

IDEFO

Activity Modeling

What is an Activity Model?

- ◆ A representation of the activities and the relationships between and among those activities in an existing or planned system.
 - ◆ A collection of diagrams, glossary, and text along with the context, viewpoint, and purpose statements.

Benefits of Activity Modeling

- ◆ Document current activities for standardization and provide guidelines for new activity users to reduce the learning curve.
- ◆ Capture and analyze AS-IS activities.
- ◆ Design/Redesign activities for TO-BE scenarios.

What is IDEF0?

- ◆ An activity modeling method.
- ◆ Supports descriptions at any desired level of detail through Decompositions.
- ◆ Provides both a process and a language for constructing a model of the activities and their interrelationships.

Generic Activity Modeling Tool?

- ◆ Automates the IDEF0 method.
- ◆ Adheres to the method standard.
- ◆ Provides background quality checking and advisory support.
- ◆ Employs a SmartDraw capability.

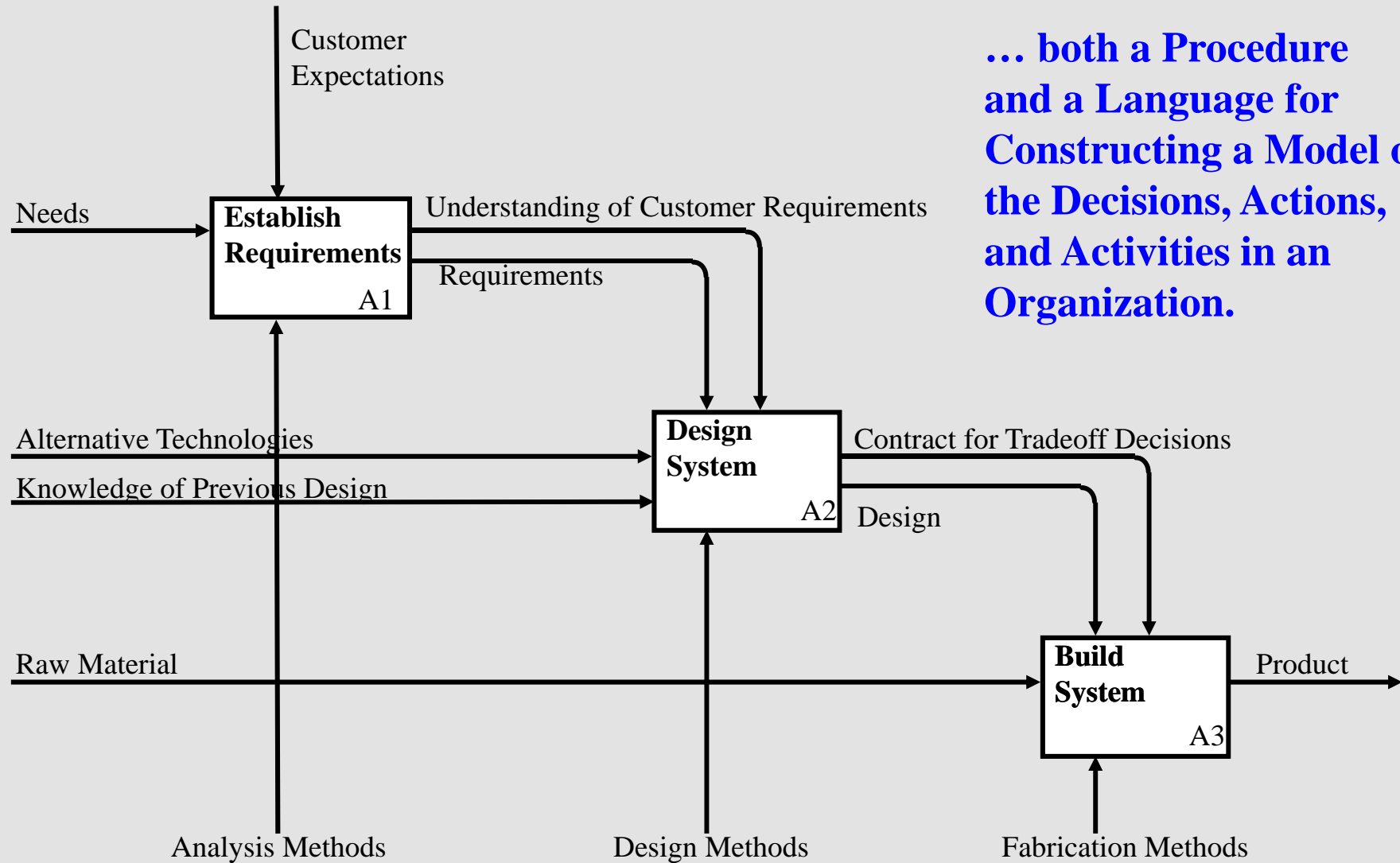
Why Develop An Activity Model?

- ◆ To identify, document, and communicate what an enterprise does.
- ◆ To facilitate the collection of data needed to perform functional analysis.
- ◆ To identify value added and non-valued added activities.
- ◆ To identify activities or functions that need to be improved.

What IDEF0 Represents

- ◆ Functions - Decisions, Actions, or Activities of the domain
- ◆ Objects - Physical or conceptual of the domain
- ◆ Roles that objects stand-in relative to functions
- ◆ Relations between functions formed by objects
- ◆ Relations between functions formed by the composition relationship

IDEFO Provides ...



... both a Procedure and a Language for Constructing a Model of the Decisions, Actions, and Activities in an Organization.

Components

Context, Purpose, & Viewpoint

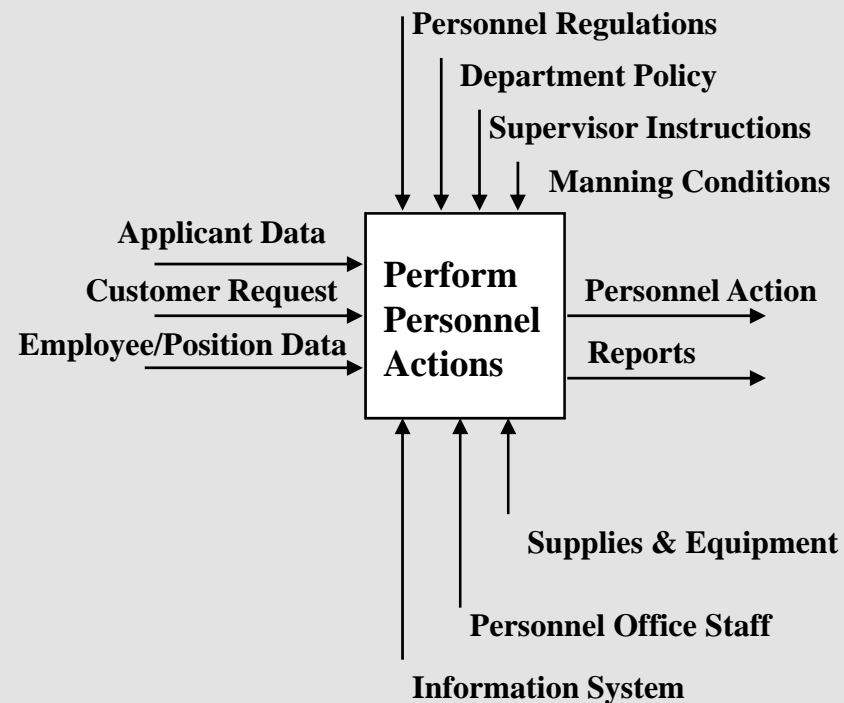
Establishing the Model Objectives

- ◆ Viewpoint
 - ◆ Determines what can be seen and from what slant.
- ◆ Purpose
 - ◆ Establishes the goal of the communication intended by the model.
 - ◆ Defines why the model is being developed.
 - ◆ Specifies how the model will be used.
- ◆ Context
 - ◆ Establishes the scope of a model.
 - ◆ Establishes the subject as part of a larger whole.
 - ◆ Creates a boundary with the environment.

Context

- ◆ The context defines the boundaries of your model or what you will include in the model.

Employee/ Position Data
comes from outside the model.



- ◆ Scopes the model and defines the boundaries.
 - ◆ If the scope is too big, the model becomes too complex and resource-intensive.
 - ◆ If the scope is too small, the model becomes trivial.

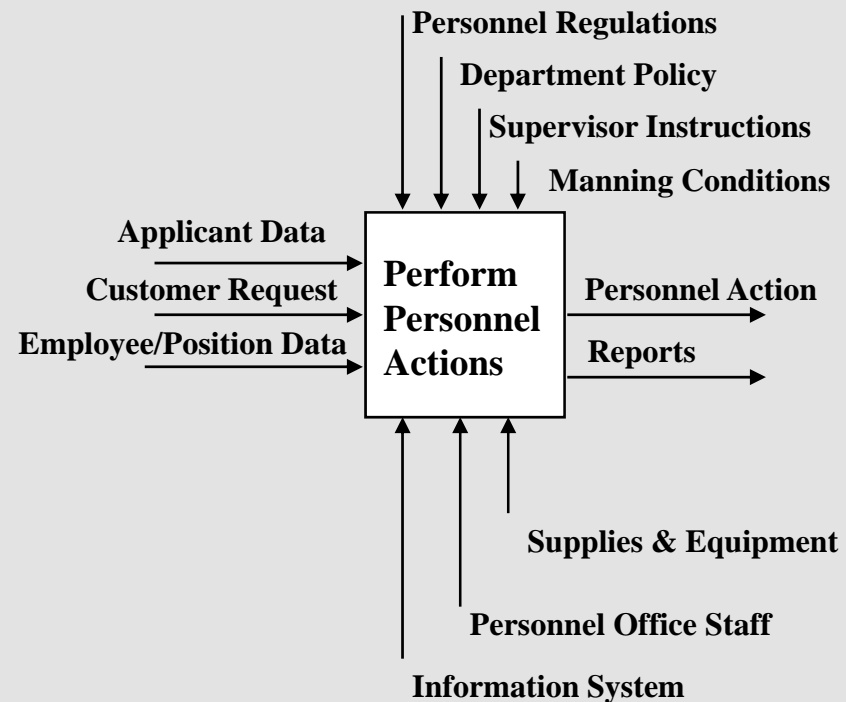
**Determining the context is
the most critical step in
Activity Modeling.**

Purpose

- ◆ The purpose defines the reason to develop this particular activity model.

Purpose:

To document the activities associated with managing Personnel Actions and identify non-value added activities that might be eliminated.



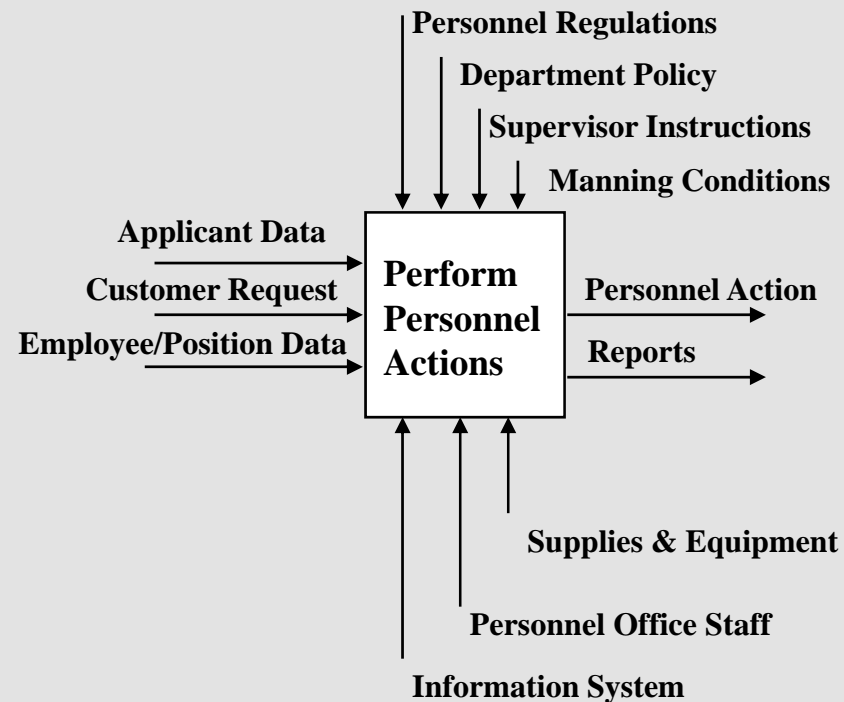
Viewpoint

- ◆ The viewpoint describes the perspective of the person or group developing the model.

