AD NUMBER
ADB062459
LIMITATION CHANGES
TO: **Approved for public release; distribution is unlimited.**
FROM: **Distribution authorized to U.S. Gov't. agencies only; Test and Evaluation; 15 JUN 1981. Other requests shall be referred to Air Force Wright Aeronautical Laboratories, Wright-Patterson AFB, OH 45433.**
AUTHORITY
AFWAL ltr, 21 Jan 1983

AFWAL-TR-81-4023
VOLUME VI

INTEGRATED COMPUTER-AIDED MANUFACTURING (ICAM) ARCHITECTURE PART II

VOLUME VI - DYNAMICS MODELING MANUAL ($IDEF_2$)

SofTech, Inc.
460 Totten Pond Road
Waltham, MA 02154

June 1981

FOR EARLY DOMESTIC DISSEMINATION

Because of its significant early commercial potential, this information, which has been developed under a U. S. Government program is being disseminated within the United States in advance of general publication. This information may be duplicated and used by the recipient with the expressed limitations that it not be published nor released to foreign parties without appropriate licenses. Release of this information to other domestic parties by the recipient shall be made subject to these limitations. This legend shall be marked on any reproduction of this data in whole or in part. These limitations shall be considered void after two years after the release date on the data.

SUBJECT TO EXPORT CONTROL LAWS

This document contains information for manufacturing or using munitions of war. Export of the information contained herein, or release to foreign nationals within the United States, without first obtaining an export license, is a violation of the International Traffic-in-Arms Regulations. Such violation is subject to a penalty of up to 2 years imprisonment and a fine of $100,000 under 22 USC 2778.

Include this notice with any reproduced portion of this document.

DISTRIBUTION STATEMENT

Distribution limited to United States Government agencies only; Test and Evaluation Data; Statement applied June 15, 1981. Other requests for this document must be referred to AFWAL/MLTC, WPAFB, OH 45433.

MATERIALS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

NOTICES

Note that this document bears the label "FEDD", an acronym for "FOR EARLY DOMESTIC DISSEMINATION". The FEDD label is affixed to documents that may contain information having high commercial potential.

The FEDD concept was developed as a result of the desire to maintain U.S. leadership in world trade markets and encourage a favorable balance of trade. Since the availability of tax supported U. S. technology to foreign business interests could represent an unearned benefit, research results that may have high commercial potential are being distributed to U. S. industry in advance of general release.

The recipient of this report must treat the information it contains according to the conditions of the FEDD label on the front cover.

This technical report has been reviewed and is approved for publication.

Steven R. Le Clair

STEVEN R. LE CLAIR, Captain, USAF
Project Manager
Computer Integrated Mfg Branch

N G Tupper

NATHAN G. TUPPER
Chief
Computer Integrated Mfg Branch

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization, please notify AFWAL/MLTC, Wright-Patterson AFB, Ohio 45433 to help us maintain a current mailing list."

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER AFWAL-TR-81-4023, Volume VI	2. GOVT ACCESSION NO. AD-B062	3. RECIPIENT'S CATALOG NUMBER 456L
4. TITLE (and Subtitle) INTEGRATED COMPUTER AIDED MANUFACTURING (ICAM) ARCHITECTURE PART II Volume VI - Dynamics Modeling Manual ($IDEF_2$)		5. TYPE OF REPORT & PERIOD COVERED Final - 29 September 1978 through 10 May 1981 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s)		8. CONTRACT OR GRANT NUMBER(s) F33615-78-C-5158
9. PERFORMING ORGANIZATION NAME AND ADDRESS SofTech, Inc. 460 Totten Pond Road Waltham, MA 02154		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 78011F
11. CONTROLLING OFFICE NAME AND ADDRESS Materials Laboratory (AFWAL/MLTC) AF Wright Aeronautical Laboratories (AFSC) Wright-Patterson AFB, OH 45433		12. REPORT DATE June 1981 13. NUMBER OF PAGES 195
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution limited to United States Government agencies only; Test and Evaluation Data; Statement applied 15 June 1981. Other requests for this document must be referred to AFWAL/MLTC, WPAFB, OH 45433.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Manufacturing Architecture
Model of Manufacturing
Integrated Computer Aided Manufacturing
Subsystem Integration
ICAM
IDEF
$AUTOIDEF_0$
MFG0
DESIGN0
MFG1
SMC2
MCMM0/MFG0
SMC0/MFG0

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The overall objective of the Integrated Computer Aided Manufacturing (ICAM) Architecture Part II was to utilize and expand the manufacturing architecture. Included in this objective were the support and direction of future ICAM projects, logical design of manufacturing systems, validation and verification of ICAM modules, integration of ICAM modules and the orderly transition of ICAM modules into ICAM systems. This volume, Volume VI, is the manual given to students learning the $IDEF_2$ Dynamics Modeling Methodology for describing the time varying behavior of functions and information.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

The report is presented in the following eleven volumes:

1. Volume I - Architecture Part II Accomplishments
2. Volume II - Architecture - A Structured Approach to Manufacturing
3. Volume III - Integration Using Architecture
4. Volume IV - Function Modeling Manual ($IDEF_0$)
5. Volume V - Information Modeling Manual ($IDEF_1$)
6. Volume VI - Dynamics Modeling Manual ($IDEF_2$)
7. Volume VII - Composite Function Model of "Manufacture Product" (MFG0)
8. Volume VIII - Composite Function Model of "Design Product" (DESIGN0)
9. Volume IX - Composite Information Model of "Manufacture Product" (MFG1)
   Part 1 - MFG Development
   Part 2 - MFG1 Model
10. Volume X - Dynamics Model of a Sheet Metal Center Subsystem (SMC2)
11. Volume XI - ICAM Library Maintenance and Distribution Procedures

FOREWORD

This report was prepared by SofTech, Inc., Waltham, Massachusetts under USAF Contract F33615-78-C-5158. This is the final report describing the Part II work performed on the: ICAM Architecture of Manufacturing; Information Modeling; Subsystem Integration; Tools Development; User Interface Requirements; and the Architecture of Design. This work was performed during the period of 29 September 1978 through 10 May 1981 and was initiated under the direction of the ICAM Program Manager, Mr. Dennis E. Wisnosky, and sponsored by the Manufacturing Technology Division, Materials Laboratory, Air Force Wright Aeronautical Laboratories at Wright-Patterson AFB, Ohio. The Air Force Project Managers for this project were: Mr. Richard Mayer through 30 June 1979 and Captain Steven R. LeClair through completion. The prime contractor for the project was SofTech, Inc. The Project Manager for SofTech was Mr. Reuben Jones. Primary Coalition Team Companies participating on this project were: Rockwell International, Vought Corporation, Hughes Aircraft Company, Dan Appleton Company, Northrop Corporation, Boeing Computer Services, Boeing Commercial Airplane Company, Pritsker & Associates, Higher Order Software, and Control Data Corporation.

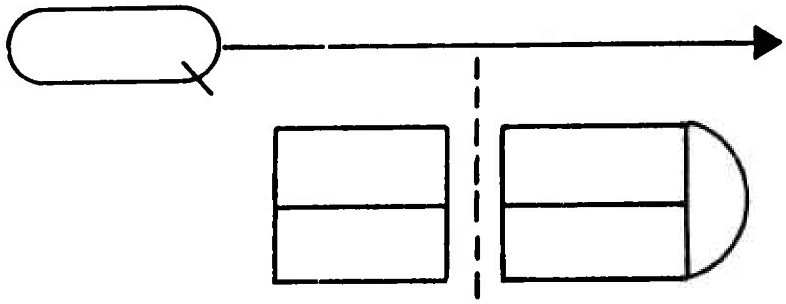
PREFACE

The fundamental concepts of $IDEF_2$ are based on computer modeling and simulation techniques. $IDEF_2$ provides a vehicle for describing the elements of manufacturing systems whose behavior varies over time. When complete, $IDEF_2$ will contain an analysis capability to portray the time varying status of the elements and the integrated performance associated with the manufacturing system. The methodology is based on network modeling concepts; facility models; and graphical modeling, analysis and display procedures. The understanding of the dynamic aspects of a system has led to increased productivity of manufacturing systems.

The major contributors to the development of $IDEF_2$ methodology are:

Robin J. Miner	Pritsker & Associates	Developed the syntax and semantics associated with $IDEF_2$
A. Alan B. Pritsker	Pritsker & Associates	Developed the syntax and semantics associated with $IDEF_2$
John F. Ippolito	Pritsker & Associates	Developed the syntax and semantics associated with $IDEF_2$
Rick Mayer	AFWAL/MLTC	Contributed his insight concerning Air Force needs and requirements
Charles Patrick	SofTech, Inc.	Reviewed the proposed methodology and made suggestions regarding its improvement.

TABLE OF CONTEXT



SECTION 1

INTRODUCTION

INTRODUCTION

The U. S. Air Force Program for Integrated Computer Aided Manufacturing (ICAM) is directed toward increasing manufacturing productivity through the systematic application of computer technology. The ICAM Program approach is to develop structured methods for applying computer tecnnology to manufacturing and to use those methods to better understand how best to improve manufacturing productivity.

The ICAM Program identified a need to better communicate and analyze manufacturing for the people involved in improving productivity. To satisfy that need, the ICAM Program developed the IDEF (ICAM Definition) method to address particular characteristics of manufacturing. IDEF is comprised of three modeling methodologies which graphically characterize manufacturing:

$IDEF_0$ is used to produce a function model which is a structured representation of the functions of a manufacturing system of environment, and of the information and objects which interrelate those functions.

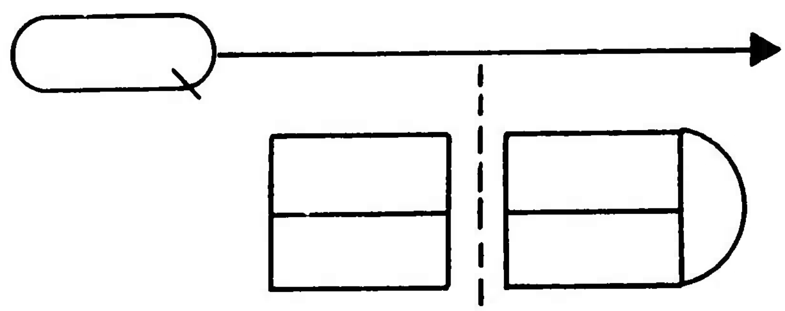
$IDEF_1$ is used to produce an information model which represents the structure of information needed to support the functions of a manufacturing system or environment.

$IDEF_2$ is used to produce a dynamics model which represents the time varying behavior of functions, information and resources of a manufacturing system of environment.

Each of the three models individually or any group of models can form an "ARCHITECTURE" when the environment of system being modeled is comprised of component systems, organizations and/or technologies which must work together to accomplish the overall objective (production) of the manufacturing environment or system. The significance of the models being referred to as "ARCHITECTURES" is that they are blueprints or frameworks which define graphically the fundamental relationships - the functional interfaces, identification of common, shared and discrete information, and dynamic interaction of resources. It is important to recognize that the IDEF models become Architectures when used to better understand, communicate and analyze not only the subject environment or system (manufacturing) but how its constituent components (system, organizations and technologies) fit together.

IDEF is a method, Architecture is a means and improving manufacturing productivity is the end which the ICAM Program is pursuing within the U. S. aerospace industry.

The following material is a discussion of the fundamental concepts, techniques and procedures regarding the use of $IDEF_2$ to produce a dynamics model.



SECTION 2

$IDEF_2$ CONCEPTS

$IDEF_2$ CONCEPTS

2.1 Background

The problems facing manufacturers continue to grow in size and complexity. Tools and equipment are more sophisticated, often involving a computer to perform their functions. As a result, personnel with specialized training are required to operate the equipment, and complicated manufacturing procedures must be defined. In addition, natural resources are less available, and manufacturing is constrained by the limited supply and expense of raw materials. Increases in technology and decreases in natural resource availability are only two causes for increased complexity in manufacturing. There are many others, all of which result in new, large, and complex problems which have created a need for new procedures and techniques for understanding and solving manufacturing problems. The ICAM Definition (IDEF) methods are intended to meet this need.

When modeling systems, the definition of the system is relative to the purpose for constructing the model, the objectives or goals of the modeling effort, and the particular modeler's viewpoint. Generally, a system can be defined as a set of interrelated objects which perform a specific function. In a given situation a particular set of interrelated objects may be only a small part of a larger system, that is, a subsystem. In another situation that same collection of interrelated objects may be the primary focus of interest and would be considered as the system. To help alleviate this confusion, a modeler usually specifies a modeling purpose that is based on the problem to be solved. In addition the objectives of the modeling effort are stated in terms of the particular problem being addressed. Then the boundaries of the system and level of modeling detail are established based on the purpose and objectives. A schematic of this approach to modeling is shown in Figure 2-1.

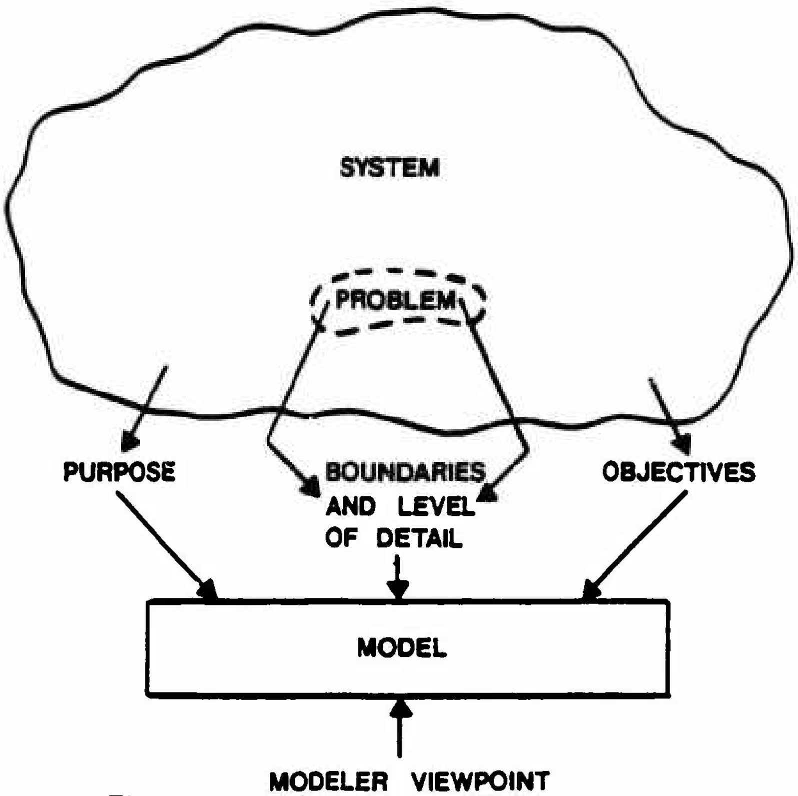


Figure 2-1. Approach to Model Building

$IDEF_2$ is the ICAM dynamics modeling methodology. It is a methodology which has been designed to allow one to describe the time-varying behavior of manufacturing systems in such a way that the descriptions can be analyzed using computer simulation to generate measures of manufacturing system performance. System models which generate measures of system performance can be used to make better decisions regarding real systems.

It is important when building an $IDEF_2$ model to clearly and explicitly define the problem, modeling objectives, modeling purpose and the modeler's viewpoint. To develop a good model it is essential that the modeler also understand the structure and operating rules of the system in order to extract the essence of the system without including unnecessary detail. The amount of detail included in a model should be based on the modeling purpose and objectives and only elements which could cause differences in decision making should be considered. The definition of level of modeling detail involves the definition of model elements, the specification of modeling assumptions, and the definition of model element interactions.

An $IDEF_2$ model can be employed in five ways depending on the modeling purpose:

- As an explanatory device to define a system or problem
- As a communications vehicle to determine system elements, components or issues
- As a documentation medium
- As a design assessor to synthesize and evaluate proposed solutions to problems
- As a predictor to forecast and aid in planning future developments

The first three of these uses of $IDEF_2$ models emphasize their use as descriptive tools, while the second two emphasize their use as analysis tools. As a descriptive tool, an $IDEF_2$ model is used to identify the components of systems which affect or cause a system's behavior to change over time. A clear and complete understanding of the behavior of an existing system is essential in developing good solutions to existing problems. In addition, a complete description of new systems early in their design will help insure a successful new system. $IDEF_2$ provides a structure and syntax which will aid in acquiring an accurate understanding of system elements and their operation.

The last two uses of $IDEF_2$ emphasize the use of $IDEF_2$ models as analysis tools. Once an $IDEF_2$ model of a system is built, one can experiment with the model via simulation to draw inferences about the real system. There are several advantages to this type of modeling. First, a proposed system design may be evaluated in terms of its expected performance without actually building the system. Second, one may experiment with existing systems without disturbing the systems, incurring unnecessary costs or creating unsafe conditions. Thirdly, a systems limit or capacity may be determined without destroying the system. Thus modeling can be used for design, procedural analysis, and performance assessment.

Manufacturing systems can be viewed and modeled in many different ways. $IDEF_0$ views a system as the set of functions it performs. Alternatively, $IDEF_1$ views a system by studying information it contains. $IDEF_2$ on the other hand views the time-varying behavior of a system. These three views of systems are not contradictory but are complimentary. Each view provides a modeler with a tool to focus on a particular aspect of a system according to his needs and modeling purpose.

In $IDEF_2$ a modeler has the capability to describe the elements of manufacturing systems which vary over time or cause other elements to vary over time. In order to characterize the time-varying behavior of systems it is first important to describe those elements which change over time.

Some of these elements are represented by $IDEF_0$ and $IDEF_1$, although their behavior over time is not. Consider the functions performed by a system which are described by $IDEF_0$. While $IDEF_0$ focuses on the static aspects of these functions, they also have dynamic aspects. For example, each activation of a particular function takes a specific amount of time to complete. In addition, if all inputs, controls, and mechanisms required for an activation are not present then the function would have to wait until they become present. Thus, while $IDEF_0$ does not describe it, there is time-varying behavior associated with the functions performed by a system. $IDEF_2$ can represent this time-varying behavior.

Also consider the information required to support a system which is characterized by $IDEF_1$. While $IDEF_1$ focuses on the static aspects of information, there can be variation over time of the information itself or in the relationships between various types of information. These dynamic aspects of information are not represented in $IDEF_1$, but can be characterized using $IDEF_2$.

In addition to the components of $IDEF_0$ or $IDEF_1$ models which may vary over time, there are many other dynamic or time-varying aspects to manufacturing systems which $IDEF_2$ allows one to characterize. The functions represented in $IDEF_0$ consist of activities, queues, and decisions which all affect the dynamic behavior of manufacturing systems. Each instance of an activity has time-varying behavior associated with it because it begins at a particular time, has a time duration, and completes at a particular time. Queues contain parts/jobs, information or other types of entities which arrive to and depart from queues at particular instants in time. There are also many types of decisions made based on available information which affect activities being performed and queues which hold entities to be processed.

The ability of a manufacturing system to perform effectively depends to a large extent on the interactions between system components and on external controlling influences. The performance of the real manufacturing system is measured in terms of its effectiveness and efficiency in converting system inputs into system outputs. The same performance measures (with the possible exception of output quality which we are unable to measure with a model) apply to a model of a manufacturing system. A manufacturing decision maker is concerned with efficiency and effectiveness as measured by the quantity of throughput, the degree to which deadlines are met, the utilization of resources, and the quantities of in-process inventory maintained. Figure 2-2 shows typical performance measures in each of these categories. The $IDEF_2$ modeling methodology was designed so that $IDEF_2$ models could be analyzed to quantify these and other problem specific measures of manufacturing system performance.

2.2 Objectives

The overall objective of the $IDEF_2$ modeling methodology is to provide a method which allows one to describe the time-varying behavior of manufacturing systems in such a way that the description can be analyzed via computer simulation to get measures of manufacturing system performance. Thus there are two functions which are performed by an $IDEF_2$ model. First, it allows a modeler or designer to create a graphic description of the system he is considering. This graphic description provides a communications vehicle which can be used in the model development to describe and document a system. In addition the completed

A. Measures of Throughput
   1. The number of entities produced/processed by the total system in a specified period of time
   2. The quantity of entities produced/processed by the total system per unit time (production rate)
   3. The number of entities produced/processed by a particular subsystem in a specified period of time
   4. The quantity of entities produced/processed by a particular subsystem per unit time
   5. The time between departures of entities from the total system or a particular subsystem

B. Measures of Ability to Meet Deadlines
   1. Time to produce/process a specified number of entities (makespan)
   2. Time in the system for entities (flowtime)
   3. Entity lateness
   4. Entity tardiness

C. Measures of Resource Utilization
   1. Fraction of time a particular resource is busy, idle, inoperative, or blocked
   2. The number or proportion of resources which are busy, idle, inoperative, or blocked

D. Measures of In-Process Inventory
   1. The number of entities waiting for a particular resource
   2. The number of entities waiting for any resource
   3. The number of entities which balk (spillover) from a given storage area

Figure 2-2. Categorization of Typical Manufacturing System Performance Measures

model will serve as a means of communicating a system definition to management. The second function performed by an $IDEF_2$ model is to represent a system in an analyzable form. That is, the $IDEF_2$ model is an analyzable representation of the time-varying behavior of systems which provides a means of measuring the dynamic performance of the system.

As part of the overall objective, $IDEF_2$ is to represent a wide class of manufacturing systems. The methodology was designed to allow a modeler to represent the dynamic aspects of: hardware systems such as production lines or group technology cells; software systems such as part program generation for computer controlled robots; or communications systems and procedural systems such as design processes or shop order release systems. In addition, $IDEF_2$ allows the representation of these systems at any level of detail desired. High level models may aggregately represent activities and decisions as a single activity while models at a lower level of detail will explicitly represent each activity and decision.

In addition to allowing one to represent many types of manufacturing systems, $IDEF_2$ was designed to allow one to solve various types of manufacturing problems including scheduling, capacity planning, inventory management, raw material requirements planning, quantity control program assessment and capital investment analysis.

As an example of the use of $IDEF_2$ to address these types of problems consider scheduling. Scheduling is the process of determining the starting time of jobs and on which machines the jobs are to be performed. The scheduling process involves the following stages: aggregate planning, loading, sequencing, and detailed scheduling.

Aggregate planning is the activity of determining the overall level of output for a given time period and the level of resources to be deployed during the time period of interest. For example, a production manager's aggregate plan may call for producing 1000 units for the month of July using 10 machines and 20 workers. The aggregate plan does not specify starting times, the order of production, the machines to be used, or the workers to be employed. The representation of part flow which includes resource allocation allows one to model aggregate planning in $IDEF_2$.

Loading is the action of allocating work to machine areas, and it establishes the work load each machine center will carry under a given aggregate plan. A typical loading specification is "50 jobs of type 1 will be performed in work area B." Loading does not involve specification of the order in which jobs are to be performed. Loading would be modeled in $IDEF_2$ through branching operations which route jobs to machine centers (servers and resources) based on the achievement of the loading objectives.

Sequencing or "work dispatching" is used to establish priorities for processing jobs at work centers. The term job sequencing refers to the determination of which job is to be performed next when a machine becomes available. Queue ranking rules and queue selection procedures are the means for modeling job sequencing in $IDEF_2$. The term machine sequencing is used for the process of determining which machine at a given work center is to be used next if more than one machine is idle. Server selection procedures and Resource Disposition Trees are the means for modeling machine sequencing in $IDEF_2$.

Identification of start and finish dates for jobs at a work center is the final stage of scheduling and is called detailed scheduling. Detailed scheduling utilizes the determinations made during aggregate planning, loading, and sequencing and is often a direct consequence of these activities. For detailed scheduling, mathematical programming or other such techniques may be required. If a job has a schedule date for completion, it is maintained as an attribute of the entity representing the job. The Entity Flow Network and the Resource Disposition Tree employ this attribute in the decision processes for the job. In this way, the scheduling procedures inherent in developing the due dates are followed. The decision processes are modeled directly using the $IDEF_2$ syntax.

Clearly, there are many different ways to schedule a production system. The success of any scheduling process depends on whether it enables the production system to perform well with respect to the performance measures: throughput, ability to meet due dates, resource utilization, and levels of in-process inventory. $IDEF_2$ will aid in the decision making process by making these quantities available to the decision maker.

The $IDEF_2$ methodology fulfills several needs within the life cycle of system development described by the ICAM System Development Methodology. $IDEF_2$ can be used in the requirements definition step, the preliminary design step, and the detailed design step of system development. Within the requirements definition step, $IDEF_2$ provides a framework and syntax in which a modeler can describe the time varying aspects of the AS-IS manufacturing system. The process of describing the time-varying behavior of the AS-IS system is an important step in the development of a TO-BE system because it forces system designers to detail the operational elements of the AS-IS system. Thus, the task of developing an $IDEF_2$ model will force designers to consider the details of the current system before considering any new structures for the TO-BE system. An AS-IS $IDEF_2$ model can then be validated via simulation to insure that the $IDEF_2$ model performs as the real system does. This validation will insure that the system designers have an accurate perception of problems associated with the AS-IS system before attempting to alleviate them in the TO-BE system design.

In addition to its applicability in the requirements definition step, $IDEF_2$ will also be useful in the preliminary design step of system development. Within the preliminary design step $IDEF_2$ models of design alternatives are built to evaluate the effectiveness of the TO-BE system high level design alternatives. These $IDEF_2$ models aid in determining and quantifying system design alternatives. These $IDEF_2$ models aid in determining and quantifying system design problems early in the development of the systems. Potential problems which can be identified include inadequate throughput, processing bottlenecks and high in-process inventories. In addition, performance/cost tradeoffs and capability/cost tradeoffs can be made. In this manner $IDEF_2$ can be used to make tradeoffs with respect to cost, time, and technology. After building $IDEF_2$ models of various design alternatives and performing tradeoff studies, the modeler or designer will choose the design which appears most promising. Thus, the selection of the "best" overall design and the initial performance testing of this"best" design could also be performed with an $IDEF_2$ model.

Within the detailed design step of system development, the $IDEF_2$ model of the best high level design developed in the preliminary design step is expanded to include the details developed in the detailed design step. The modeler can use the $IDEF_2$ model to evaluate alternate detailed system designs including new algorithms or operating rules which will govern the TO-BE system. The detailed $IDEF_2$ model allows the modeler or designer to determine the details of system operation as well as the cost of system operation. Therefore, in the preliminary design phase the components of the system and their costs are identified while the operating rules and associated costs are determined in the detailed design phase. Thus, using $IDEF_2$, a system designer will have some assurance by the end of the detailed design step that the TO-BE system will indeed perform as desired and meet the needs of its users.

2.3 Approach to $IDEF_2$ Modeling

A manufacturing system produces products and information based on orders and requests subject to rules, procedures and the availability of raw materials, facilities, material handling equipment, manpower, and machines. To describe a manufacturing system in $IDEF_2$, the manufacturing system is decomposed into four submodels as shown in Figure 2-3. An $IDEF_2$ model consists of a Facility Submodel, an Entity Flow Submodel, a Resource Disposition Submodel, and a System Control Submodel.

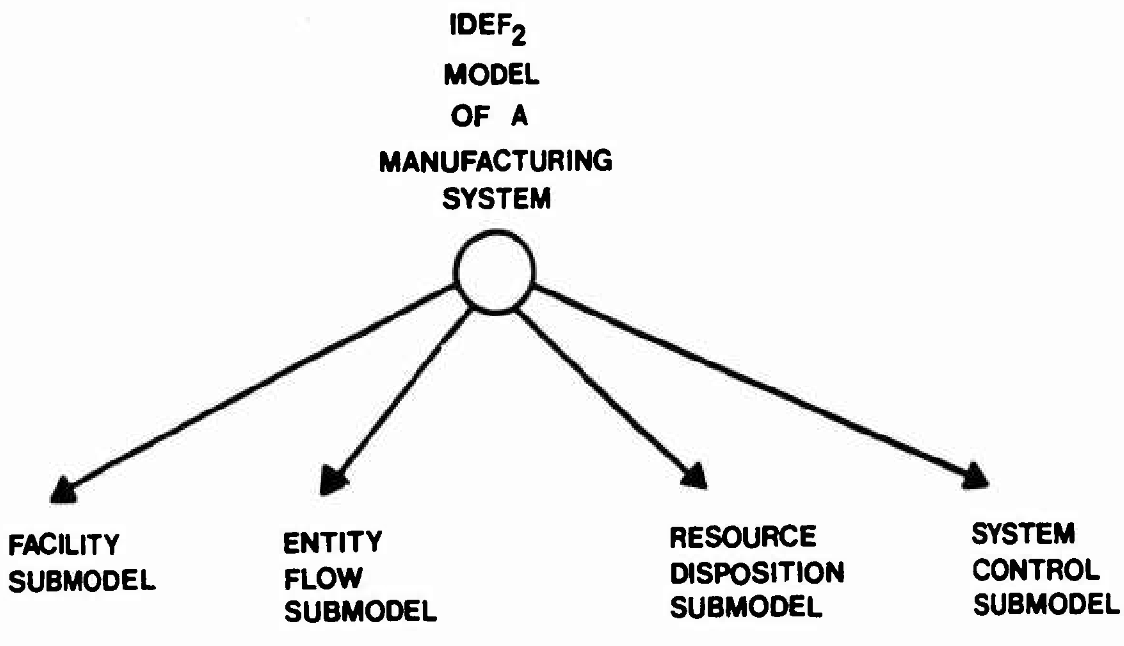


Figure 2-3. $IDEF_2$ Manufacturing System Decomposition

The $IDEF_2$ Facility Submodel describes the resources which are used by the system to produce the final products or information. These resources may be physical, logical or cognitive. Physical resources are any physical components of systems such as people, machines and materials. Logical resources are any logical components of systems which determine their operation, such as computer software or manufacturing procedures. Cognitive resources are those resources which are required for thought processes such as experience and creativity.

The $IDEF_2$ Entity Flow Submodel is the means by which the flow of products or information through facilities is described. An entity may be real or conceptual. Examples of entities are products or information which are operated on or produced by the system being modeled. It is in the Entity Flow Submodel that the actual processing of products or information is described. The activities which are performed as well as the decisions regarding alternate flow of entities are also described.

The $IDEF_2$ Resource Disposition Submodel is used to describe the disposition of resources when they become available. A resource in $IDEF_2$ is any part of the system which must be present to perform an activity. The $IDEF_2$ modeler uses tree structures to ask questions concerning the status of the system and to specify what actions should be taken with respect to the available resource.

The fourth $IDEF_2$ submodel is the System Control Submodel which describes the occurrences of activities which control but do not prescribe the flow of entities. Situations handled by System Control Submodels include the breakdown and repair of resources, the arrival of entities, the alteration of resource capacities, the initiation and termination of shifts and the alteration of job priorities.

The approach to building $IDEF_2$ models is to build the four sub-models which when combined describe the time-varying behavior of a system. This approach is different than the integrated modeling approach which is typically used when modeling systems. However, there are several advantages to this approach. First, by breaking a large system down into smaller components attention can be focused on a few details at a time. In this manner, large projects can be divided among several modelers or sequential modeling can be performed. Second, the data collection for large systems will be simplified because procedures can be developed which organize and structure the data in the same way the submodels are structured. Third, the difficult statistical and modeling concepts are segregated from the description of entity flow. Modelers are not required to use difficult statistical and modeling concepts. A fourth advantage to the decomposition approach is that certain submodels could potentially be used with different types of analysis mechanisms. For example, the Facility Submodel could potentially be used for plant layout analysis as well as for simulation analysis. A fifth advantage to the decomposition approach is that certain submodels may be standardized and used in several $IDEF_2$ models without requiring that they be redeveloped. This decomposition approach to dynamics modeling results in structured data collection and modeling efforts.

2.4 Disciplined Teamwork

The creation of a model is carried out by authors. It is a dynamic process which requires the participation of more than one person. Throughout a project, the draft versions of the submodels are distributed to one or more other project members for review and comment. The discipline requires that each person expected to make comments about a submodel makes them in writing and submits them to the author of the submodel. This cycle continues until the submodels, and eventually the entire $IDEF_2$ model, are formally approved. During the process, incorrect or unacceptable analysis and design results are usually spotted early, and oversights and errors are detected before they can cause major disruptions.

The IDEF modeling process includes procedures which retain written records of all decisions and alternate approaches as they unfold during the project. Initial diagrams are created by an author and then critiqued by knowledgeable commenters. Commenters document their suggestions directly onto copies of the author's models. Authors respond to each comment in writing on the same copy. Suggestions are accepted or rejected in writing along with the reasoning used. As changes and corrections are made, outdated versions of diagrams are retained in the project files - nothing is thrown away.

The end effect of this process for organized teamwork is a high assurance that final $IDEF_2$ models are valid and are well expressed. The diagrams are changed to reflect corrections and valid comments. More detail is added and more diagrams are created. More comments are made. More changes are included. The final model represents the agreement of the authors and commenters on a representation of the system being modeled from a given viewpoint and for a given purpose. The viewpoint states the source of information and who will use the model, thereby indicating which points will be emphasized and what terminology will be employed; the purpose indicates the underlying rationale for constructing the model. Together they set the framework of the model for both authors and commenters. The continuous awareness of the model's purpose and viewpoint will guide the model during construction toward an internally consistent, usable final form.

2.5 Basic Concepts

The building of $IDEF_2$ models will follow a slightly different procedure than that used for building $IDEF_0$ and $IDEF_1$ models because the goal of building an $IDEF_2$ model is to analyze as well as describe the dynamic behavior of a manufacturing system.

