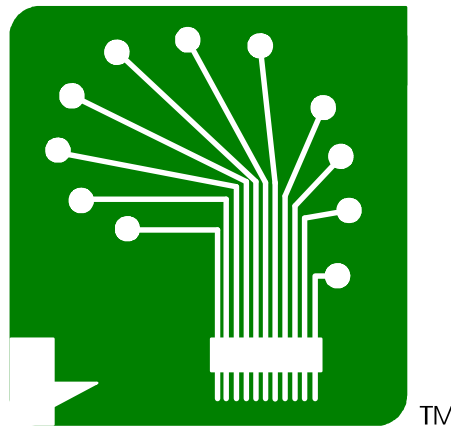


---

# Introduction to IDEF0 Modeling

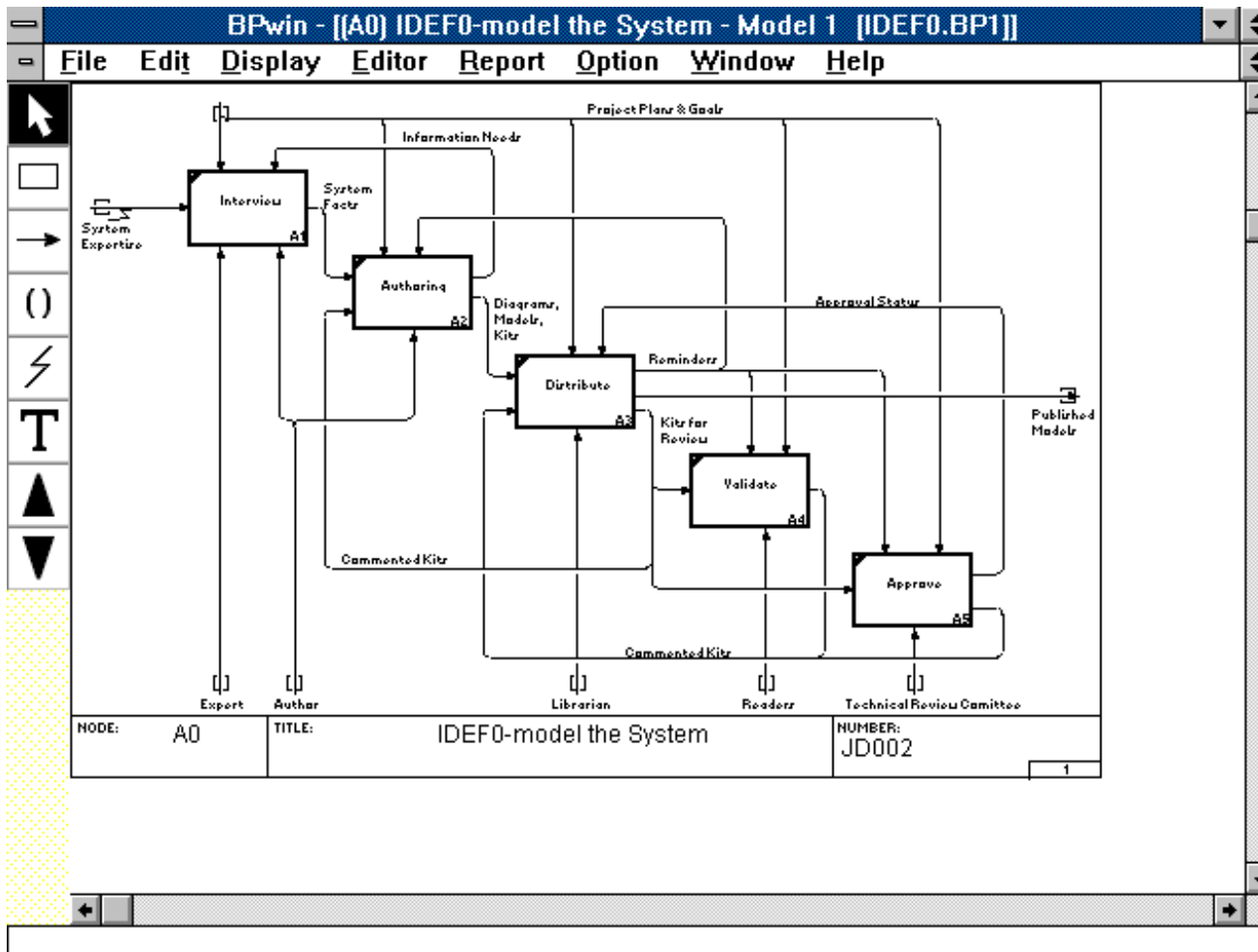
---



**Dr. James P. Davis**  
**Knowledge Based Silicon Corporation**  
**EDA&T Conference - August 1995**

---

# Overview of IDEF0 Methodology

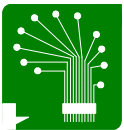


### Activities:

- Interview business experts
- Author the IDEF0 model
- Distribute model for review
- Validate model against business
- Approve the model