Purpose:

To document the activities associated with managing Personnel Actions and identify non-value added activities that might be eliminated.

Viewpoint: Personnel Officer



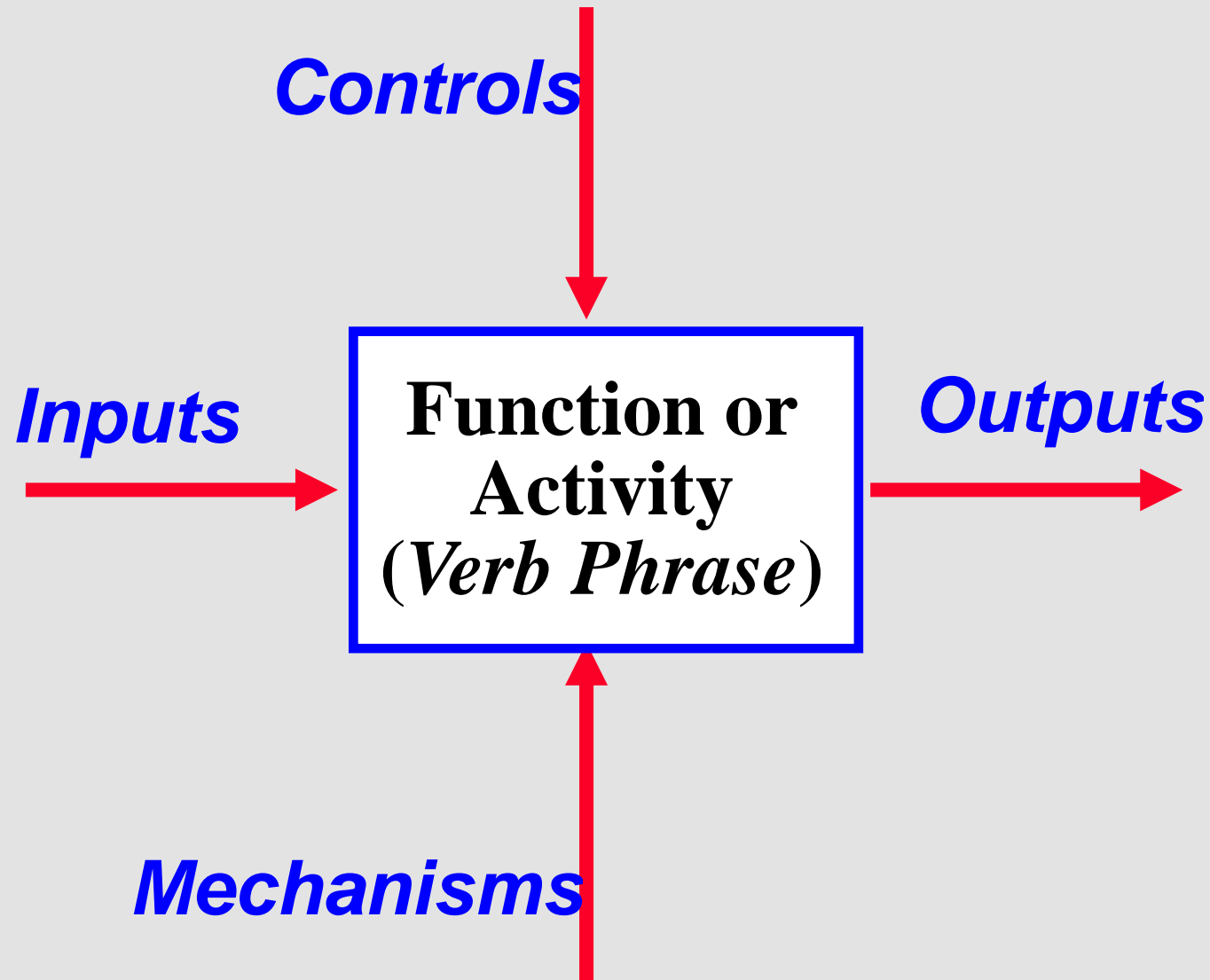
Purpose & Viewpoint Interaction

- ◆ Different purpose and viewpoint result in different models.

		<i>Purpose</i>	
		Non-value added Personnel Actions	Non-value added Separation Actions
<i>Viewpoint</i>	Personnel Officer	all personnel actions accomplished by multiple activities	a single type of personnel action and multiple activities
	Personnel Clerk	a few personnel actions accomplished by multiple activities	a single type of personnel action and a small set of activities

IDEF0 Graphical Modeling Language

Diagram Syntax



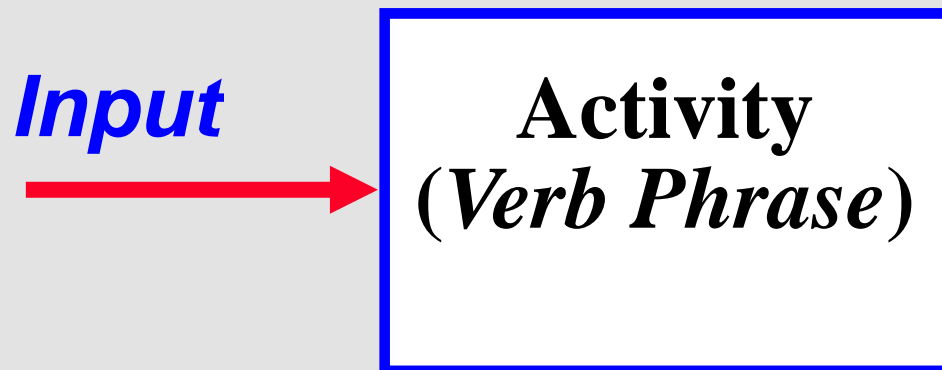
Activity

- ◆ An action, function, or operation. Represented by a box and labeled as a verb phrase.

Activity
(Verb Phrase)

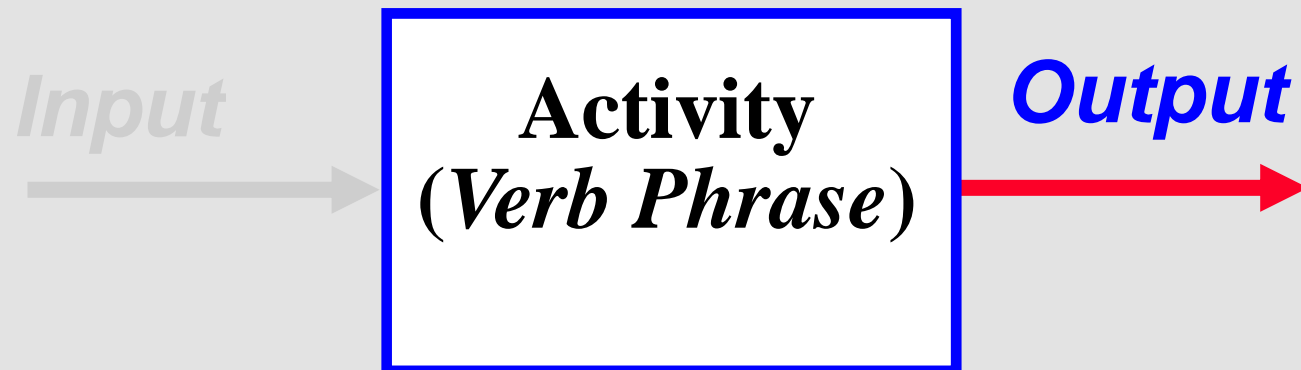
Input

- ◆ Any real object or data needed to perform an activity. Transformed through the completion of the activity.

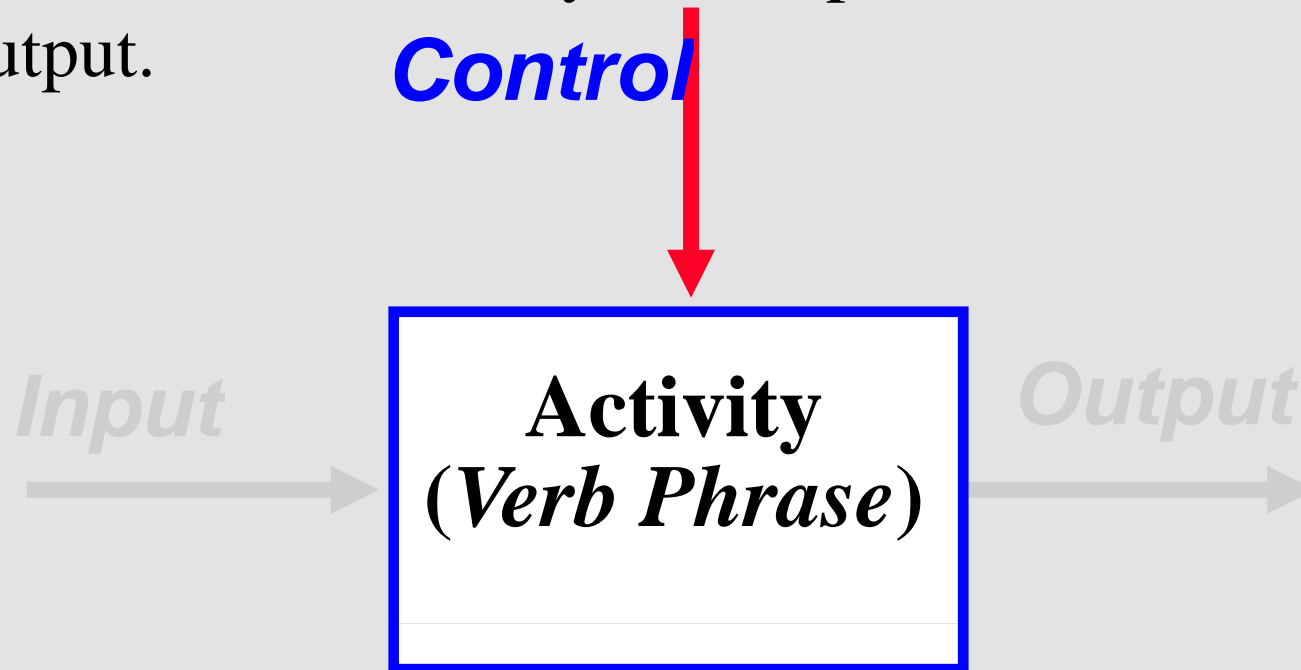


Output

- ◆ Results from the completion of the activity.