An $IDEF_0$ model of the use of $IDEF_2$ to DESCRIBE AND ANALYZE MANUFACTURING SYSTEM DYNAMICS is shown in Figures 2-4 through 2-6. The A-0 diagram is shown in Figure 2-4. The input to the function DESCRIBE AND ANALYZE MANUFACTURING SYSTEM DYNAMICS is SYSTEM INFORMATION. This includes all the data required to structure, support, and validate an $IDEF_2$ model. The things which control the DESCRIBE AND ANALYZE activity are the $IDEF_2$ METHODOLOGY, MODELING

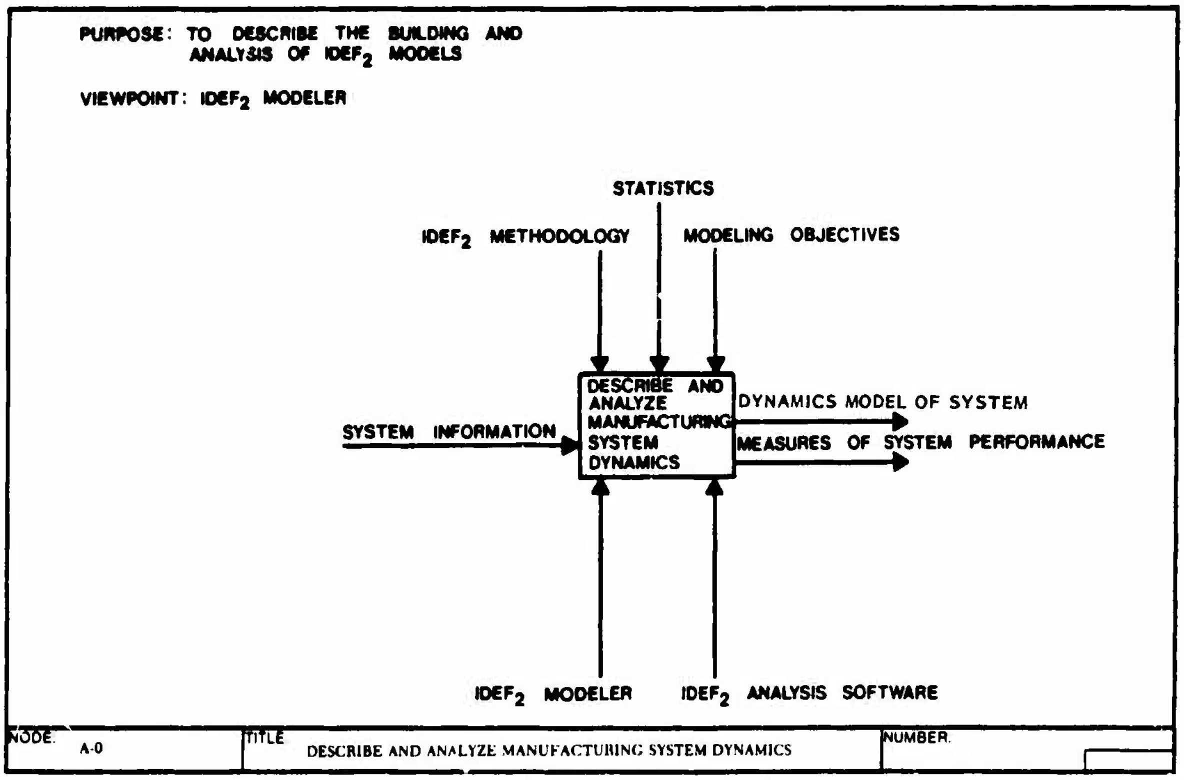


Figure 2-4. Describe and Analyze Manufacturing System Dynamics

OBJECTIVES, and STATISTICS. The $IDEF_2$ METHODOLOGY and the MODELING OBJECTIVES will be used to build the $IDEF_2$ MODEL of the system. The MODELING OBJECTIVES and STATISTICS control the analysis of the $IDEF_2$ model. The mechanisms used to perform this function are the $IDEF_2$ MODELER and the $IDEF_2$ ANALYSIS SOFTWARE. The outputs of the DESCRIBE AND ANALYZE function are the DYNAMICS MODEL OF SYSTEM and MEASURES OF SYSTEM PERFORMANCE. The MEASURES OF SYSTEM PERFORMANCE will be the result of analyzing the $IDEF_2$ model.

Figure 2-5 contains the A0 diagram of the function DESCRIBE AND ANALYZE MANUFACTURING SYSTEM DYNAMICS. This function breaks down into four subfunctions at this level: BUILD DYNAMICS MODEL, VALIDATE MODEL, DESIGN ANALYSIS EXPERIMENT, and PERFORM ANALYSIS OF DYNAMICS MODEL. All of the inputs, controls,

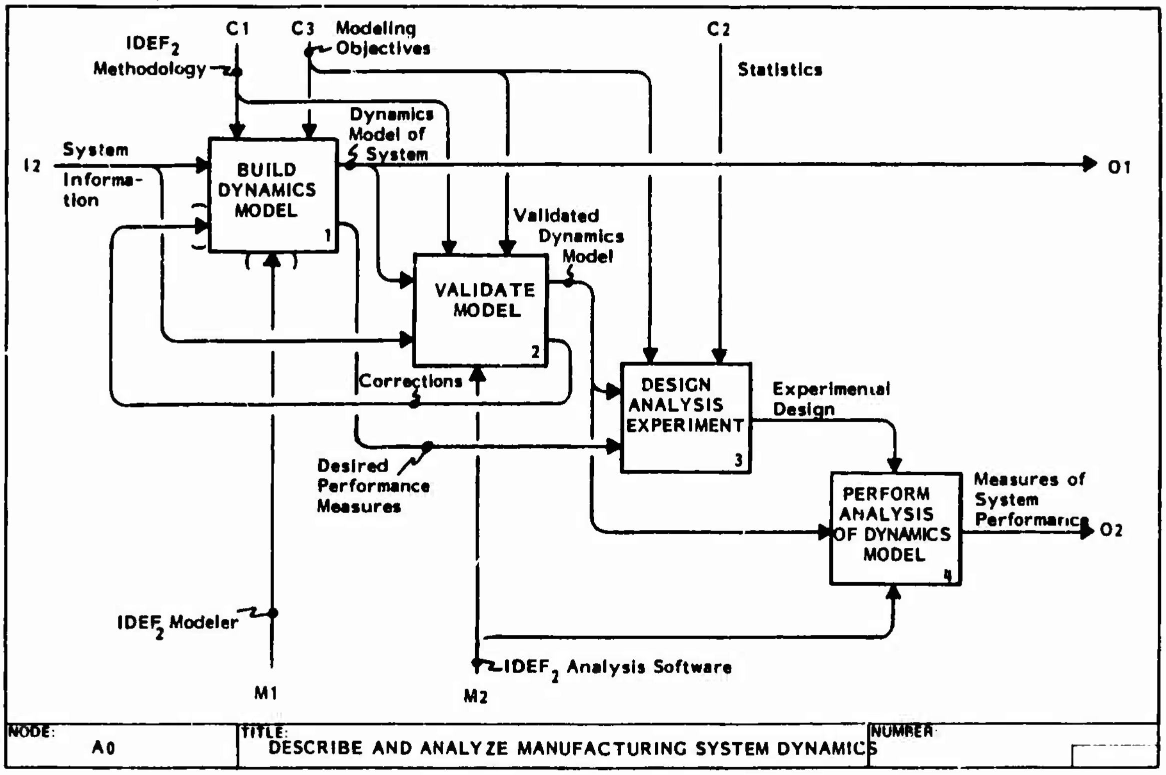


Figure 2-5. Describe and Analyze Manufacturing System Dynamics

mechanisms and outputs from the A-0 diagram are shown in this diagram as they affect the subfunctions of the function DESCRIBE AND ANALYZE MANUFACTURING SYSTEM DYNAMICS.

Figure 2-6 illustrates the subfunctions of the function BUILD DYNAMICS MODEL. The first step in building a DYNAMICS model is to DETERMINE DESIRED PERFORMANCE MEASURES. That is, determine what measures of performance are used to judge the ability of the real system to perform its job. These performance measures then control the determination of system boundaries. When modeling a system with $IDEF_2$, all details concerning a manufacturing system are not represented. Only those which could affect the dynamic behavior of the system are included. Once the system boundaries are defined, they are used with the desired performance measures to determine the DATA REQUIREMENTS for the DYNAMICS model. These DATA REQUIREMENTS control the COLLECT REQUIRED DATA function whose input is the raw SYSTEM INFORMATION which includes other IDEF models, if available, and

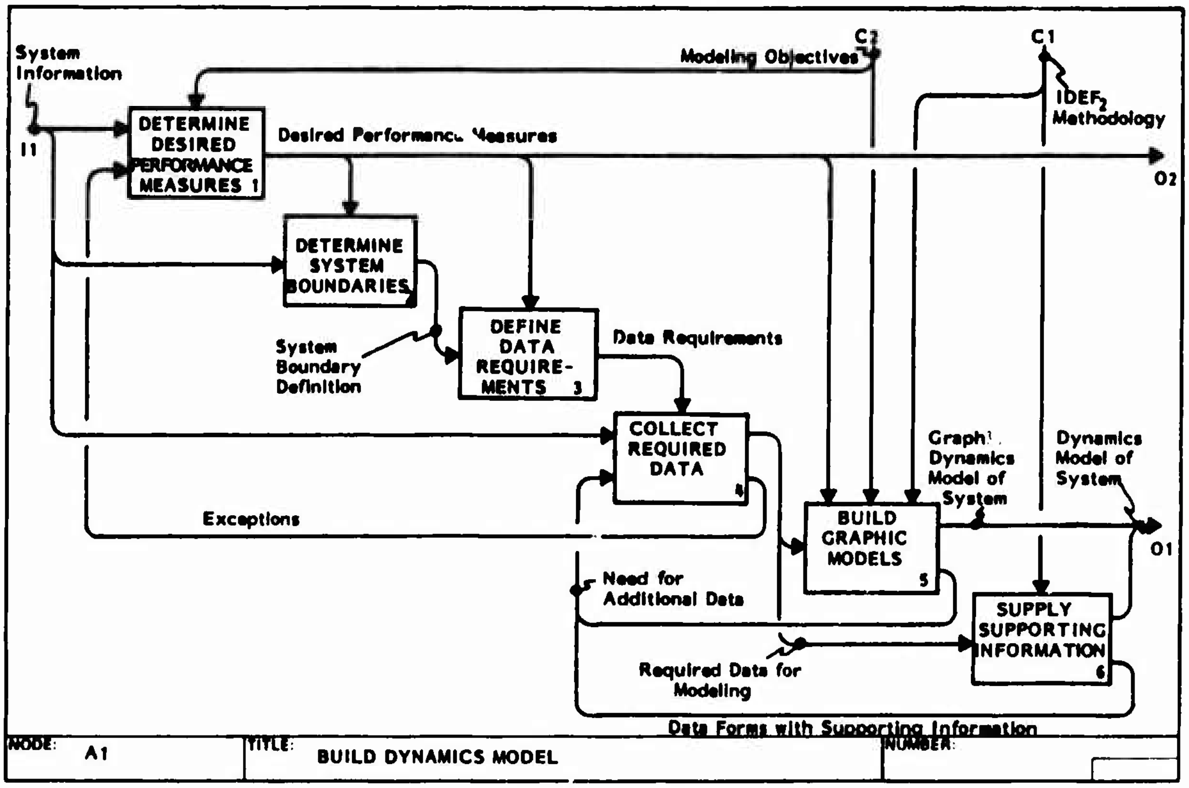


Figure 2-6. Build $IDEF_2$ Model

whose output is the data required for $IDEF_2$ modeling. Next the MODELING OBJECTIVES, DESIRED PERFORMANCE MEASURES, $IDEF_2$ METHODOLOGY, and REQUIRED DATA for modeling are used to build the graphic portions of the DYNAMICS submodels. If additional data is required to build the graphic portions of the submodels then additional data must be collected. The REQUIRED DATA for modeling and the $IDEF_2$ METHODOLOGY are used by the function SUPPLY SUPPORTING INFORMATION to produce the DATA FORMS WITH SUPPORTING INFORMATION which join with the GRAPHIC DYNAMICS MODEL to produce the complete DYNAMICS MODEL. If all required data is not available to SUPPLY SUPPORTING INFORMATION then additional data may need to be collected.

Figure 2-7 contains an $IDEF_1$ diagram which describes the information characteristics of a DYNAMICS model. In building a DYNAMICS model the MODELING PURPOSE is used to determine the MODEL OBJECTS as well as to determine the PERFORMANCE MEASURES. PERFORMANCE MEASURES are also determined by the MODELER'S VIEWPOINT. The MODEL BOUNDARIES and MODELING OBJECTS generate DATA REQUIREMENTS which are satisfied by SYSTEM DATA. The SYSTEM DATA is then used to build the DYNAMICS MODEL which is scoped by the MODEL BOUNDARIES.

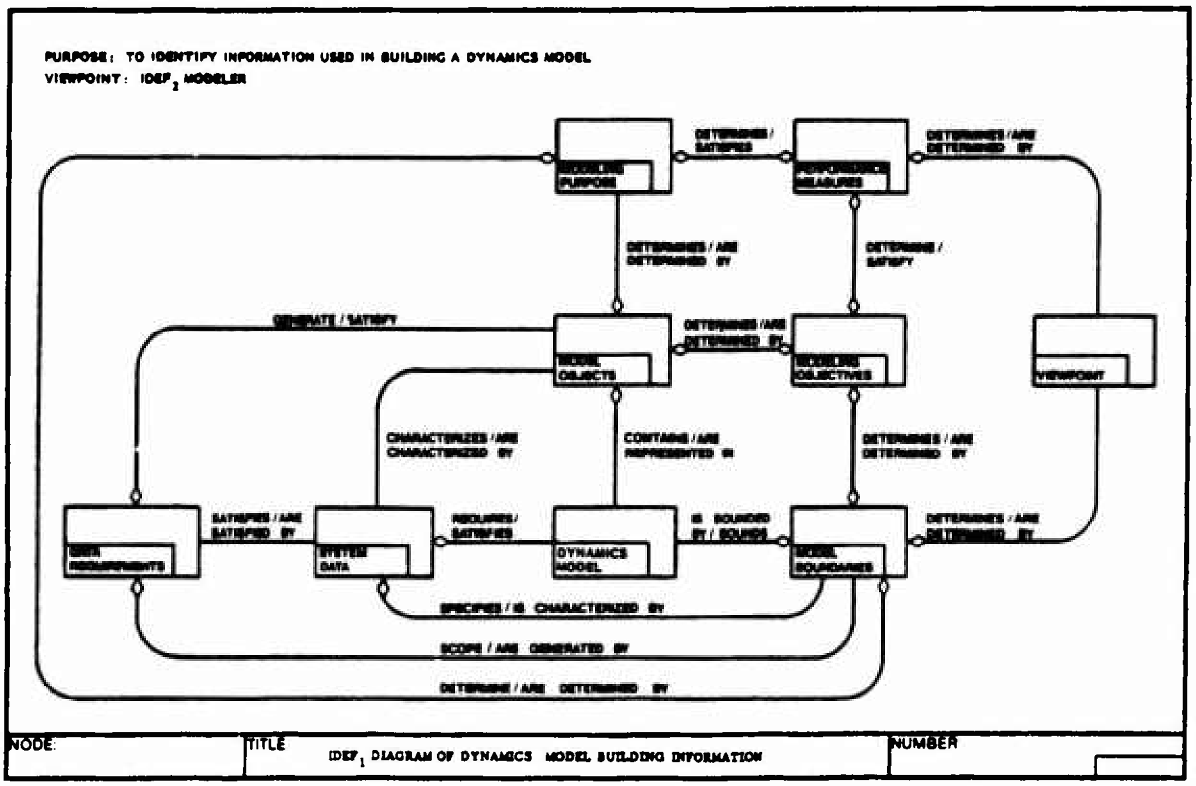
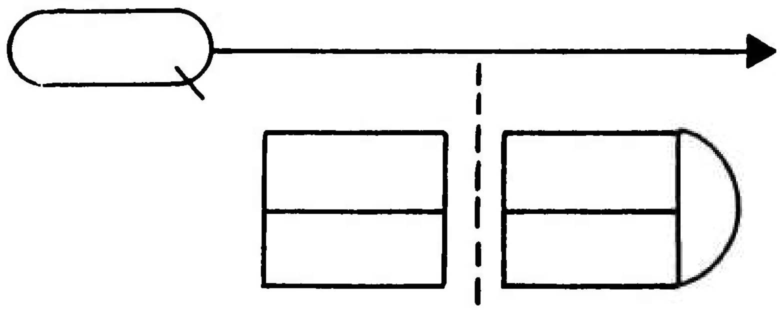


Figure 2-7. IDEF1 Diagram of Dynamics Model Building Information



SECTION 3

UNDERSTANDING IDEF$_2$ DIAGRAMS

UNDERSTANDING $IDEF_2$ DIAGRAMS

Before a discussion of the conventions of the $IDEF_2$ modeling methodology, it is appropriate to first review the basic concepts behind the modeling technique. When building an $IDEF_2$ model, the modeling purpose and objectives, along with the modeler's viewpoint aid in defining the boundaries and components of the system to be modeled. The system to be modeled is then decomposed into a Facility Submodel, an Entity Flow Submodel, a Resource Disposition Submodel and a System Control Submodel.

3.1 System Breakdown

Each of the submodels within an $IDEF_2$ model contains a graphical component and supporting documentation contained on forms. The graphical components of these submodels each have a symbol set designed to facilitate their construction in a straightforward and understandable manner. The following paragraphs describe the respective symbologies for the Entity Flow Network, Resource Disposition Trees, System Control Networks, and Facility Diagrams, the graphic components of the four $IDEF_2$ submodels. The section is concluded with a discussion of the supporting documentation required to support each of these graphical tools.

3.1.1 Entity Flow Network

An Entity Flow Network is a graphical network representation of all possible paths an entity can take in flowing through a system along with the resources that are required for passage through the system. Each entity flow path represents a sequence of activities. Arrows are used to graphically portray activities. Nodes are used to separate activities and represent the decision elements, queues, and milestones associated with the modeling of entity movement through a system.

An entity is something that forms an important part of the real world being modeled. Entities may be either real or conceptual. Examples of real entities are parts, jobs, and materials. Examples of conceptual entities are ideas, requests, and pieces of information. Entities have characteristics that are referred to as attributes. Example attributes of the entity "part" are its number, class, lot number, and priority. The concept of entities within $IDEF_2$ is consistent with the $IDEF_1$ definition.

$IDEF_2$ models system behavior by examining the manner in which entities flow through the system, and the reaction of the system to entity flow. For example, a part is said to flow along a production line as operations are carried out on it. If the production line is the system being modeled, the parts produced can be seen as entities flowing through the system. In $IDEF_2$, activities in which entities engage are represented with arrows. The arrows are separated by nodes.

Each entity flows over the network and each may have a different route as branching from nodes can occur probabilistically or conditionally. Also, rules (procedures) are specified to select from among competing elements such as parallel queues and servers, alternative resources, and subordinate activities. If two different entity types require exactly the same processing they might have identical Entity Flow Submodels. However, if their flow through the system depends on the entity type, a separate Entity Flow Submodel may be required for each entity type.

Symbology of Entity Flow Networks

As an introduction to understanding Entity Flow Networks, let us consider a simple production system in which parts arrive, wait to be processed, are operated on by a single resource (a drill), and depart the system. A schematic of such a system is shown in Figure 3-1. In this system, parts arrive requiring a drilling operation. If the drill is free when a part arrives, it is processed immediately. If the drill is inoperative or occupied with another part, the arriving part must await service in a queue. As soon as a part completes processing, the first part waiting in the queue is processed while the completed part exits the system.

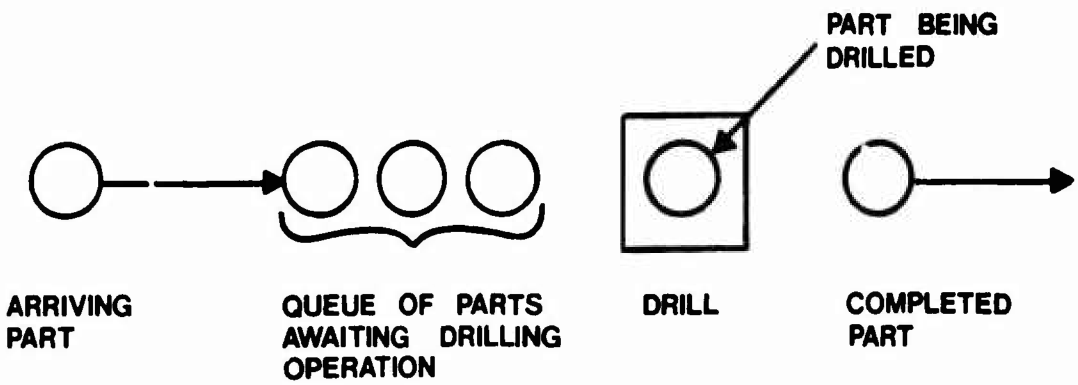
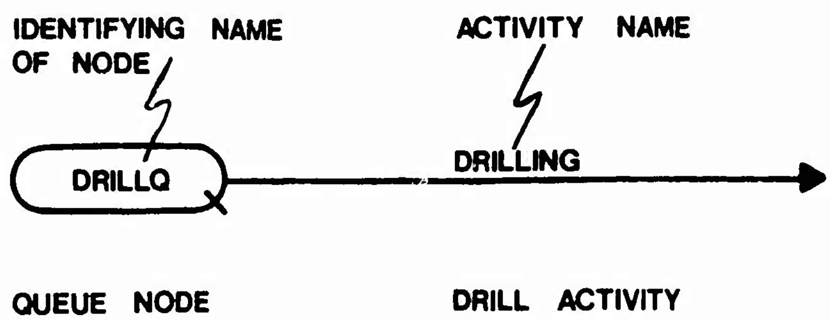
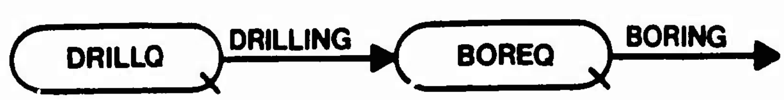


Figure 3-1. A Schematic of a Simple Production System

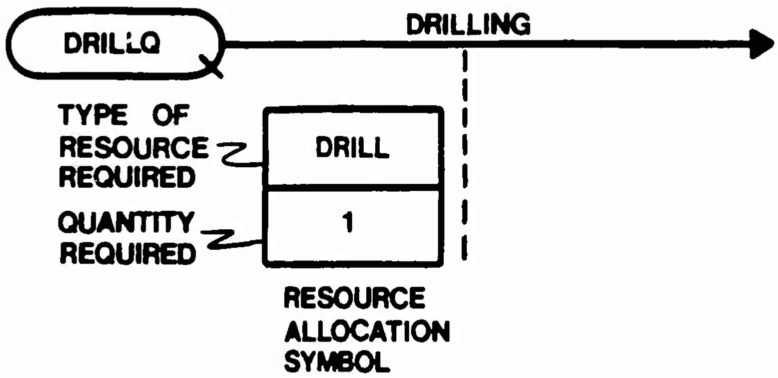
The passage of time in Entity Flow Networks is represented by an ACTIVITY. Arrows are the graphical representation of ACTIVITIES. In our example, the drilling operation takes time, and is thus graphically modeled by an arrow. If a part is being drilled when another part arrives, the arriving part must wait until the drill becomes available. Waiting occurs at QUEUE nodes. The figure shown below illustrates the Entity Flow Network representation of the drilling operation and the location of parts awaiting service.



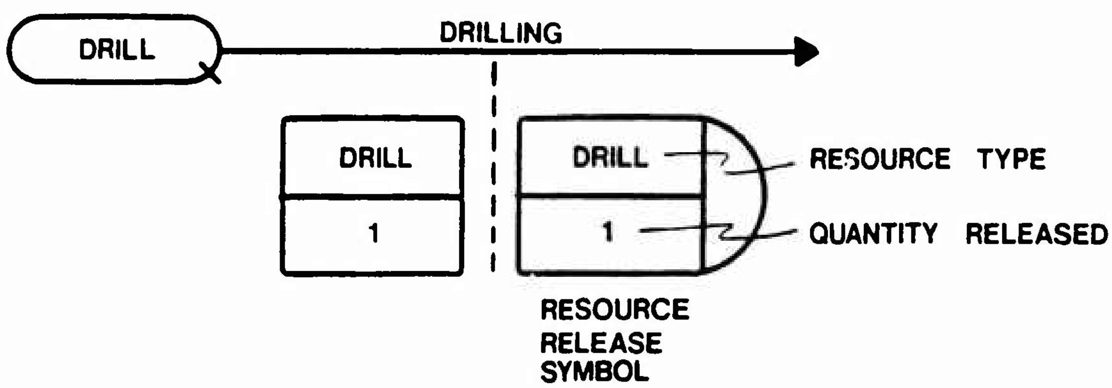
If another operation succeeds the drilling operation and parts may again wait for processing, the network would look like the one shown below.



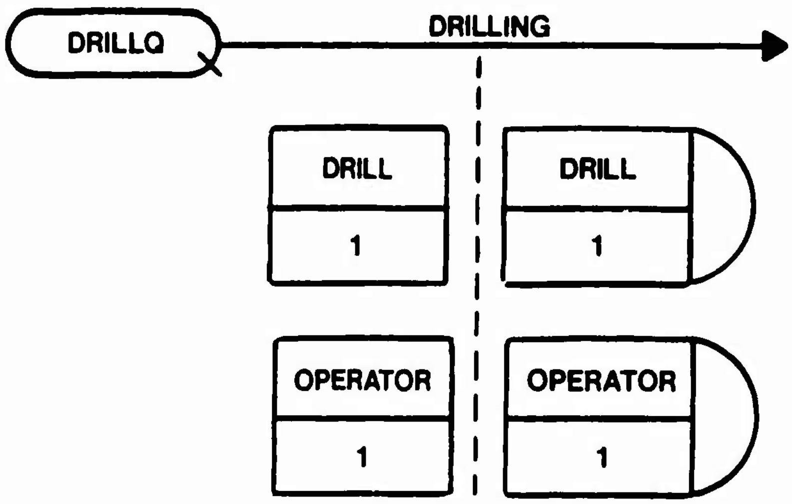
DRILLING and BORING are the names of the drilling and boring operations. In $IDEF_2$ any resources which are required to perform ACTIVITIES are shown on Entity Flow Networks. Resources are allocated to entities at the beginning of ACTIVITIES and are freed at the completion of ACTIVITIES. The allocation of a resource is accomplished with RESOURCE ALLOCATION nodes, which specify the type and quantity of the resource being allocated. In the network shown here, a single drill is allocated to parts to perform a drilling operation.



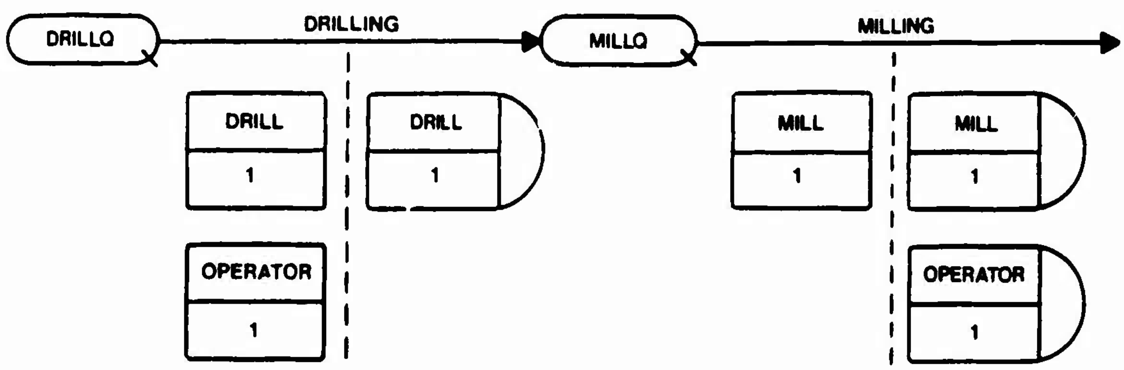
If, at the end of an ACTIVITY, the resource is no longer required by the part, it is freed via a RESOURCE RELEASE symbol. The resource type being freed and the quantity being released are specified on the symbol, as shown below



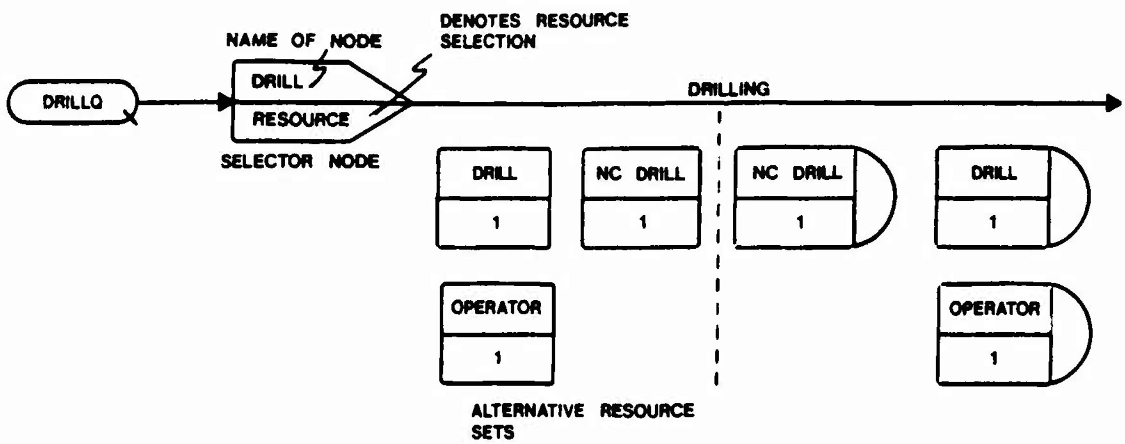
Some activities require more than one resource type. The DRILLING ACTIVITY in our example may require the presence of a human operator in addition to the drill. If so, another RESOURCE ALLOCATION symbol is used beneath the first one and another RESOURCE RELEASE symbol is required at the ACTIVITY's end.



Resources may remain allocated to entities for successive ACTIVITIES. If, for instance, a single operator can operate multiple machine types, the network shown below may be appropriate. Here, the resource OPERATOR is allocated to a part in order to perform two successive operations. Since he is not released after the DRILLING ACTIVITY, he is assumed to be also required for the milling operation. At the end of the MILLING ACTIVITY, both the MILL and OPERATOR resources are released.



Now consider an ACTIVITY where different types of resources can perform the same ACTIVITY. An example of this is the case where an ACTIVITY can be performed either manually or with automated machinery. Alternative resource sets are modeled by using adjacent columns of RESOURCE ALLOCATION and RESOURCE RELEASE symbols as shown in the figure below. The resource set that is selected to perform the ACTIVITY is released upon completion of the ACTIVITY. The SELECTOR node will choose the appropriate set of resources to perform the drilling operation based on a selection rule specified by the modeler. SELECTOR nodes are used to select between competing resources, activities, branches, and queues. In the example shown below, the RESOURCE label indicates that a selection must be made between alternative resource sets.



BOUNDARY nodes identify locations where entities may enter a network, exit a network, or be transferred to another location within a network. Three types of BOUNDARY nodes are available to $IDEF_2$ modelers: START, END, and GO TO. Adding BOUNDARY nodes to our last example network, we have the network shown in Figure 3-2.

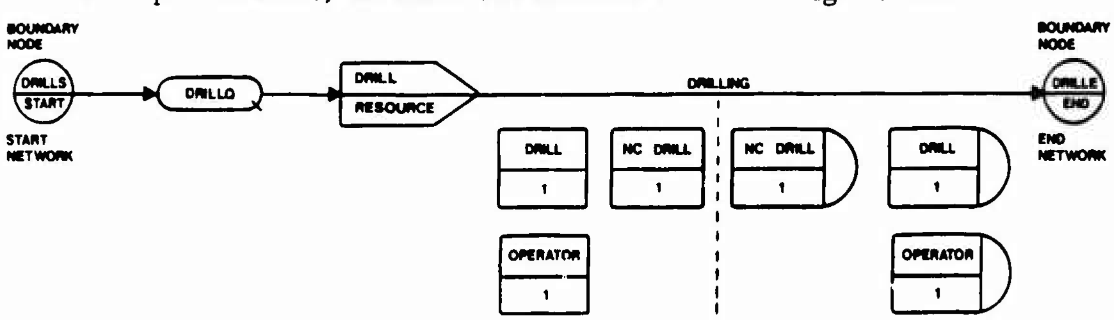


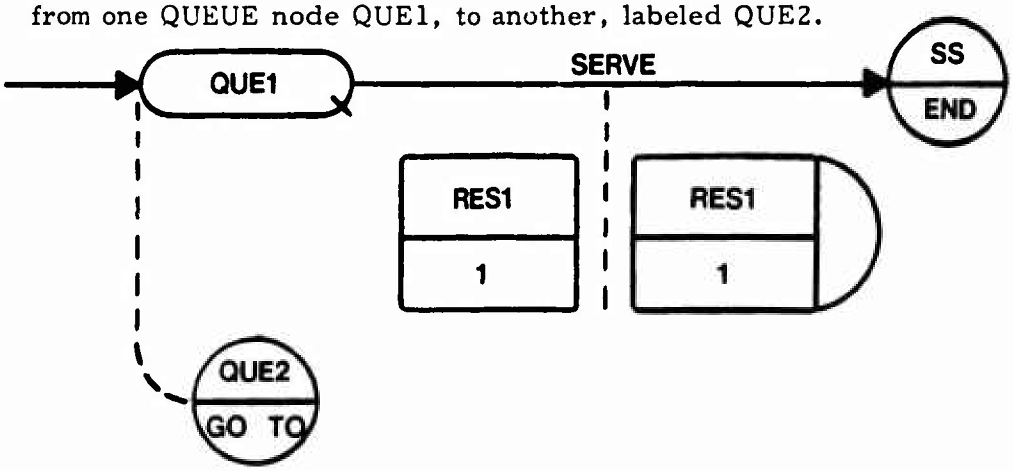
Figure 3-2. Example Entity Flow Network

These brief introductory examples were intended to briefly expose the beginning $IDEF_2$ modeler to the concept of entity flow, represented by network elements. Let us now begin to study the elements used in the Entity Flow Networks in more detail to better understand their meaning and the modeling concepts they facilitate.