### Roles:

- Domain experts
- Business analyst (model "author")
- Librarian
- Model "readers" (peer/team reviewers)
- Technical Review Committee (QA)



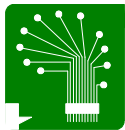
---

---

# Enterprise Modeling Using IDEF0

---

- Create description of enterprise for the purpose of gaining understanding, and of being able to answer questions about the enterprise.
- Used to describe enterprise and its environment prior to, or in conjunction with, defining requirements.
- Used to precisely define boundaries of system (i.e., what is in and out of scope for the project under consideration).
- Model the enterprise from a particular "viewpoint", or perspective, so as to keep the activity focused on the goal of effort and on pertinent characteristics of interest in enterprise.
- Create a description of the enterprise with a single subject, single purpose (exemplified by questions to be answered about the enterprise), and single viewpoint.
- Note that, during Project "scoping" activity, the viewpoint is most likely that of looking at the enterprise from the standpoint of the client-server application to be deployed in the enterprise.



---

# Overview of IDEF0 Notation

---

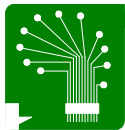
1. The IDEF0 notation consists of BOXes, representing Activities in the system or enterprise, and ARROWs, describing relationships between activities.

2. Characteristics of Activity Boxes:

- represent key activities of the enterprise, from the selected viewpoint
- are named using "active" verbs or verb phrases
- are decomposed into groups of lower-level activities
- grouped by no more than 6 activities to a single level in model
- numbered in sequence in a given level, according to "dominance":
  - influence one activity has over others in level
  - dominance based on (1) first in sequence, or (2) most influential

3. Characteristics of Arrows:

- represent collections of things ("artifacts")
- named using nouns or noun phrases
- connect activities together, and connect them to interfaces
- arrows clasified into following categories:
  - inputs (I), outputs (O), controls (C), mechanisms (M)



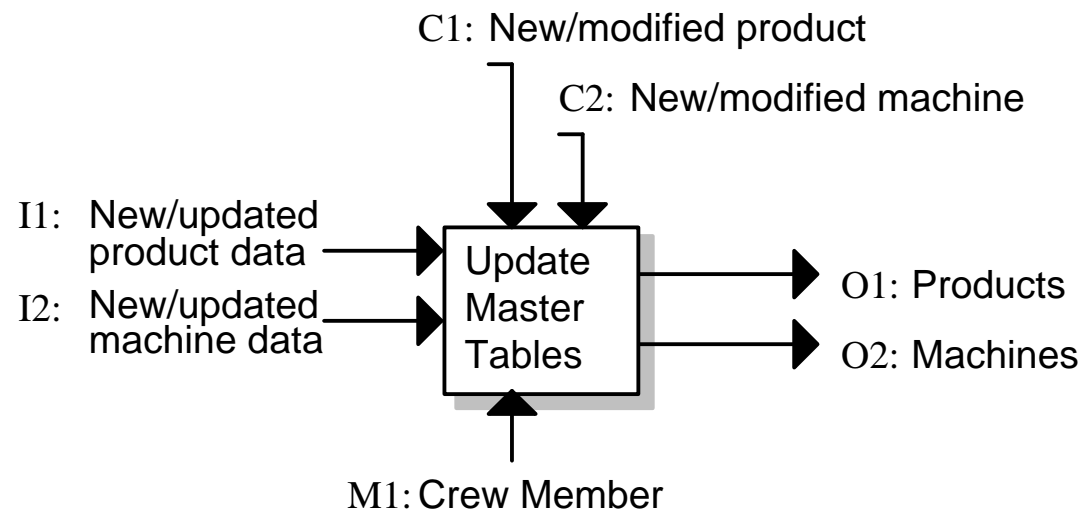
---

# Overview of IDEF0 Notation

---

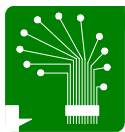
**Inputs:** transformed by Activity into outputs