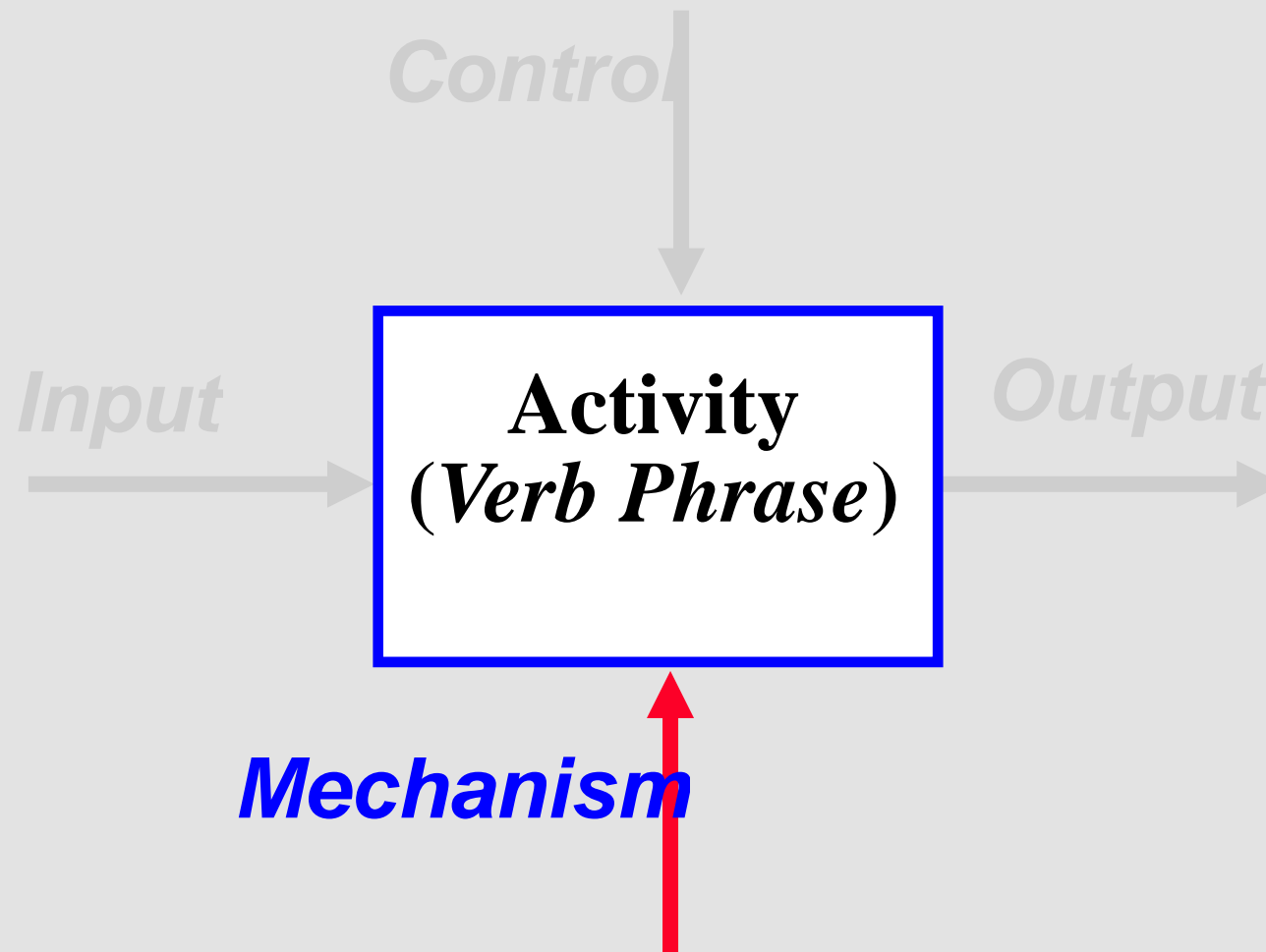


- ◆ Directs, guides, or initiates the activity. May also combine in some way with input(s) to result in an output.

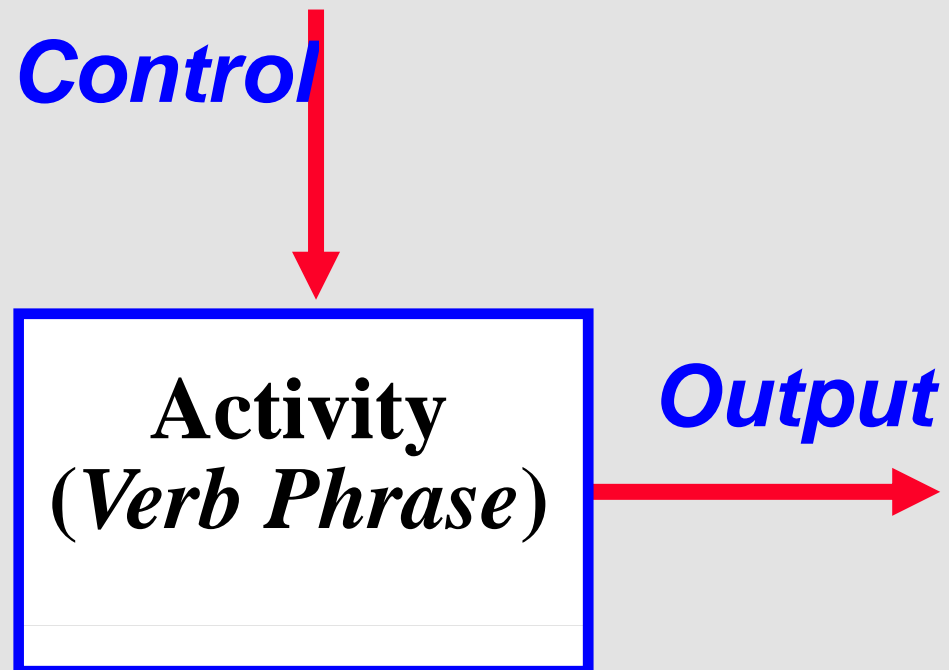


Mechanism

- ◆ Indicates how the activity is accomplished.



Minimal Requirements

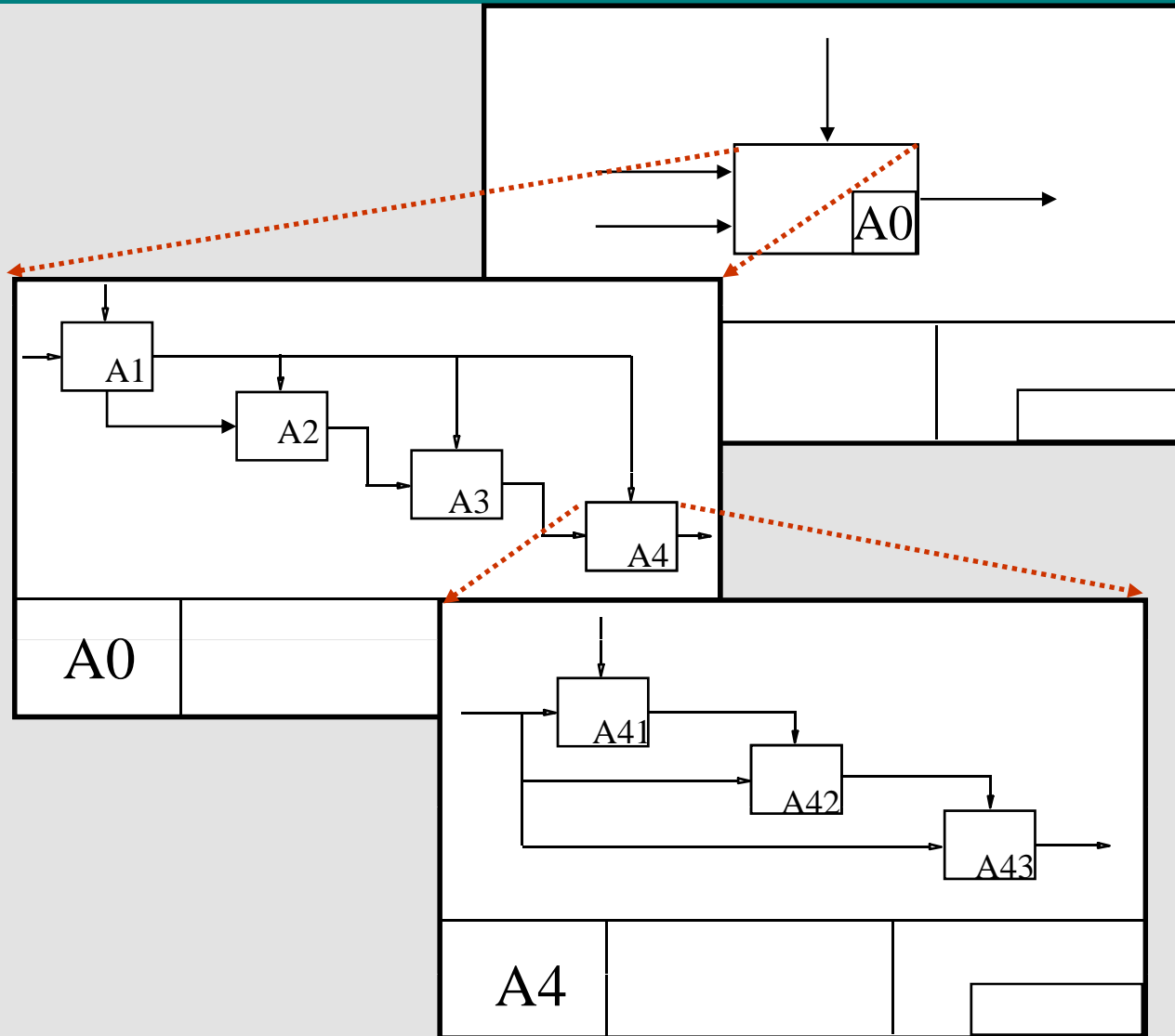


Functional Decomposition

Functional Decomposition

- ◆ Further defines an activity by dividing it into its sub-activities.
- ◆ Ensures the gradual, systematic exposition of detail required to understand and communicate what activities are being performed.

Decomposition



**More
General
(Parent)**



**More
Detailed
(Child)**

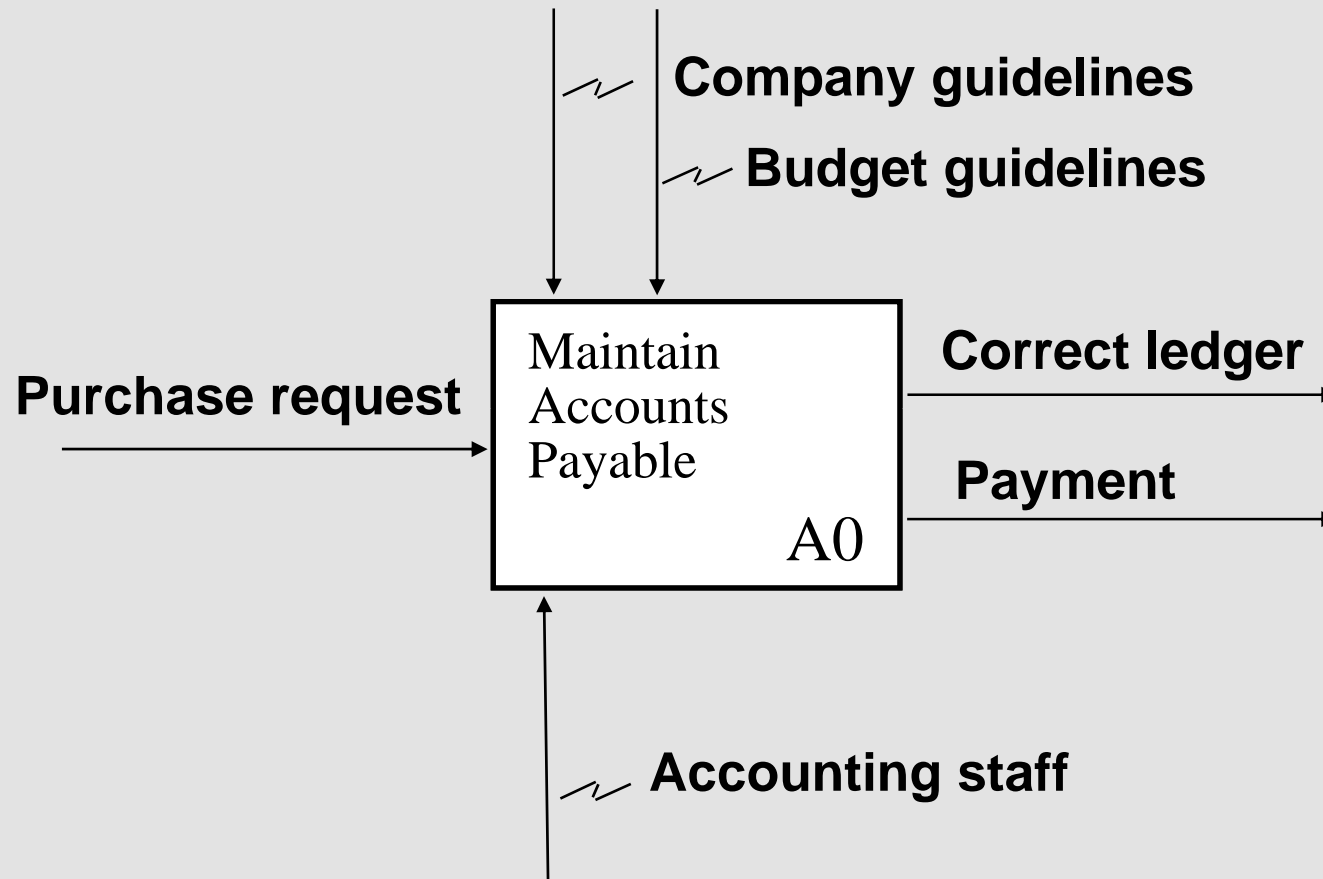
Functional Decomposition

- ◆ Each activity is composed of distinguishable sub-activities.
- ◆ A “parent” activity is decomposed into three to six “child” activities.
- ◆ Each child can become a parent and be further sub-divided.