QUEUE nodes

A QUEUE node is a location in a network where entities wait. When an entity enters a QUEUE node, its disposition depends on the status of the resource(s) for which it is waiting. The relative order in which entities wait in QUEUE nodes is determined by a specified queue ranking rule. The rules available for use in $IDEF_2$ are FIFO (First-in, first-out), LIFO (last-in, first-out), low-value first based on an entity attribute value, or high-value first based on an entity attribute value. If a QUEUE node is used to model a limited quantity of waiting space, a maximum capacity is specified.

When an entity arrives to a QUEUE node which is at its capacity, the disposition of the entity must be specified. The disposition is based on a specification as to whether the entity should balk or be blocked. In the case of balking, the entity can be routed to another node of the network using a GO TO boundary node. The node is specified by providing the label of the node as shown below. Here, an entity balks from one QUEUE node QUE1, to another, labeled QUE2.

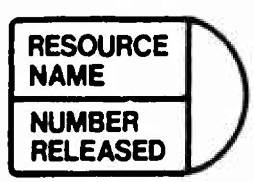


When an entity is blocked from entering a full QUEUE node, it waits in the previous QUEUE or ACTIVITY until space is available in the QUEUE. When space becomes available in the QUEUE, the entity will join the QUEUE. The symbol for blocking at a QUEUE node is shown below.

RESOURCE ALLOCATION and RELEASE Symbols

RESOURCE ALLOCATION SYMBOL

Resources are allocated to perform ACTIVITIES on entities. If more than one unit of resource type could potentially service an entity, the resource type must be specified on the RESOURCE ALLOCATION symbol. The number of units required for the ACTIVITY can be a variable or a constant. An example of an entity requiring a variable amount of resources is a batch of parts arriving to a machining center containing several machines of the same type. The number of machines required to service the batch depends on the size of the batch. $IDEF_2$ facilitates modeling this situation by permitting the modeler to specify the number of resource units with an $IDEF_2$ variable. Figure 3-3 lists the $IDEF_2$ variables which can be used in Entity Flow Networks.



RESOURCE RELEASE SYMBOL

RESOURCE RELEASE symbols release one or more units of a resource type upon completion of an ACTIVITY. The number of units released is specified in the same manner as in RESOURCE ALLOCATION symbols.

VARIABLE	*DEFINITION*
TIME*	Current Time
attribute	User-Defined Attribute Name of Current Entity
variable	User-Defined Variable Name
NNACT (NAME)*	Number of Active Entities in ACTIVITY "NAME" at Current Time
NNCNT (NAME)*	The Number of Entities that have Completed ACTIVITY "NAME"
NUMRES (NAME)*	Current Number of Units of Resource "NAME" Available
CONVEYOR	Status of Conveyor at Current Time: 0 → Down; 1 → Up
SHIFT	Status of Shift at Current Time: 0 → Down; 1 → Up
NNQ (NAME)*	Number of Entities in Queue "NAME" at the Current Time

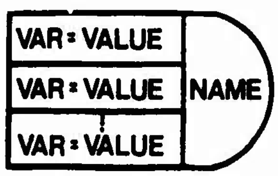
*These variables may not be assigned new values by the $IDEF_2$ modeler.

Figure 3-3. $IDEF_2$ Variables

ASSIGNMENT Node

The ASSIGNMENT node is used to prescribe values to the attributes of entities passing through the ASSIGNMENT node or to prescribe values for system variables that pertain to the model in general. Some $IDEF_2$ variables can be assigned a value in ASSIGNMENT nodes.

The symbol for the ASSIGNMENT node is shown below.



The values assigned to variables at ASSIGNMENT nodes can take on a variety of forms. The value can be a constant, an $IDEF_2$ variable, an arithmetic expression containing constants and/or $IDEF_2$ variables. or a sample from a probability distribution. A list of probability distributions available to $IDEF_2$ modelers is shown in Figure 3-4.

VARIABLE/FUNCTION	*DEFINITION*
EXPONENTIAL (MEAN)	A Sample from an Exponential Distribution with Mean MEAN
UNIFORM (LOW,HIGH)	A Sample from a Uniform Distribution in the Interval LOW to HIGH
WEIBULL (BETA,ALPHA)	A Sample from a Weibull Distribution with Scale Parameter BETA and Shape Parameter ALPHA
TRIANGULAR (LOW,MODE,HIGH)	A Sample from a Triangular Distribution in the Interval LOW to HIGH with Mode MODE
NORMAL (MEAN,DEVIATION)	A Sample from a Normal Distribution with Mean MEAN and Standard Deviation DEVIATION
LOGNORMAL (MEAN,DEVIATION)	A Sample from a Lognormal Distribution with Mean MEAN and Standard Deviation DEVIATION
ERLANG (MEAN,N)	A Sample from an Erlang Distribution which is the Sum of N Exponential Samples each with Mean MEAN
GAMMA (BETA,ALPHA)	A Sample from a Gamma Distribution with Parameters BETA and ALPHA
BETA (THETA,PHI)	A Sample from a Beta Distribution with Parameters THETA and PHI
POISSON (MEAN)	A Sample from a Poisson Distribution with Mean MEAN

Each Parameter for a Distribution can be Specified as a Constant, as a User-Defined Attribute, or as a User-Defined Variable. Distribution types and corresponding parameters may be determined via standard statistical methods discussed in the following:

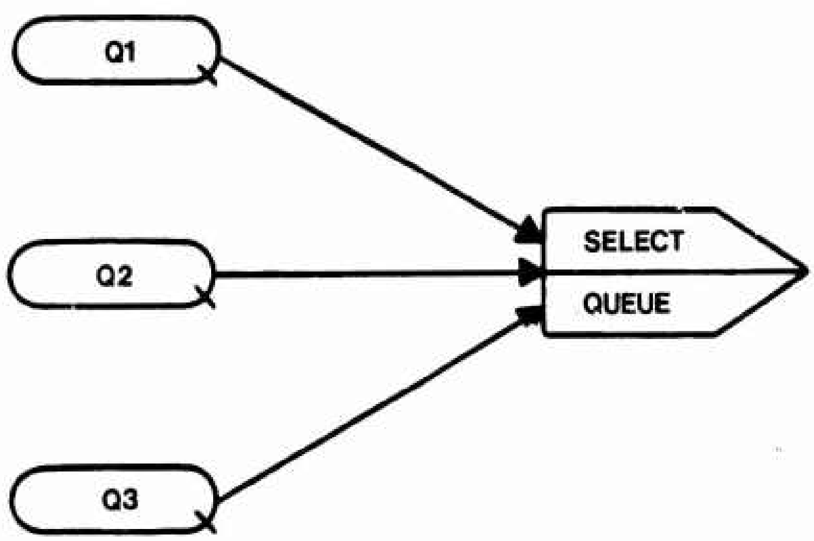
1. AID Users Manual, in preparation, Pritsker & Associates, Inc.
2. Hahn, C. J. and S. S. Shapiro, *Statistical Models in Engineering*, John Wiley & Sons, Inc., New York, 1967.
3. Hastings, N.A.J. and J. B. Peacock, *Statistical Distributions*, Butterworth & Co., Ltd., London, 1975.
4. McGrath, E. J. and D. C. Irving, *Techniques for Efficient Monte Carlo Simulation: Volume I: Selecting Probability distributions*, NTIS, 1973.
5. Pritsker, A. Alan B. and C. D. Pegden, *Introduction to Simulation and SLAM*, Halsted Press, New York, 1979.

Figure 3-4. Probability Distributions Available for use in $IDEF_2$ Models

SELECTOR Node

The SELECTOR node provides flexibility in modeling entity flow. In effect, SELECTOR nodes are used to model basic decision making within Entity Flow Networks. SELECTOR nodes can select: a.) an entity from parallel QUEUES; b.) a QUEUE to hold entities; c.) resources to perform ACTIVITIES; d.) ACTIVITIES for entities to engage in; and e.) BRANCHES for entities to take.

Consider first the selection of an entity from parallel queues. To accomplish this selection the modeler must specify a selection rule. Selection rules available for the QUEUE selection capability are shown in Figure 3-5. The network representation of this situation is shown below.

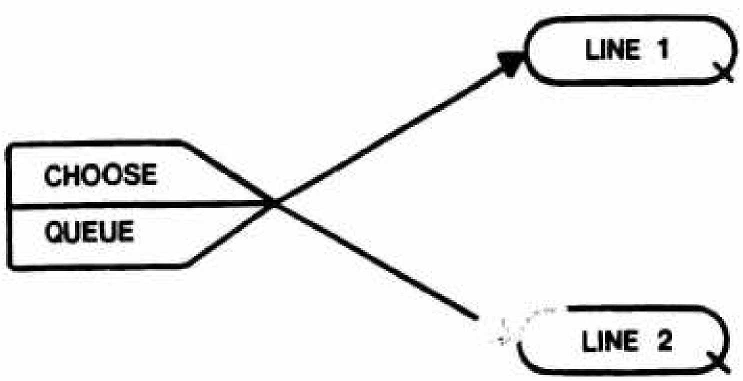


1. Preferred order – select entities from QUEUES in a specified order.
2. Cyclic priority – select from first QUEUE node containing an entity, beginning with the last QUEUE node selected.
3. Random priority – select from QUEUES in a random order.
4. Largest number priority – select from the QUEUE having the largest number of entities.
5. Smallest remaining capacity – select from the QUEUE having the least remaining capacity.

Figure 3-5. Selecting Entities from Paralleled Queues

If the selection rule specified is Preferred Order, the SELECTOR node will attempt to take entities from the QUEUE nodes in the order that they appear (top down) on the network. In this case, an attempt would first be made to remove an entity from Q1. If none were present, Q2 would be examined and an attempt would be made to remove an entity from it. Again, if none were present Q3 would be examined and, if an entity is present, it will be removed by the SELECTOR node.

The second type of selection involves the choice of a single QUEUE from several QUEUES to hold an entity. The selection rules available to model this entity disposition are listed in Figure 3-6. The symbolic representation of this situation is shown below.

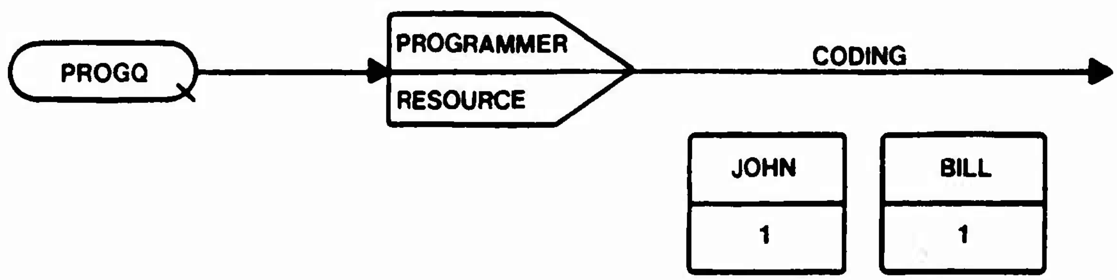


1. Preferred order—consider QUEUES in a specified order to select one in which to put the entity.
2. Cyclic priority—put the entity in the first QUEUE having capacity, beginning with the last QUEUE node selected.
3. Random priority—select a QUEUE at random in which to put the entity.
4. Smallest number priority—select the QUEUE with the smallest number of entities in it.
5. Largest remaining capacity—select the QUEUE with the largest remaining storage space.

Figure 3-6. Selecting a Queue to Hold Entities

This network example could be modeling the arrival of a customer to checkout lines at a supermarket. The selection rule used would be Smallest Number Priority which means select the QUEUE with the smallest number of entities in it. With this selection rule, the SELECTOR node would route arriving entities to the QUEUE node containing the least entities.

A third type of selection involves the selection of the resource(s) to perform an ACTIVITY. This selection is required when two or more non-identical resources can be used for the same ACTIVITY. An example of this situation is illustrated with Entity Flow Network symbols below.

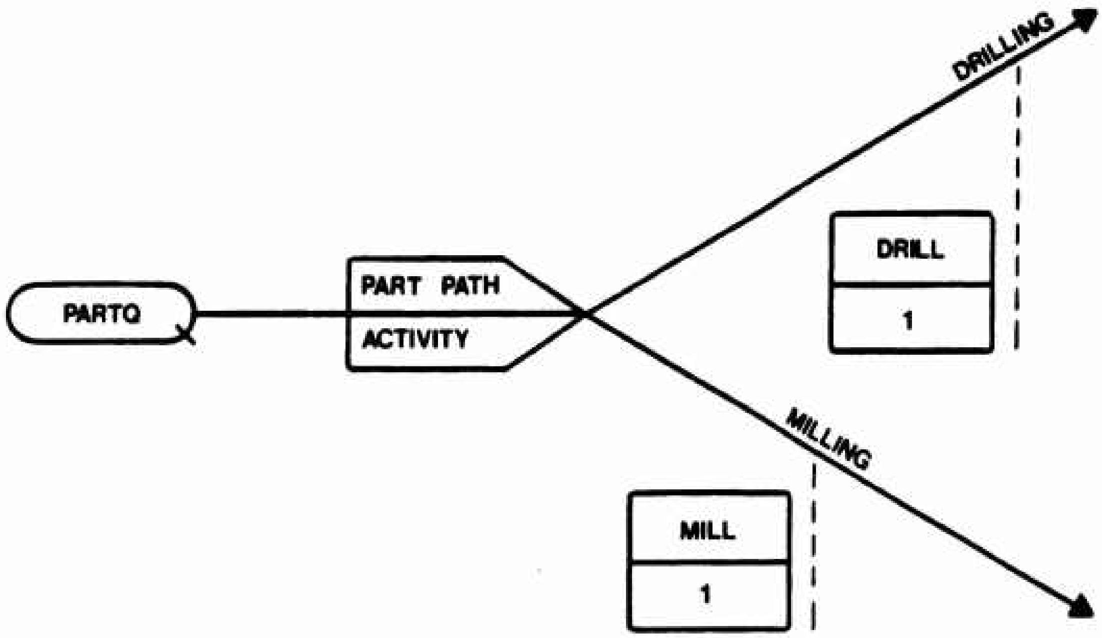


In this system, a certain programming task must be performed and two different programmers, John and Bill, can perform the task. A selection can be made to choose the programmer who will perform the programming task. A list of the selection rules available for this type of selection is shown in Figure 3-7. For example, if the selection rule was specified as Least Amount of Usage, the programmer that has been utilized the least would be allocated to an arriving entity. In the event that he is currently busy, the other programmer would be examined, and, if free, would be allocated to the entity.

1. Preferred order – similar to QUEUE selection
2. Cyclic order – similar to QUEUE selection
3. Random order – similar to QUEUE selection
4. Least amount of usage – select resource with least amount of busy time to date
5. Longest idle time – select resource that has been idle the longest

Figure 3-7. Resource Selection

If an entity can engage in two or more different ACTIVITIES at a particular point in time, a selection as to which ACTIVITY will be engaged next is required. This case is exemplified by a part that requires several machining operations when the order in which the operations are to be performed is unspecified. The part may reach a point in its process where two or more operations could be performed next if the appropriate resources are available. A choice must be made as to in which ACTIVITY the part will next engage. In network form, this translates to a decision concerning which path the entity will follow. The Entity Flow Network for this sample case is illustrated below. The selection rules available to $IDEF_2$ modelers for ACTIVITY selections are presented in Figure 3-8.

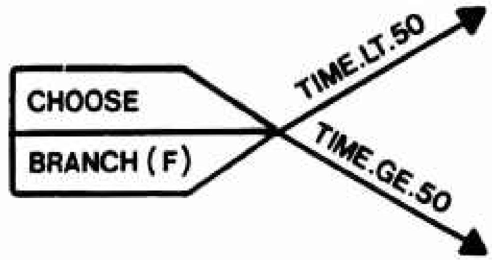


1. Preferred order—similar to resource selection
2. Cyclic order—similar to resource selection
3. Random order—similar to resource selection
4. Least amount of usage—similar to resource selection

Figure 3-8. Selection of Activities

In this example, entities in the QUEUE node labeled PARTQ may be processed by either a drill or a mill. If, for example, the selection rule is specified as Random Order, no preference is given to either ACTIVITY. That is, the SELECTOR node will randomly select one of the ACTIVITIES and check the status of the required resource. If the resource is occupied with another part, the status of the remaining resource will be examined, and if free, the part will be routed over the ACTIVITY which is performed by that resource.

Finally, consider the selection of alternate BRANCHES on which an entity may travel. The selection of BRANCHES is performed conditionally or probabilistically. Conditional BRANCHING refers to an entity taking a BRANCH only if a specified condition is met. There are two types of conditional BRANCHING specifications: these are conditional-take-first; and conditional-take-all. The former requires that only the first BRANCH on which the specified condition is met be taken. The latter states that all BRANCHES meeting the specified condition are to be taken. The symbolic representation of an example of conditional-take-first BRANCHING is shown below.

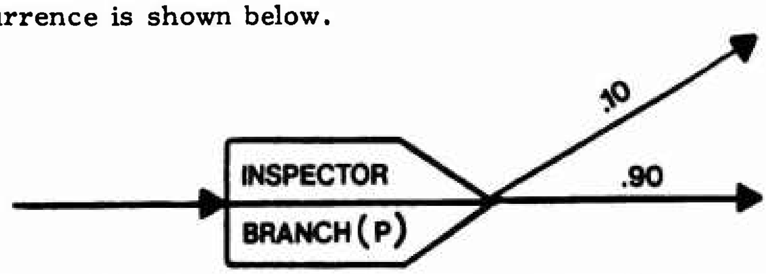


In this example, one of two BRANCHES will be taken based upon the conditions specified for each BRANCH. The BRANCH (F) specification states that conditional-take-first branching is the basis for determining the BRANCH which arriving entities will take. The condition on the first BRANCH states that it should be taken if the current time has a value of less than 50 time units. The second conditional statement permits the BRANCH to be taken if the current time has a value greater than or equal to 50 time units. Any $IDEF_2$ variable may be used to construct a conditional BRANCH (see Figure 3-3). The relational operators which may be used are listed in Figure 3-9. The $IDEF_2$ modeler is responsible for specifying a condition for any situation which occurs at that point in the network.

RELATIONAL OPERATOR	*MEANING*
.EQ.	EQUAL TO
.LT.	LESS THAN
.LE.	LESS THAN OR EQUAL TO
.GT.	GREATER THAN
.GE.	GREATER THAN OR EQUAL TO
.NE.	NOT EQUAL TO

Figure 3-9. Operators Which may be used in Conditional Branching

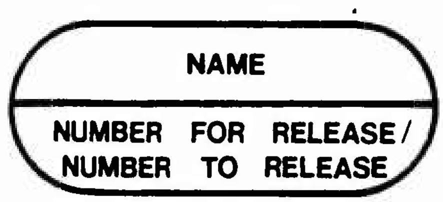
Probabilistic BRANCHING refers to the instance where an entity is routed over a BRANCH a prescribed percentage of the time. An example of this is an inspection operation where parts pass inspection ninety percent of the time. The remaining ten percent of the parts are rerouted to an adjustment station for readjustment. The symbolic representation for this occurrence is shown below.



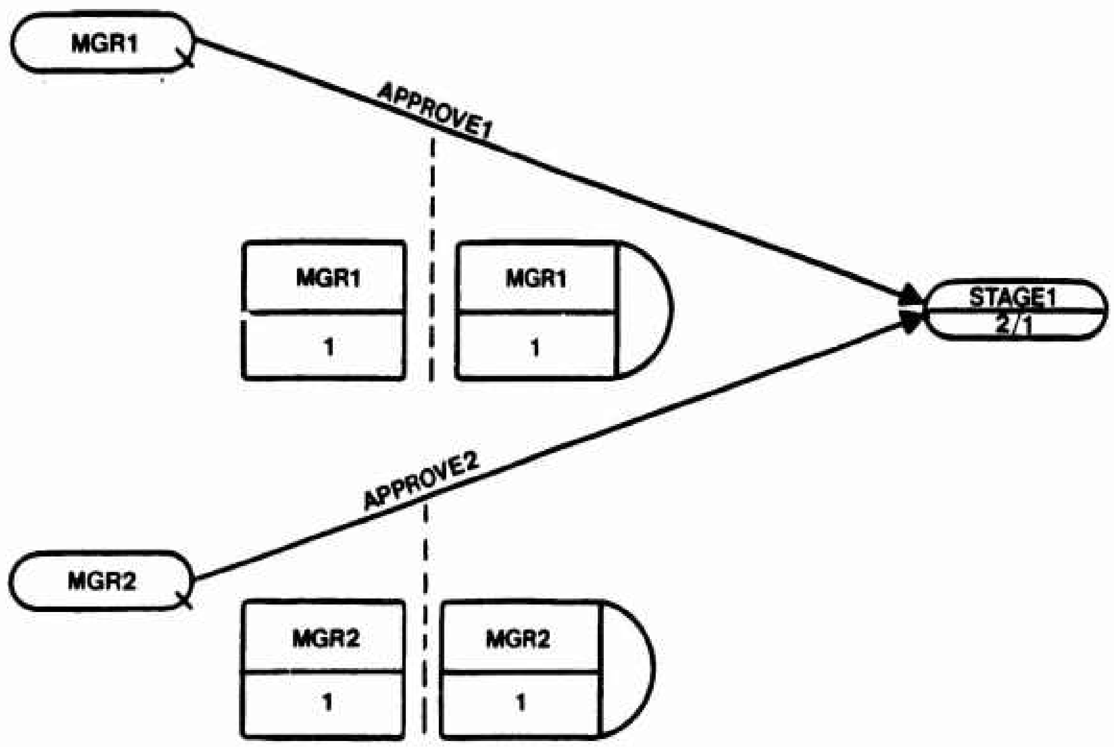
The modeler is responsible for insuring that the probabilities on all probabilistic BRANCHES eminating from a node sum to one.

ACCUMULATE node

An ACCUMULATE node models the grouping of entities. An example of its use is a situation where parts are processed individually and transported in batches. Entities enter the node and are released when a specified number of entities have arrived. The symbol for the ACCUMULATE node is shown below.

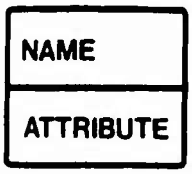


On this node one must specify the number of entities required to arrive to the node (NUMBER FOR RELEASE) as well as the number of entities to leave the node upon its release (NUMBER TO RELEASE). To illustrate an application of the ACCUMULATE node, consider the example shown below. Here we have modeled one step in the life of an appropriation request (AR). Several signatures are required at a particular level of approval. The AR is ready for approval at the next level only after the signatures of MGR1 and MGR2 have been received. The approved AR continues in its approval schedule after accumulating the signatures of MGR1 and MGR2.



MATCH node

A MATCH node matches entities residing in specified QUEUE nodes that have equal values of a specified attribute. When each QUEUE node preceding a MATCH node has an entity with the specified common attribute value, the MATCH node removes the matched entities from the QUEUE nodes and routes them to other nodes. Note that each entity is routed individually. An example of the symbology for the MATCH node is shown below.



To illustrate the application of MATCH nodes, consider the example shown in Figure 3-10. Here we have the situation where an entity is required from each of the nodes Q1, Q2, and Q3, with matching values of attribute PART NUMBER. Upon finding an entity in each of the QUEUE nodes with a common specified value for attribute PART NUMBER, the MATCH node will remove those entities from the QUEUE nodes and route those from Q1 and Q2 to the ACCUMULATE node named HOLDER. At the same time, the entity from Q3 will be routed to the ACCUMULATE node named FILE.

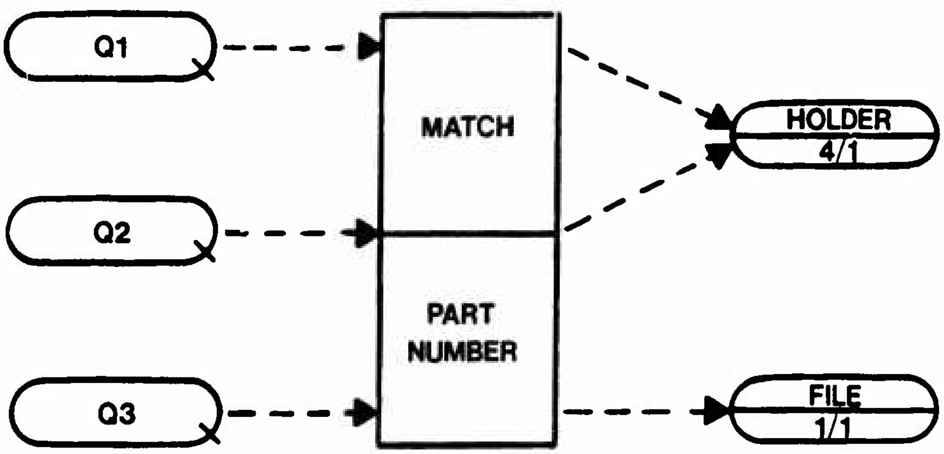


Figure 3-10. Example Application of Match Node

ASSEMBLY node

ASSEMBLY nodes are used to model situations where several entities, each coming from different QUEUE nodes, are physically assembled into a single entity having a single set of attributes. This is similar in function to the MATCH node in that entities are removed from preceding queues by this node. However only one entity exits an ASSEMBLY node, whereas all incoming entities exit a MATCH node. The symbol for the assembly node is shown below.

ACTIVITIES

ACTIVITIES are modeled graphically with arrows and represent the passage of time. The duration of the ACTIVITY is specified on a form which will be described later. The symbol for an ACTIVITY is shown below.

SUMMARY

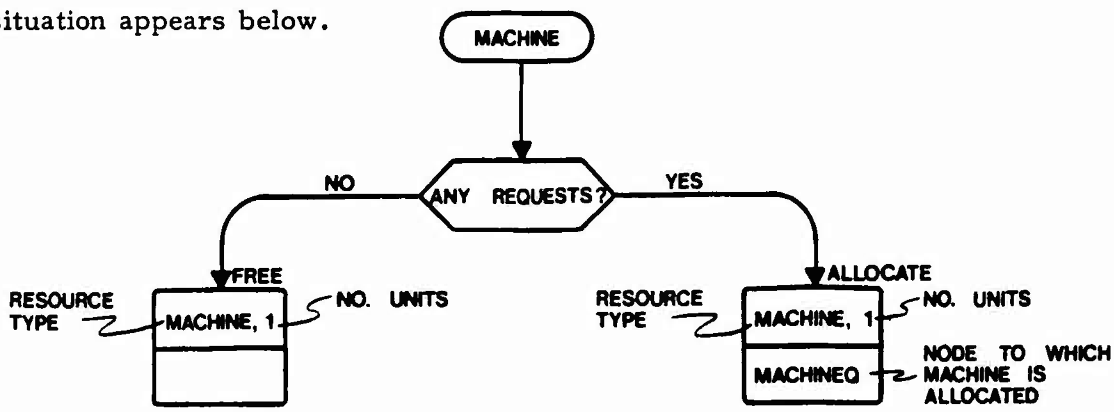
Entities form an important part of the real world systems modeled with $IDEF_2$. They may be real or conceptual. Entity Flow Networks model the flow of entities through manufacturing systems. They represent the paths entities follow through systems and specify the resources required for entity passage. The concepts involved in modeling entity flow are embodied in a set of node types and the graphical representation of ACTIVITIES.

2.1.2 Resource Disposition Trees

When a resource completes an ACTIVITY, its disposition depends on the status of the system. That is, it may become idle if no other entities require its attention; it may be allocated to an entity that requires its use to engage in an ACTIVITY; or it may wait for other resources to become available in order to perform an ACTIVITY requiring multiple resources. The decision regarding a resource's disposition is made via a Resource Disposition Tree.

A Resource Disposition Tree consists of nodes containing QUESTIONS, the answers to which determine either the next QUESTION or what ACTIONS are to be taken on the available resources. In this manner, the leaf nodes of each tree contain ACTIONS to be taken on the available resources.

To illustrate the use of pertinent QUESTIONS in Resource Disposition Trees, consider the case of a machine that processes parts that queue in front of it. Each time the machine completes a part, the disposition of the machine must be determined. If no parts are awaiting processing, the machine becomes idle; if parts are waiting, the machine is allocated to the first part in its QUEUE. The Resource Disposition Tree for this situation appears below.



In this example, the QUESTION "ANY REQUESTS?" asks if any requests for the services of the resource are outstanding i.e., are any parts awaiting processing? If the answer to this is "NO," the arrow labeled "NO" is taken and a FREE ACTION is specified. The information in the top row of the box states that resource type MACHINE is affected with one unit being freed. If the question is answered "YES," the arrow labeled "YES" is taken and the action specified is to ALLOCATE one unit of resource type MACHINE to the node named MACHINEQ.

FREE and ALLOCATE are examples of ACTIONS that may be specified in Resource Disposition Trees. Four types of ACTIONS are available to $IDEF_2$ modelers, as shown in Figure 3-11.

ALLOCATE

FREE

PREEMPT

ERROR

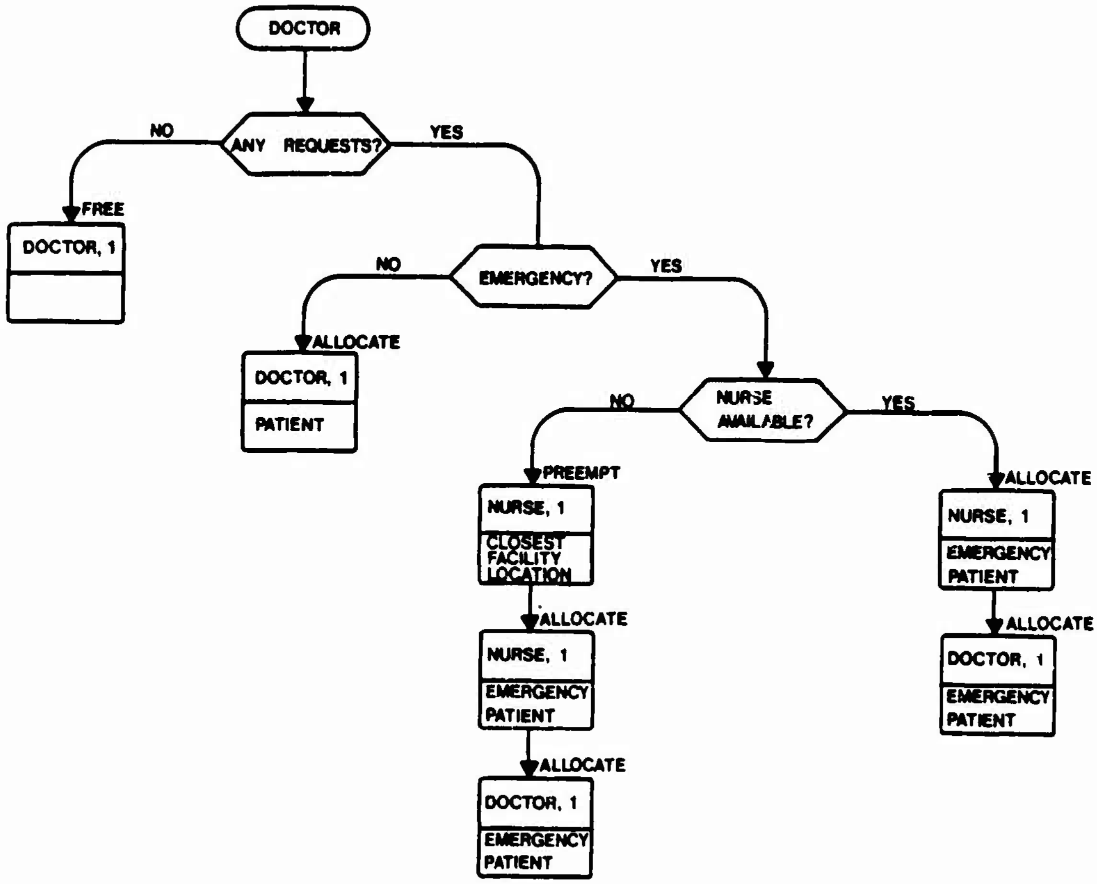
Figure 3-11. Possible Action Types for $IDEF_2$ Resource Disposition Trees

Using Resource Disposition Trees, the analyst can ask questions about the status of the system in order to decide the disposition of resources. Several typical questions are shown in Figure 3-12. If a modeler wants to ask a question that does not appear on the list, he may create questions which better suit his purpose. Most modelers will find the list presented quite adequate for their modeling needs.

Let us examine another example of resource disposition. Consider a hospital where a limited number of doctors and nurses are on duty. Ordinarily, nurses have their own set of tasks to perform. However, in some emergency cases, a doctor will preempt a nurse from her other duties to assist in caring for a patient. The Resource Disposition Tree for the resource DOCTOR is shown on Page 50.

