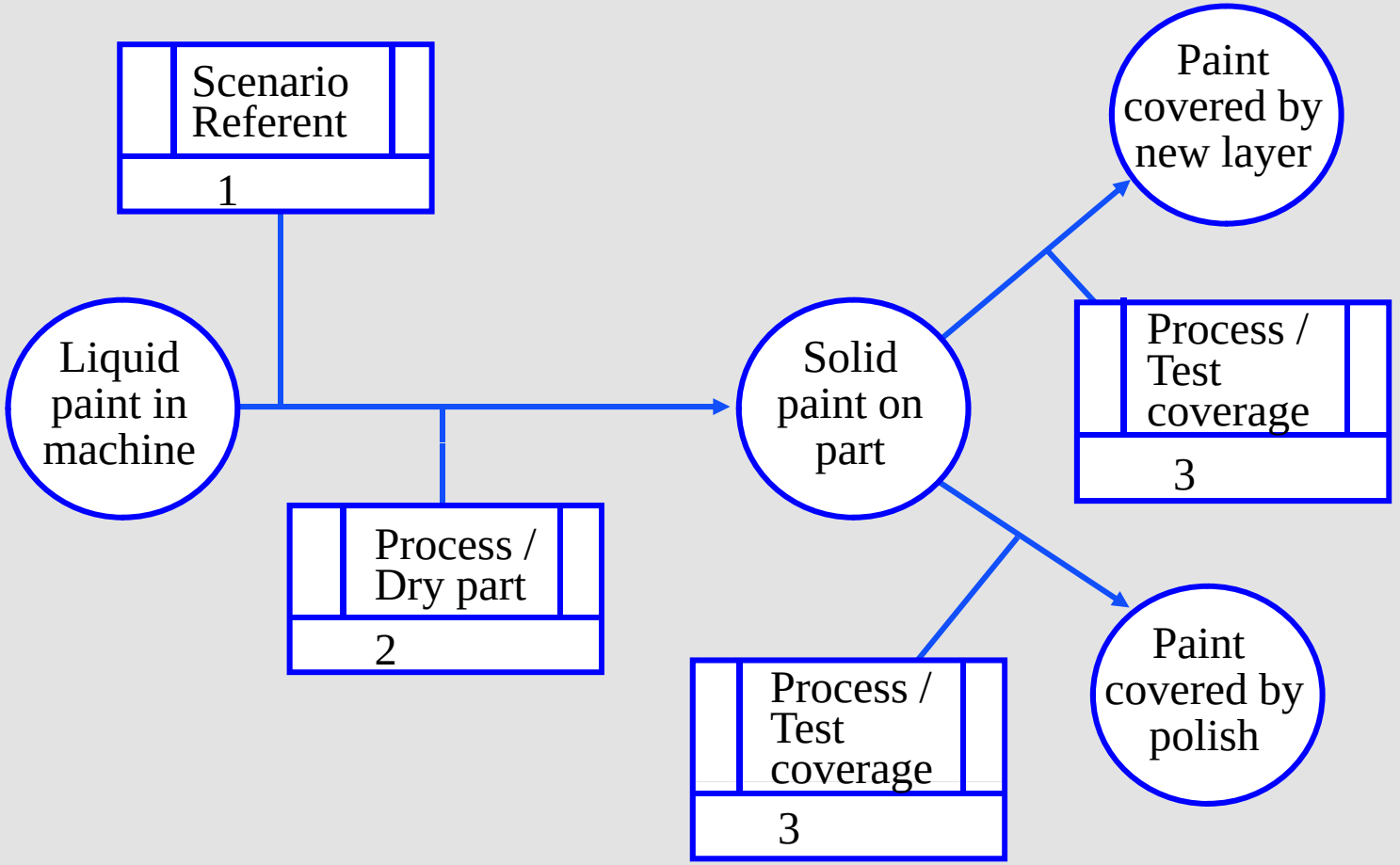


OSTN Diagram

- Allows construction of an object-centered view.
- Summarizes allowable transitions of an object in the domain.
- Used to document data life cycles.
- Cuts across the process flow diagrams.
- Characterizes dynamic behavior of objects.



Paint OSTN (Focus Object: Paint)



Eight Process Design Principles

Principle I

Process design is a design activity.

- Primarily creative in nature
  - Find, copy, and adapt best practices
- Primarily iterative in execution
- Requires cost/performance/benefit/risk tradeoffs
  - Simulation analysis
  - ABC analysis
- No one single solution
- Not complete until specifications are produced

Principle II

Process design expertise is made up of a set of skills and the knowledge of how to apply those skills opportunistically.

- Constraint management / satisfaction
- Recognize difference between requirements and design goals
- Not a flow chart
- Progress not necessarily made in a linear fashion
- Should result in multiple alternatives that are subject to tradeoff analysis

Principle III

"Object design" plays a central role in the process design.

- Inputs and outputs
- Resources
- Intermediate objects
- Interface objects
- Object state transitions
- Object "quality" measures

Principle IV

Processes must be specified to a level that can allow allocation to specific resources available in the execution environment.

- Decomposition into sub processes
- Termination condition of process design
- Processes will change as the skills and capabilities of the people and machines change

Principle V

Physical and logical input/output contiguity must be maintained (Conservation Law).

- Input/output of each process unit must be specified and matched with the input available and the output required at the position of the process unit in the process flow
- Drives decomposition
- Highly dependent on object design

Principle VI

There will always be failures that must be addressed.

- Failure mode identification
- Failure mode analysis
- Failure detection sub process design
- Failure handling sub process design
- Robustness relative to failures

Principle VII

Process design includes the design of process steps for by-product management.

- waste or scrap
- identify
- collect
- dispose

Principle VIII

Process design includes design of process steps and objects for execution coordination and management.

- Concurrent processes
- Resource allocation
- Work item prioritization
- Status, performance, traceability, data collection
- Interface management

IDEF3 Models

Reading

Building

Reading IDEF3 Models

- Study the context, purpose, and viewpoint to understand the scope of the model.
- Read process flow diagrams from left to right, starting with the leftmost process(es).
  - Reading a diagram in this manner is called “performing a walkthrough.”
- Examine carefully the description and elaboration form of each element.

Building IDEF3 Models

Some practical guidelines—

- Do *not* follow an XOR fan-out junction with an AND fan-in junction.
- Avoid multiple leftmost processes in a diagram: their interpretation is ambiguous.
  - Use a fan-out junction preceding the multiple leftmost processes to clarify the process flow.
- When possible, avoid nested fan-out junctions to simplify diagrams.
- A fan-out junction immediately following a fan-in junction can indicate a missing process in the diagram.

Conclusion

- IDEF3 documents current processes for standardization and provides guidelines for new process members to reduce the learning curve.
- IDEF3 provides a mechanism to capture the temporal sequence of a process, the decision logic effecting the process, and the state transitions of objects within the process.
- IDEF3 serves as a tool to analyze existing processes and design and test new processes before embarking on expensive changes.

IDEF3 Process Modeling

- Review & Questions