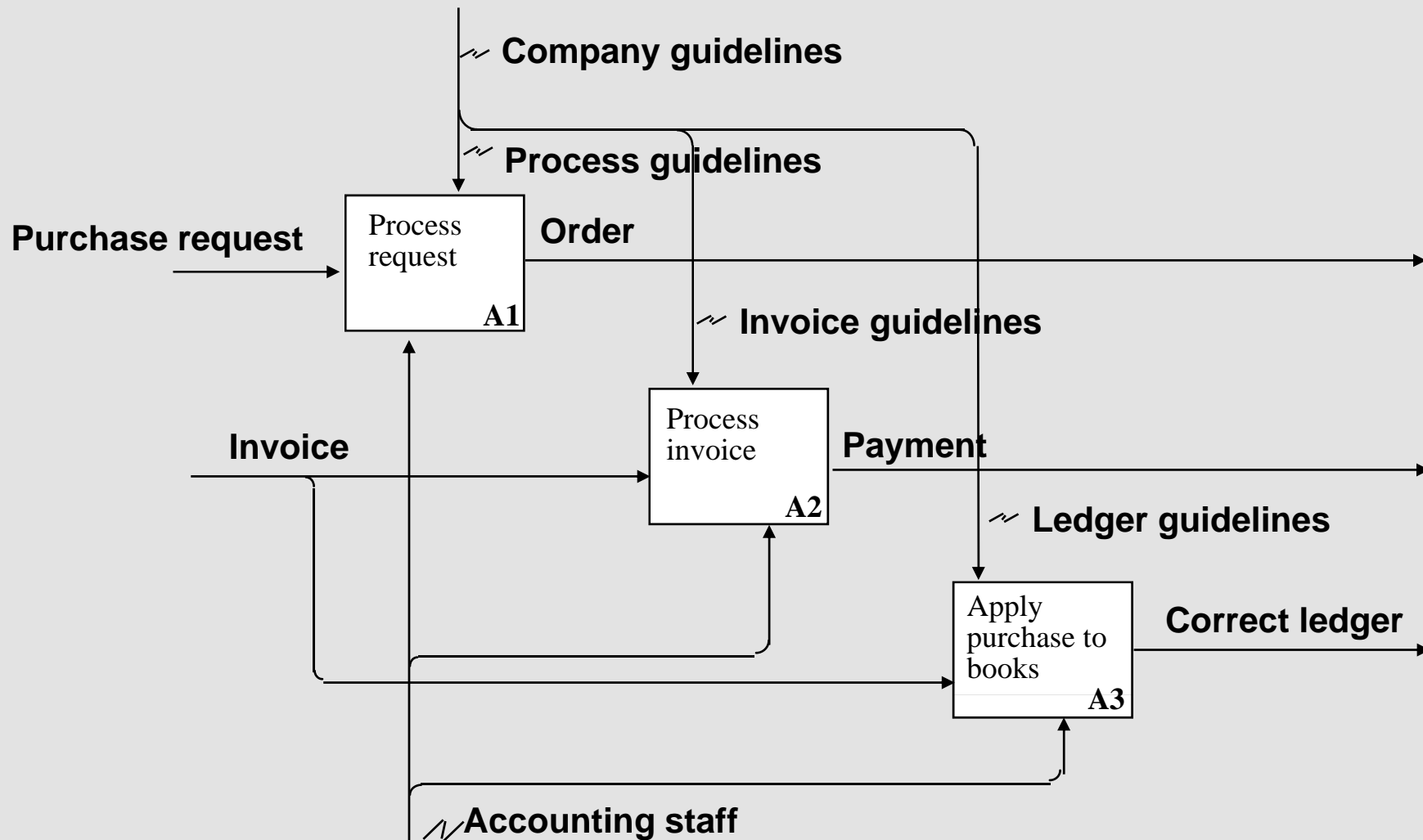
Functional Decomposition

- ◆ Each activity in the model is unique and not represented multiple times.
 - ◆ The Sales Dept., Accounting Dept., and Engineering Dept. may all submit Monthly Expense Reports...
 - ◆ ...but in the activity model there is only one activity called “Create Monthly Expense Reports.”

Decomposition



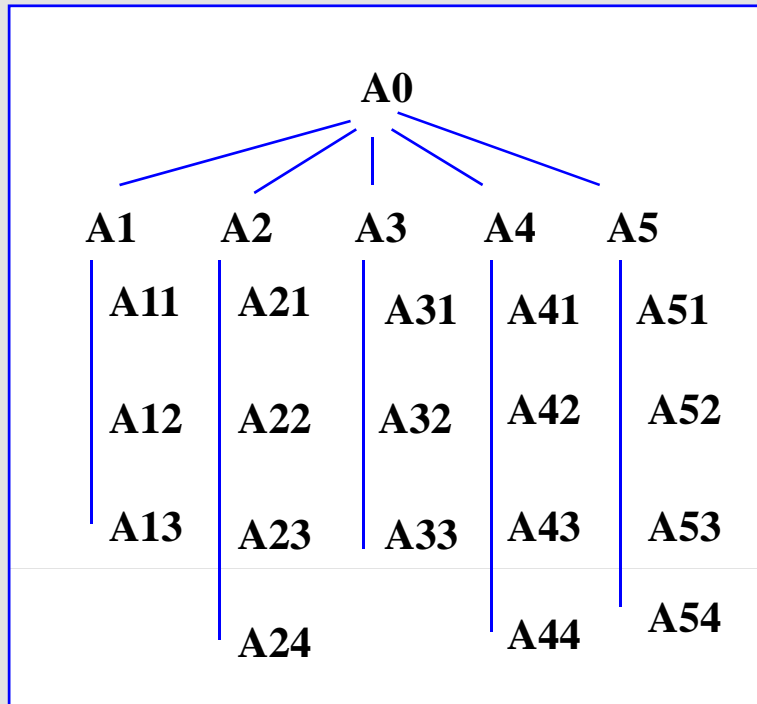
Decomposition



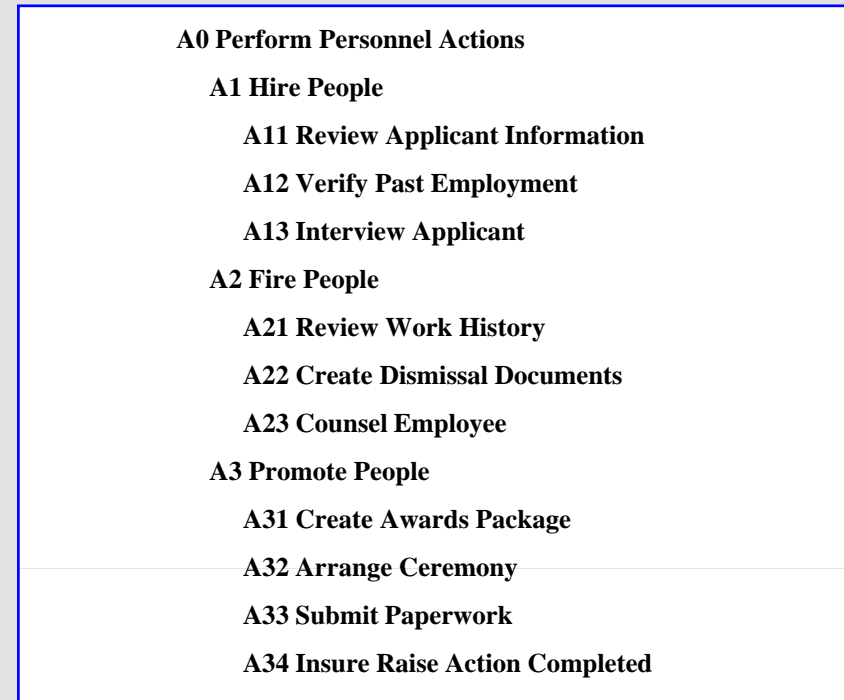
Activity Hierarchy

- ◆ Each activity in a model is uniquely identified with an Activity Number (A0, A1, A12, etc.).
- ◆ Each activity can be uniquely placed within a model according to its relative decomposition number.
- ◆ An activity is depicted only once in an activity model.

Activity Hierarchy within a Model



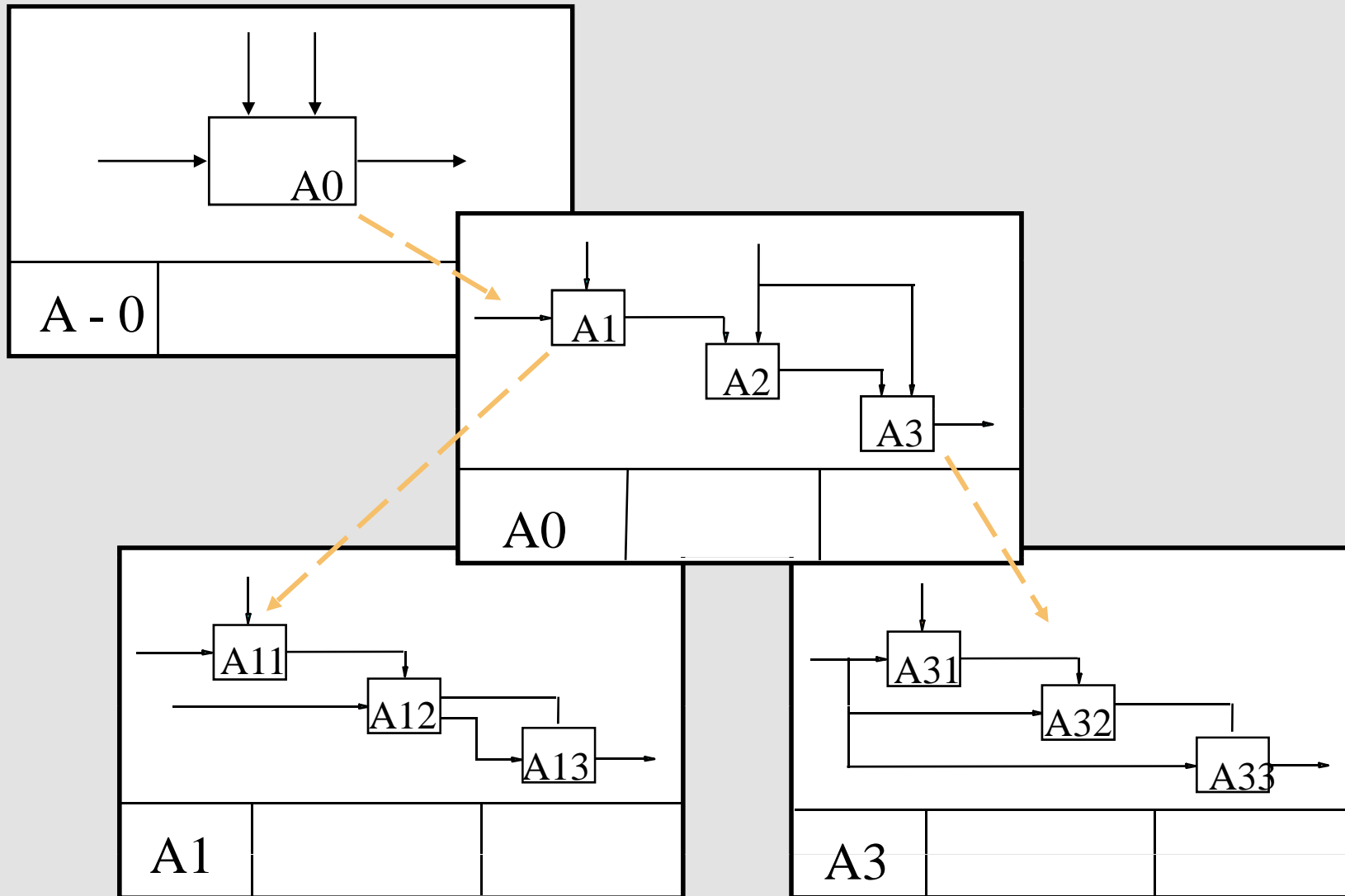
Node Tree



Indented List

An activity is depicted only once in an activity model.