QUESTION	*CLARIFICATION*
ANY REQUESTS?	Are there any entities currently waiting to use this resource?
ANY SINGLE REQUESTS?	Are there any entities which are waiting to use only this resource?
ANY MULTIPLE REQUESTS?	Are there any entities requiring this and other resources?
ANY LAST RESOURCE REQUESTS?	Are there any entities requesting this resource which have had all other required resources allocated to them?
ANY RESOURCE REQUIRE-MENTS SATISFIED?	Are there any entities requesting multiple resources where all required resources are either available or already allocated to the entity?
RESOURCE R IS FREE?	Are any resource R's free?
RESOURCE R WILL BE AVAILABLE IN N TIME UNITS?	Will resource R be released from the activity it is currently performing in N time units?
IS UTILIZATION (R) .RC. VALUE?	How does the utilization of resource R compare to the given value where RC = {EQ,NE,LT,GT,LE,GE}?
IS THE NUMBER IN <QUEUE NAME> .RC. VALUE?	How does the number in queue <queue name> compare to the given value where RC = {EG,NE,LT,GT,LE,GE}?
PROCESSED BY ACTIVITY Y?	Are there any entities which have been processed by activity Y?
NOT PROCESSED BY ACTIVITY Y?	Are there any entities which have not been processed by activity Y?
LAST PROCESSED BY ACTIVITY Y?	Are there any entities whose most recent activity was activity Y?

Figure 3-12. A List of Questions for use in Modeling Resource Disposition



In this example, the first QUESTION asks if there are any outstanding requests for DOCTOR. If not, the doctor is freed. If there is a request, the next QUESTION asks whether or not it is an emergency. If the answer to this is "NO," the doctor is allocated to the node labeled PATIENT. If an emergency exists, the next QUESTION checks the availability of a nurse. If a nurse is available, she and the doctor are allocated to the node named EMERGENCY PATIENT. If no nurses are available, the tree specifies a preemptive action, seizing the nurse at the closest location to the emergency. CLOSEST FACILITY LOCATION is an example of a preemptive rule. Several other rules for preemption are listed in Figure 3-13.

PREEMPT RESOURCES FROM THE:

ENTITY NEAREST COMPLETION
ENTITY NEAREST END OF OPERATION
ENTITY FARTHEST FROM COMPLETION
ENTITY FARTHEST FROM END OF OPERATION
CLOSEST FACILITY LOCATION (TO ENTITY PREEMPTING)
CLOSEST TO <FACILITY LOCATION>
HIGHEST PRIORITY ENTITY
LOWEST PRIORITY ENTITY
LEAST RESOURCES REQUIRED
MOST RESOURCES REQUIRED

Figure 3-13. Possible Preemption Rules

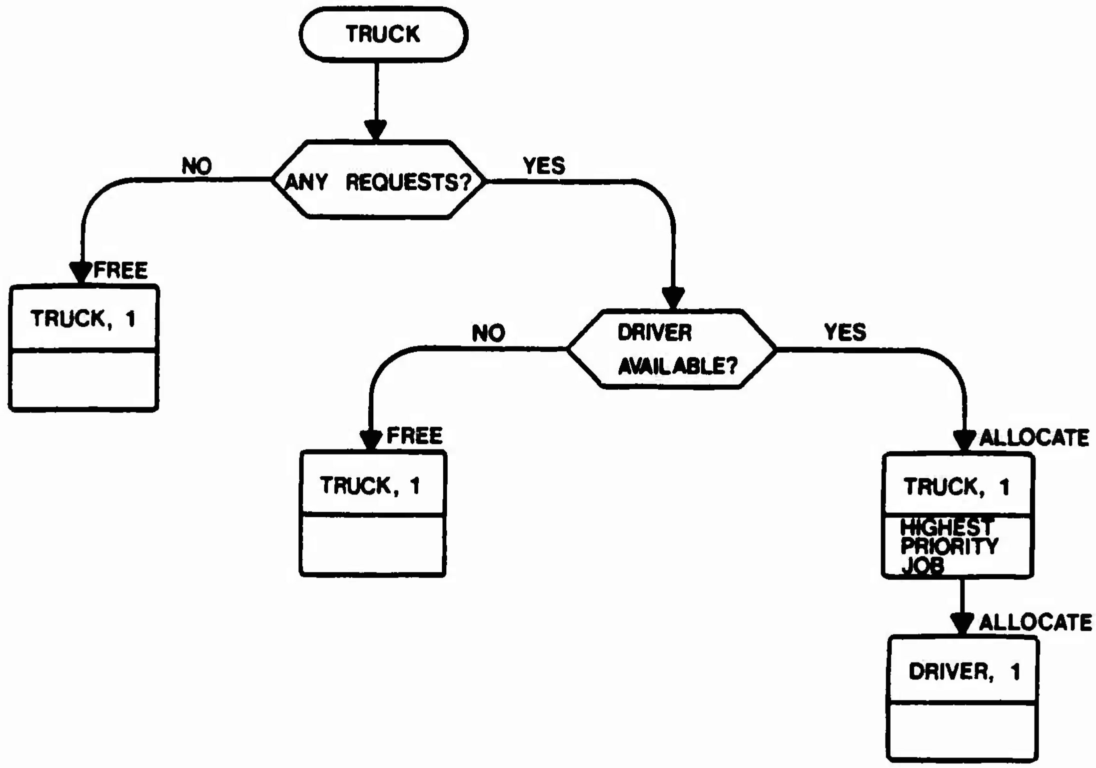
In our examples thus far, resources have been allocated to specified nodes. $IDEF_2$ permits resource allocation to be specified with many allocation rules which are listed in Figure 3-14. Consider another example.

ALLOCATE

SINGLE REQUEST
LAST RESOURCE
RESOURCE REQUIREMENTS SATISFIED
A QUEUE
AN ACTIVITY
CLOSEST FACILITY LOCATION
HIGHEST PRIORITY ENTITY
LOWEST PRIORITY ENTITY
FEWEST RESOURCES REQUIREMENTS
MOST RESOURCES REQUIREMENTS
LEAST ADDITIONAL RESOURCES REQUIRED
MOST ADDITIONAL RESOURCES REQUIRED

Figure 3-14. Possible Allocation Rules

A tool and die shop makes several types of dies for a fabricating manufacturer. Many of the dies are heavy enough to prohibit manual transportation between machine stations. These transfers between stations are accomplished by a single lift truck which handles all transfer requests with the shop. The Resource Disposition Tree for this lift truck is shown below.



Upon a "YES" answer to the first QUESTION, the availability of a driver is examined. If no drivers are free, the truck is FREED. If there is a driver available, TRUCK is assigned to the highest priority job requesting a transfer. Both TRUCK and DRIVER are allocated to that job.

Summary

Resource Disposition Trees are used to determine the disposition of resources that have become available. The trees consist of nodes containing QUESTIONS concerning the status of the system, whose answers either specify another QUESTION or the ACTION(S) to be taken on the available resources. By combining QUESTIONS and ACTIONS in the manner illustrated in this section, $IDEF_2$ modelers can specify both simple and complex resource disposition procedures.

3.1.3 System Control Networks

System Control Networks portray activities or conditions which may affect the status of the system and flow of entities but do not directly cause entity flow. They can be used to create entities, alter attributes of entities, and to alter the capacity of resources. In general, System Control Networks are used to initiate entity, resource, and facility changes.

The activities or conditions represented in System Control Networks are triggered by events; these are happenings, occurrences, milestones, or decisions. Event times are the instants at which the status of the system may change. Immediately prior to an event occurrence, it is not possible to predict the future status of the system with certainty. Events may occur randomly or at specified intervals.

An illustrative example will provide an introduction to the concept of System Control Networks. Consider the arrival of entities to a system. Entity arrival is an event because it causes a change in system status, i.e., the system contains one more entity after the entity arrives. Once the entity enters the system, its flow is modeled with an Entity Flow Network. However, its insertion into the system is modeled through a System Control Network. Figure 3-15 illustrates the representation of entity arrivals and insertions into Entity Flow Networks with System Control Network symbology.

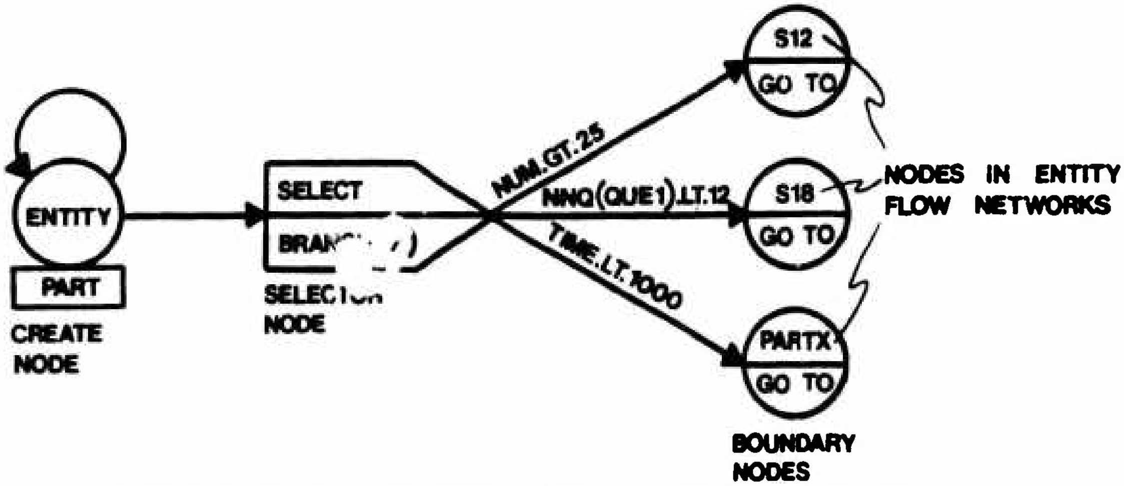


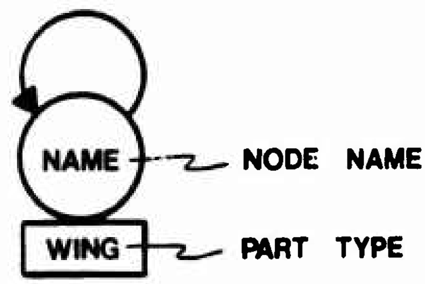
Figure 3-15. System Control Network for Entity Creations

The network shown in Figure 3-15 demonstrates several features of System Control Networks. The SELECTOR and BOUNDARY nodes serve the same purpose as those introduced in entity flow networking. The CREATE node models the arrival of entities to the system. The arc on the CREATE node denotes the time between arrivals of entities to the system.

The SELECTOR node dispenses entities to BOUNDARY nodes which enter the entities into Entity Flow Networks. BRANCH(A) specifies conditional, take-all branching as the basis for branch selection. The first branch will be taken if the user-defined variable NUM has a value greater than 25 and an entity will be entered into the Entity Flow Network where the node named S12 appears. The second branch will be taken if the number of entities in the node QUE1 is less than twelve, while the third branch is taken if the current time is less than 1000.

Several symbol types used in Entity Flow Networks are also used in System Control Networks. The list of common symbol types include ASSIGN nodes, RESOURCE RELEASE symbols, ACCUMULATE nodes, and BOUNDARY nodes. Descriptions of these symbols given in Section 3.1.1 are directly applicable to System Control Networks. Node types that are unique to System Control Networks are described below.

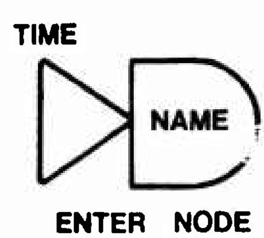
CREATE node



The CREATE node is used to generate entities. It is used to model arrivals of any type of entity to a system, including parts and any type of information that might "arrive" or be requested at specified intervals of time. The time between creations of entities may be specified by a constant value, a probability distribution, a historical list of events, a table, or graph. If the entities are always to be routed directly to a particular Entity Flow Network, the create node should be followed by GO TO nodes which transfer control to the appropriate START node in the Entity Flow Network.

ENTER node

The ENTER node shown below is used to denote the first occurrence of an event modeled in a System Control Network. It is used to gain entry into the network; thus, the first node in networks in which it is used is always an ENTER node. ENTER nodes can be used to mark or initiate an event occurrence since the time of its initial release is user-specified directly above the symbol.



ACTIVATE/DEACTIVATE nodes

The ACTIVATE node is used to activate resources, that is, make resources available for use in the $IDEF_2$ model. An ACTIVATE node will activate all units of the resource type specified. The DEACTIVATE performs the opposite function, deactivating all units of the specified resource type.

Special features of these nodes allow the user to specify the key words SHIFT or CONVEYOR. Using SHIFT instead of a resource type will extend the activate/deactivate function to all resource types. This feature is designed to conveniently facilitate the modeling of shift changes, as illustrated in Figure 3-16. Using the key word CONVEYOR represents a switch to activate/deactivate a conveyor's movement. The network symbols for ACTIVATE and DEACTIVATE nodes are shown below.

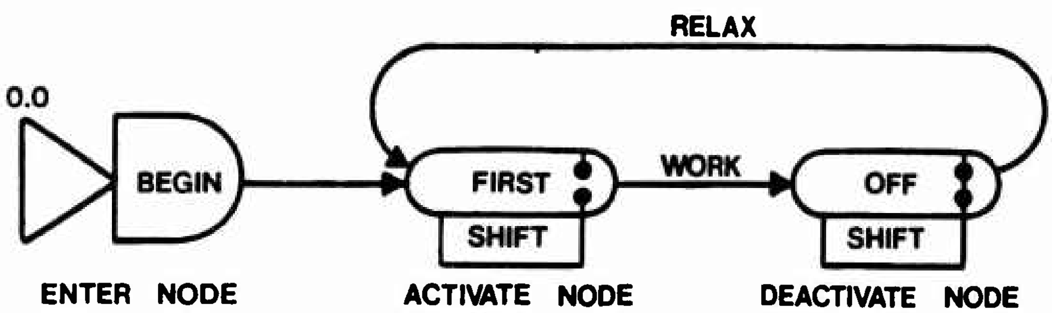
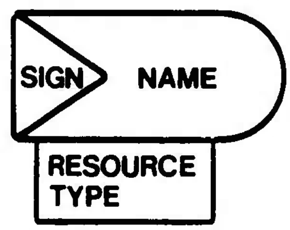


Figure 3-16. Shift Changes

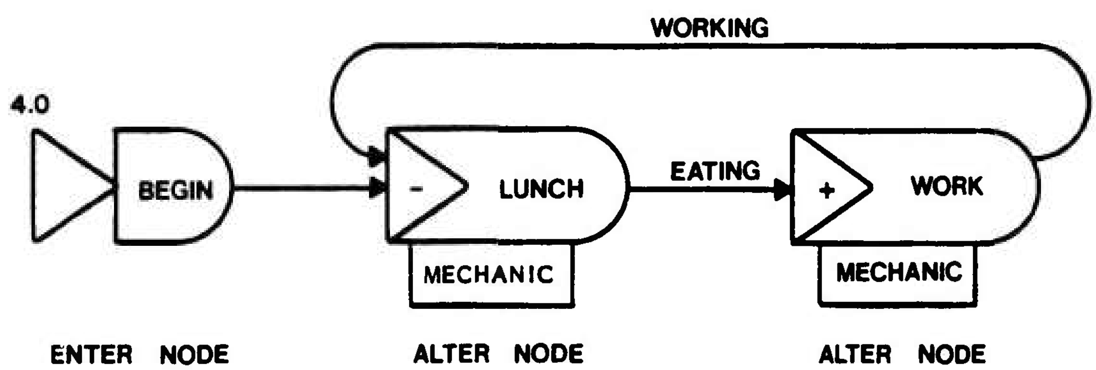
In this example, the ENTER node initiates the network at time 0.0, releasing the ACTIVATE node. With the SHIFT designation, all resources in the model are activated by this single ACTIVATE node. The ACTIVITY eminating from this node represents the duration of the first and only working shift. When the DEACTIVATE node is released, all resources are deactivated until the end of the ACTIVITY labeled RELAX which represents the time between working shifts.

ALTER node

ALTER nodes are used to make discrete changes in the capacity of resources. For example, the addition of a machine to a machine shop constitutes a change in resource capacity. This could be accomplished with an ALTER node. The symbol for the ALTER node is shown below.



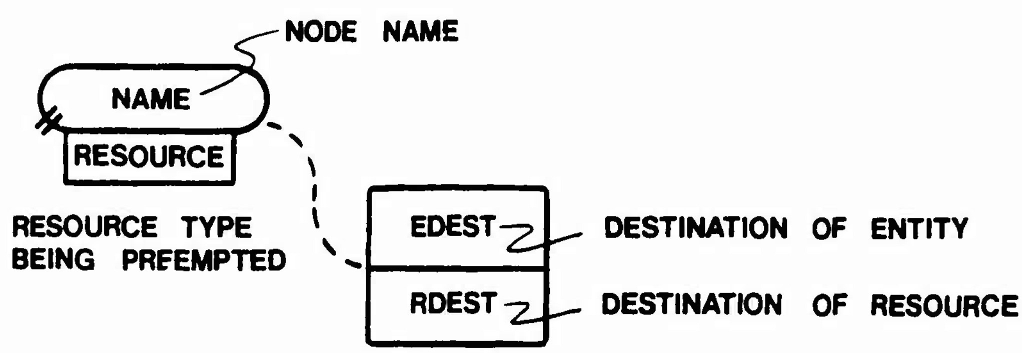
The modeler must specify the sign of the change, i.e., positive or negative. In addition, the resource type being altered must be specified on the node. To illustrate the use of ALTER nodes, consider the figure shown below.



This figure demonstrates an approach to modeling lunch breaks. Here, the resource type MECHANIC is being altered to model the change in the number of mechanics available to work during the lunch hour. The 4.0 above the ENTER node triggers the initiation of the network at time 4.0. The first ALTER node reduces the number of resources of type MECHANIC, indicated by the minus sign (-) on the node. The arrow labeled EATING represents the time that mechanics are on lunch break. The second ALTER node increases the number of mechanics, indicated by the plus sign (+) in the node. The arrow labeled WORKING represents the time until the next lunch break. Notice that the network is self regenerative in that once the network is initiated, it will maintain itself and need not be triggered again.

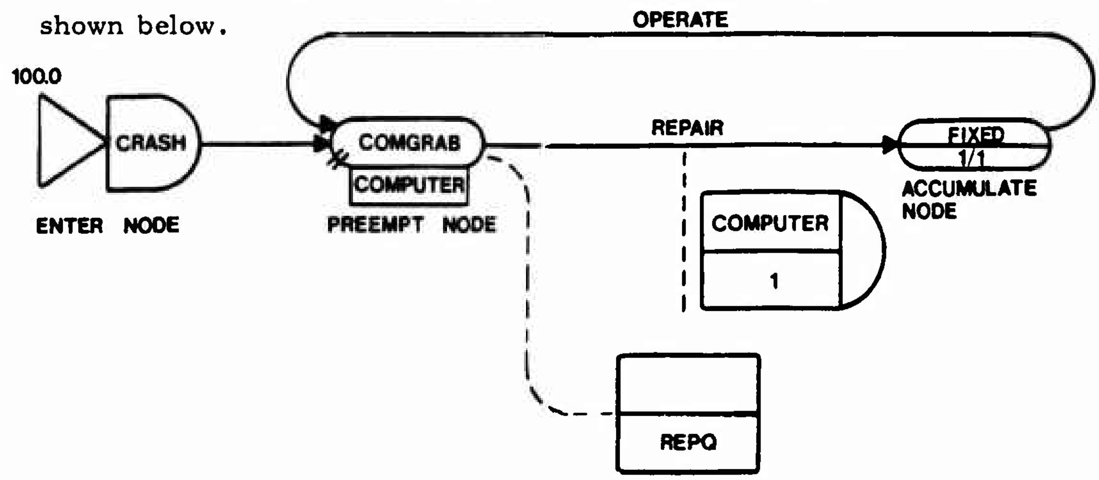
PREEMPT node

PREEMPT nodes are used to remove resources from the activity in which they are currently engaged, for employment in another activity. For example, a machine may be pulled from its service on one job to begin work on another job of higher priority. The System Control Network symbol for the PREEMPT node is shown below.



A PREEMPT node conceptually takes a resource out of an Entity Flow Network and places it in a System Control Network. Once the resource has been preempted in this way, it is routed to the network location specified for it in the bottom half of the symbol shown above (RDEST). Any entities being processed by the resource, at the time of its preemption are routed to the network location specified in the top half of the symbol (EDEST). If no location is provided by the user for the entity destination, it is assumed to be placed at the head of the queue for the preempted resource. Consider the following example of the use of a PREEMPT node.

The computer in a DNC machining center experiences periodic failure, requiring repair. When failure occurs, the entire machining center goes down because the computer controls the system. This situation can be modeled with the use of a PREEMPT node in the manner shown below.



In this example, the ENTER node initiates the network at time 100.0. At this time, the PREEMPT node will preempt the computer from the Entity Flow Network in which it normally operates. The computer will be immediately routed to the node labeled REPQ to be repaired. Since no entities are processed directly by the computer, no destination is specified for inprocess entity disposition. The resources being used to service entities are unavailable while the computer is being repaired and any entities being processed by the resources will be placed at the head of the queues waiting for the resources to become available again. At the end of the activity labeled REPAIR, the computer is released from this network with a RESOURCE RELEASE symbol as shown. The ACCUMULATE node is released at the end of every REPAIR ACTIVITY and initiates the OPERATE ACTIVITY which represents the time until the next computer failure. This network is self regenerating after its initiation and will model the recurring breakdown of the computer.

DETECT node

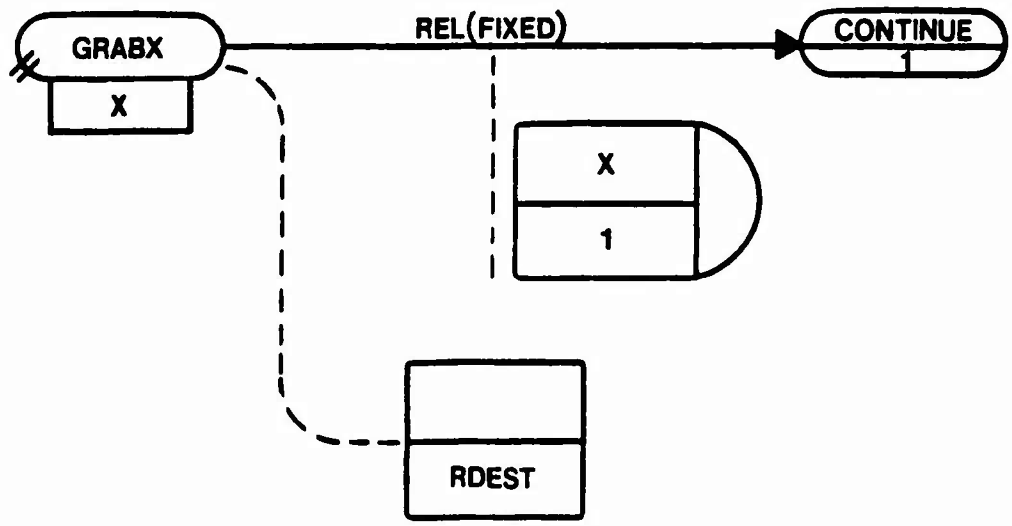
DETECT nodes are used to detect crossings of threshold values by continuous variables. Continuous variables are those whose value changes continuously over time. The symbol for the DETECT node is shown below.

FACILITY, ENTITY, OR RESOURCE NAME	ATTRIBUTE NAME	NODE NAME

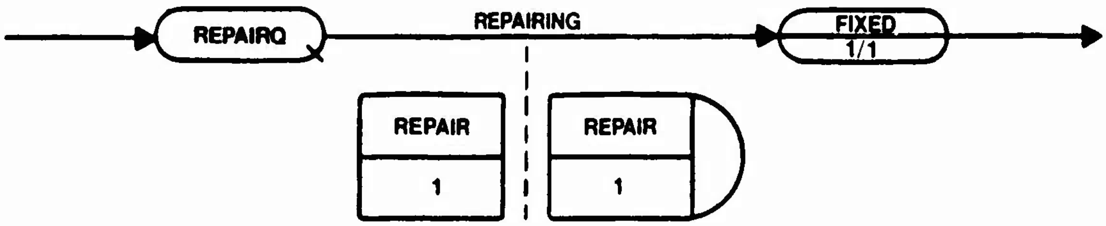
Continuous variables are modeled with equations that describe the variable's value over time. These variables are maintained as attributes of facilities, entities, or resources. For example, consider the continuous variable "tool life," which is a function of the time the tool is used. The life of the tocl is described as a function of its time in use. After a prescribed percentage of the tool's life has been depleted, the tool must be repaired or replaced. A DETECT node could be used to detect the point at which the tool must be repaired.

Activities in System Control Networks

Arrows are used to model ACTIVITIES in System Control Networks. ACTIVITIES represent the passage of time in System Control Networks as they do in Entity Flow Networks. The discussion given in Section 3.1.1 describing methods for specifying ACTIVITY durations applies directly to System Control Networks. In addition, ACTIVITIES of indefinite duration may be modeled in System Control Networks. ACTIVITIES whose duration are dependent on the status of another ACTIVITY may be modeled as shown on Page 61.



This example depicts the preemption of resource X to model a repair. X is routed to RDEST by the PREEMPT node where it will be repaired by another resource. Since it is not known whether the repairing resource is available to begin work on X at the time of preemption, it is not possible to predict when X will be repaired and back in service. Therefore, REL(FIXED) is used to specify the duration of time until X is again operational. REL(FIXED) signifies that the ACTIVITY will end when the node named FIXED, located elsewhere in the model, is released (i.e., next reached). The Entity Flow Network which contains FIXED and the repair ACTIVITY is shown below.



When the resource REPAIR is available, the REPAIRING ACTIVITY begins. Upon its completion, the node named FIXED is released, ending the ACTIVITY in the System Control Network.

SELECTOR node

The ACTIVITY selection and BRANCH selection capabilities of the SELECTOR node may be used in System Control Networks as well as in Entity Flow Networks. A discussion of the use of the SELECTOR node in Entity Flow Networks appeared in Section 3.1.1. The portions of the discussion pertaining to ACTIVITY selection and BRANCH selection also apply in System Control Networks. Figure 3-8 lists the selection rules for the selection of ACTIVITIES. The selection of BRANCHES may be conditional-take-first (BRANCH(F)), conditional-take-all (BRANCH(A)), or probabilistic (BRANCH (P)). The $IDEF_2$ variables which may be used in System Control Network conditional BRANCHING are listed in Figure 3-3. A list of the relational operations which may be used are listed in Figure 3-9.

SUMMARY

System Control Networks model influences that may change the status of manufacturing systems. They permit the analyst to model highly irregular activities based upon the status of related system elements. The flexibility of System Control Networks provides the $IDEF_2$ modeler with powerful techniques for modeling the occurrence of events.

3.1.4 Facility Diagrams

To understand the characteristics of manufacturing systems, modeler's usually find it useful to begin by sketching the system including all of the components which they desire to model. Regardless of the modeling vehicle or language to be used, this graphical representation aids the modeler in identifying the components of the system which need to be represented in the model. Although, this "sketching" step is not usually considered to be part of the actual modeling process, it is a useful aid to the modeler.

The goal of $IDEF_2$ is to provide a vehicle for describing and modeling ICAM systems. Since the construction of a graphic representation of the ICAM system will often times be done by the modeler anyway, it is appropriate to consider it as part of the actual modeling process and give the modeler the tools to make his modeling, communication, and documentation job easier. Thus, the construction of a Facility Diagram is one step toward building an $IDEF_2$ model of ICAM system dynamics.

Within $IDEF_2$, the objectives for describing the facility are threefold:

1. To provide a framework by which the modeler can relate to the system.
2. To provide a convenient means of specifying the relative locations of system components.
3. To provide a means for specifying the characteristics of the components of the system.

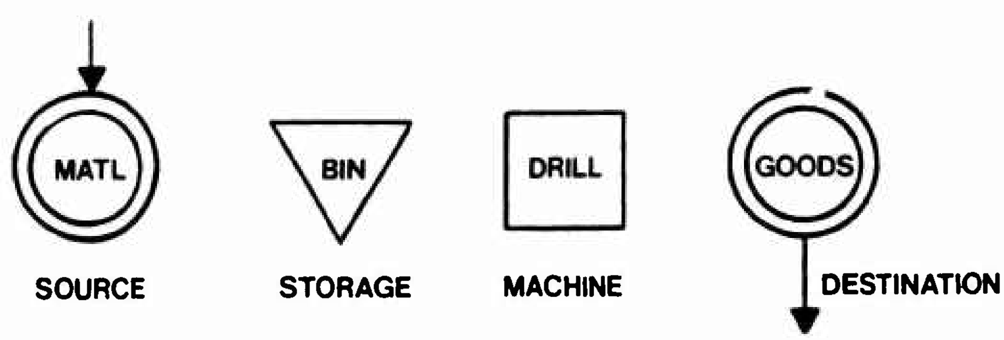
A diagram of the system to be modeled accomplishes the first two of these objectives, but not the third. To provide a means for specifying the characteristics of the components of the system, $IDEF_2$ uses a Facility Component Attribute Definition Form. The form provides a structured means of specifying the characteristics of facility components, and thus accomplishes the third objective of a facility description.

For many types of manufacturing systems there exist symbols commonly used for drawing diagrams of the systems. Among these are production systems and computer systems. $IDEF_2$ symbol sets have been developed for production and computer systems which follow the commonly used industry-wide symbol set. However, for many other types of manufacturing systems, no industry-wide standard symbol set exists. Each company or plant within a company uses its own particular symbol set. Because the purpose of the Facility Diagram is to communicate the components and their relative locations, the standard $IDEF_2$ Facility symbol set should be used for production and computer systems. When these symbol sets are not applicable the modeler is allowed to use the symbol set with which he is most familiar for the system being modeled. In this case, all symbols used in the Facility Diagram must be documented and explained.

In most industrial plants, facility layouts are constructed at one time or another, but no standard methods are available. Many plants use different types of layout techniques. However, the emphasis when constructing a layout is usually on identifying the flow of materials through the facility. One common type of layout diagram used is a flow planning diagram. An example of a flow planning diagram is shown in Figure 3-17. Since flow planning diagrams accomplish the first two objectives of the Facility Diagram, a modified version of the flow planning diagram has been adopted for Facility Diagrams of production systems.

As with production systems there are no industry standards for a computer hardware symbol set, however, the symbol set commonly used are similar. The $IDEF_2$ symbol set for computer systems utilizes the common elements of the frequently used computer system symbol sets. This symbol set should be used when constructing $IDEF_2$ Facility Diagrams of computer systems. If additional symbols are used in Facility Diagrams of computer systems, they must be documented and explained.

The symbology associated with Facility Diagrams is generic in nature and contains elements that will be familiar to most $IDEF_2$ modelers. Figure 3-18 lists each of the standard Facility Diagram components along with its symbolic representation. As an introduction to the use of Facility Diagram components, consider the example shown below.



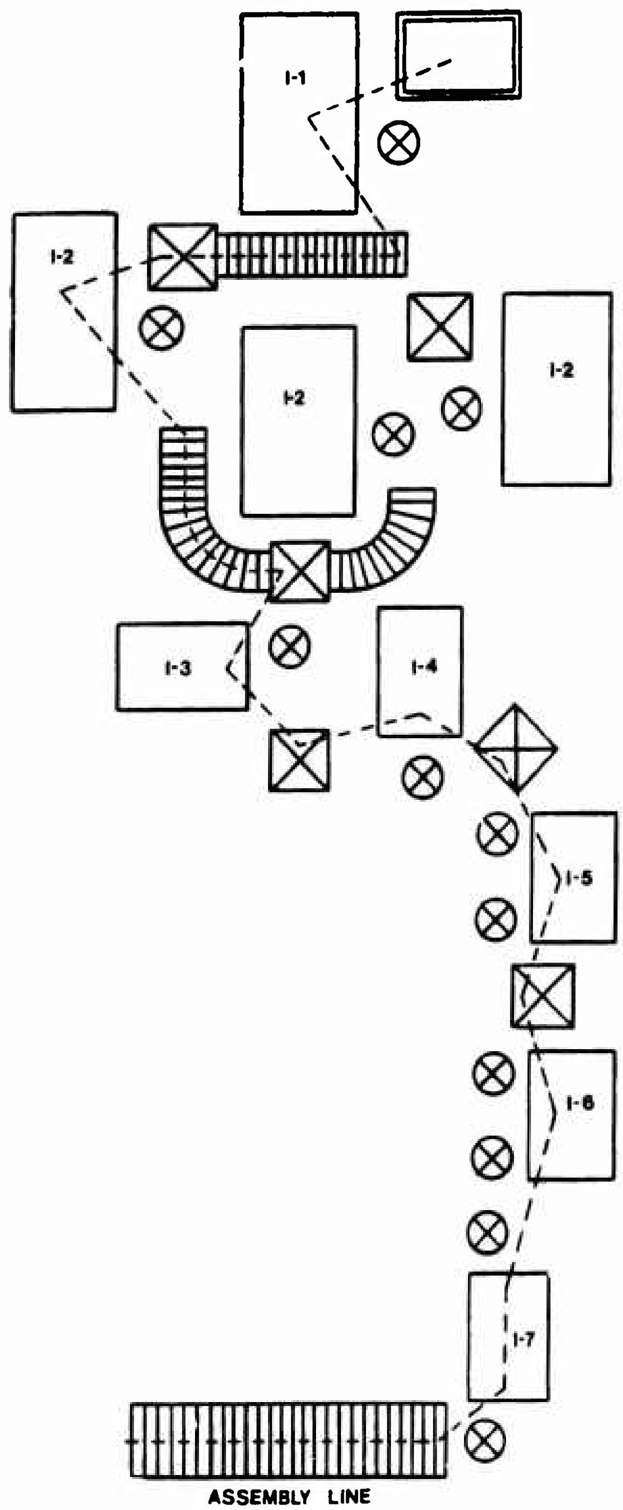


Figure 3-17. An Example of a Flow Planning Diagram used for Plant Layout

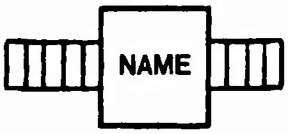
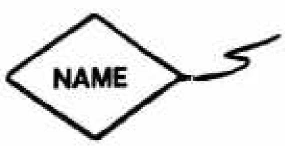
SYMBOL	*COMPONENT*	*DESCRIPTION*
	Machine	Any device which performs activities on entities
	Manpower	Any type of personnel including operators, repairmen, foremen, and managers.
	Material Handling Equipment	Stand alone devices
	Material Handling Equipment	Devices that move on tracks
	Material Handling Equipment	Fixed path devices
	Storage	Racks, totes, bins, etc.
	Inspection Station	Any location where entities are inspected.
	Computer	Any computer which controls other devices in a manufacturing environment.
	Scrap Production	Any area where scrap is produced.
	Source	Material source.