**Controls:** dictate & constrain under what conditions transformations occur

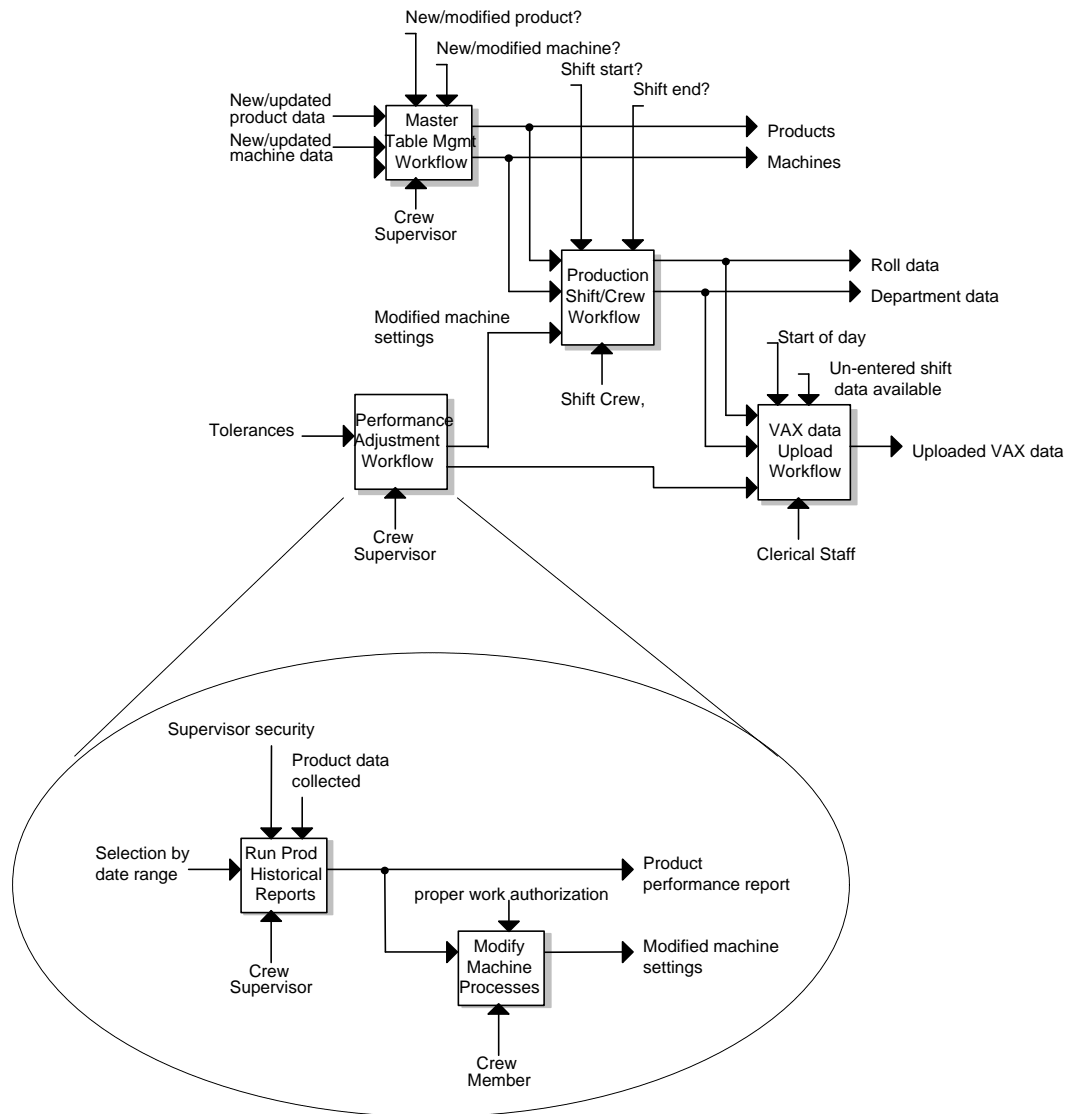


**Mechanisms:** describe how the activity is to be carried out (resource, process, etc.)

**Outputs:** created as a result of activity transformations.

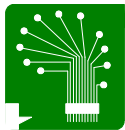


# Characteristics of IDEF0 Activities

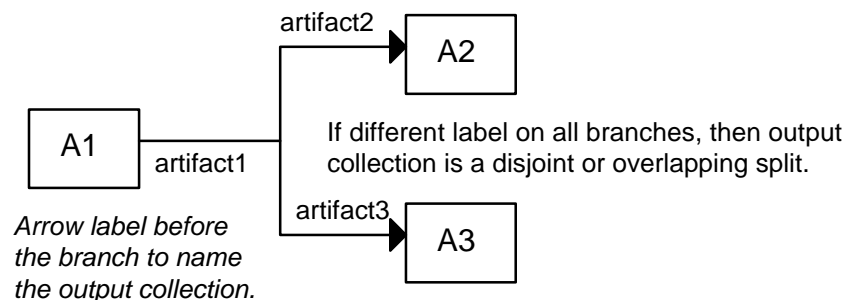
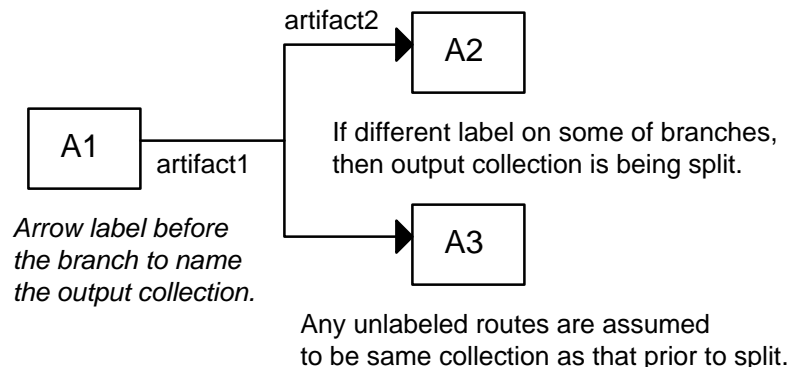
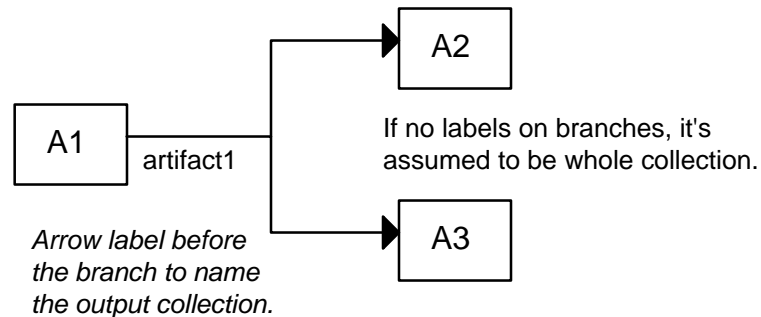


**Decomposition:** Taking an activity and diving into it in order to define its internal structure and organization into related sub-activities. The "parent" is decomposed into "children" activities and interactions.

**Sequencing:** Grouping activities at a given level, using arrows to tie them together, forming the activity sequences giving detail of the parent activity. Activities precede and follow one another.



# Characteristics of IDEF0 Arrows - Fan-out & Split

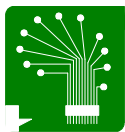


**Arrow flows:** collection of entities or artifacts, serving as inputs, outputs, controls and mechanisms for activities.