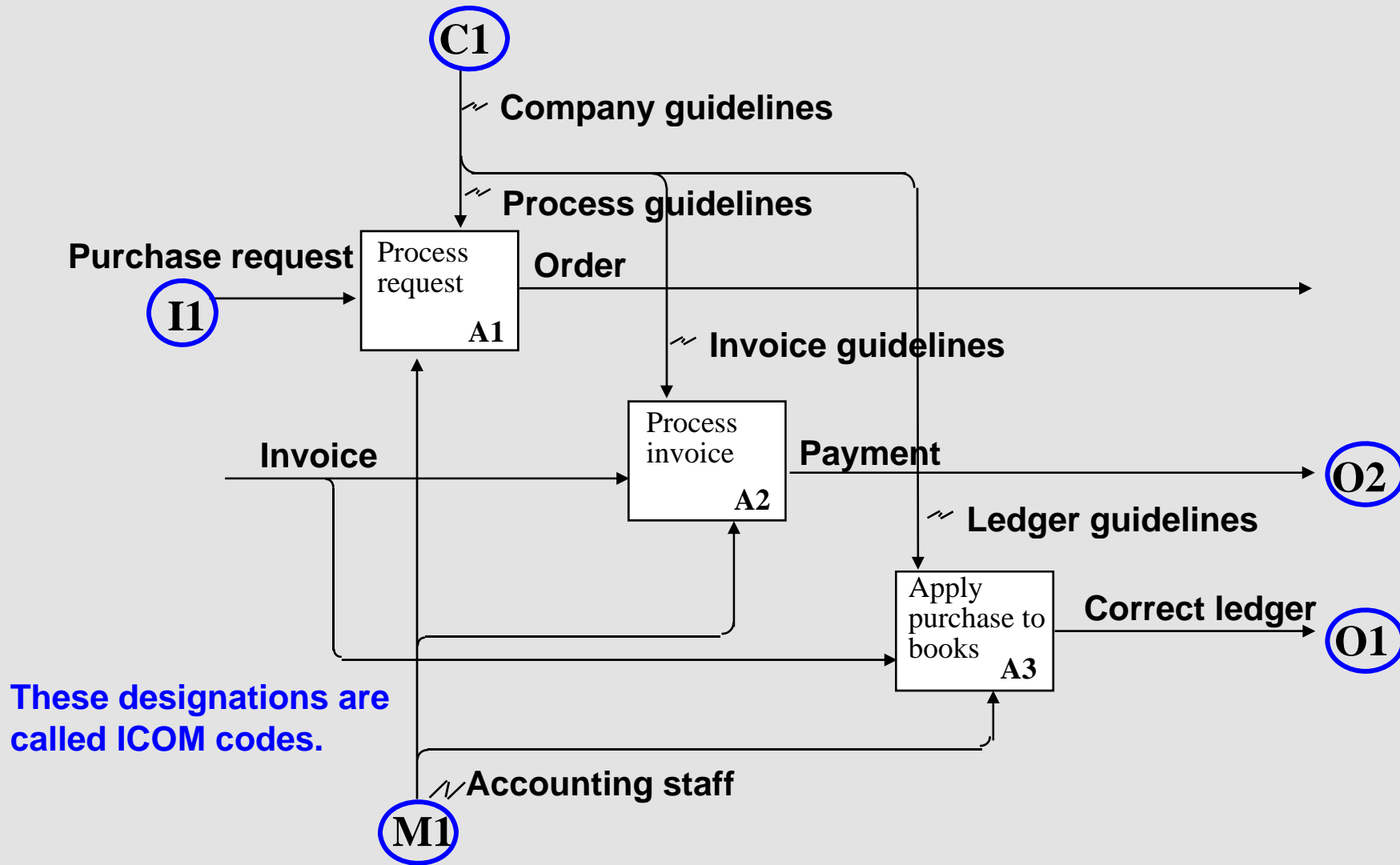
Diagram Numbering



Concept Hierarchy within a Model

- ◆ Concepts (inputs, controls, outputs, and mechanisms) are not uniquely identified within a model, but are identified between parent and child activities.
- ◆ Each concept is identified by a letter and number combination that specifies the concept's relative position on the parent diagram.

Concept Hierarchy within a Model

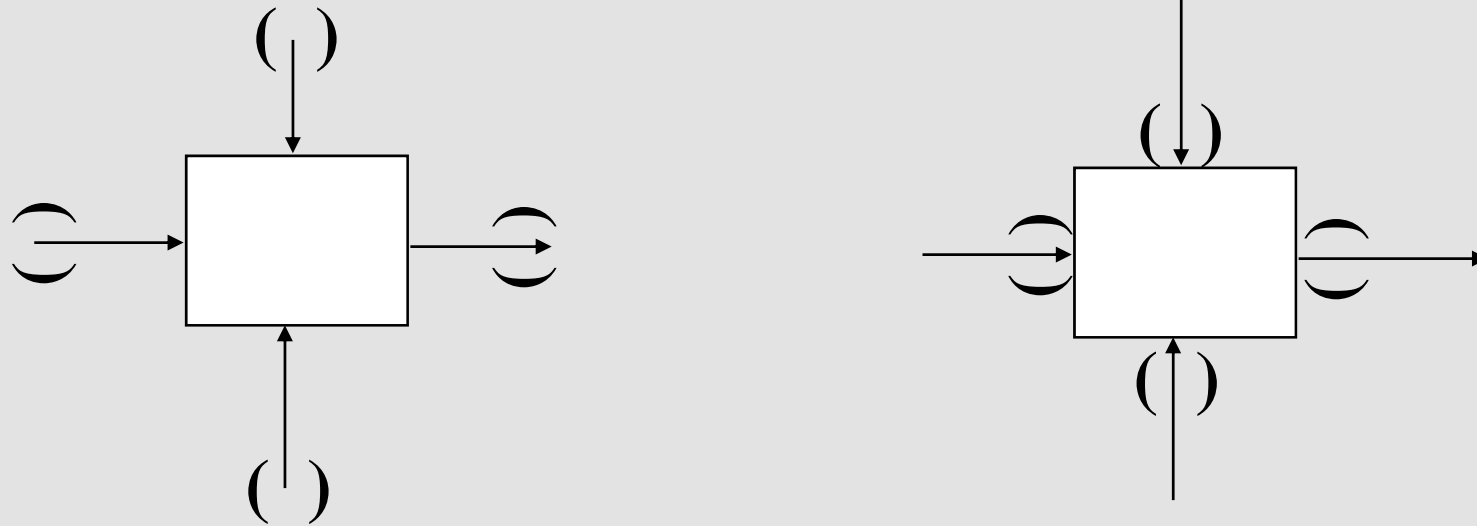


Tunneling

Tunneled concepts ...

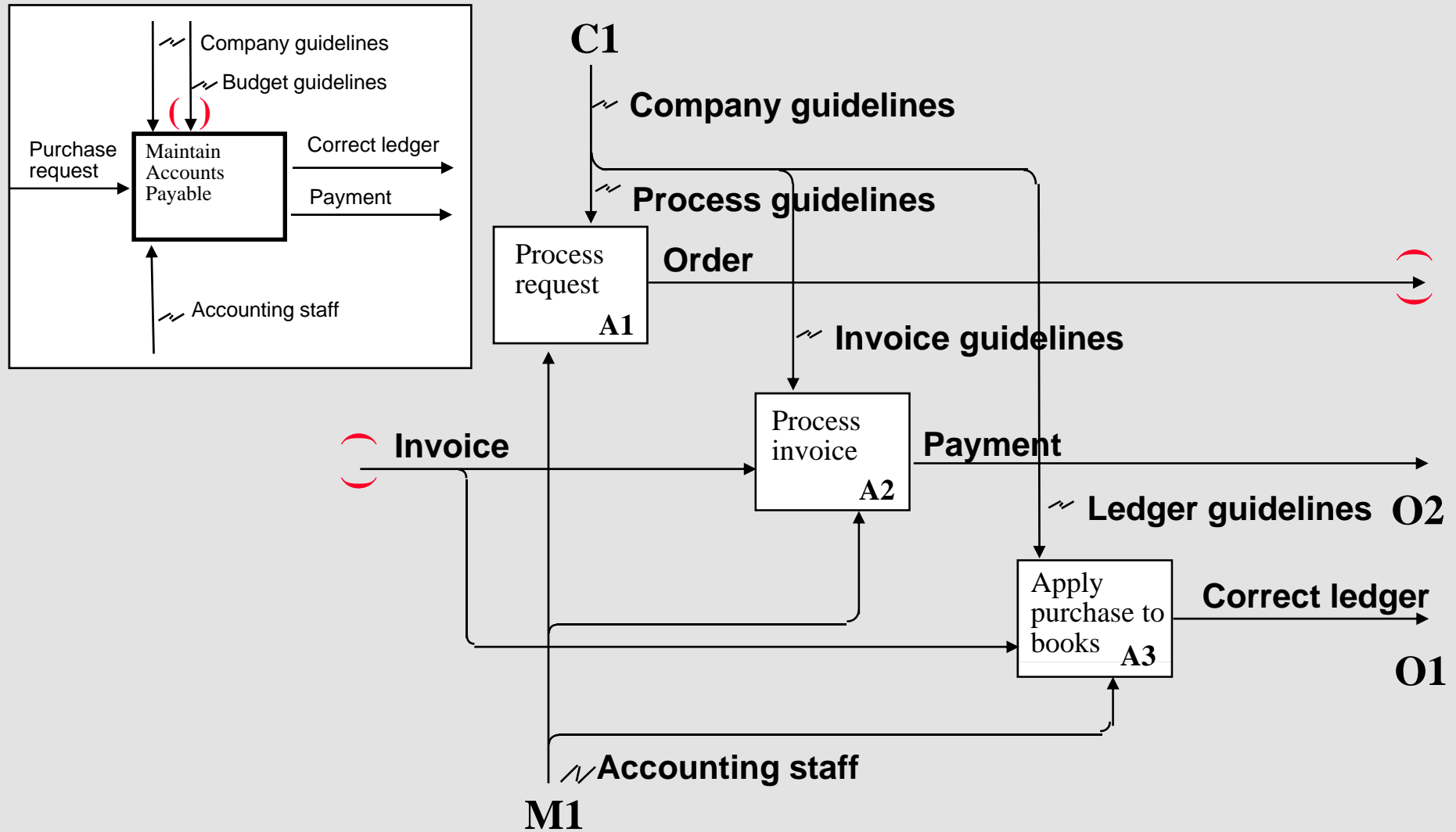
- ◆ Are intended to simplify a diagram.
- ◆ Communicate functional relationships between activities without cluttering every diagram in-between.
- ◆ Are not intended to be used as a means of “eliminating” unnecessary concepts from a model.

Tunneling



- ◆ A concept tunneled at the unconnected end indicates that the concept will not be shown at a higher level.
- ◆ A concept tunneled at the connected end indicates that the concept will not be shown at a lower level.