Figure 3-18. Components of Facility Diagrams

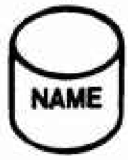
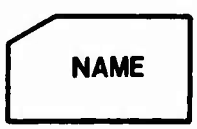
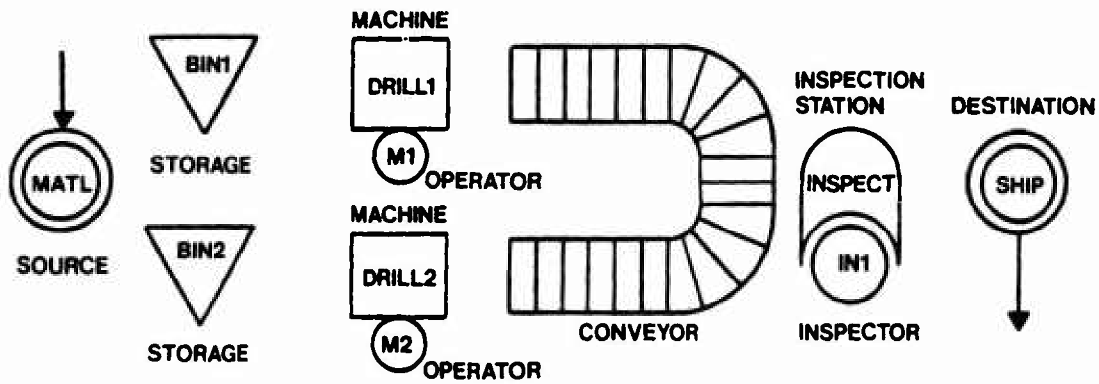
SYMBOL	*COMPONENT*	*DESCRIPTION*
	Destination	End product destination.
	Memory	Mass storage device.
	Punched Tape	Any punched tape used or produced by a computer system.
	Magnetic Tape	Any magnetic tape used by a computer system.
	Printed Output	Any hard copy report used or generated.
	Terminal	CRT, teletype, etc.
	Communication Link	Any data line or path over which data may be transmitted.
	CPU	Central Processing Unit
	Input	Any source of input to a computer system.

Figure 3-18. Components of Facility Diagrams (Continued)

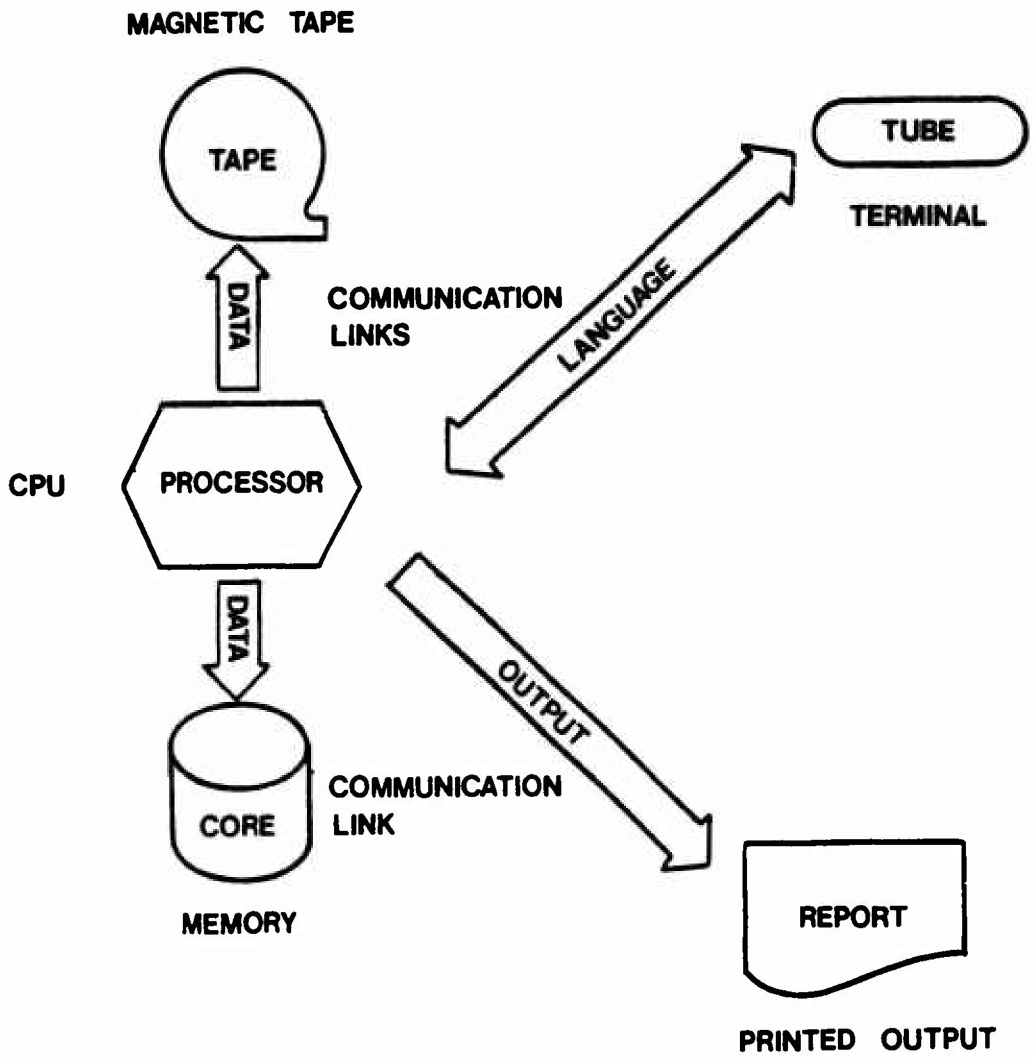
Here we have represented a simple system in which raw materials enter the system to be stored in a bin. The bin feeds materials to the drilling operation, after which completed parts leave the system. This Facility Diagram is quite simple but it represents the basic elements of the real system. A slightly more complex system would be represented with the following Facility Diagram.



In this system, raw material enters the system from the left as indicated by the SOURCE element. Upon entry to the system, it is stored in two bins to await processing by either of two drills. A human operator performs the drilling operation at each drilling station. Parts are then rolled to the inspection station on a curved roller conveyor. At the inspection station, a human inspector inspects all parts before sending them to shipping.

From these two examples, it is clear that Facility Diagrams can be used to represent a variety of production systems. Odd-shaped devices such as the C shaped roller conveyor are easily represented by altering the shape of the basic material handling track device symbol to suit the user's particular needs. In addition, the flexible design of Facility Diagrams does not prohibit the use of symbols not shown in Figure 3-18.

Facility Diagrams also can be used to represent computer systems. Several symbols have been incorporated into the Facility Diagram symbology which are specially suited to computer systems. Consider the sample Facility Diagram for the computing system shown on the following page.



In this system, users access the computer with a terminal, labeled TUBE. The terminal allows a user to communicate with the computer through a communication link, which is labeled LANGUAGE. This link is shown to be bi-direct onal to represent interaction between the central processor and the termi al. CORE storage is used in the system as well as a magnetic tape drive for data files. These devices are linked to the CPU via data lines, represented with communication links labeled DATA. Finally, the CPU sends messages to a hard copy device which prints an output report. The device that produces the reports is represented by a PRINTED OUTPUT symbol.

Facility Diagrams model the components of manufacturing systems. It is important to recognize that Facility Diagrams do not explicitly show material flow. The arrangement of physical elements may imply the direction of flow, but Facility Diagrams do not model the actual flow of entities. The task of modeling explicit entity flow is accomplished with Entity Flow Networks.

Summary

Facility Diagrams can help initiate the modeling process by placing boundaries on the system to be modeled. The elements included in a Facility Diagram define the scope of a model by explicitly defining the system being modeled. By defining the model's scope and specifying the system elements to be included in the model, Facility Diagrams offer $IDEF_2$ modelers an excellent start in the modeling process.

3.2 Supporting Documents

In order to understand and analyze the dynamic behavior of a complex manufacturing system, we must characterize elements of the system and portray the interactions between the system elements. The graphical portion of $IDEF_2$, i.e., the Entity Flow Networks, Resource Disposition Trees, System Control Networks, and Facility Diagrams provide a visual display of the behavior of the system. For instance, the presence of a QUEUE node indicates that entities are awaiting service by a resource. But important characteristics of the QUEUE such as its maximum size, its ranking priority scheme, and its status at the beginning point in the analysis are not included on the graphical image. For this reason, supporting documents are required on which to specify additional characteristics of the graphical elements on $IDEF_2$ models.

Resource Disposition Trees are self documenting in that all the information required to process the disposition of resources is found on the trees. However, Entity Flow Networks, System Control Networks, and Facility Diagrams contain graphical images which require characterization via supporting documents.

All $IDEF_2$ graphical elements except Resource Disposition Tree elements have attributes for which values must be defined. These required attributes define the basic characteristics of the system elements they represent. Graphical symbols, without characterization of the physical system elements that they portray cannot be used for analysis of performance. These required attributes along with the $IDEF_2$ diagrams complete the requirements for the construction of $IDEF_2$ models. Any required attributes which are not contained on the $IDEF_2$ symbols must be supplied on $IDEF_2$ forms.

Figures 3-19 through 3-21 list the required and optional attributes to be specified for Entity Flow Networks, System Control Networks, and Facility Diagrams, respectively.

In addition to the attributes of the Entity Flow Nodes, System Control Nodes, and Facility Diagram Components, $IDEF_2$ models must describe the attributes of entities which flow through Entity Flow Networks, the initial disposition of entities in Entity Flow Networks, the initial disposition of all resources, and the initial values of any continuously valued attributes. This information is to be provided on Entity Definition forms, Initial Entity Disposition forms, Initial Resource Disposition forms and Initial Continuous System Definition forms, respectively. These forms are included in Section 4 of this document.

COMPONENT	*REQUIRED ATTRIBUTES*	*OPTIONAL ATTRIBUTES*
QUEUE node	1. Identifying name 2. Ranking rule 3. Maximum capicity 4. Initial number in queue	5. Full condition 6. Selector node priority
RESOURCE ALLOCATION symbol	1. Resource name 2. Number of units required	
RESOURCE RELEASE symbol	1. Resource name 2. Number of units to be released	
ASSIGNMENT node	1. Identifying name 2. Variable to which an assignment is to be made 3. Value or expression for assignment . . repeats of 2 and 3 .	
SELECTOR node	1. Identifying name 2. Selection type 3. Selection rule	
BOUNDARY node	1. Identifying name 2. Type of boundary	3. Inventory shrinkage costs 4. Unit cost 5. Unit profit 6. Lost sales cost 7. Inventory carrying cost
ACCUMULATE node	1. Identifying name 2. Number for release 3. Number to release 4. Attribute save criteria	5. Number of entities required for the first node release if it is different from "number for release".
MATCH node	1. Identifying name 2. Matching attribute	
ASSEMBLY node	1. Identifying name 2. Attribute save criteria	
ACTIVITY symbol	1. Identifying name 2. Duration (see Figure 3.22)	3. Priority
BRANCH symbol	1. Branch type 2. Conditional expression or probability	

Figure 3-19. Attribute Specification for Components of Entity Flow Networks

COMPONENT	*REQUIRED ATTRIBUTES*	*OPTIONAL ATTRIBUTES*
CREATE node	1. Identifying name 2. Start node name 3. Time between creations or historical trace 4. Time of first creation 5. Entity type	6. Cost of initial purchase 7. Cost of initial transportation
ALTER node	1. Identifying name 2. Resource type 3. Change in capacity/new quantity	
ASSIGNMENT node	1. Identifying name 2. Variable to which an assignment is to be made 3. Value or expression for assignment · · repeats of 2 and 3 ·	
PREEMPT node	1. Identifying name 2. Resource type 3. Entity destination 4. Resource destination 5. Remaining Processing Time	6. List of activities from which resource can be preempted
ACTIVATE node	1. Identifying name 2. Resource type/SHIFT/CONVEYOR	
DEACTIVATE node	1. Identifying name 2. Resource type/SHIFT/CONVEYOR	
DETECT node	1. Identifying name 2. Attribute name 3. Facility, entity, or resource 4. Direction of crossing (positive, negative) 5. Threshold value 6. Tolerance	
SELECT node	1. Identifying name 2. Activity selection rule	
ACCUMULATE node	1. Identifying name 2. Number for release 3. Number to release 4. Attribute save criteria	4. Number of entities required for the first node release if it is different from attribute 2.
ACTIVITY symbol	1. Identifying name 2. Activity priority 3. Duration (see Figure 3.22)	

Figure 3-20. Attribute Specification for Components of System Control Networks

COMPONENT	*REQUIRED ATTRIBUTES*	*OPTIONAL ATTRIBUTES*
BOUNDARY node	1. Identifying name 2. Type of boundary	
RESOURCE RELEASE symbol	1. Resource name 2. Number of units to be released	
ENTER node	1. Identifying name 2. Time of entry	

Figure 3-20. Attribute Specification for Components of System Control Networks (Continued)

COMPONENT	*REQUIRED ATTRIBUTES*	*OPTIONAL ATTRIBUTES*
MACHINE	1. Identifying name	2. Machine type 3. Capacity 4. Manpower type 5. Location coordinates 6. Cost of utilities per time unit 7. Cost of fuel per time unit 8. Upstream & downstream distances 9. Replacement cost
MANPOWER	1. Identifying name	2. Type 3. Machines he can operate 4. Material handling equipment he can operate 5. Location coordinates 6. Salary 7. Overhead
MATERIAL HANDLING EQUIPMENT	1. Identifying name	2. Type of device 3. Capacity 4. Speed 5. Location coordinates 6. Cost of utilities per time unit 7. Cost of fuel per time unit 8. Replacement cost 9. Upstream & downstream distances
STORAGE	1. Identifying name	2. Type of storage 3. Capacity 4. Location coordinates 5. Cost of storage per time unit
INSPECTION STATION	1. Identifying name	2. Type of inspection station 3. Capacity 4. Type of manpower required 5. Type of entities inspected 6. Location coordinates 7. Probability of finding defective part 8. Probability of finding non-defective part
COMPUTER	1. Identifying name	2. Type 3. Capacity 4. Processing rate 5. Units controlled 6. Location coordinates 7. Cost of operation per time unit

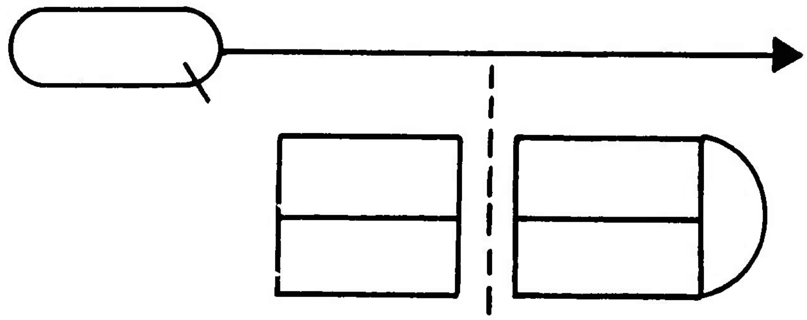
Figure 3-21. Attribute Specifications for Components of Facility Diagrams

COMPONENT	*REQUIRED ATTRIBUTES*	*OPTIONAL ATTRIBUTES*
SCRAP PRODUCTION	1. Identifying name	2. Type of scrap produced 3. Location coordinates 4. Material cost
SOURCE	1. Identifying name	2. Type of material arriving 3. Location coordinates
DESTINATION	1. Identifying name	2. Type of entity leaving 3. Location coordinates
MEMORY	1. Identifying name	2. Type of input 3. Input rate 4. Location coordinates 5. Cost of input per unit
PUNCHED TAPE	1. Identifying name	2. Type of tape 3. Punch speed 4. Read speed 5. Location coordinates 6. Cost per unit of tape
MAGNETIC TAPE	1. Identifying name	2. Type of tape 3. Capacity 4. Writing speed 5. Reading speed 6. Location coordinates 7. Cost per entity stored
PRINTED OUTPUT	1. Identifying name	2. Type of device 3. Size of output 4. Printing speed 5. Location coordinates 6. Cost per unit printed
TERMINAL	1. Identifying name	2. Data transfer rate 3. Location coordinates 4. Cost per unit transferred
COMMUNICATIONS LINK	1. Identifying name	2. Data transfer rate 3. Percent of data lost 4. Location coordinates 5. Cost per unit transferred 6. Power requirements 7. Cost of power
CENTRAL PROCESSING UNIT (CPU)	1. Identifying name	2. Processing rate 3. Polling rate 4. Location coordinates 5. Cost per unit processed per unit time
INPUT	1. Identifying name	2. Type of input 3. Input rate 4. Location coordinates 5. Cost of input per unit

Figure 3-21. Attribute Specifications for Components of Facility Diagrams (Continued)

ACTIVITY DURATION	*EXAMPLE*
A CONSTANT	5.0
A SAMPLE FROM A PROBABILITY DISTRIBUTION	NORMAL(5.2,2.0)
A REFERENCE TO THE RELEASE OF A NODE	REL(node name)
A REFERENCE TO A USER-DEFINED VARIABLE	Variable name
A REFERENCE TO AN ENTITY ATTRIBUTE	A. attribute name
A REFERENCE TO A FACILITY ATTRIBUTE	F. facility component (attribute name)
A VALUE FROM A GRAPH	G. graph name
A VALUE FROM A TABLE	T. table name

Figure 3-22. Activity Duration Specifications Allowed in $IDEF_2$



SECTION 4

READING $IDEF_2$ DIAGRAMS

READING $IDEF_2$ DIAGRAMS

An $IDEF_2$ model is made up of a series of submodels: Facility Submodel, Entity Flow Submodel, Resource Disposition Submodel and System Control Submodel.

When published, an $IDEF_2$ model contains, in sequence, each of the submodels in their entirety. Each submodel is identified in the node field at the bottom of the standard IDEF form, as shown in Figure 4-1.

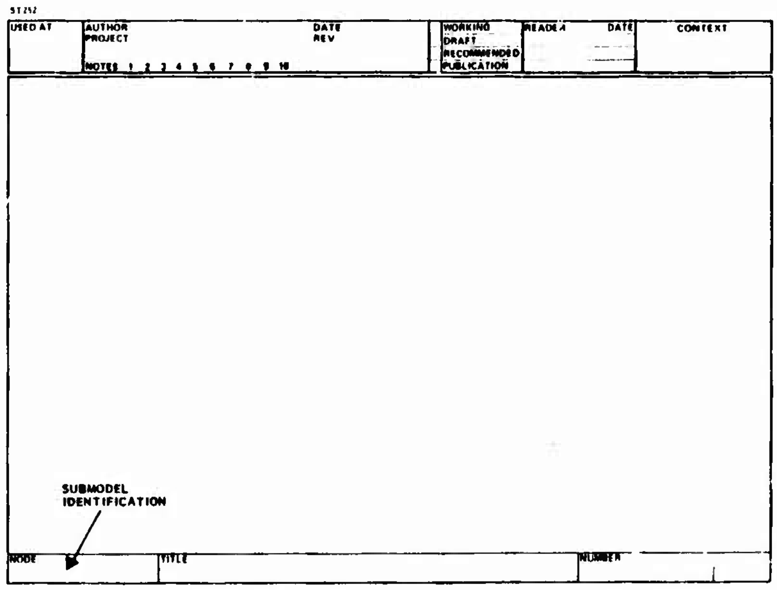


Figure 4-1. IDEF Form

Figure 4-2 shows how the structure of the diagram number conveys the information about the submodel.

Node Number Model Name/Submodel ID/Topic, Page	Submodel
MODEL/FD/Facility Name, i	Facility
MODEL/EN/Entity Name, i	Entity Flow
MODEL/RT/Resource Name, i	Resource Disposition
MODEL/CN/Control Condition, i	System Control

Figure 4-2. Identification of Submodel Information

4.1 Diagram Reading Steps

The precise information about a system is in the diagrams themselves. The following reading sequence is recommended:

1. Check the information given in the node field to determine submodel identity.

2. Review graphic symbols used for the submodel being read and, for the facility submodel, identify any special symbols adapted by the author.

3. Mentally walk through the information given on the diagram. Note if symbols are used correctly. Check to see if the information on the diagram is appropriate to the submodel in which it is contained.

4. Read the supporting documentation accompanying the diagrams.

4.1.2 Semantics of $IDEF_2$

The symbols used in $IDEF_2$ submodels are reviewed in Section 3. If the author of a model introduces special symbols, these should be explained in the documentation accompanying the model.

Each of the submodels of $IDEF_2$ should be read with the objective and scope of the entire model in mind. The patterns of the diagrams can easily lead to a review of details which, whether accurate or inaccurate, are not relevant to the model. Such irrelevance can be spotted and avoided if the habit of checking the objective and scope is maintained.

4.1.2.1 Facility Submodel

When interpreting Facility Submodels, check the following points:

- Are the diagrams accurate?
- Are the facilities shown comprehensive?
- Are their any facilities which were omitted?
- Are there unnecessary details or irrelevant topics?

4.1.2.2 Entity Flow Submodel

When reading an Entity Flow Submodel, check the following information before reading the diagrams:

- Glossary definition of the entity under consideration.
- The extent of the network. Is it complete or do boundary nodes indicate that only part of the flow has been documented?

When reading the diagrams, check the following points:

- Is each operation that must occur shown?
- Are they shown in proper sequence?
- Are there any unnecessary operations included?
- Are all decisions and alternate routes documented by the networks selector nodes?

Other auxiliary questions should be developed based on the objectives of the model being read. Examples of such questions are:

- Are travel time and resources considered?
- Are assembly, disbursing and regrouping of batches displayed?

4.1.2.3 Resource Disposition Submodel

When reading a Resource Disposition Submodel, ascertain the following information before reading:

- Is the tree intended to portray the way resources would be used optimally?
- Is the tree intended to portray actual events?
- Is the tree a representation of one of a number of schemes to be tested by simulation?

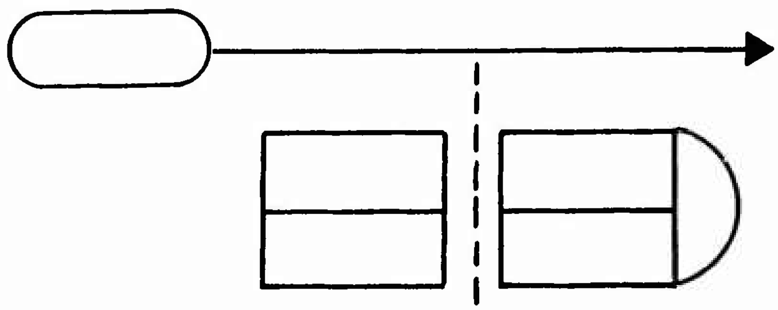
When reading the submodel, check the following points:

- Are the most common cases covered?
- Are the most complex cases covered?
- Are there any cases that have been omitted?

4.1.2.4 System Control Submodel

When reading a System Control Submodel, check the following points:

- Have any sources of an entity been overlooked?
- Have all minor sources (returns, cannibalized parts), relevant to the objective of the model been included?
- Have all general features (seasons, holidays, shifts) relevant to the objective of the model been included?
- Have all detailed features (tool wear, breakdown of repair equipment) relative to the objective of the model been included?
- Have any features which are not relevant to the objective of the model been included?



SECTION 5

IDEF FORMS AND PROCEDURES GUIDE

5.1 IDEF Teamwork Discipline

The development of any IDEF model ($IDEF_0$, $IDEF_1$, and $IDEF_2$) is a dynamic process which requires the participation of more than one person. Throughout a project the draft portions of a model are created by authors and distributed to other project members for review. These draft portions of a model are called Kits and may contain diagrams, text, glossary or any other information the author feels is pertinent to the development of the model.

The IDEF teamwork discipline identifies all persons interested in the review of a model as reviewers. Reviewers who are expected to make a written critique of a Kit are called commenters. Reviewers who receive a Kit for information only, are not expected to make written comments and are called readers.

The discipline requires that each person expected to make comments about a Kit shall make them in writing and submit them to the author of the Kit. The author responds to each commenter in writing on the same copy. This cycle continues, encompassing all Kits pertaining to a particular model, until the model is complete and recommended for publication.

The evaluation of a model is recorded by disseminating a model (with most recent changes) every 3 months in the form of a Kit and sent to readers to assist them in maintaining current information about the model.

The end effect of this process for organized teamwork is a high assurance that final IDEF models are valid and are well expressed. The Kits are changed to reflect corrections and valid comments. More detail is added by the creation of more diagrams, text and glossary. More comments are made; more changes are included. The final model represents the agreement of the author and reviewers on a representation of the system being modeled from a given viewpoint and for a given purpose.

5.2 The IDEF Kit Cycle

In creating a document, materials written or gathered by an author are distributed to commenters in the form of a Standard Kit. Commenters review the material and write comments about it. The commenters return the Kit to the author who reacts to comments and may use the comments to revise or expand the material. The Kit is returned to the commenter with the reactions from the author. This is known as a Kit Cycle. The steps of the Kit Cycle are as follows:

- The author assembles the material to be reviewed into a Standard Kit*. A cover sheet is completed. Copies of the kit are distributed to each of the commenters, and to the author. The original is filed for reference.

- Within the response time specified, the commenter reads the kit and writes comments directly on the copy. The kit is returned to the author.

- The author responds in writing directly on each commenter's copy. The author may agree with the comment, noting it on his working copy, and incorporating it into the next version of the model. If there is disagreement, the author notes the disagreement on the kit and returns it to the commenter.

- The commenter reads the author's responses and, if satisfied, files the kit. (Commented Kits are always retained by the Commenter.) If the commenter does not agree with the author's responses, a meeting is arranged with the author to resolve differences. If this cannot be done, a list of issues is taken to appropriate authority for decision.

This cycle continues until a document is created which represents the careful consideration of all project members. In addition, a complete history of the process has been retained.

The results of this Kit Cycle are a document to which author and commenters have contributed, and, if necessary, a list of issues that require management action.

Throughout the cycle, a project librarian handles copying, distribution, filing, and transfer of kits between authors and commenters.

*Types of IDEF Kits are explained in Section 5.3.

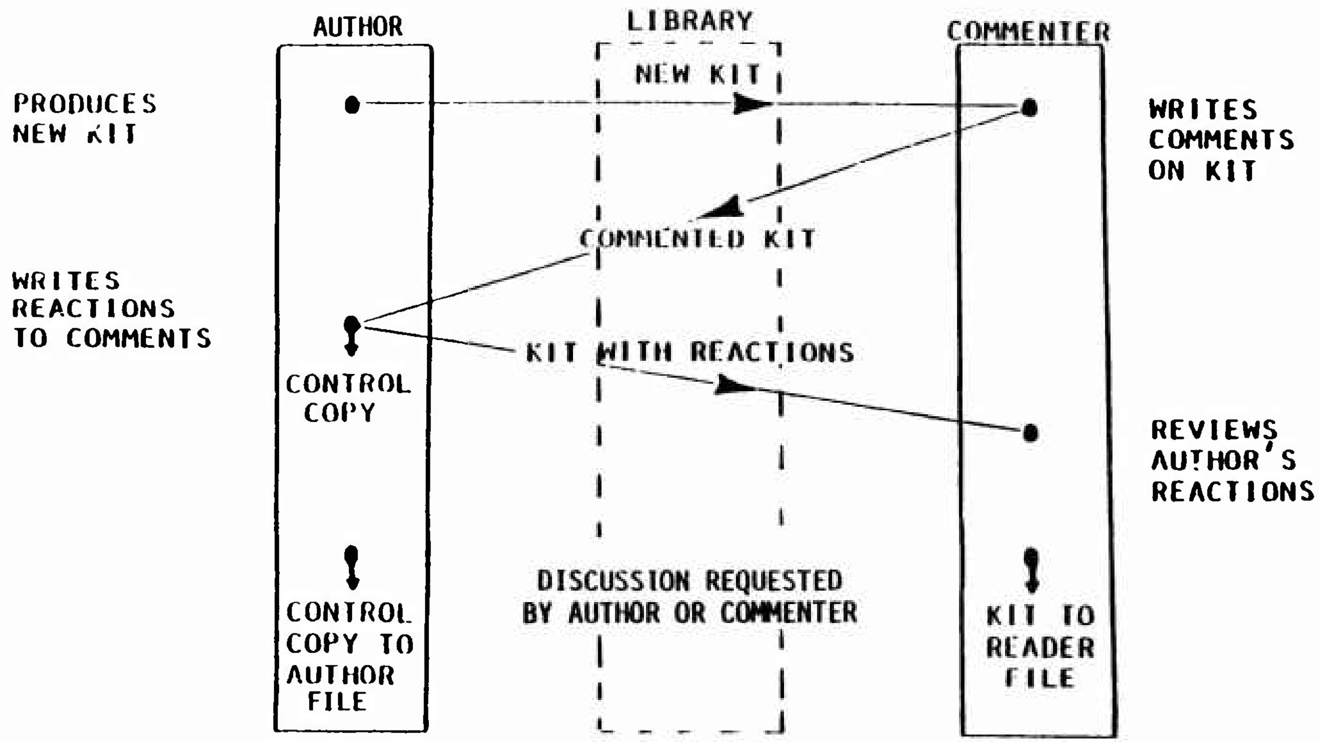


Figure 5-1. Kit Cycle

5.2.1 Personnel Roles

The roles and functions of people involved are:

- **Authors (Modelers)** — People who prepare any IDEF model.
- **Commenters (Experts)** — People knowledgeable of the subject being modeled from whom authors may have obtained information by means of interviews, and have enough training in an IDEF technique to offer structured comments in writing.*
- **Readers (Experts)** — People knowledgeable of the subject being modeled from whom authors may have obtained information by means of interviews, and review documents for information but are not expected to make written comments.
- **Librarian** — A person assigned the responsibility of maintaining a file of documents, making copies, distributing kits and keeping records.

It is important to note that these are generic roles.

A "role" has nothing to do with someone's job title, and the same person may be asked to perform several roles. Thus, each individual's participation is, in fact, unique and depends upon the kit involved.

5.2.1.1 Authors

An author interviews experts and creates documents. However, an author may or may not be the source of the technical content of a document. An author may serve only as a technical writer or scribe to record material gathered from other sources. An author often operates in a role which is largely editorial: identifying, sorting out, and organizing the presentation of knowledge obtained from experts.

---

*Comments between commenter and author are considered privileged information. Commented kits are not duplicated for distribution to anyone else on the program. The library does not retain a file of commented copies.

5.2.1.2 Commenter

Commenters read material produced by authors and verify its technical accuracy. Commenters are responsible for finding errors and suggesting improvements. The role of a commenter is the key to producing high quality results. The commenter determines whether the author has followed the IDEF techniques consistently, whether the viewpoint and purpose have been adhered to and whether errors or oversights exist which should be brought to the author's attention.

5.2.2 Guidelines for Authors and Commenters

5.2.2.1 Commenter Guidelines

No set pattern of questions and rules can be adequate for commenting, since subject matter, style, and technique vary so widely. However, guidelines do exist for improving quality. The major criteria for quality are: Will the document communicate well to its intended audience? Does it accomplish its purpose? Is it factually correct and accurate, given the bounded context? Overall guidelines for commenting are:

- Make notes brief, thorough and specific. As long as the author understands that niceties are dropped for conciseness, this makes for easier communication and less clutter.

- Use the ⓝ notation to identify comments. To write ⓝ-note, check the next number off the NOTES list number the note, circle the number, and connect the note to the appropriate part with a squiggle "↗." (See Section 5.4 Standard Diagram Form)

- Make constructive criticisms. Try to suggest solutions, not just make negative complaints.

- Take time to gather overall comments. These may be placed on the cover or on a separate sheet. (But don't gather specific points onto this sheet when they belong on the individual pages.) Agenda items for author/commenter meetings may be summarized. Make agenda references specific.

The length of time spent critiquing depends on a variety of things: familiarity with what is being described, the number of times something has been reviewed, the experience of the commenter and author, etc. A kit returned to an author with no comments means that the commenter is in total agreement with the author. The commenter should realize that there is a shared responsibility with the author for the quality of the work.

5.2.2.2 Author/Commenter Interchanges

When a commenter returns a kit, the author responds by putting a "√" or "X" by each ⓝ-note. "√" means the author agrees with the commenter and will incorporate the comment into the next version of the kit. "X" means the author disagrees. The author must state why in writing where the comment appears. After the author has responded to all comments, the kit is returned for the commenter to retain.

After reading the author's responses, it is the commenter's responsibility to identify remaining points of disagreement and to request a meeting with the author. This specific list of issues forms the agenda for the meeting.

5.2.2.3 Meeting Rules

Until comments and reactions are on paper, commenters and authors are discouraged from conversing.