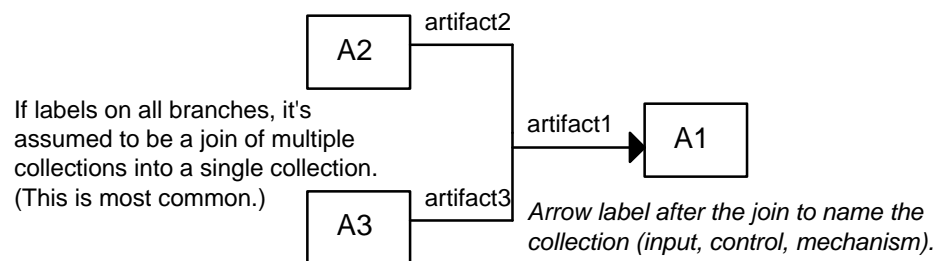
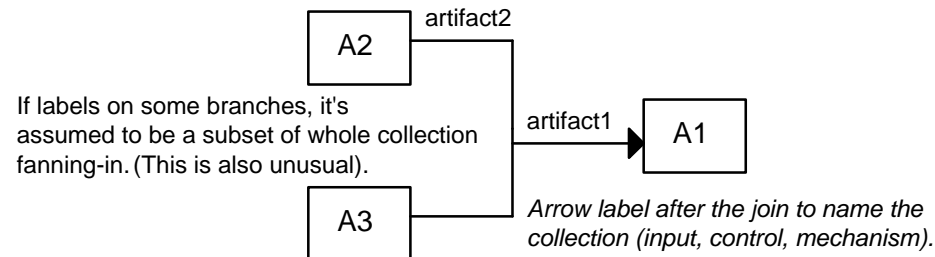
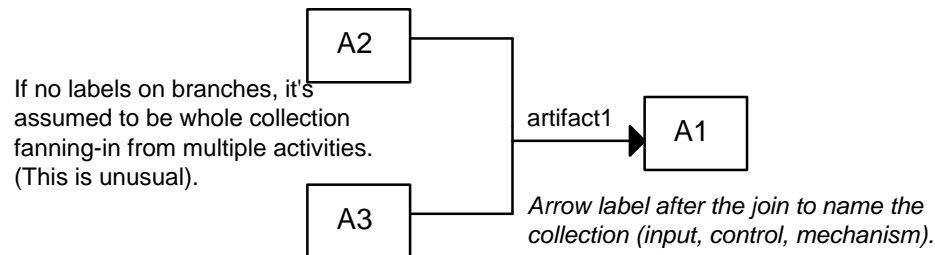
**Fan-out:** an output collection is routed to multiple inputs of other activities.

**Split:** an output collection is divided up in some manner, directing some of output collection of artifacts to a different destination. The labeled artifacts are a subset of total entering the branch.

Disjoint or overlapping split collections: When all branches are labeled, it implies split into disjoint or overlapping subsets from the original collection.

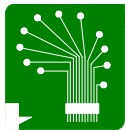


# Characteristics of IDEF0 Arrows - Fan-in & Join



**Fan-in:** collections of things are flowing into a single collection, to be used as input, control or mechanism. The collection is the same on both sides of the joining arrow junction.

**Join:** an output collection is aggregated up in some manner, directing a combined collection of artifacts from multiple sources to a single destination. The labeled arrows are a subset of total resulting from the join.

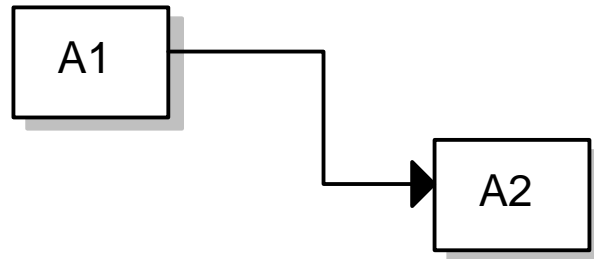




---

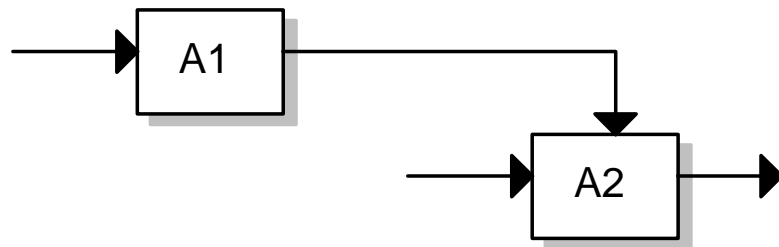
## Characteristics of IDEF0 Arrows - Interconnect

---

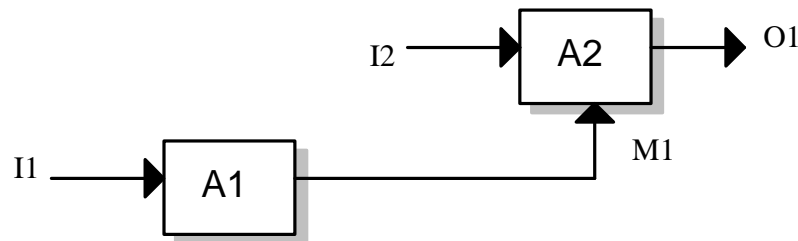


There are different relationships between activities implied by interconnect patterns.