Tunneling



Tunneling

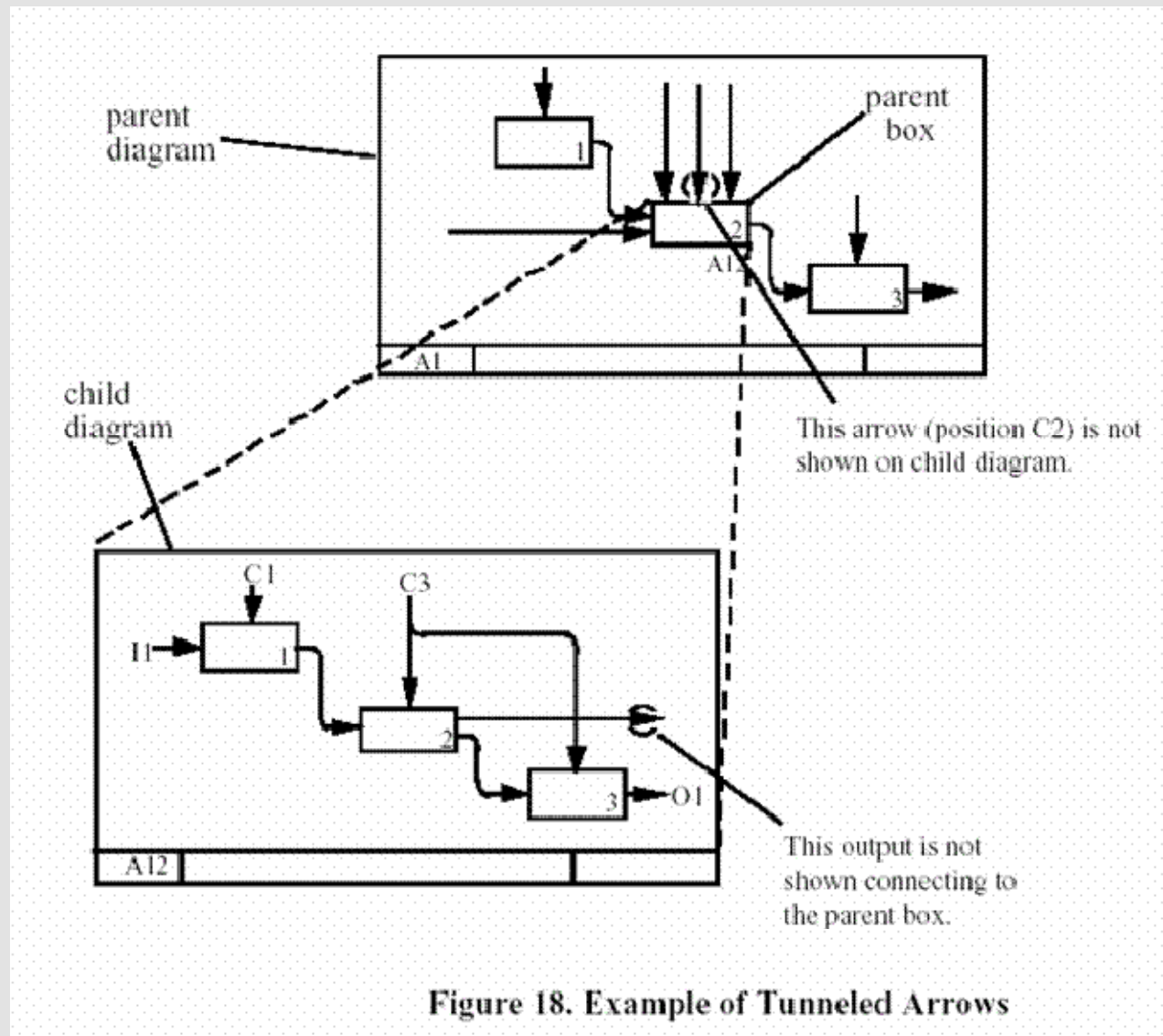
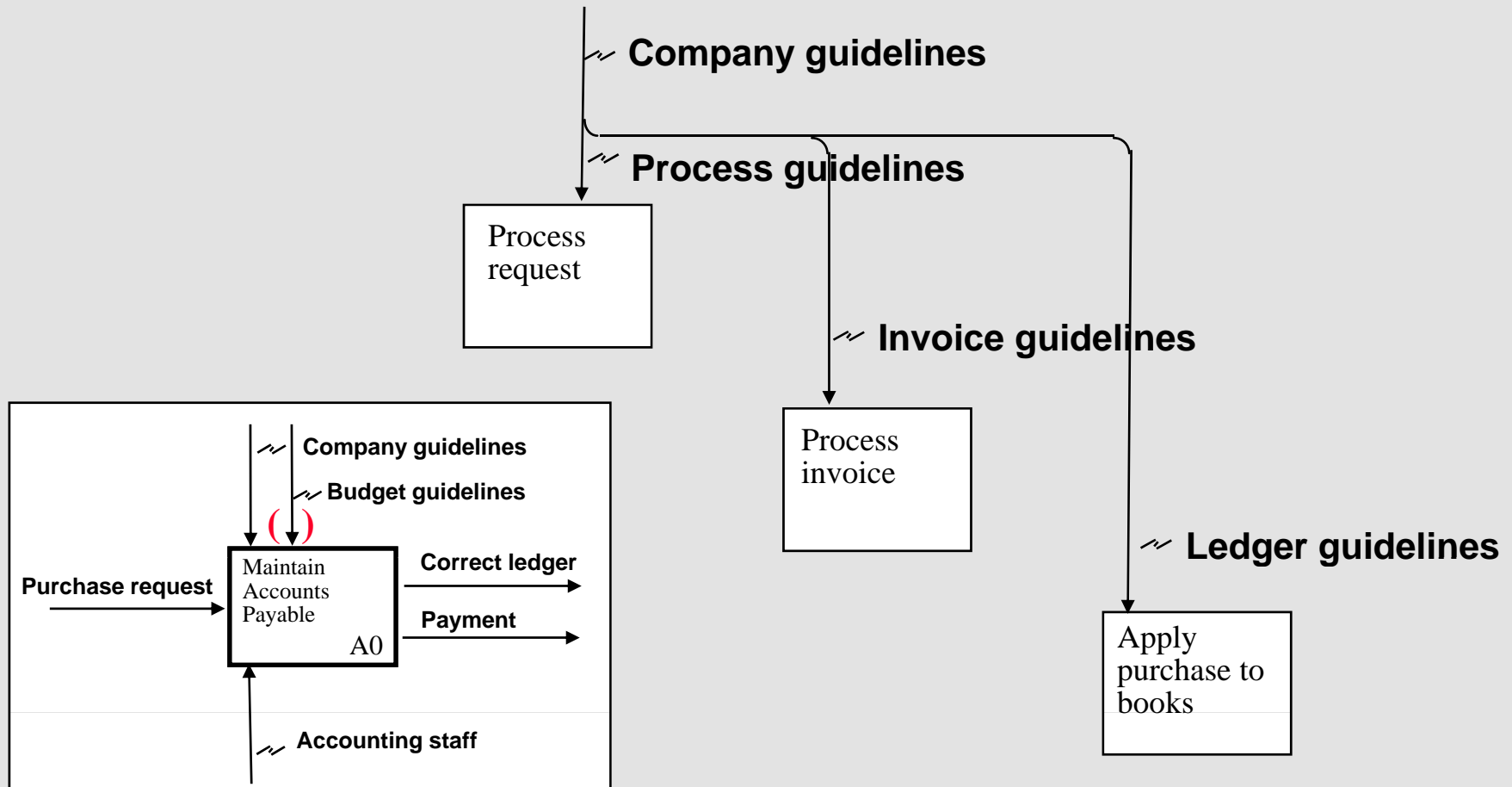


Figure 18. Example of Tunneled Arrows

Bundling & Unbundling

- ◆ Bundling allows us to group several concepts into a larger “set” of concepts.
- ◆ Unbundling allows us to decompose a general concept into its component concepts.

Bundling & Unbundling



Bundling & Unbundling

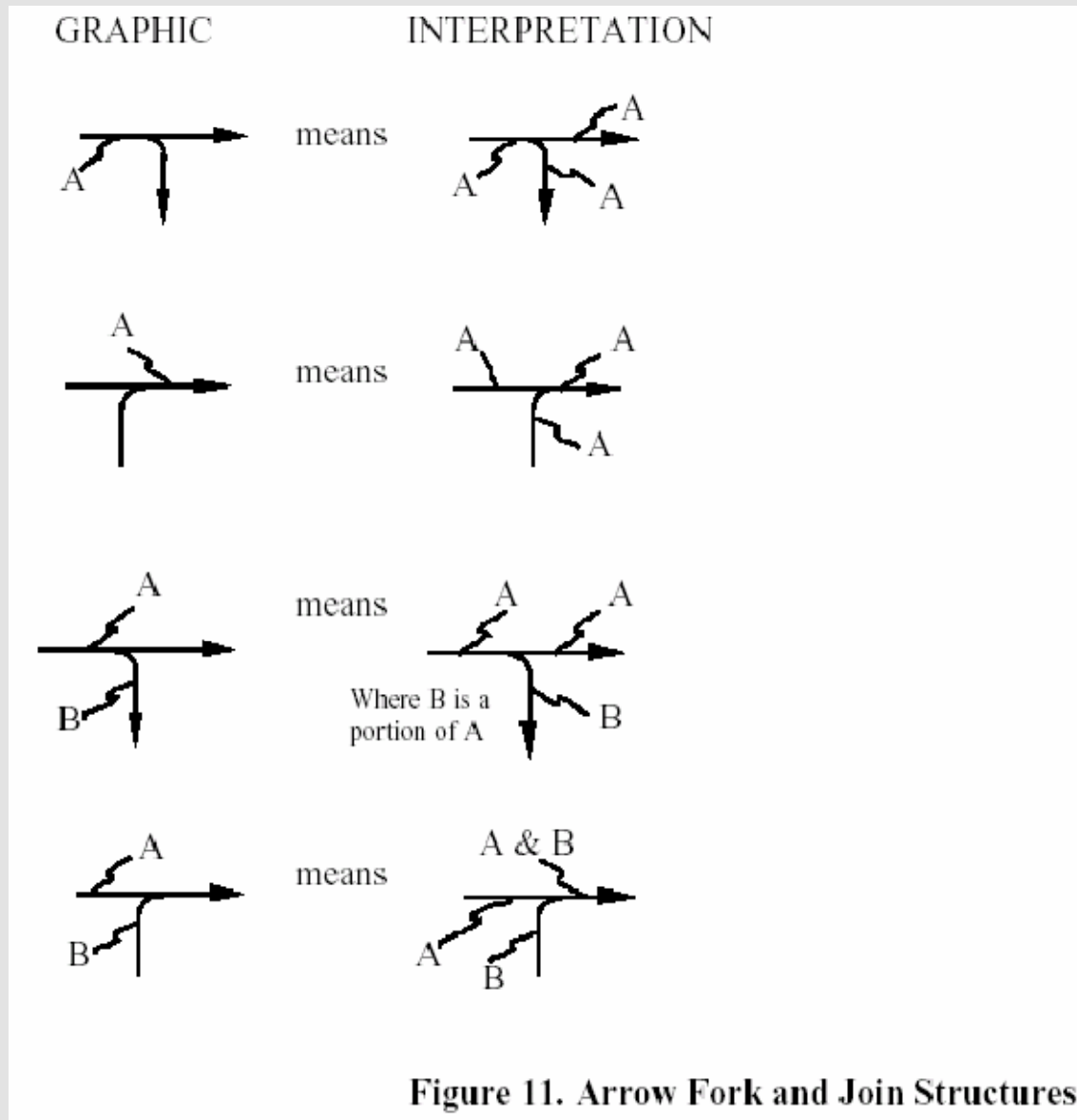
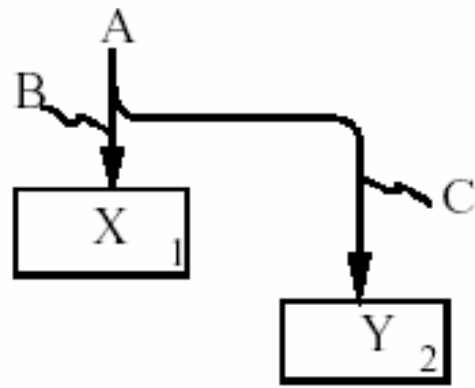


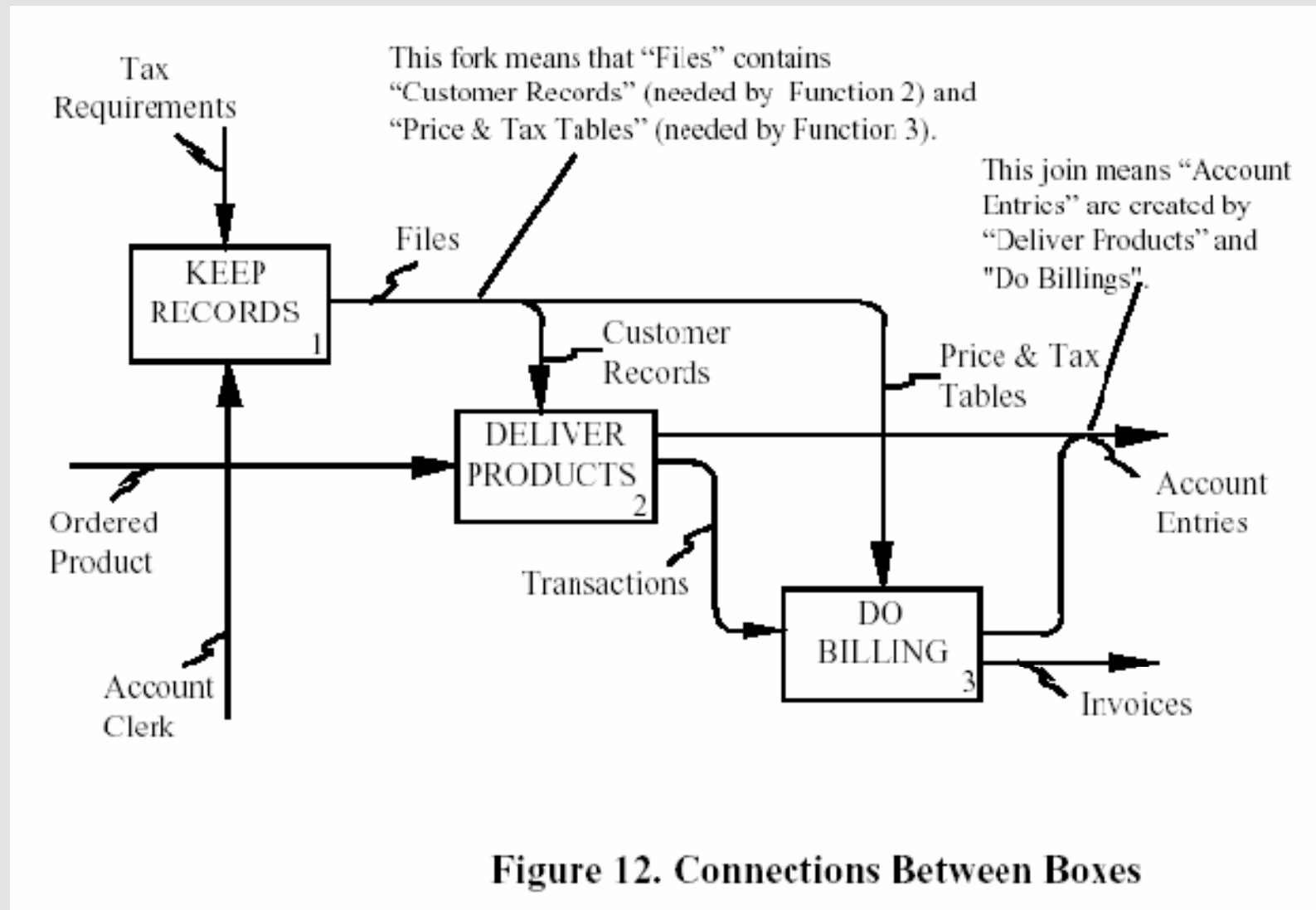
Figure 11. Arrow Fork and Join Structures