When a meeting is required, the procedure is as follows:

1. Each meeting should be limited in length.
2. Each session must start with a specific agenda of topics to be considered and must stick to these topics.
3. Each session should terminate when the participants agree that the level of productivity has dropped and individual efforts would be more rewarding.

4. Each session must end with an agreed list of action items which may include the scheduling of follow-up sessions with specified agendas.

5. In each session, a "scribe" should be designated to take minutes and note actions, decisions, and topics.

6. Serious unresolved differences should be handled professionally, by documenting both sides of the picture.

The result of the meeting should be a written resolution of the issues or a list of issues to be settled by appropriate managerial decision. Resolution can take the form of more study by any of the participants.

5.3 IDEF Kits

A kit is a technical document. It may contain diagrams, text, glossaries, decision summaries, background information, or anything packaged for review and comment.

An appropriate cover sheet distinguishes the material as a kit. The cover sheet has fields for author, date, project, document number, title, status, and notes.

There are two types of IDEF Kits:

- Standard Kit - All kits to be distributed for comment. It is considered a "working paper" to assist the author in refining his total model and is limited to 20 pages.
- Summary Kit - Contains the latest version of a model. It is sent for information only and is designed to aid in maintaining current information about the total model while portions of the model are being processed through the Kit Cycle.

Standard Kits contain portions of a model and are submitted frequently as work progresses. Standard Kits are submitted through the IDEF Kit Cycle for review and are the type referred to in this manual.

Summary Kits are submitted every three months. These kits contain the latest version of the model. Recipients of Summary Kits are not expected to make comments on them although they may choose to do so. Summary Kits are kept by the recipients for their files. A description of Summary Kits is found in the "ICAM Library User's Guide."

5.3.1 Completing a Cover Sheet for a Standard Kit

Complete one cover sheet for each kit submitted. (No reproductions). Fill in the following fields on the Cover Sheet (Figure 5-2).

1. MODEL/DOCUMENT DESCRIPTION:

   Title - Should be descriptive of the kit
   Life Cycle Step - "AS IS" or "TO BE"
   IDEF Method - 0, 1 or 2
   System - Acronym for System or Subsystem
   Distribution Type - Specify if other than Standard Kit Distribution*

2. PROJECT INFORMATION:

   Author - Name of person submitting kit**
   Date - Date sent to Library
   Company - Name of company submitting kit
   A.F. Project No. -
   Task No. -

3. KIT INFORMATION:

   Check Standard Kit, indicate document number assigned by Library if this is a revision to a Standard Kit

4. REVIEW CYCLE:

   To be signed and dated after review by commenter and author.

---

*Types of Distribution available are discussed in Volume XI of this report.

**In cases where a Standard Kit is submitted as a group effort (i.e., task team, committee, or co-authors) one individual from the group assumes responsibilities as "author."

IDEF COVER SHEET FORM

MODEL/DOCUMENT DESCRIPTION	PROJECT INFORMATION	KIT INFORMATION	REVIEW CYCLE
TITLE: Entity Class Definition	AUTHOR: D. Stone DATE:	X STANDARD KIT	REVIEWER DATE
LIFE CYCLE STEP "AS IS"	COMPANY: SofTech	SUMMARY KIT	
IDEF METHOD: 1 SYSTEM: IMCV	AF PROJECT NO. 112 TASK NO. II	SUPERSEDED OR REVISED DOCUMENT NUMBER	AUTHOR DATE
DISTRIBUTION TYPE.			

COPY FOR	REVIEWERS NAME	COMPANY	PROJECT NUMBER	COPY FOR	REVIEWERS NAME	COMPANY	PROJECT NUMBER

LOG

FILE

AUTHOR

KIT CYCLE DATES	
RECEIVED BY LIBRARY	
KIT TO REVIEWER	
COMMENTS DUE BACK TO LIBRARY	
COMMENTS TO AUTHOR	
AUTHOR RESPONSES DUE BACK TO LIBRARY	
AUTHOR RESPONSE TO COMMENTER	
KIT CYCLE COMPLETE	

COPYING INSTRUCTIONS

___ copies of ___ pages ___ total

NODE INDEX/CONTENTS

Pg.	Node	Title	C-Number	Status
1		IDEF COVER SHEET	DSC1094	
2		Node Index/Contents	DSC1116	
3	PI/T3	Intro. to F.V. "FEO"	DSC1114	
4	PI/F1	Part Pro. Req. Cluster	DSC1115	
5	PI/E1	Cost Account	DSC7	
6	PI/E6G1	Part	DSC9	
7	PI/E12G1	Engineering Drawing	DSC16	
8	PI/E13G1	Engineering Spec.	DSC17	
9	PI/E22G1	Purchase Request	DSC26	
10	PI/E23G1	Purchase Req. Item	DSC27	
11	PI/E24G1	Purchase Req. Change	DSC28	
12	PI/E30G1	Approved Supplier	DSC34	
13	PI/E81G1	Contract	DSC187	
14	PI/E106G1	Approved Carrier	DSC547	

COMMENTS/SPECIAL INSTRUCTIONS

Return Kit through ICAM Program Library

C-NUMBER DS 1094 PAGE 1

NOMENCLATURE | DOCUMENT/MODEL TITLE | DOCUMENT NUMBER 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

ST 352 Rev. 8/80

Fig. -2. IDEF Cover Sheet

5. NODE INDEX/CONTENTS:

Node number, title and C-number of each page of the document (including the cover sheet) CONTENTS SHEET, Figure 4-3 (if needed) is always Page 2.

6. COMMENTS/SPECIAL INSTRUCTIONS:

Any other information for the reviewers. This can also be used for special instructions to the library about the handling of the document. The library also uses this field for special instructions to receiver of kits.

5.3.2 How to Prepare a Standard Kit

To avoid oversights, review the kit as if that were the only information available. Catch any typographical errors. Add points of clarification that come to mind as brief notes on the kit itself. Glossary definitions for terms that appear in the kit should always be appended as support material.

On occasion an $IDEF_2$ model author may construct an $IDEF_2$ model which does not conform to all of the $IDEF_2$ modeling procedures. Such models or diagrams are designated as For Exposition Only (FEO) diagrams or models. There are three primary uses of the FEO convention in $IDEF_2$. First, an $IDEF_2$ diagram, network or tree is labeled as an FEO when it does not conform to all $IDEF_2$ model construction rules. Secondly, each page of an $IDEF_2$ submodel which does not contain required supporting forms is designated as an FEO. Thirdly, all pages of a partial $IDEF_2$ model, such as a model consisting only of an Entity Flow Submodel, are designated as FEO's.

Gather helpful materials and append these for the commenter's benefit. Never use this supplemental material to convey information which should properly be conveyed by the diagrams themselves. Whenever possible, use the most natural means of communication - diagrams - to show details that are important for the reader in understanding the concepts. Combine all material with a completed Cover Sheet and Contents Sheet and submit to the Library.

IDEF KIT CONTENTS FORM

DOCUMENT NUMBER	AUTHOR D. Stone	DATE 7/24/80	X STANDARD KIT	REVIEWER	DATE
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	COMPANY SofTech		SUMMARY KIT		
	AF PROJECT NO 112	TASK NO. II	SUPERSEDED OR REVISED DOCUMENT NUMBER	AUTHOR	DATE

Pg.	Node	Title	C-Number	Status
15	PI/E115G1	Department	DSC1095	
16	PI/E116G1	Employee	DSC1096	
17	PI/E120G1	Part Specification	DSC1097	
18	PI/E170G1	Purchase Request Approval Authority	DSC1098	

Pg.	Node	Title	C-Number	Status

NOMENCLATURE	TITLE	C-NUMBER	PAGE
	NODE INDEX/CONTENTS SHEET	DS 1116	

ST352-1

Figure 5-3. IDEF Node Index/Contents Sheet

5.3.3 $IDEF_2$ Kit Assembly Convention

An $IDEF_2$ Kit contains submodels which should be assembled and numbered consistently.

A typical $IDEF_2$ Kit contains:

1. Cover Sheet
2. A-0 Function Model of the system to be modeled, including modeling purpose and the modeler's viewpoint.
3. Glossary
4. Text
5. User Definitions (variables, graphs, tables or algorithms)
6. Facility Diagram(s)
7. Facility Forms
8. Entity Flow Network(s)
9. Entity Flow Forms
10. Resource Disposition Tree(s)
11. Initial Resource Disposition Forms
12. System Control Network(s)
13. System Control Forms

- Cover Sheet

  The cover sheet indicates who developed the kit, who will review the kit, what information is contained in the kit as well as kit distribution information

- A-0 Function Model

  The A-0 function model will provide the initial high level definition of the system whose dynamics are modeled in the $IDEF_2$ kit. This diagram should contain the purpose of the $IDEF_2$ model as well as the viewpoint of the modeler. The A-0 function model is drawn on a standard IDEF diagram form.

- Glossary

All $IDEF_2$ models should contain a glossary section in which key terms relating to the model are defined. This will aid readers and commenters in understanding models which describe unfamiliar areas of manufacturing. In addition, it clarifies the use of unfamiliar terms or terms which people may use differently.

- Text

$IDEF_2$ models also contain a textual description of the model which describes the overall operation of the system modeled in $IDEF_2$. This textual description aids the readers and commenters in understanding the overall operation of the system and explains any complex situations described in the model.

- User Defined Variables

$IDEF_2$ is very flexible in allowing model authors to use the variable and attribute names of their choice. However, this flexibility will result in confusion if these variables and their allowable values are not clearly defined for readers and commenters. Therefore, any user-defined variables, entity attributes, facility attributes, graphs, tables or algorithms used in the model are defined in this section.

- Facility Submodel

The Facility Submodel appears next in an $IDEF_2$ kit. The Facility Submodel consists of the graphic Facility Diagram and forms containing additional characteristics about the facility components. The Facility Diagram is drawn on standard IDEF diagram forms and the additional characteristics are provided on Facility Component Attribute Definition forms. Brief explanatory text may be included on a Facility Diagram to enhance or clarify the $IDEF_2$ model. When developing $IDEF_2$ kits, it is important that the diagrams and forms be numbered consistently so that readers will be able to follow and understand the models. Facility Diagrams are numbered according to the following convention: "model name/FD/facility name, i" where "model name" is a one word name for the entire $IDEF_2$ model, "FD" stands for Facility Diagram, "facility name" is a one word name of the facility on the form and "i" is the page number of the Facility Diagram. The same "name" is used throughout an $IDEF_2$ model. Examples of Facility Diagram numbering are SMC/FD/RIVITING,1,SMC/FD/RIVITING,2, ... etc. The title of each page of the Facility Diagram should be a unique title which includes the name of the facility. Facility Diagram titles are entered in the "TITLE" box on the standard IDEF form. Examples of Facility Diagram titles are "PREPARATION AREA OF PLANT 1" or "ASSEMBLY AREA OF PLANT1." The additional characteristics of the facility components are provided on Facility Component Attribute Definition Forms. An example of a Facility Component Attribute Definition Form is provided in Figure 5-4. The numbering convention for Facility Component Attribute Definition Forms is "model name/FA/facility name,i,j" where "model name" is a one word name for the entire $IDEF_2$ model, "FA" stands for Facility Attribute, "facility name" is a one word name of the facility being modeled, "i" is the page number of the corresponding Facility Diagram, and "j" is the number of the Facility Component Attribute Definition Form being defined. Examples of the numbering of Facility Component Attribute Definition Forms are:

SMC/FA/RIVITING,1,1, SMC/FA/RIVITING,1,2,...SMC/FA/RIVITING,1,10,
SMC/FA/RIVITING,2,1, SMC/FA/RIVITING,2,2,...SMC/FA/RIVITING,2,10

...etc. The titles of Facility Component Attribute Definition Forms should correspond to the titles of the Facility Diagrams which they describe. The titles should be entered in the "TITLE" box on the standard IDEF form. Both the facility page numbers and the facility attribute page numbers are written in the lower left hand corner of the appropriate forms in the box labeled "NODE."

- Entity Flow Submodel

The Facility Submodel is followed by the Entity Flow Submodel. The Entity Flow Submodel consists of graphic Entity Flow Networks and forms containing supporting information regarding entity flow. The Entity Flow Networks are drawn on standard IDEF diagram forms. Brief explanatory text may be included on a page of an Entity Flow Network to enhance or clarify the description of entity flow. The additional supporting information consists of the definition of the $IDEF_2$ entities, the initial disposition of $IDEF_2$ entities, and information regarding the nodes of the Entity Flow Networks. This additional supporting information is provided on Entity Definition Forms, Initial Entity Disposition Forms and Entity Flow Node Attribute Definition Forms, respectively. Examples of these forms are shown in Figures 5-5, 5-6, and 5-7.

USED AT	AUTHOR PROJECT NOTES 1 2 3 4 5 6 7 8 9 10	DATE REV	WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT

FACILITY COMPONENT ATTRIBUTE DEFINITION

FACILITY COMPONENT	IDENTIFYING NAME	NUMBER AVAILABLE	ATTRIBUTES	
			IDENTIFIER	VALUE

NODE	TITLE	NUMBER

Figure 5-4. Facility Component Attribute Definition Form

USED AT	AUTHOR PROJECT NOTES 1 2 3 4 5 6 7 8 9 10	DATE REV	WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT

ENTITY DEFINITION

ENTITY TYPE	ATTRIBUTES	
	IDENTIFIER	VALUE AT CREATION

NODE	TITLE	NUMBER

Figure 5-5. Entity Definition Form

USED AT	AUTHOR PROJECT NOTES 1 2 3 4 5 6 7 8 9 10	DATE REV	WORKING DRAFT RECOMMENDED PUBLICATION	READER	DATE	CONTEXT

INITIAL ENTITY DISPOSITION

ENTITY TYPE	ATTRIBUTES		NUMBER OF UNITS	RESOURCES		ACCUMULATE, QUEUE OR ACTIVITY NAME	REMAINING ACTIVITY TIME
	IDENTIFIER	VALUE AT CREATION		TYPE	NUMBER		

NODE	TITLE	NUMBER

Figure 5-6. Initial Entity Disposition Form

USED AT	AUTHOR PROJECT NOTES 1 2 3 4 5 6 7 8 9 10	DATE REV	WORKING DRAFT RECOMMENDED PUBLICATION	READER	DATE	CONTEXT

ENTITY FLOW NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES	
		IDENTIFIER	VALUE

NODE	TITLE	NUMBER

Figure 5-7. Entity Flow Node Attribute Definition Form

As with Facility Submodels it is important that Entity Flow Submodels be numbered consistently. Pages of Entity Flow Networks are numbered according to the following convention: "model name/EN/entity name,i", where "model name" is a one word name of the entire $IDEF_2$ model, "EN" stands for Entity Network, "entity name" is a one word name of the entity flowing through the network and "i" is the page number of the Entity Flow Network. Examples of Entity Flow Network numbering are: SMC/EN/PART,1, AMC/EN/PART,2,...SMC/EN/PART,10. Each page of an Entity Flow Network should be given a unique title which includes the name of the entity and the type of processing it is undergoing in the network. An example of an Entity Flow Network title is "SETUP AND ASSEMBLY OF WINGS." The numbering of the forms associated with entity flow is similar. For Entity Definition Forms the convention is "model name/ED/entity name,i," where "model name" is the name of the entire $IDEF_2$ model, "ED" stands for Entity Definition, "entity name" is a one word name of the entity flowing through the network model and "i" is the page number of the Entity Definition Form. Examples of Entity Definition Form numbering are: SMC/ED/PART,1, SMC/ED/PART,2... SMC/ED/PART,10. Titles of Entity Definition Forms should read "DEFINITION OF ENTITY" followed by the name of the entity being defined. For Initial Entity Disposition Forms the numbering convention is "model name/EI/entity name,i" where "model name" is the name of the entire $IDEF_2$ model, "EI" stands for Entity Initialization, "entity name" is a one word name of the entity flowing through the network and "i" is the page number of the Initial Entity Disposition Form. Titles of Initial Entity Disposition Forms should read "INITIAL DISPOSITION OF" followed by the name of the entity whose disposition is being described. Examples of Initial Entity Disposition Form numbering are SMC/EI/PART,1, SMC/EI/PART,2,... SMC/EI/PART,10. For Entity Flow Node Attribute Definition Forms the numbering convention is "model name/EA/entity name,i,j" where "model name" is the name of the entire $IDEF_2$ model, "EA" stands for Entity Attribute, "entity name" is a one word name of the entity flowing through the network, "i" refers to the Entity Flow Diagram for which additional information is being described, and "j" refers to the page number of the Entity Flow Node Attribute Definition Form. Entity Flow Node Attribute Definition Form titles should correspond to the titles of the Entity Flow Networks they are describing. If the same title applies to multiple forms "CONTINUED" should be used on all titles subsequent to the first. Examples of Entity Flow Attribute Definition Form numbering are:

SMC/EA/PART,1,1,SMC/EA/PART,1,2,...SMC/EA/PART,1,10, SMC/EA/PART,2,1,SMC/EA/PART,2,2,...SMC/EA/PART,2,10, ...etc.
All page numbers are written in the lower left corner of the forms in the box labeled "NODE." All titles are written in the "TITLE" box at the bottom of the form.

- Resource Disposition Submodel

The Entity Flow Submodel is followed by the Resource Disposition Submodel. The Resource Disposition Submodel consists of a set of Resource Disposition Trees and forms containing information about the initial disposition of resources. The Resource Disposition Trees are drawn on standard IDEF forms and the initial disposition of resources is specified on Initial Resource Disposition Forms. Brief explanatory text may be included on a page containing a Resource Disposition Tree to enhance or clarify the disposition procedures. The numbering convention for Resource Disposition Trees is "model name/RT/resource name, i" where "model name" is the name of the entire $IDEF_2$ model, "RT" stands for Resource Tree, "resource name" is a one word name of the resource whose disposition is being described and "i" is the page number of the Resource Disposition Tree. Examples of Resource Disposition Tree numbering are: SMC/RT/OPERATOR,1, SMC/RT/OPERATOR,2,...SMC/RT/OPERATOR,10. Titles of Resource Disposition Trees should read "DISPOSITION PROCEDURES FOR" followed by the name of the resource whose disposition procedures are being described such as "DISPOSITION PROCEDURES FOR OPERATORS." The numbering convention for the Initial Resource Disposition Forms is "model name/RI/resource name,i" where "model name" is the name of the entire $IDEF_2$ model, "RI" stands for Resource Initialization, "resource name" is a one word name of the resource whose initial disposition is being defined and "i" is the number of the Initial Resource Disposition Form. Examples of Initial Resource Disposition Form numbering are: SMC/RI/OPERATOR,1, SMC/RI/OPERATOR,2,...SMC/RI/OPERATOR,10. Titles of Initial Resource Disposition Forms should read "INITIAL DISPOSITION OF" followed by the type of resources whose disposition is being defined such as "INITIAL DISPOSITION OF MILLS" or "INITIAL DISPOSITION OF ALL RESOURCES". Both the Resource Disposition Tree number and the Initial Resource Disposition Form number are written in the lower left corner of their corresponding forms in the box labeled "NODE." The tree and form titles are written in the "TITLE" box at the bottom of their corresponding forms.

USED AT	AUTHOR / PROJECT / NOTES 1 2 3 4 5 6 7 8 9 10	DATE / REV	WORKING / DRAFT / RECOMMENDED / PUBLICATION	READER	DATE	CONTEXT

INITIAL RESOURCE DISPOSITION

TYPE	ATTRIBUTES: IDENTIFIER	ATTRIBUTES: VALUE AT CREATION	NUMBER OF UNITS AVAILABLE	INITIAL DISPOSITION (QUEUE, ACTIVITY, FREE)

NODE	TITLE	NUMBER

Figure 5-8. Initial Resource Disposition Form

- The System Control Submodel follows the Resource Disposition Submodel and completes an $IDEF_2$ kit. A System Control Submodel consists of System Control Networks and additional information about the nodes of the System Control Networks. The System Control Networks are drawn on standard IDEF forms, and the additional information regarding the system control nodes is provided on System Control Node Attribute Definition Forms. Brief explanatory text may be provided on a page of a System Control Network to enhance or clarify the system control function modeled. The numbering convention for System Control Networks is "model name/CN/control condition,i" where "model name" is the name of the entire $IDEF_2$ model, "CN" stands for Control Network and "i" refers to the page number of the System Control Network. In this case "control condition" refers to the control condition modeled by the network such as breakdown, arrival, or shift change. Examples are: SMC/CN/BREAKDOWN,1, SMC/CN/SHIFTCHANGE,2, or SMC/CN/ARRIVAL,3. The title of a System Control Network should be a unique name of the system control function modeled in the network such as "BREAKDOWN OF MILLS." The numbering convention for the System Control Node Attribute Definition Forms is "model name/CA/control

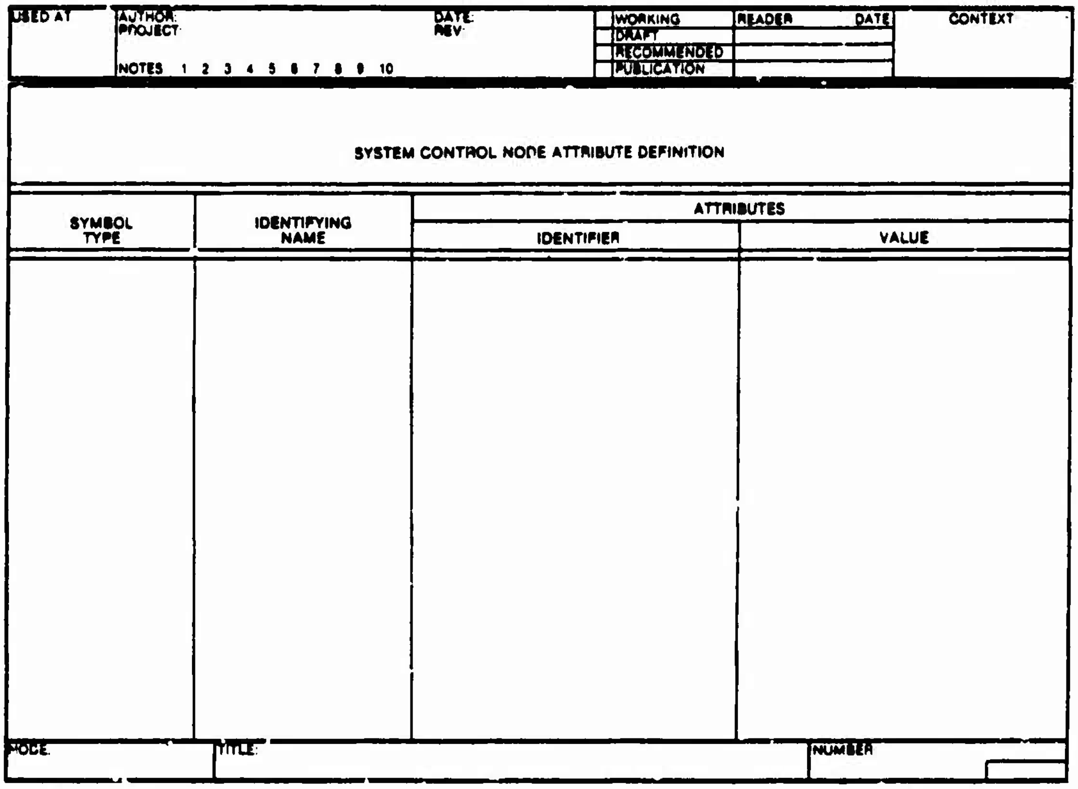


Figure 5-9. System Control Node Attribute Definition Form

condition,i,j" where "model name" is the name of the entire $IDEF_2$ model, "CA" stands for Control Attribute, "i" refers to the number of the corresponding System Control Network, and "j" refers to the page number of the System Control Node Attribute Definition Form. In this case "control condition" refers to the control condition modeled by the corresponding network such as breakdown, shift change, or arrival. Examples of System Control Node Attribute Definition Form numbering are: SMC/CA/BREAKDOWN,1, SMC/CA/ARRIVAL, 2, or SMC/CA/SHIFTCHANGE,3. Titles of System Control Node Attribute Definition Forms should correspond to the titles of the System Control Networks which they describe. Both the System Control Network number and the System Control Node Attribute Definition number are written in the lower left corner of their corresponding forms in the box labeled "NODE."

If any of the attributes of facilities, entities, or resources are defined as continuous functions, the initial values of the continuous functions are defined in the System Control

Submodel on an Initial Continuous System Definition Form. An example of an Initial Continuous System Definition form is shown in Figure 5-10. The numbering convention for these forms is "model name/CS/resource or entity name,i" where "model name" is the name of the entire $IDEF_2$ model. "CS" stands for Continuous System initialization and "i" is the page number of the Initial Continuous System Definition Form. The resource or entity name is the resource or entity for which a continuous attribute is being defined. Examples of Initial Continuous System Definition Form numbering are: SMC/CS/DRILL,1, SMC/CS/PART,1, or SMC/CS/OPERATOR,3. The titles of Initial Continuous System Definition Forms should indicate that the form contains the initial definition of variables and the type of variables being defined such as "INITIAL DEFINITION OF TOOL WEAR" or "INITIAL DEFINITION OF ALL CONTINUOUS VARIABLES." All page numbers are written in the lower left corner of the forms in the box labeled "NODE." All titles are written in the "TITLE" box at the bottom of the form.

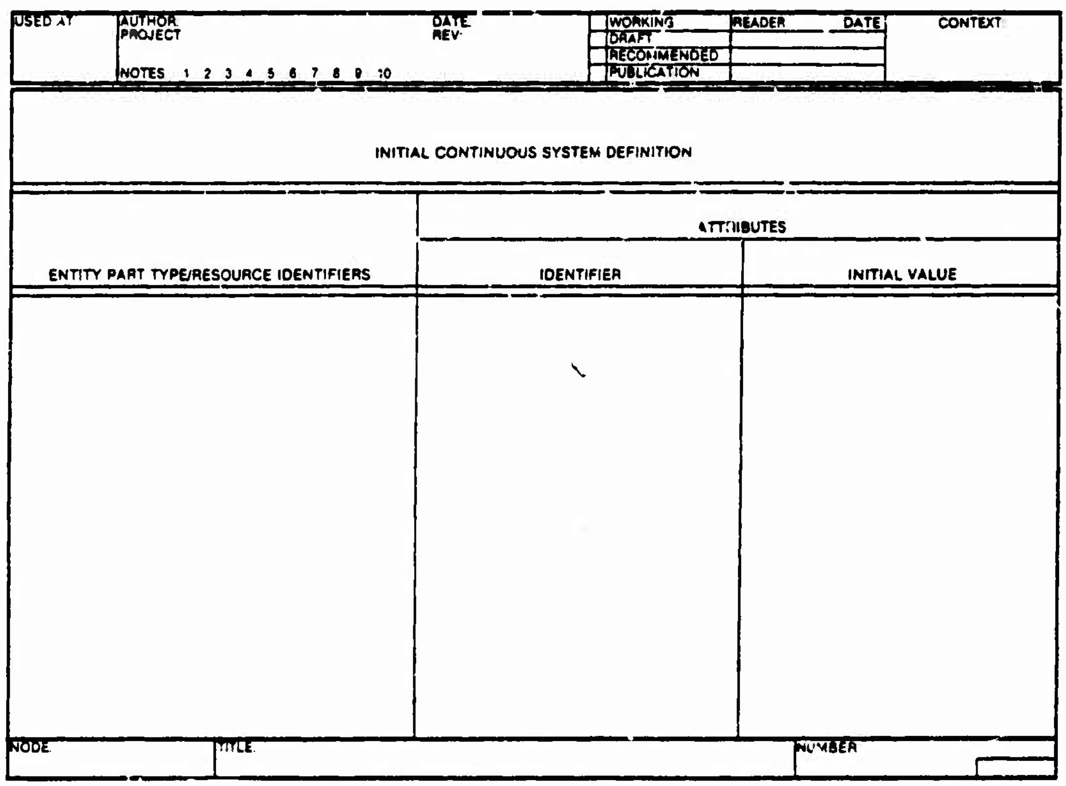


Figure 5-10. Initial Continuous System Definition Form

5.4 Standard Diagram Form

The Diagram Form (Figure 5-11) has minimum structure and constraints. The sheet supports only the functions important to the discipline of structured analysis. They are:

- Establishment of context;
- Cross-referencing between pieces of paper;
- Notes about the content of each sheet.

The diagram form is a single standard size for ease of filing and copying. The form is divided into three major sections:

- Working Information (top)
- Message Field (center)
- Identification Fields (bottom)

The form is designed so that the working information at the top of the form may be cut off when a final "approved for publication" version is completed. The diagram form should be used for everything written.

5.4.1 Working Information

The "Author/Date/Project" Field

This tells who originally created the diagram, the date that it was first drawn, and the project title under which it was created. The "date" field may contain additional dates, written below the original date. These dates represent revisions to the original sheet. If a sheet is re-released without any change, then no revision date is added.

The "Notes" Field

This provides a check-off for ⓝ notes written on the diagram sheet. As comments are made on a page, the notes are successively crossed out. The crossing out provides a quick check for the number of comments, while the circled number provides a unique reference to the specific comment.

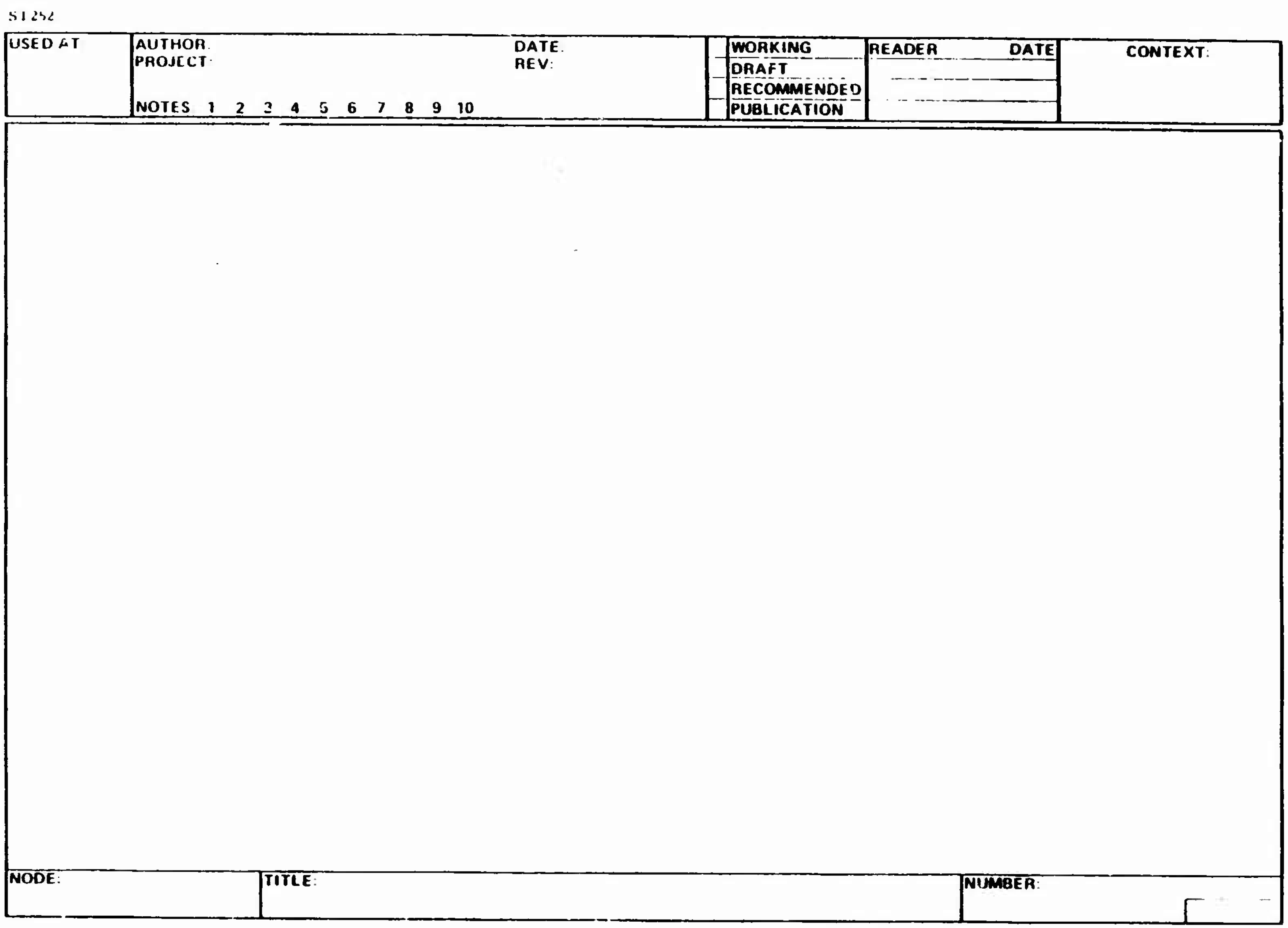


Figure 5-11. Standard Diagram Form

The "Status" Field

The status classifications provide a ranking of approval. They are:

WORKING	The diagram is a major change, regardless of the previous status. New diagrams are, of course, working copy.
DRAFT	The diagram is a minor change from the previous diagram, and has reached some agreed-upon level of acceptance by a set of readers. Draft diagrams are those proposed by a task leader, but not yet accepted by a review meeting of the technical committee or coalition.
RECOMMENDED	Both this diagram and its supporting text have been reviewed and approved by a meeting of the technical committee or coalition, and this diagram is not expected to change.
PUBLICATION	This page may be forwarded as is for final printing and publications.

The "Reader/Date" Field

This area is where a commenter should initial and date each form.

The "Context" Field

The Context field is not used in $IDEF_2$.

The "Used At" Field

This is a list of diagrams , other than the immediate context, which use this sheet in some way.