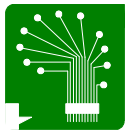
**Output to input:** output of one activity becomes input to box of lesser dominance.



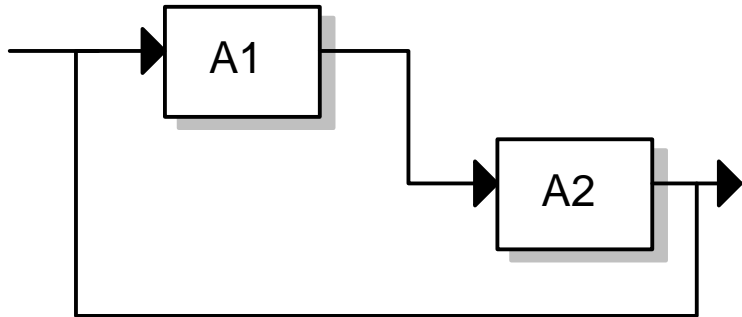
**Output to control:** output of one box directly constrains a box of lesser dominance. Implies tight coupling of activities.



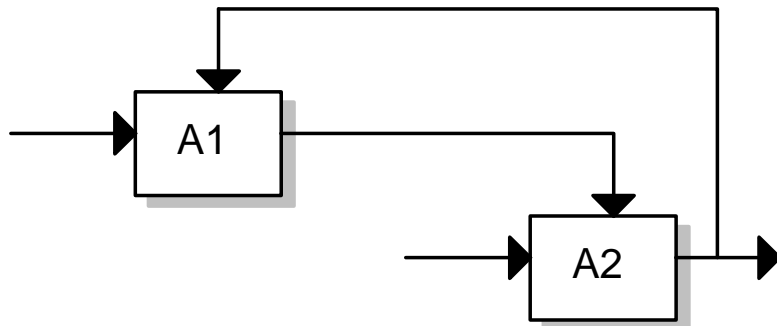
**Output to mechanism:** output of one activity becomes means by which another activity gets done. Implies a strong sequencing relationship.



# Characteristics of IDEF0 Arrows - Feedback Loops

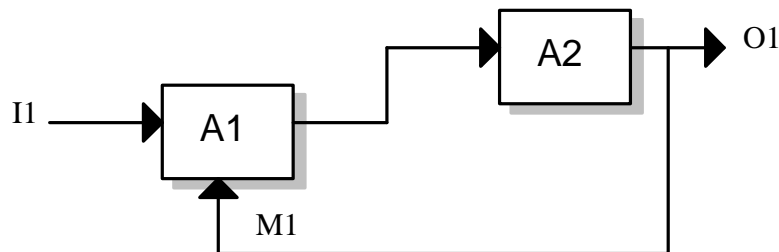


In addition to feed-forward relationships (outputs affect activities of lesser dominance), there are feedbacks into the model affecting activities of greater dominance.

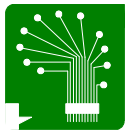


**Feedback to input:** output of one activity is input to box of greater dominance. Implies iteration, error correction, rework.

**Feedback to control:** output of one box directly constrains a box of greater dominance.



**Feedback to mechanism:** output of one activity becomes means by which more dominant activity gets done. Only occurs in tandem with iteration of feedback to input.



---

---

# Characteristics of IDEF0 Diagrams

---

1. **Context Diagram:** defined the top-level of enterprise being modeled using IDEF0. It delineates the boundaries of the portion of the enterprise under consideration, and is defined for the viewpoint.
2. **C-numbers:** chronological numbering scheme for diagrams, and includes *author's initials + diagram version number* (e.g., JD001). This is method for version control of individual IDEF0 diagrams.
3. **Node numbers:** means for identifying and tracking individual activities in the model. Provides a means for associating activity boxes in a parent diagram with the diagram and activity components of children (e.g., EPR/A12, indicates EPR project, activity 1, sub-activity 2 of the decomposed top-level diagram).
4. **ICOM codes:** Input, Control, Output, Mechanism number scheme, so as to track these relationships through the levels of hierarchy, as they cross the boundary from current level off the page.
5. **Tunneling:** In order to reduce the amount of detail that either gets pushed down or carried up in the decomposition hierarchy, you can have arrows in the diagram that aren't associated with arrows in next level above or below.