Bundling & Unbundling



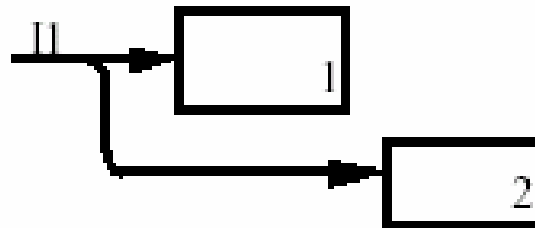
Pipeline A splits into B and C to provide controls to X and Y.

Bundling & Unbundling

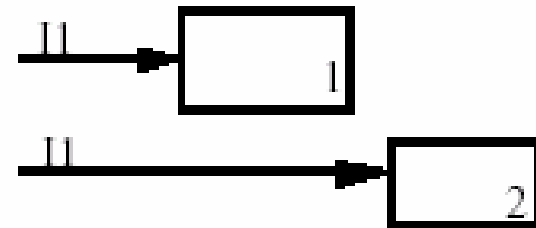


Bundling & Unbundling

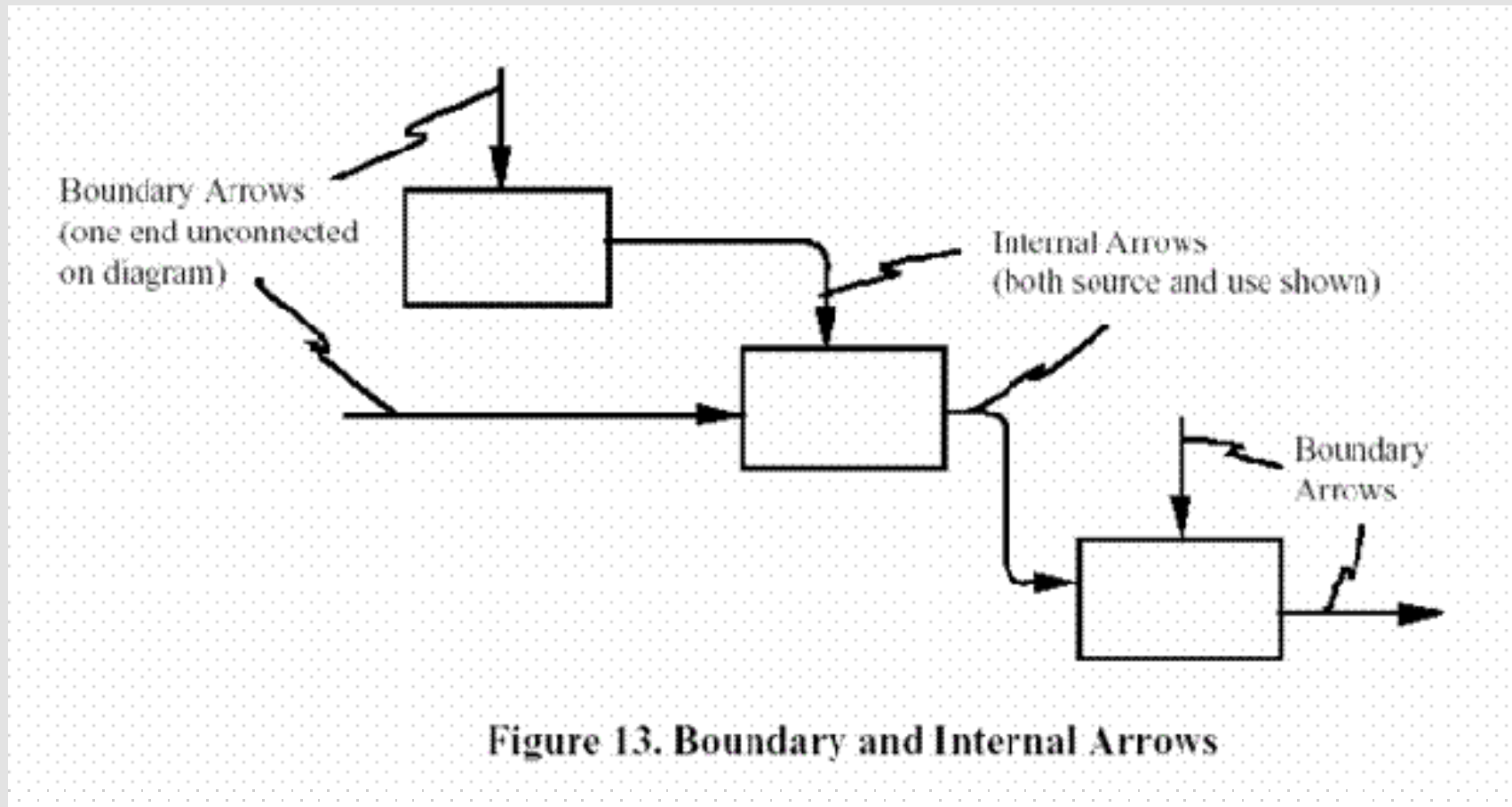
This is preferred



over this



Bunary and Internal Arrows



Boundary and Arrows Correspondence

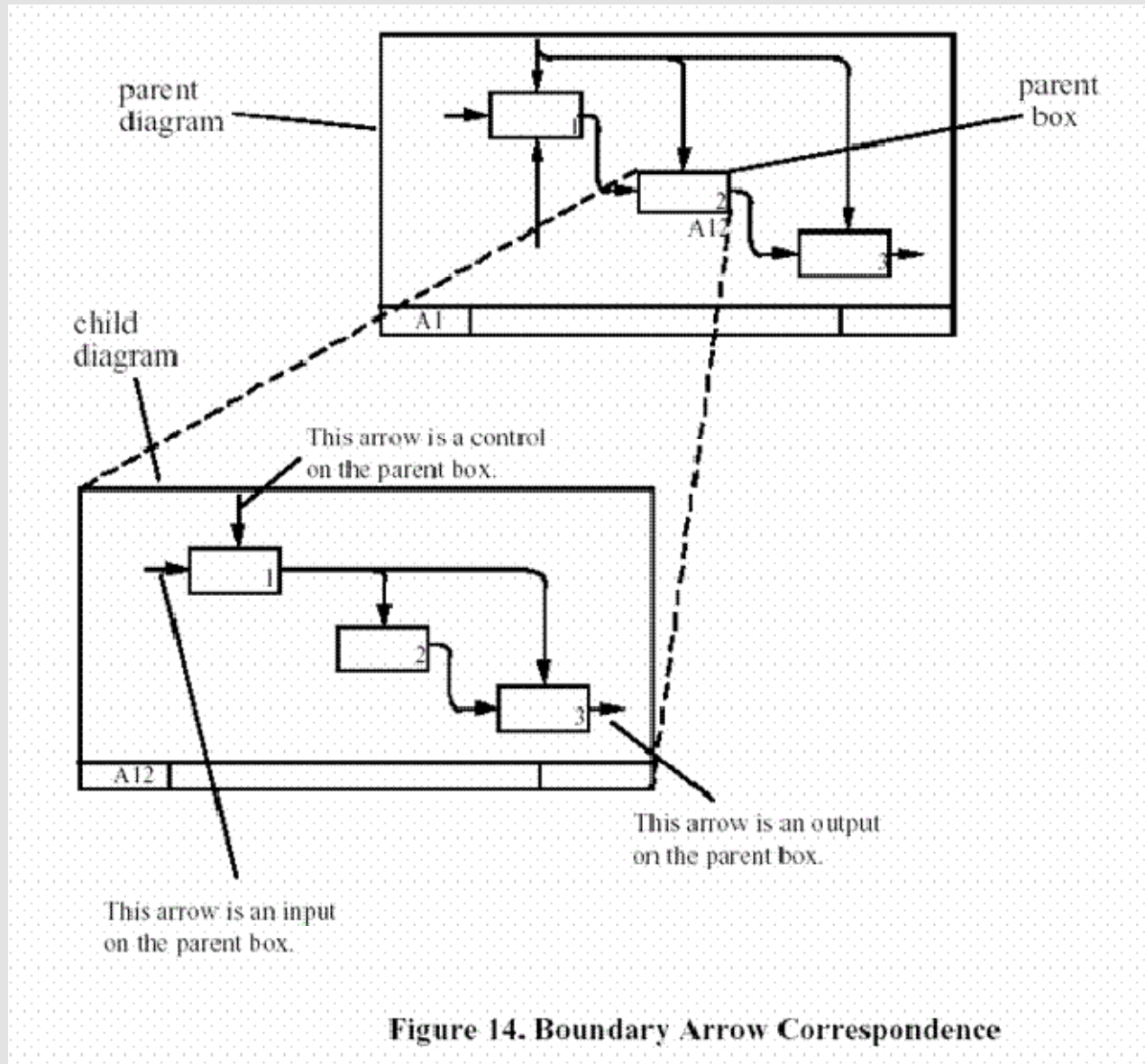
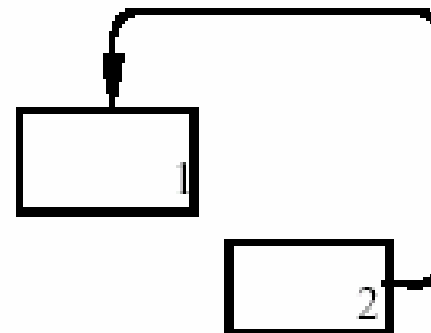


Figure 14. Boundary Arrow Correspondence

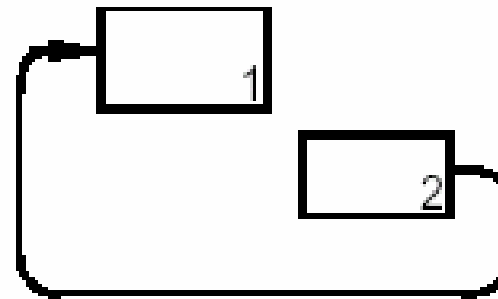
Control feedbacks

Control feedbacks shall be shown as “up and over”.



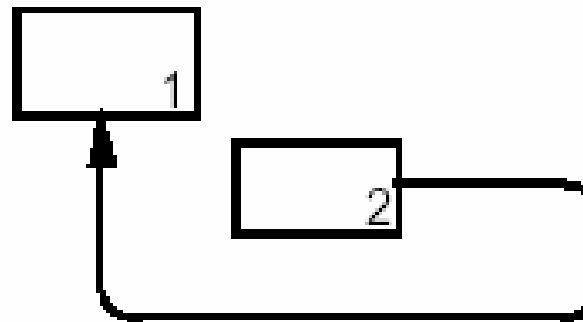
Input feedbacks

Input feedbacks shall be shown as “down and under”.



Mechanism feedbacks

Mechanism feedbacks shall be shown as “down and under”.