5.4.2 The "Message" Field

The Message field contains the primary message to be conveyed. The field is normally used for diagramming. However, the field can be used for any purpose: glossary, checklists, notes, sketches, etc. The author should use no paper other than diagram forms.

5.4.3 The "Title" Field

The Title field contains the name of the material presented on the Diagram Form.

5.4.4 The "Number" Field

This field contains all numbers by which this sheet may be referenced.

C-Number

The C-number is composed of two or three letters of the author's initials followed by a number sequentially assigned by the author. This C-number is placed in the lower left corner of the Number field and is the primary means of reference to a sheet. Every diagram form used by an author receives a unique C-number. When a model is published, the C-number may be replaced by a standard sequential page number (e.g., "pg. 17").

Page Number

A kit page number is written by the librarian at the right hand side of the Number Field. This is composed of the document number followed by a number identifying the sheet within the document.

5.5 Keeping Files

Each person participating in a project should maintain files of the documents received. The librarian maintains the master and reference files for each kit submitted during the course of a project. A complete explanation of library files is given in the "ICAM Program Library Maintenance Procedures," Volume XI of this report.

Variations in the filing process may occur based on individual preferences but it is recommended that these files be maintained:

- Standard Kit Files, maintained by authors and commenters
- Summary Kit, maintained by authors, commenters, and readers
- Working Files , maintained by authors

5.5.1 Standard Kit File

This file contains the Standard Kits issued or received. A record of kits filed should be maintained and should include any information that allows convenient access to the contents of the kit.

5.5.2 Summary of Kit File

This file contains the Summary Kits issues or received. A record of these kits should also be maintained.

5.5.3 Working File

This file contains all documentation that has not been submitted in a kit. Work in progress and notes should be kept in this file.

5.6 The IDEF Model Walk-Through Procedures

In addition to the Kit Cycle, a Walk-Through Procedure has been developed. This procedure may be used when the participants in building a model can be assembled for commenting.

1. Present the model to be analyzed by listing the extent of each submodel.

2. Present a Glossary of Terms. This will allow each reviewer to replace his own meanings of words and those that the presenting team has chosen. The meanings should not be questioned at this point. A change in meaning now would require many changes in the diagrams.

The diagram walk-through process is an orderly, step-by-step process where questions can be asked that may identify potential weaknesses in the diagram. Six steps of a structured walk-through follow below.

Diagram corrections may be proposed at any step. These corrections may be noted for execution at a later date or adopted immediately.

Step 1: SCAN THE ENTITY FLOW NETWORK DIAGRAMS

This step allows the reader to obtain general impressions about the context of the diagram. Typically, the reader will have reviewed the procedures or processes that the entity flow network diagrams are portraying.

CRITERIA FOR ACCEPTANCE:

1. The entity flow network diagrams reflect the procedures or processes being modeled.

2. Ensure that the diagrams conform to the $IDEF_2$ syntax and that special symbols and deviations from syntax are explained.

Step 2: SCAN THE SYSTEM CONTROL SUBMODEL

This model will portray the activities or conditions that may affect the resources available to the process being modeled and the entry of entities into the process.

CRITERIA FOR ACCEPTANCE:

1. Review any listed assumptions and make a judgement as to their validity.

2. Ensure that all entities the entity flow network diagrams reflect have a "Create Entity" System Control Model.

Step 3: SCAN THE RESOURCE DISPOSITION SUBMODEL

This model attempts to describe the required resources, initial dispositions and actions taken upon them as they become available or are released for redisposition.

CRITERIA FOR ACCEPTANCE:

1. Ensure that all pertinent resources have been included.

2. Ensure that the resource is cross-referenced to one or more activities in the entity flow network.

3. Ensure that the resource is cross-referenced to a facility unit depicted in the facility submodel.

4. Review the Resource Disposition Tree to ensure that the decision points are logically correct.

Step 4 : SCAN THE FACILITY SUBMODEL

This model will describe graphically and textually the operational arrangement of the facility components being modeled.

CRITERIA FOR ACCEPTANCE:

1. Ensure that the activities modeled in the entity flow network model are reflected in the facility submodel.

2. Ensure that the resources depicted in the resource disposition submodel are reflected in the facility where they are actually located.

3. Ensure that the entities the facility processes are depicted.

Step 5: READ THE SUPPORTIVE DOCUMENTATION

This step examines any documentation the author has included for clarity or to support the process modeled. This documentation is not required for $IDEF_2$ but will usually aid the reviewer in his examination of the model.

CRITERIA FOR ACCEPTANCE:

1. The documentation is cross-referenced to the model to which it relates.

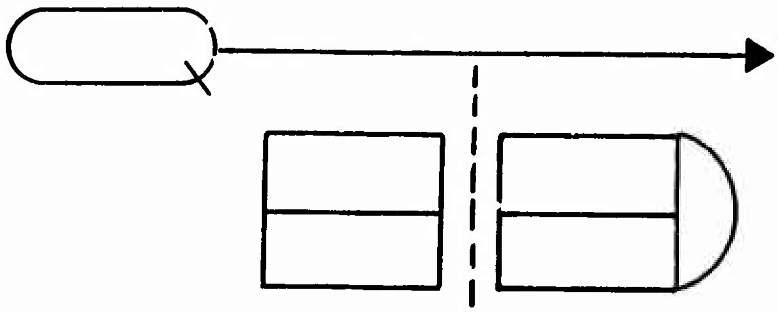
Step 6: SET THE STATUS OF THE DIAGRAM

1. Recommended as it stands.

2. Recommended as modified.

3. Draft: Too many changes made, a redraw is necessary, and future review is required.

4. Not Accepted: A complete re-analysis is required.



SECTION 6

AUTHOR'S GUIDE TO CREATING $IDEF_2$ DIAGRAMS

AUTHOR'S GUIDE TO CREATING IDEF2 DIAGRAMS

The reader has now been exposed to all of the basic concepts of $IDEF_2$. It is now appropriate to present an example. It is assumed that the reader is now familiar with the $IDEF_2$ syntax so we will begin with the statement of the problem. The purpose in presenting the example in this manner is twofold. First, to reinforce the reader's understanding of $IDEF_2$ concepts, and second to illustrate an approach to solving a problem using $IDEF_2$. By stating the problem first and then walking through the model construction, the type of problem $IDEF_2$ can be used to address will be illustrated, as well as the use of $IDEF_2$ as a problem solving aid.

6.1 The ABC Manufacturing Company

The ABC Manufacturing Company produces a number of steel products. Each product undergoes a multi-stage process, and typically requires three to four machining operations. In the past ABC has managed to fill the majority of its customer's orders in a timely fashion without experiencing excessive in-process inventory levels. Recently however, the production control department of ABC has experienced some problem in estimating the time to fill orders. In fact, they have experienced both excessive in-process inventories and delays in the projected completion time of customer orders. The surplus in-process inventories have forced occasional subcontracting to external job shops.

The problem, in the opinion of the production control manager, is that he just does not know the true capacity of their manufacturing facilities. The problem has not been apparent until now because ABC has always had the machine capacity to fill customer orders. However, the recent inventory problem experienced and the projected increase in product demand has prompted the production control manager to request a tool to aid him in estimating the capacity of the system. The tool must consider enough detail to provide information concerning in-process inventories and machine utilization, but it must be flexible enough to permit changes in system configurations so that alternative system designs can be evaluated.

The management of ABC has decided that a model of the manufacturing facility for one part type should be built in order to test the usefulness of the modeling technique before applying it to the entire ABC facility. They feel that modeling the system is the best approach to the problem because they can build a model without disturbing the actual system, and test alternative configurations of the system which might alleviate the problems being experienced without incurring production delays due to the system changes. $IDEF_2$ has been selected as the modeling vehicle because the problem involves understanding the dynamic behavior of the system i.e., the interactions between elements of the system with the passage of time.

6.2 The System Description

In order to test the utility of the $IDEF_2$ modeling methodology, the ABC Company decided to model a fairly simple segment of their facility. With the objective of determining the capacity of this system, the project team assigned to this task has already performed the initial data collection. From this initial data gathering they have learned that the system produces a single part type, which they denoted PART1. Included in the system are two saws, two drills, a boring machine, and five machine operators, each dedicated to a station. Also, two repairmen are present to service machines that break down. Material storage is allocated in front of each machine station. Parts are fed to each station by hand from bins except the boring operation which is fed parts from drilling stations via chutes. Materials are initially stored on racks; the operators who perform the operations are responsible for pulling totes of the material off the racks and bringing the material to their stations on roller conveyors. Parts are stored in bins between operations.

The part that is processed in this system requires three operations: a sawing operation, a drilling operation, and a boring operation. In order to perform a machining operation, the machine must be operative and the associated human operator must be present. Machines that break down must wait to be repaired by a repairman.

Parts arrive to the system in batches and remain in batches of five parts throughout the system. The system operates for a single shift per day. All personnel take a one-half hour lunch break four hours into the shift. When a resource becomes available by finishing work on a job, it is allocated to another job if all resources required by the waiting job are free; otherwise the available resource becomes idle. A schematic drawing of the ABC manufacturing system is shown in Figure 6-1.

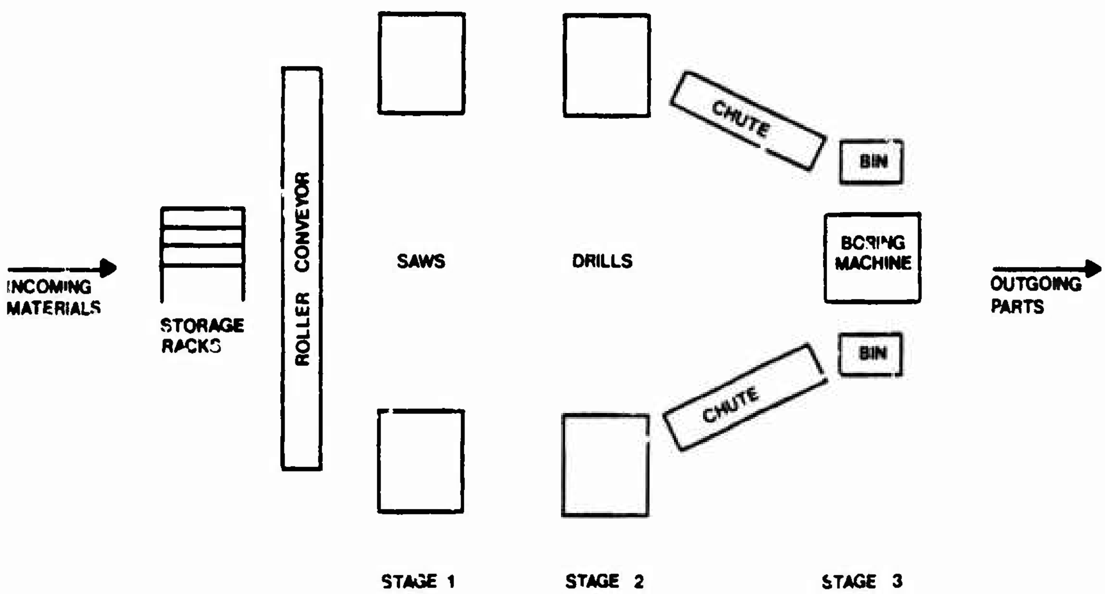


Figure 6-1. Schematic Drawing of ABC Manufacturing System

6.3 Defining the Model

At this point we face the first critical activity in the model building process. We must now define the elements of the system that are to be included in the model. The first step in this process is to clearly define the objectives of the study. This will determine the types of elements that we need to include in the model and will begin to give us an idea of which measures of the system's performance we would like to obtain.

In the problem statement given earlier, the management of the ABC Company stated that they wanted a capacity planning tool. However, comments made at a later point indicated that in reality, two types of problems are present in the manufacturing system. The first is a capacity planning problem, in that the production control department cannot accurately predict completion dates. They also mentioned another problem or, more correctly, a symptom of a problem: bulging in-process inventories. The problem that is indicated by excessive in-process inventories is a process bottleneck. Therefore, to meet the needs of the ABC management, the objectives of our study are to build a model that will facilitate not only capacity planning, but also bottleneck analysis.

In order to evaluate the capacity of a manufacturing system, certain measures of the system's performance are required. For instance, quantitative information concerning the system's production rate, the time parts spend in the system, utilization rates for each resource, the amount of time spent waiting for processing, the amount of time spent being processed, and the size of queues in front of each machine represent the types of performance measures which are needed to perform capacity and bottleneck analyses.

Now that we have defined the objectives of our modeling study, and have determined the types of performance measures desired, the initial definition of the model of the system can begin. That is, we can determine which elements of the physical system we wish to represent in the $IDEF_2$ model of the system. By considering the output information we expect from the model, we can determine which elements of the system must be included in the model. We do this by including only those system elements which will affect the system's performance in terms of those performance measures that we have listed above.

For instance, since we desire utilization factors for each of the machines in the system, we must model each of those machines. Similarly, since we desire information concerning the size of part queues, we must include storage areas in front of each machine in the model. If a machine breaks down, it can no longer process parts until it is repaired. Therefore, repairmen are an important part of the system's performance, and consequently are included in the model. Continuing this type of analysis, we will soon arrive at the Facility Diagram shown in Figure 6-2. Note the arrows on the Facility Diagram which have been included to indicate the general flow of the system. The detailed flow of the materials through the system is of primary concern in the next step of the modeling process.

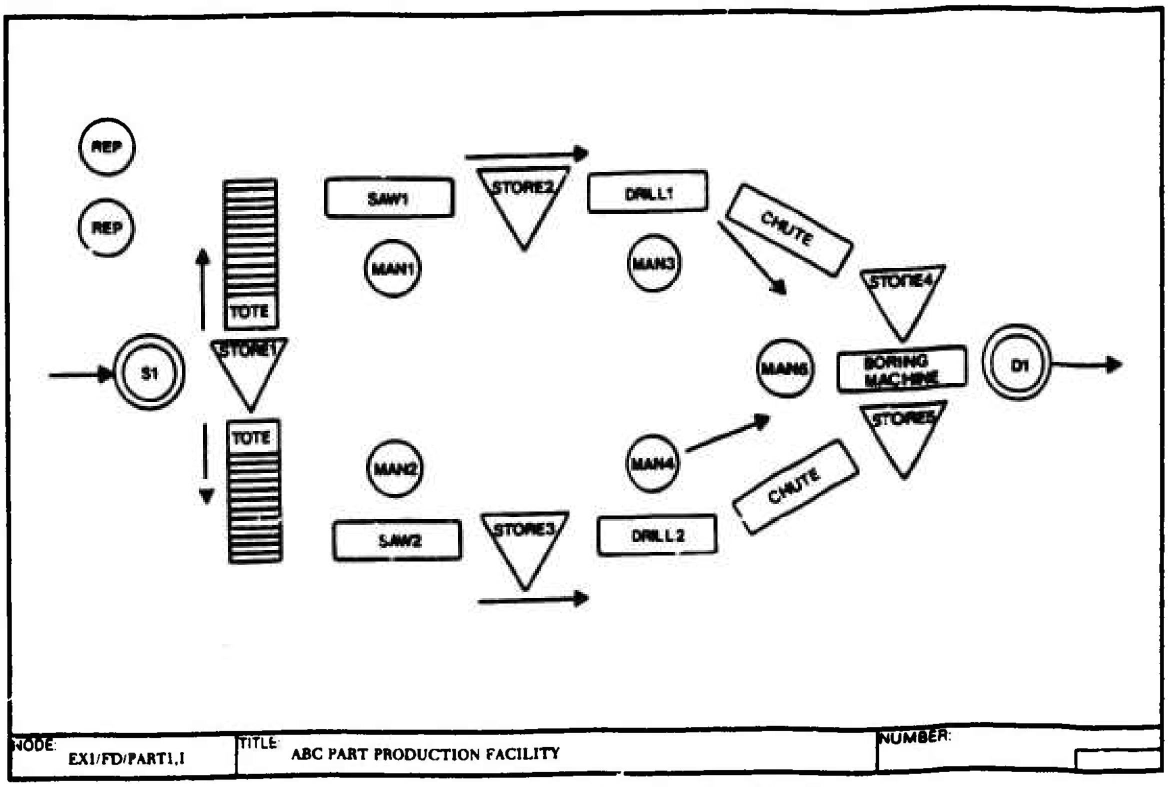


Figure 6-2. Part Production Facility

6.4 Modeling Entity Flow

The entity of prime concern in the ABC model is the single part type that is processed by this system. This part type, which is named PART1, flows through the system in a prescribed manner. Specifically, three operations are performed on it in a particular order: sawing drilling and boring. Let us begin to construct the Entity Flow Network for the part being processed by the ABC manufacturing system.

FACILITY COMPONENT ATTRIBUTE DEFINITION

FACILITY COMPONENT	IDENTIFYING NAME	NUMBER AVAILABLE	ATTRIBUTES IDENTIFIER	ATTRIBUTES VALUE
SOURCE	S1	1	TYPE	PART1
MANPOWER	REP	2	TYPE MACHINES	REPAIRMEN ALL
STORAGE	STORE1	1	TYPE CAPACITY	RACK ∞
MATERIAL HANDLING	TOTE	2	TYPE CAPACITY	TOTE 1
MANPOWER	MAN1	1	TYPE	OPERATOR

NODE: EX1/FA/PART1.1.1 TITLE: ABC PART PRODUCTION FACILITY NUMBER:

Figure 6-3. Part Production Facility Component Attribute Definition Form - Page 1

FACILITY COMPONENT ATTRIBUTE DEFINITION

FACILITY COMPONENT	IDENTIFYING NAME	NUMBER AVAILABLE	ATTRIBUTES IDENTIFIER	ATTRIBUTES VALUE
MANPOWER	MAN2	1	TYPE	OPERATOR
MACHINE	SAW1	1	TYPE CAPACITY	SAW 1
MACHINE	SAW2	1	TYPE CAPACITY	SAW 1
STORAGE	STORE2	1	TYPE CAPACITY	IN-PROCESS 100
STORAGE	STORE3	1	TYPE CAPACITY	IN-PROCESS 100

NODE: EX1/FA/PART1.1.2 TITLE: ABC PART PRODUCTION FACILITY NUMBER:

Figure 6-4. Part Production Facility Component Attribute Definition Form - Page 2

FACILITY COMPONENT ATTRIBUTE DEFINITION

FACILITY COMPONENT	IDENTIFYING NAME	NUMBER AVAILABLE	ATTRIBUTES	
			IDENTIFIER	VALUE
MACHINE	DRILL1	1	TYPE CAPACITY	DRILL 1
MACHINE	DRILL2	1	TYPE CAPACITY	DRILL 1
MANPOWER	MAN3	1	TYPE	OPERATOR
MANPOWER	MAN4	1	TYPE	OPERATOR
MATERIAL HANDLING	CHUTE	2	TYPE CAPACITY	CHUTE ∞

NODE: EX1/FA/PART1.1.3 | TITLE: ABC PRODUCTION FACILITY | NUMBER:

Figure 6-5. Part Production Facility Component Attribute Definition Form - Page 3

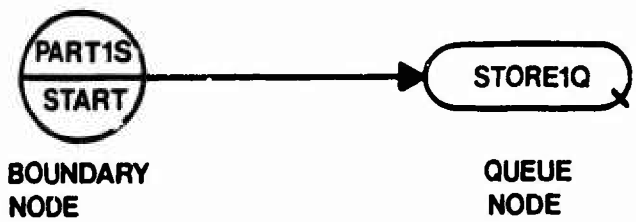
FACILITY COMPONENT ATTRIBUTE DEFINITION

FACILITY COMPONENT	IDENTIFYING NAME	NUMBER AVAILABLE	ATTRIBUTES	
			IDENTIFIER	VALUE
STORAGE	STORE4	1	TYPE CAPACITY	BIN 150
MANPOWER	MAN5	1	TYPE	OPERATOR
MACHINE	BORING MACHINE	1	TYPE CAPACITY	BORING MACHINE 1
DESTINATION	D1	1	TYPE	PART1
STORAGE	STORE5	1	TYPE CAPACITY	BIN 150

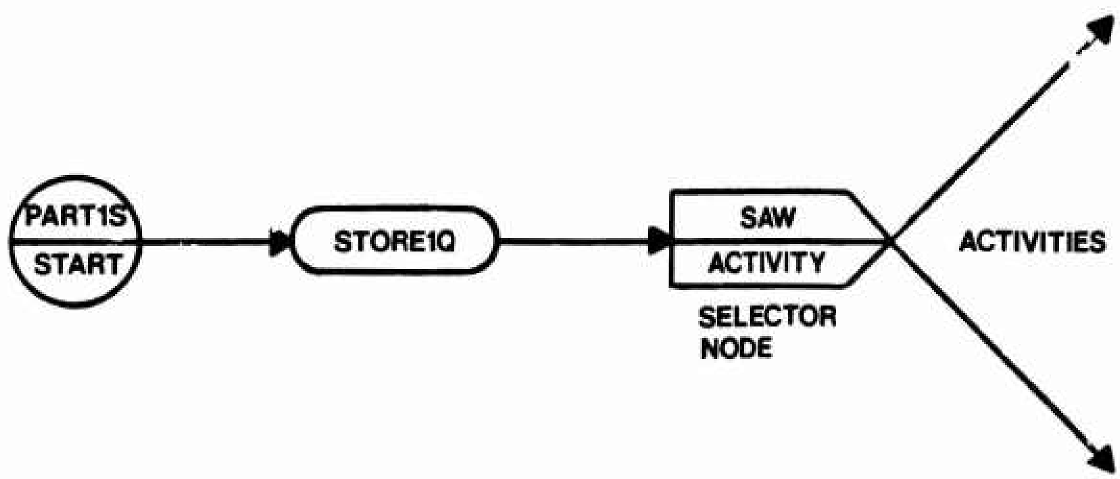
NODE: EX1/FA/PART1.1.4 | TITLE: ABC PART PRODUCTION FACILITY | NUMBER:

Figure 6-6. Part Production Facility Component Attribute Definition Form - Page 4

All Entity Flow Networks begin with a start BOUNDARY node. Therefore, we begin our Entity Flow Network with a BOUNDARY node named PART1S. As soon as a part enters the ABC system, it is stored on storage racks which we can represent with a QUEUE node. In order to denote the fact that this is the first storage area in the system, we shall name this QUEUE node with the name STORE1Q. The figure shown below illustrates our Entity Flow Network thus far.



Once the part is in the storage racks, it is available for the sawing operation which is performed next. Parts may be processed at either of the sawing stations. Therefore, our Entity Flow Network must represent the selection of the station to which the part will go. Since the sawing operation will be represented as an ACTIVITY in our Entity Flow Network, the type of selection which must be made is an activity selection. We can model this activity selection in the manner illustrated below.



The sawing ACTIVITIES are each represented by the arrows eminating from the SELECTOR node. The node was named SAW in order to denote the fact that a selection was being made between two sawing operations. In order to perform the sawing operation, two resources must be available for use. First, the saw on which the operation is to be performed must be operational and free. Secondly, the human operator that is to get the part and load it onto the saw must also be available. If both of these resources are available, they are allocated to the part and the sawing ACTIVITY is begun. We represent the allocation of resources to an entity in Entity Flow Networks with RESOURCE ALLOCATION symbols below the ACTIVITY for which they are required. Resources that are no longer required when an ACTIVITY is completed are released via RESOURCE RELEASE symbols. Since the sawing operation represents the passage of time, we indicate this on our Entity Flow Network by labeling the ACTIVITY, SAWING. With the inclusion of the ACTIVITY time and the resources required to perform the ACTIVITY we have completely described the ACTIVITY. At this point, our Entity Flow Network looks like the one illustrated in Figure 6-7.

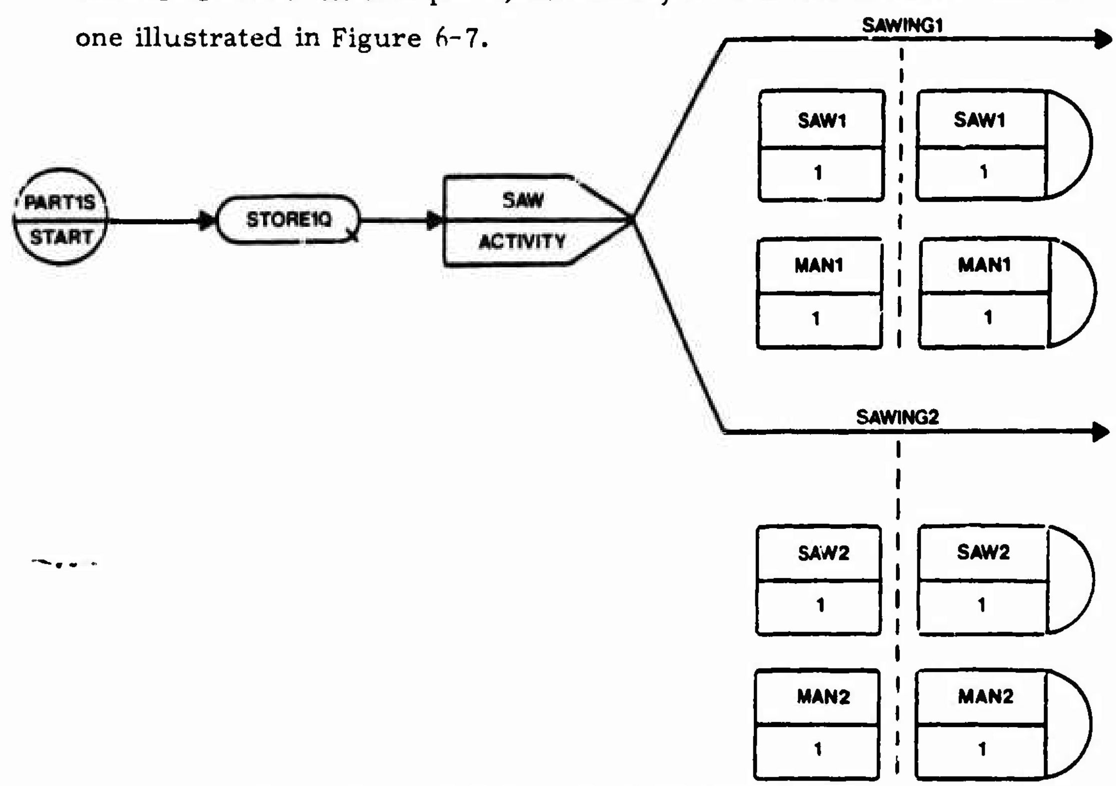


Figure 6-7. Partial Entity Flow Network for PART1

If the part was processed by SAW1 and MAN1, it will then go to the ACCUMULATE node where four entities are produced for each entity that enters the node. This represents the actual sawing of each part into four identical smaller parts. These parts then go to the QUEUE node named STORE2Q to wait for the resources required for the drilling operation. If a part was processed by MAN2 and SAW2, it will proceed to the ACCUMULATE node where again, entities are produced for each entity that enters the node. This represents the actual sawing of each part into four identical smaller parts. These parts then go to the QUEUE node labeled STORE3Q to await the drilling operation. The drilling operation is represented with the machine and operator resources both allocated and released beneath the arrow representing the ACTIVITY.

The storage areas in front of each of the drills and the boring machine have limited capacity. When a storage area is full and another part arrives, balking takes place. In the real system, an intolerably high in-process inventory level has built up and arriving parts are subcontracted to outside job shops. This is modeled in $IDEF_2$ by showing entities balking from QUEUE nodes if they arrive when the QUEUE node is at capacity. We include two END nodes named SUBC2 and SUBC3 to which entities may balk from QUEUE nodes STORE2Q and STORE3Q, respectively.

At this point, we find the size of our Entity Flow Network exceeding the space on a normal IDEF form. In order to continue our Entity Flow Network, we will use GO TO BOUNDARY nodes to denote the fact that the network is continued on another page. Two GO TO nodes directing entity flow to nodes S4 and S5, respectively, are added, resulting in the Entity Flow Network illustrated in Figure 6-8.

After the parts are drilled they go to storage areas in front of the boring machine. In our Entity Flow Network, we will represent these storage areas with two QUEUE nodes labeled STORE4Q and STORE5Q, respectively. Because of their limited capacity, balking may take place from these nodes to END nodes, SUBC4 and SUBC5, respectively.

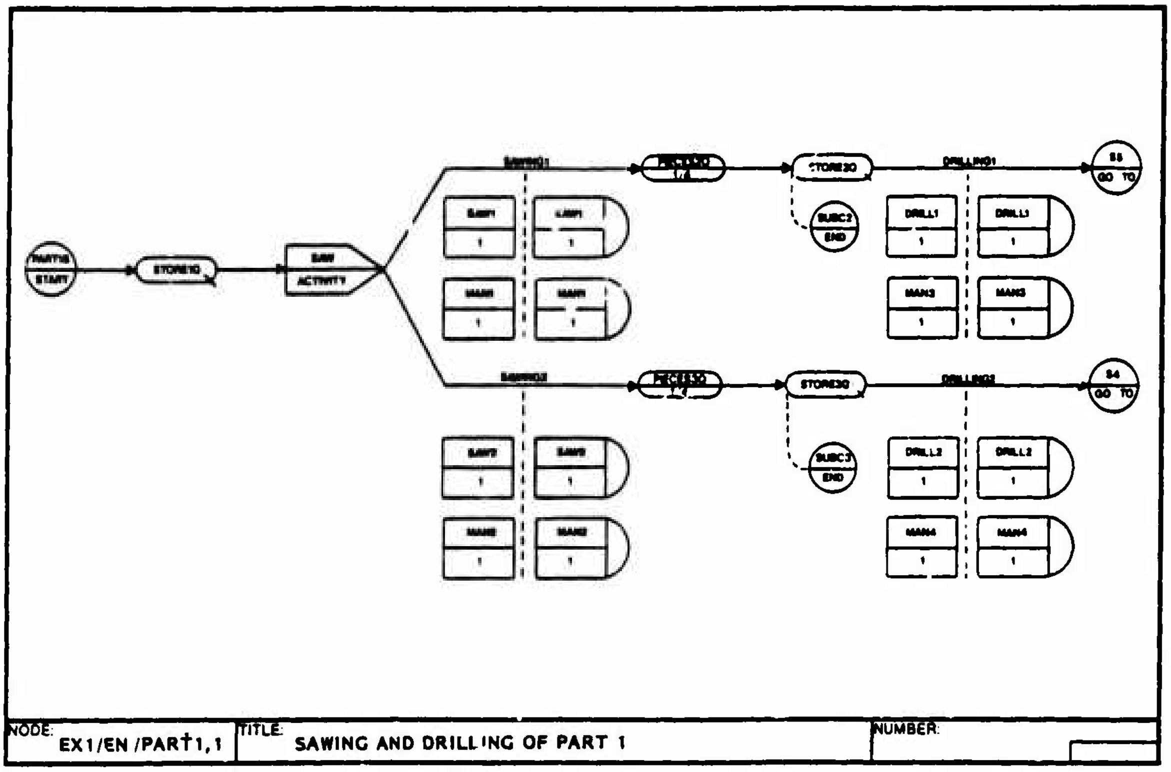


Figure 6-8. Initial Phase of PART1 Processing

These storage areas both feed the boring operation. To denote the fact that entities are selected from both of the QUEUE nodes, we must include a SELECTOR node in our network. The type of selection being made is a queue selection, i.e., we are selecting entities from one of two queues. The boring activity is represented with an arrow, with the boring machine and operator allocated via RESOURCE ALLOCATION symbols beneath the activities. The boring machine and the operator are then released at the end of the ACTIVITY. We model this by including the corresponding RESOURCE RELEASE symbols beneath the ACTIVITY. Since this is the last operation in our system we conclude the network with an END BOUNDARY node named PART1X. Figure 6-9 illustrates the portion of the Entity Flow Network just constructed.

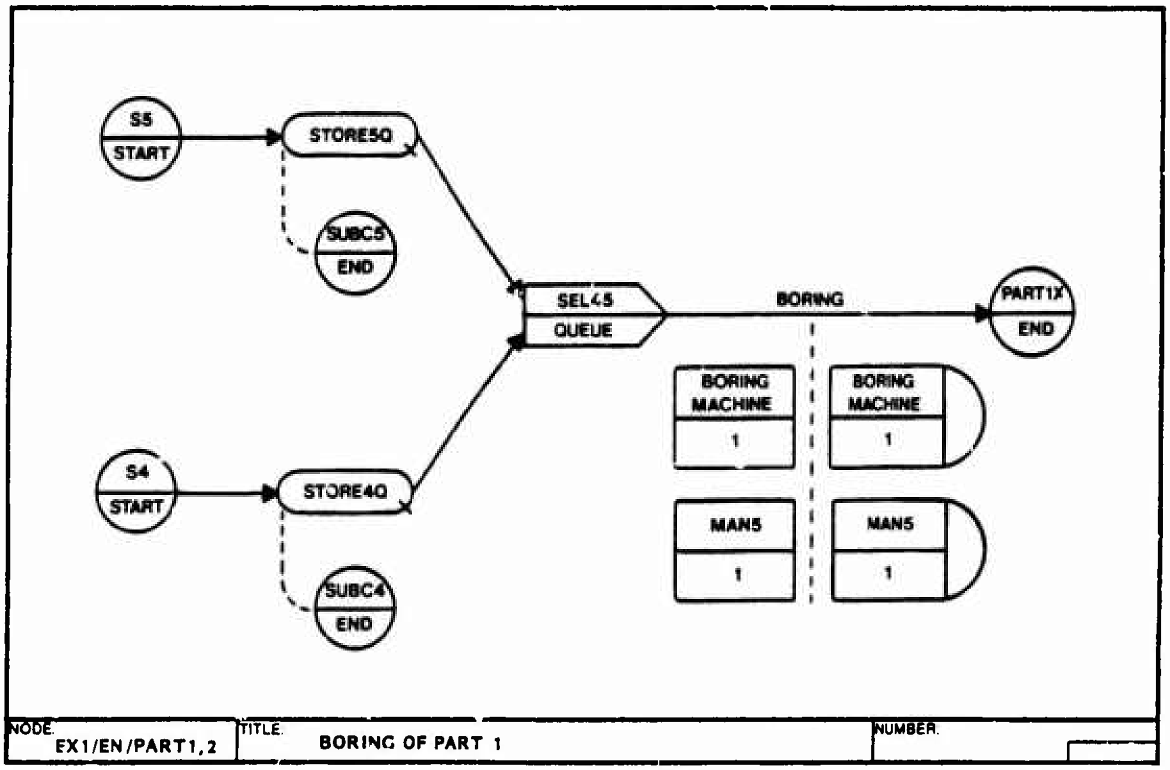


Figure 6-9. Second Phase of PART1 Processing

We have now represented the primary entity flow for our manufacturing system. We have graphically represented the flow of parts through the system by representing the activities that the parts engage in and the resources which are utilized by parts as they pass through the system. The Entity Flow Network is, in itself, a useful communicative vehicle. We could show the Entity Flow Network to people unfamiliar with our manufacturing system, and they could see the manner in which parts are processed by the system. The network is simple and uncluttered yet it contains the basic elements needed to describe a dynamic manufacturing system.