Components

Glossary ...

New Student: An employee of the company that has been directed, or volunteered, to participate in training

Instructor & Textbooks: The person responsible for teaching students and the documents, books, or other printed material used during the class

... documents the definition or characterization of one of the IDEF0 components of your effort. Each component in your model must have a glossary entry!

*What exactly do you mean by New Student?
It's defined in the glossary.*

Components

Text Elaboration ...

The input “personnel folder” is not an input to the A134 activity “Monitor Supply Consumption” because this group of information is not transformed in any way, or needed by these activities.

... is associated with a “diagram”. It describes the things that may not be apparent, but are necessary, to know to understand a diagram.

Why are these inputs only on boxes 1 and 2, but not 3?

Some Basic Rules

- ◆ Excluding the A-0 diagram, which has only one activity box, all other diagrams should have no less than three and no more than six Activity boxes.
- ◆ Each activity box must have at least one control and one output, but no more than six of each type of concept.
- ◆ Every diagram in a model must adhere to the model's overall viewpoint, purpose, and context.

Model Development

Development Process

- ◆ Establish and refine CV&P
- ◆ Collect information and artifacts
- ◆ Identify candidate functions
- ◆ Identify candidate objects
- ◆ Group functions into clusters and hierarchies
- ◆ Group objects into kind hierarchies
- ◆ Refine upwards and downwards
- ◆ Apply results
- ◆ Maintain



Validate

Establish and refine CV&P

- ◆ What are the boundaries?
 - ◆ Determine what is in and out.
 - ◆ Define the A-0.
- ◆ What is visible and what is not?
- ◆ What are the completion criteria?
 - ◆ What decisions need to be made?

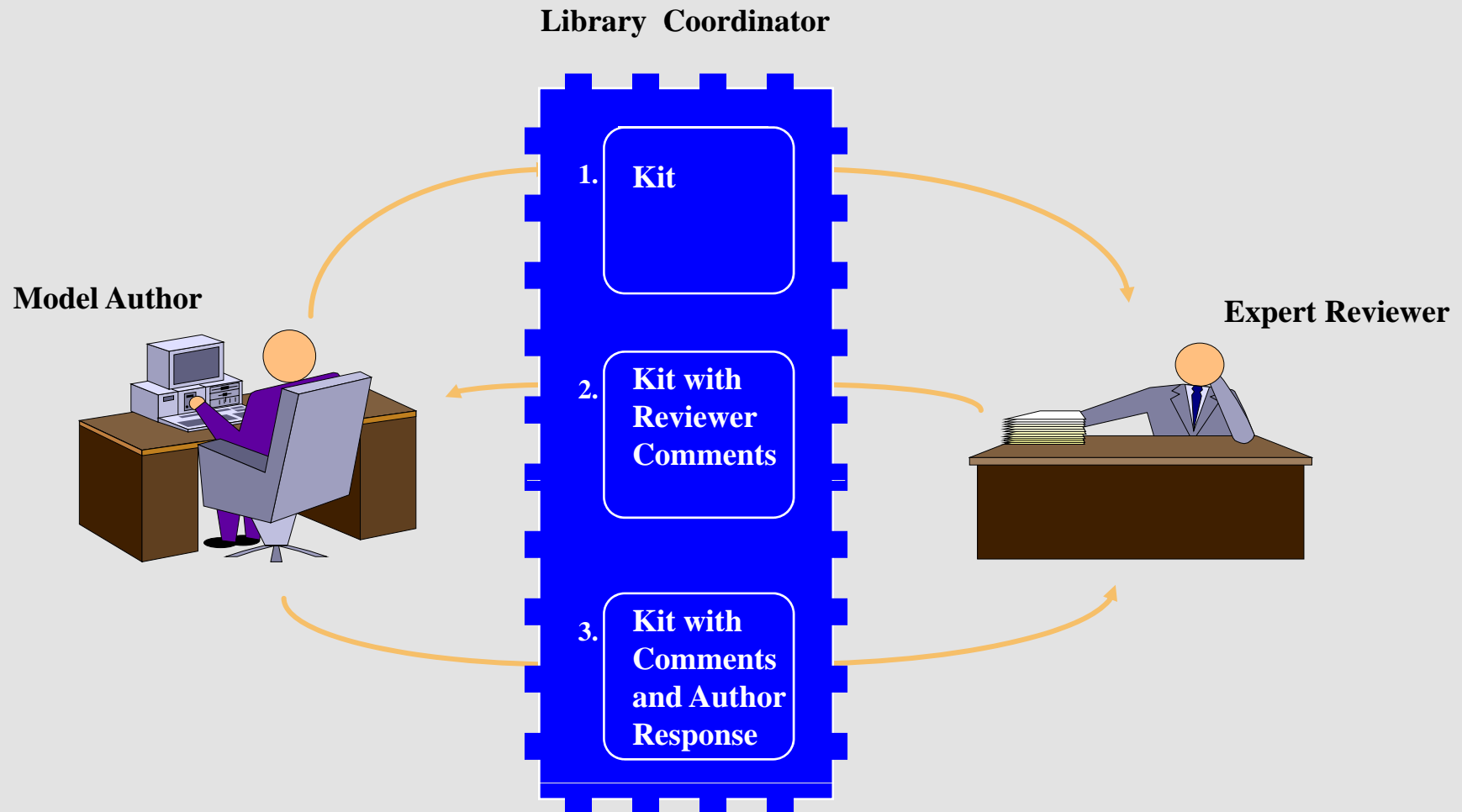
You won't get it right the first time.

You'll refine it during the course of doing the model.

Collect information and artifacts

- ◆ Identify sources and expert reviewers.
- ◆ Identify stake-holders.
- ◆ Interview.
 - ◆ Go through all levels in the organization.
 - ◆ Listen carefully.
 - ◆ Take detailed notes.
 - ◆ Collect as much as you can carry.
- ◆ Organize source material.
- ◆ Perform author-reader review cycle.

The Author-Reader Cycle



Identify Candidate Activities

- ◆ Pick out Decisions, Actions, Activities.
 - ◆ Behind each organization there must be an activity performed.
- ◆ Choose activity names carefully.
 - ◆ Use common semantics.
 - ◆ Remember the “we” rule.
 - ◆ Consider name coining an art rather than a science.
- ◆ Organize into lists.
 - ◆ By name similarity.
 - ◆ By common objects involved.
- ◆ Validate with reviewer cycle.

Identify Candidate Objects

- ◆ Pick out object references.
- ◆ Name coining key activity for many objects.
 - ◆ Definite descriptors need to be converted to names.
 - ◆ Nouns or noun phrases.
 - ◆ Be careful of state descriptors.
- ◆ Organize the lists.
 - ◆ By kind.
 - ◆ By part-of relations.
 - ◆ By subsumption relations.
- ◆ Validate with reviewer cycle.

Clusters and Hierarchies

- ◆ Collect activities into composition hierarchies.
- ◆ Collect activities together that work on the same objects.
- ◆ Avoid (where possible) type hierarchies.
- ◆ Name the group of activities (if necessary).
- ◆ Strive for at least 3 activities per group and not more than 6.
- ◆ Identify missing members of the group (where possible).

“Part-of” and “Kind” Hierarchies

- ◆ Solidify name references.
- ◆ Harmonize terminology.
- ◆ Simplify diagrams.
- ◆ Guide modeler in identification of missing activities.
- ◆ Construct new names for the super-kinds or compositions.
- ◆ Validate with experts.

Define Cells

- ◆ Associate objects with functions.
- ◆ Identify roles that objects play relative to a function.
 - ◆ Input
 - ◆ Output
 - ◆ Control
 - ◆ Mechanism
- ◆ Check object association on the next level of detail.
- ◆ Check object relevance on the same level.

Construct Diagrams

- ◆ Build what diagrams you can from the composition relationships.
- ◆ Look for inconsistent or incoherent or incomplete statements.
- ◆ Analyze for key missing relations.
- ◆ Complete the story as best able from source material.
- ◆ Validate with experts.

Refine Upwards and Downwards

- ◆ Arrange diagrams in hierarchy.
- ◆ Check consistency of interfaces.
- ◆ Is the boundary clearly defined?
 - ◆ Refine upwards.
- ◆ Do the leaf nodes contain information required to address the modeling purpose?
 - ◆ Refine downwards.
- ◆ Validate with experts.

Model Merge

- ◆ Combines separate models into a single overall model.
- ◆ Handles automatically duplicate model information according to user-specified criteria.
- ◆ Merges models from the same or different projects.
- ◆ Provides a valuable aid for group modeling projects.

Model Division

- ◆ Divides modeling tasks for a team modeling effort.
- ◆ Reduces excessively large models into sub-models of a more manageable scope.
- ◆ Allows for placing sub-model into same or different projects.

Apply Results

- ◆ Arrange periodic review with stakeholders and model users.
- ◆ Get sign-off from both groups.
- ◆ Document model application.
- ◆ Gather requirements for additional model definition.
- ◆ Gather requirements for model application.
 - ◆ Requirements definition
 - ◆ Data collection / organization
 - ◆ Training / Orientation

Maintain

- ◆ Shelf life of models is directly proportional to the use of the models.
- ◆ Models need to be maintained to continue to be useful.
- ◆ The reading of a model does not generally communicate all the understanding that the team acquired in the development of the model.
 - ◆ Members of the team giving model walk-throughs
 - ◆ Review of the source material and model evolution process

Successful Activity Modeling

Understanding IDEFØ Diagrams

- ◆ Reading is done top-down.
- ◆ Functions show what must be accomplished hence should be labeled with an active verb phrase.
- ◆ Arrows with one end unconnected indicate that the concepts are supplied, consumed, or used outside the scope of the diagram.

Understanding IDEFØ Diagrams

- ◆ Call arrows indicate a system that completely performs the function.
- ◆ Relationships: an output that is used as an I,C, or M by another function indicates that the latest function is dependent on the former but not how or when.
- ◆ Multiple ICOMS do not indicate conjunction; this is true as well for functions in a diagram.

Best Practice Guidelines

- ◆ Think control and constraint, not flow. The diagram structure must show relationships that hold regardless of sequence. Diagrams should say the right thing regardless of what steps are taken first.
- ◆ When a diagram is cluttered, it is often an indication that you put pieces of information that are at different levels of details.
- ◆ Leave out questionable concepts.

Best Practice Guidelines

- ◆ A solid abstraction is both clearer and more powerful than premature detail.
- ◆ A concept is a control unless it obviously serves as an input (is it modified?) If in doubt, make it a control.
- ◆ Input/output: what is done.
- ◆ Control: why.
- ◆ Mechanism: how.

Modeling Checklist

- ◆ In Facilitating Business Engineering, did it:
 - ◆ Define What Activities are Performed?
 - ◆ Define What is Needed to Perform those Activities?
 - ◆ Determine What the Current System Does Right?
 - ◆ Determine What the Current System Does Wrong?
 - ◆ Define Activity Interfaces (Objects & Data)?
 - ◆ Capture the Costs Related to the Activities?

Modeling Checklist

- ◆ In Facilitating Communication, did it:
 - ◆ Enhance Domain Expert Understanding?
 - ◆ Facilitate Consensus Decision-Making?
 - ◆ Promote Effective Team Activity?

Model Quality Checklist

- ◆ **Completeness**
 - ◆ Are all field-entries, labels, descriptions, purpose, and viewpoint present?
- ◆ **Conciseness**
 - ◆ Is the terminology used appropriate for the target audience?
 - ◆ Are some of the model elements redundant?
 - ◆ Are all elements clearly distinct from one another?

Model Quality Checklist

- ◆ Consistency
 - ◆ Is the terminology uniform throughout the model?
 - ◆ Are the model elements traceable to the system being modeled?
- ◆ Correctness
 - ◆ Is the model an accurate description of the system being modeled?
 - ◆ Are implied relations and constraints traceable to system constraints and relations?

Model Quality Checklist

- ◆ Complexity/Understandability
 - ◆ Is the model clear to the reviewer?
 - ◆ Is the information intended to be conveyed by the model accurately depicted via the syntax?

Conclusion

- ◆ IDEF0 documents “what” the studied system does.
- ◆ IDEF0 identifies where the unreasonable expenses are and where non-value added activities exist.
- ◆ IDEF0 concentrates upon functional dependencies, not organizational, sequential, or cause-effect relationships.

IDEF0 Activity Modeling

- ◆ Review & Questions