Returning to the context of our example, recall that the management of the ABC Manufacturing Company wanted a capacity planning tool which they could use to obtain measures of the system's performance. While our Entity Flow Network does graphically portray the passage of entities through the system, we have not yet quantified all of the dynamic characteristics of the system. To accomplish this, we must complete a set of Entity Flow Node Attribute Definition forms for each of the elements in our system. By considering the nature of our problem, we can readily suggest pieces of information that would be necessary to perform an analysis of the dynamics of our system.

For example in order to measure the length of time that parts spend in the manufacturing system, we must specify a time for each of the activities that the parts engage in. Also we must quantify the storage space available at each machine site. Practically, we know that storage space is not unlimited. For this reason, we must define the amount of storage space that is available for parts storage. If it is possible for this space to be completely utilized, we must define the disposition of parts that arrive to a queue which is already at its capacity.

The process of defining these characteristics of our systems elements typically requires more data than was gathered at the initial interviews with system personnel. However, it is usually a good idea to begin constructing $IDEF_2$ submodels as we did here to better define the types of information that will be required for the end use of the model. As we have mentioned before, the modeling objectives play an important role in the level of detail in which the system is modeled. This is also true with respect to data collection. In addition detailed information is required concerning ACTIVITY times, SELECTOR node selection rules, priority ranking for all QUEUE nodes, and other parameters of the system. Figures 6-10 and 6-11 show the Entity Flow Node Attribute Definitions for each of the elements in the Entity Flow Network.

6.6 Modeling the Disposition of Resources

When resources complete their involvement in ACTIVITIES a decision must be made concerning their disposition. In $IDEF_2$, resources that have just completed the performance of an ACTIVITY are either freed, allocated, or held until another resource or group of resources is available for which their combined application is required. The

ENTITY FLOW NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES: IDENTIFIER	ATTRIBUTES: VALUE
QUEUE	STORE1Q	RANKING RULE	FIRST-IN, FIRST-OUT
		MAXIMUM NUMBER IN QUEUE	∞
		INITIAL NUMBER IN QUEUE	1
SELECTOR	SAW	SELECTION RULE	LEAST AMOUNT OF USAGE
ACTIVITY	SAWING1	DURATION	NORMAL (5, 1.2)
QUEUE	STORE2Q	RANKING RULE	FIRST-IN, FIRST-OUT
		MAXIMUM NUMBER IN QUEUE	120
		INITIAL NUMBER IN QUEUE	0
		FULL CONDITION	BALK(SUBC2)
ACTIVITY	DRILLING1	DURATION	UNIFORM(12, 20)
ACTIVITY	SAWING2	DURATION	NORMAL(5, 1.2)
QUEUE	STORE3Q	RANKING RULE	FIRST-IN, FIRST-OUT
		MAXIMUM NUMBER IN QUEUE	120
		INITIAL NUMBER IN QUEUE	0
		FULL CONDITION	BALK(SUBC3)
ACTIVITY	DRILLING2	DURATION	UNIFORM(12, 20)

NODE: EX1/EA/PART1.1.1 TITLE: INITIAL PHASE OF PART1 PROCESSING NUMBER:

Figure 6-10. Initial Phase of PART1 Processing Entity Flow Node Attribute Definition Form

ENTITY FLOW NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES: IDENTIFIER	ATTRIBUTES: VALUE
QUEUE	STORE5Q	RANKING RULE	FIRST-IN, FIRST-OUT
		MAXIMUM NUMBER IN QUEUE	125
		INITIAL NUMBER IN QUEUE	0
		FULL CONDITION	BALK(SUBC5)
QUEUE	STORE4Q	RANKING RULE	FIRST-IN, FIRST-OUT
		MAXIMUM NUMBER IN QUEUE	125
		INITIAL NUMBER IN QUEUE	0
		FULL CONDITION	BALK(SUBC4)
SELECTOR	SEL45	SELECTION RULE	LARGEST NUMBER PRIORITY
ACTIVITY	BORING	DURATION	UNIFORM(7, 12)

NODE: EX1/EA/PART1.2.1 TITLE: SECOND PHASE OF PART1 PROCESSING NUMBER:

Figure 6-11. Second Phase of PART1 Processing Entity Flow Node Attribute Definition Form disposition of all resources shown on Entity Flow Networks must be described with Resource Disposition Trees. In the ABC manufacturing system example, resources shown on the Entity Flow Network for PART1 include each of the machine tools, the humans that operate the machines, and the repair personnel. Let us now consider the construction of a Resource Disposition Tree for the resource SAW1.

Resource Disposition Trees are comprised of one or more questions, whose answers determine the course of action to be taken to accomplish the disposition of a resource that has just completed an ACTIVITY. For the resource SAW1, the first question to be answered is "Are there any parts awaiting service by this resource?" If the answer to this question is "NO," then SAW1 may be freed and this ACTION is specified on the Resource Disposition Tree. If the answer to this question is "YES," we must then examine the status of the resource MAN1, because these resources must both be available to begin operating on a part. Thus, the next question to be asked on the tree is "Is the resource MAN1 free?" If the answer to this question is "NO," again the resource SAW1 is freed. If the answer to this question is "YES," then the situation exists where a part is ready to be processed and both the SAW1 resource and the MAN1 resource are available to begin the operation. The ACTION specified on the Resource Disposition Tree is to allocate both the saw and the operator to the first part awaiting processing in the QUEUE node named STORE1Q. The Resource Disposition Tree for SAW1 is illustrated in Figure 6-12.

The disposition of each of the resources in the ABC manufacturing system is essentially the same. That is, the first question to be asked in each of the Resource Disposition Trees is "Are there any requests for the service of this resource?" If the answer to this question is "NO," the resource is freed. If the answer to this question is "YES," another question is asked. For machine resources, the second question asks whether or not the associated human operator is available, and for the human resources, the second question asks whether or not the machine

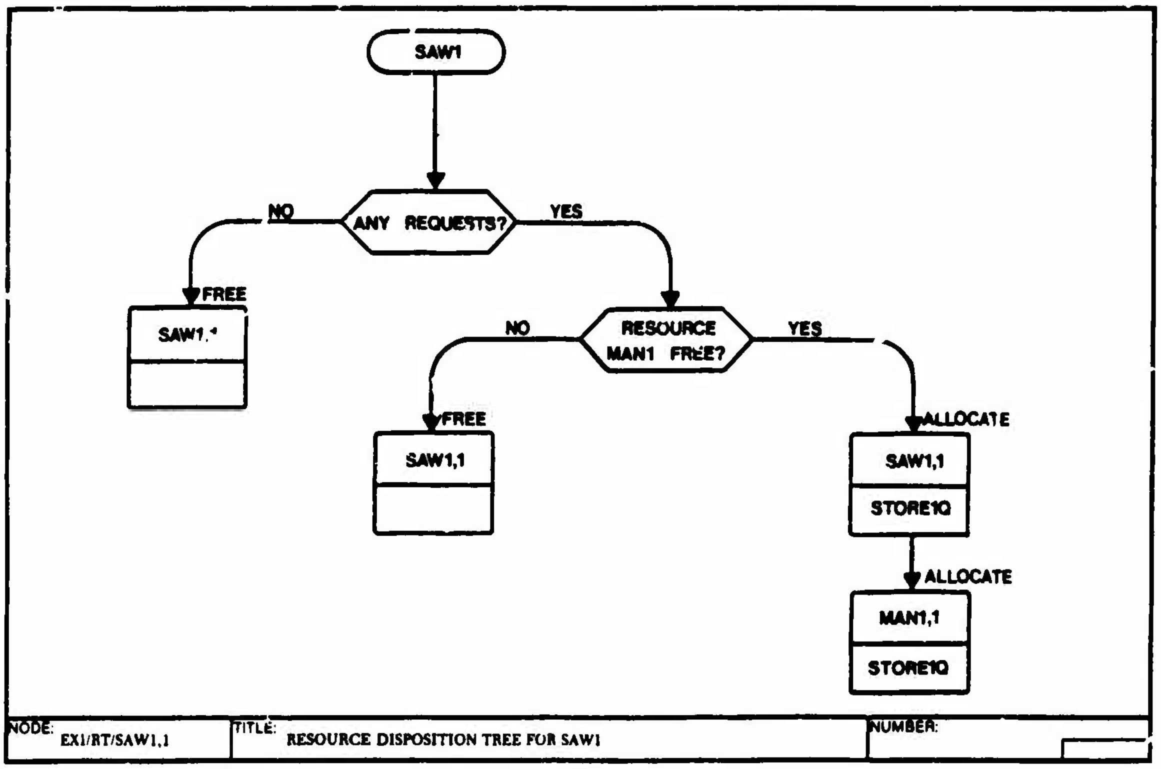


Figure 6-12. Resource Disposition Tree for SAW1

is available. If the answer to this second question is "NO," the resource is again freed. If the answer to this question is "YES," then both the human and the machine resource are allocated to the node in which a part awaits their service. The Resource Disposition Trees for each of the resources that has appeared in an Entity Flow Network thus far in the model are shown in Figures 6-13 through 6-22.

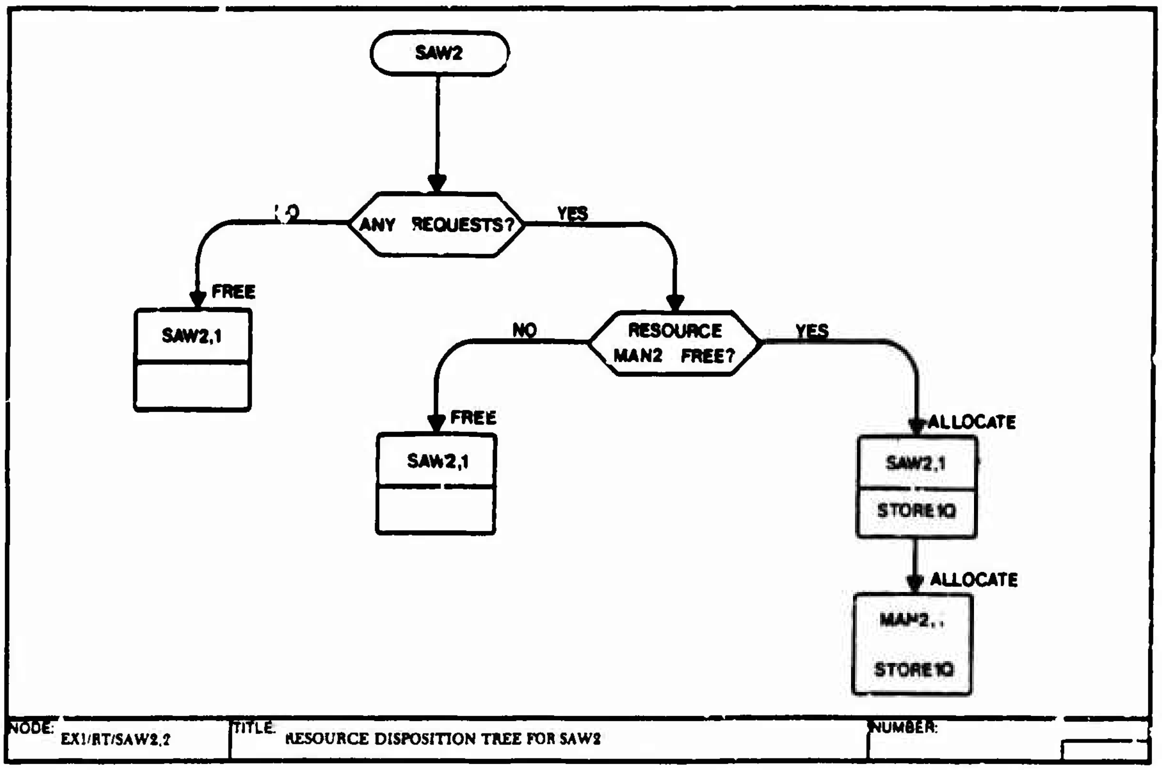


Figure 6-13. Resource Disposition Tree for SAW2

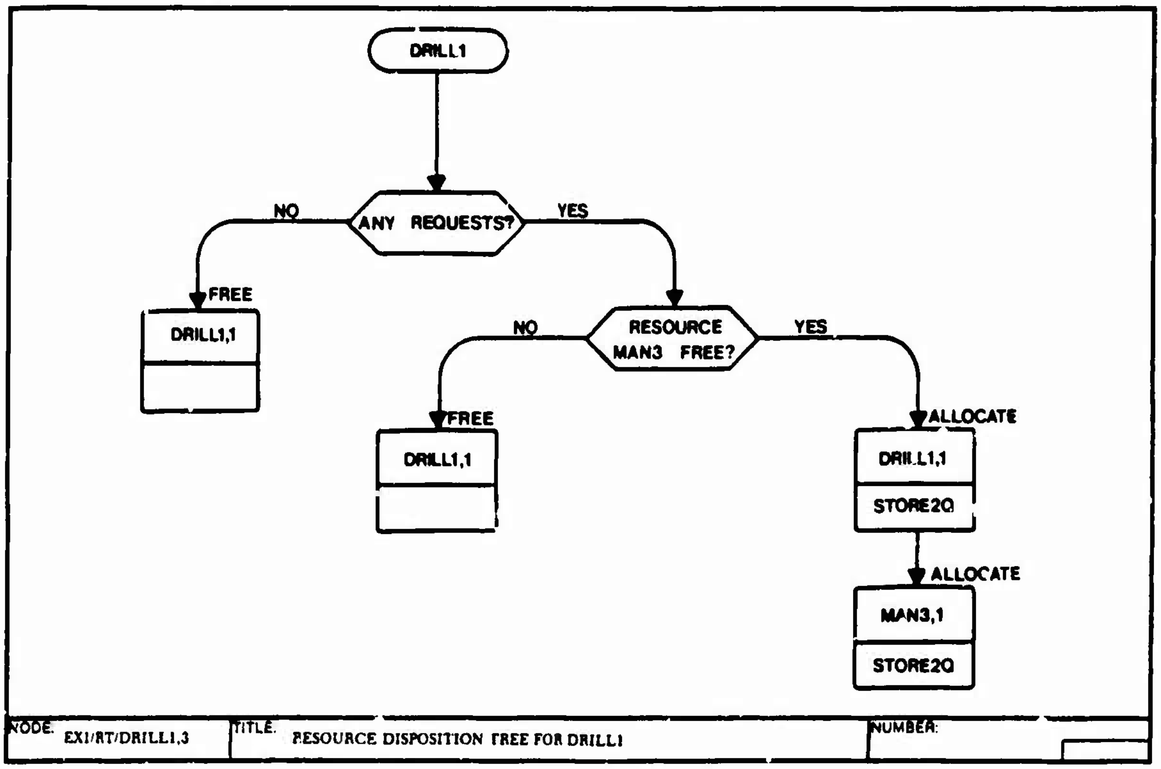


Figure 6-14. Resource Disposition Tree for DRILL1

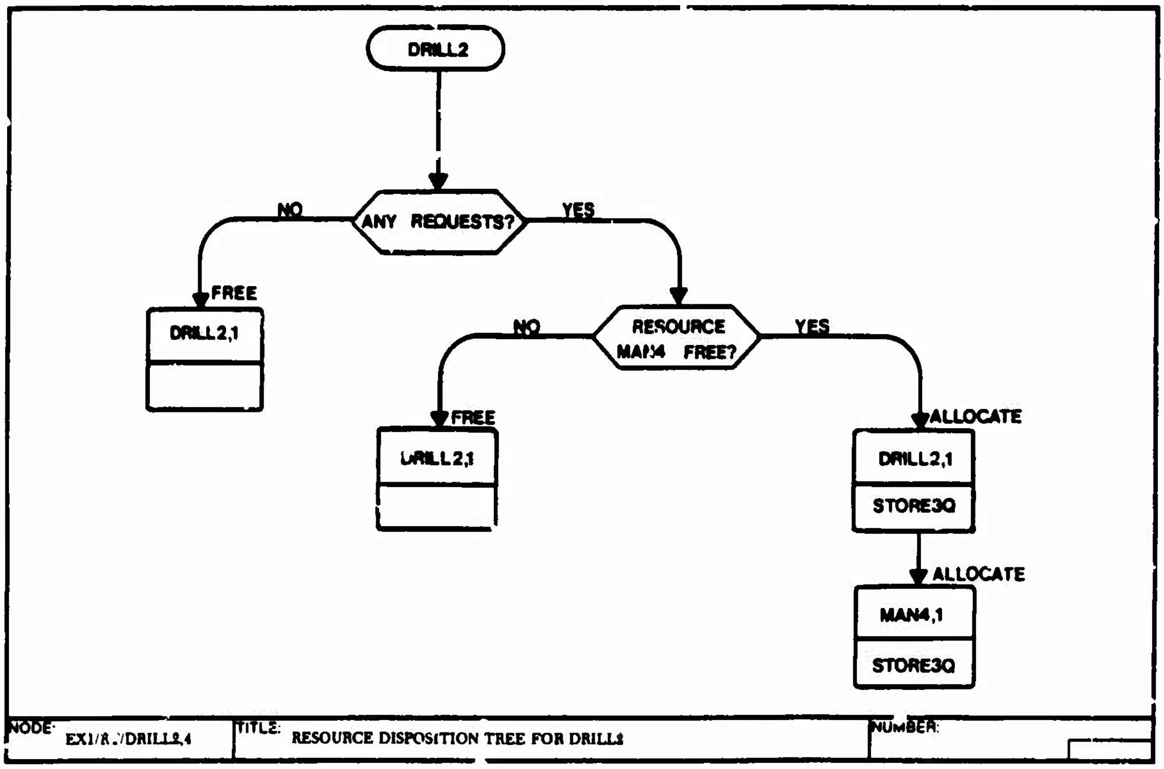


Figure 6-15. Resource Disposition Tree for DRILL2

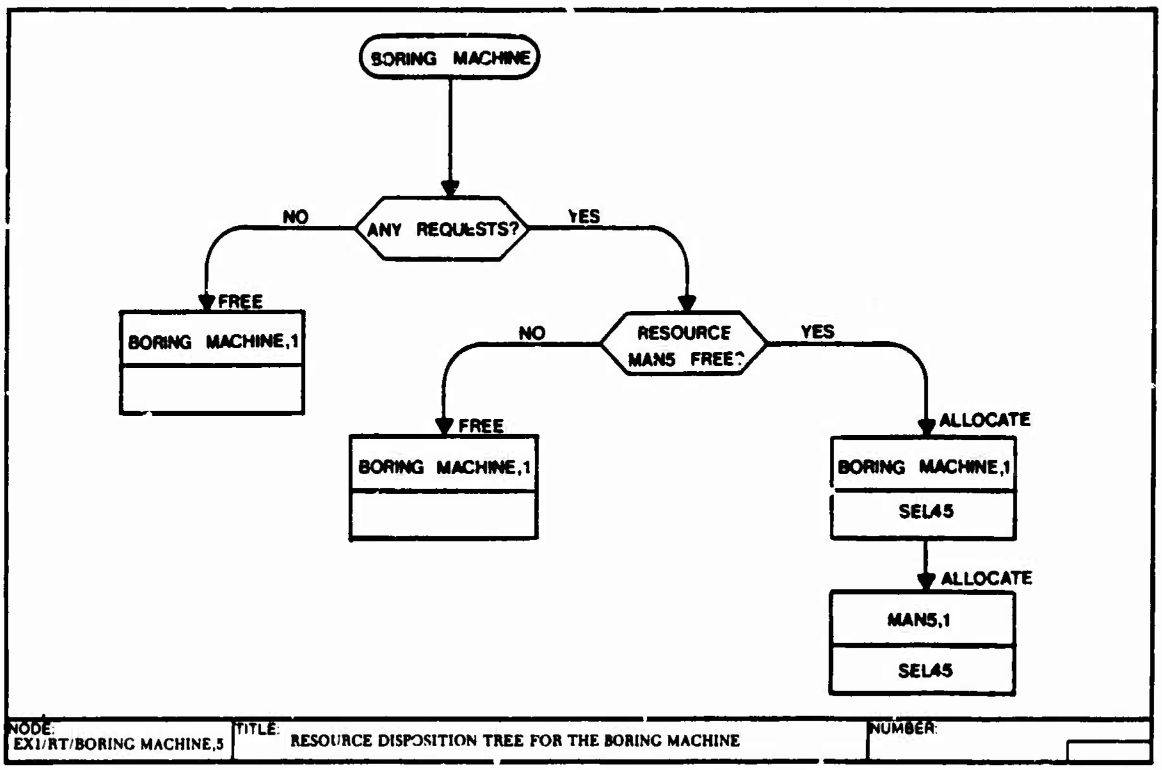


Figure 6-16. Resource Disposition Tree for the Boring Machine

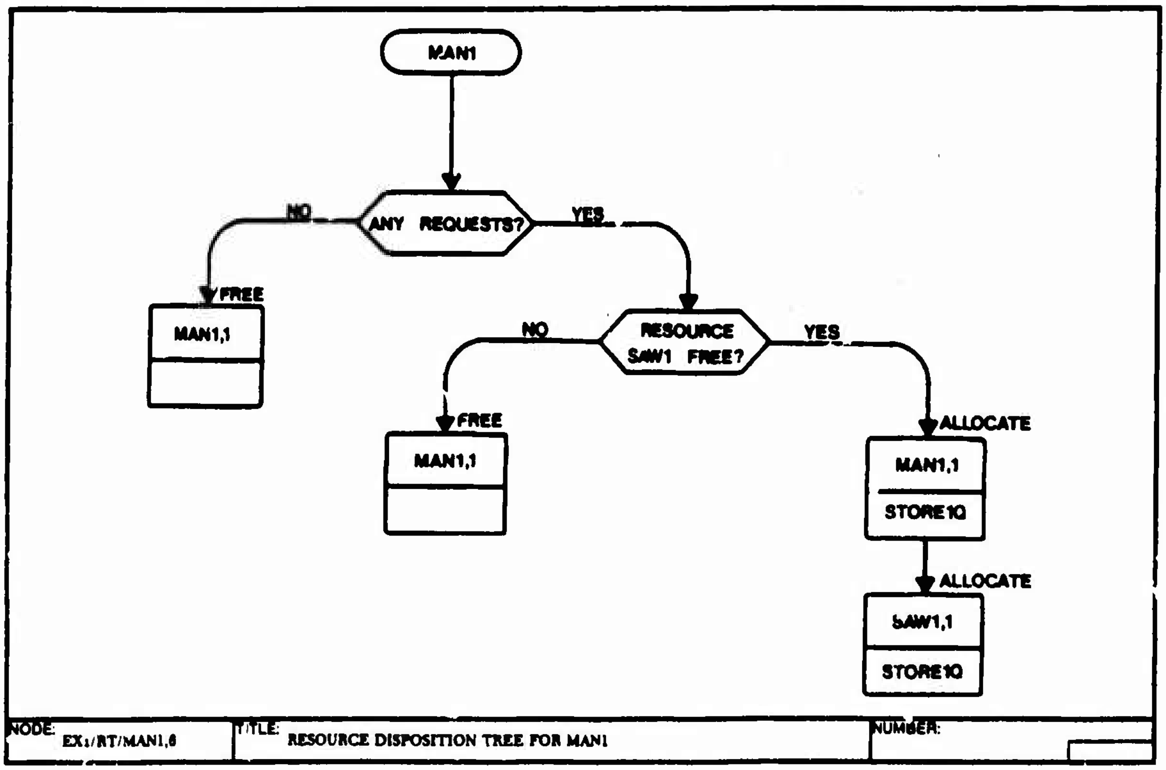


Figure 6-17. Resource Disposition Tree for MAN1

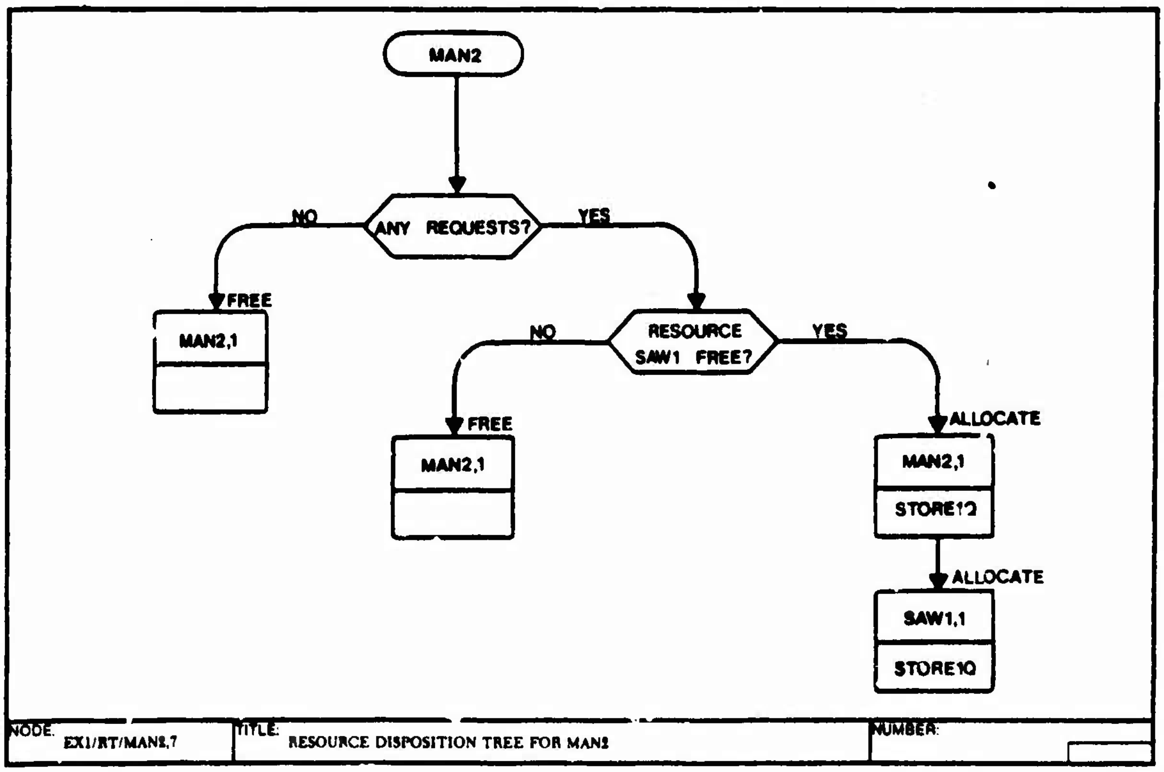


Figure 6-18. Resource Disposition Tree for MAN2

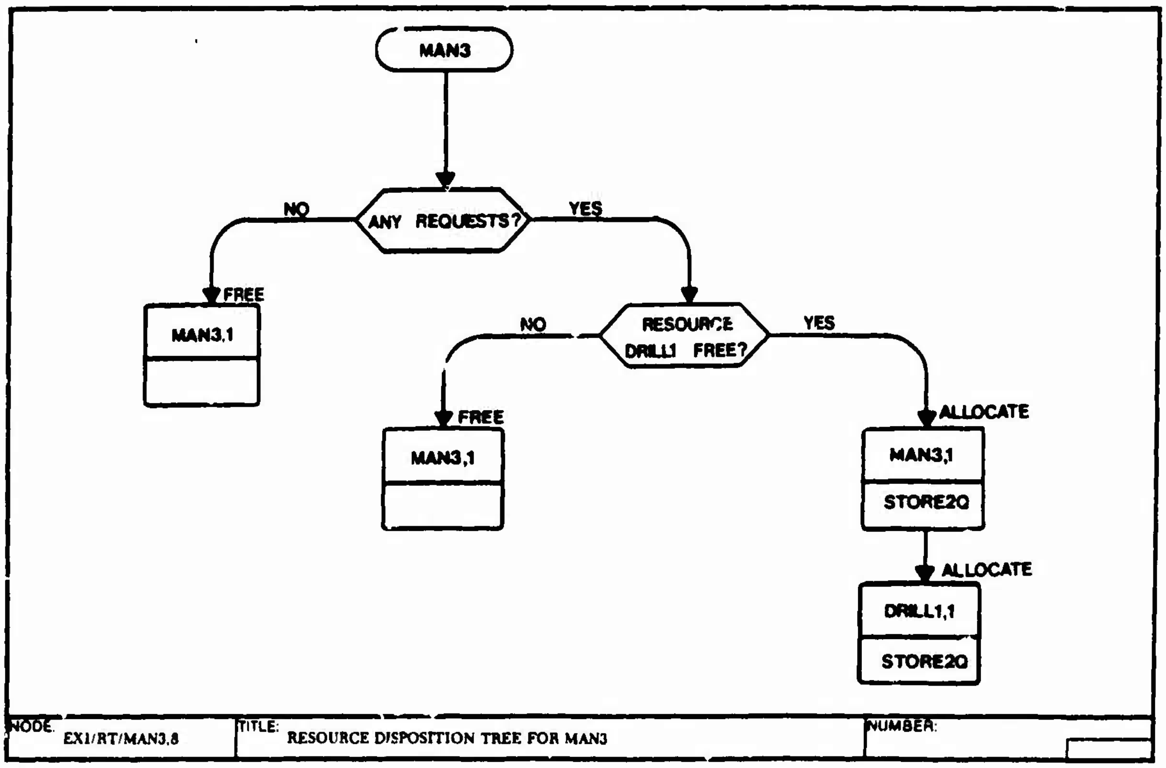


Figure 6-19. Resource Disposition Tree for MAN 3

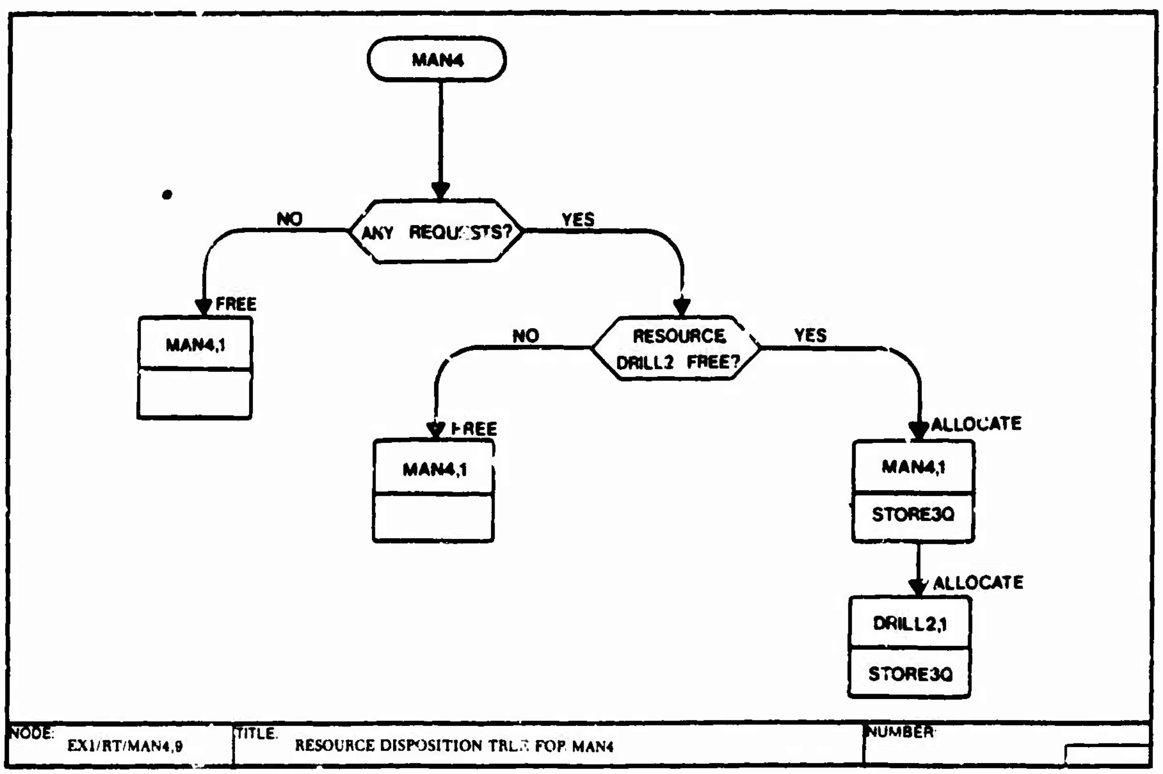


Figure 6-20. Resource Disposition Tree for MAN 4

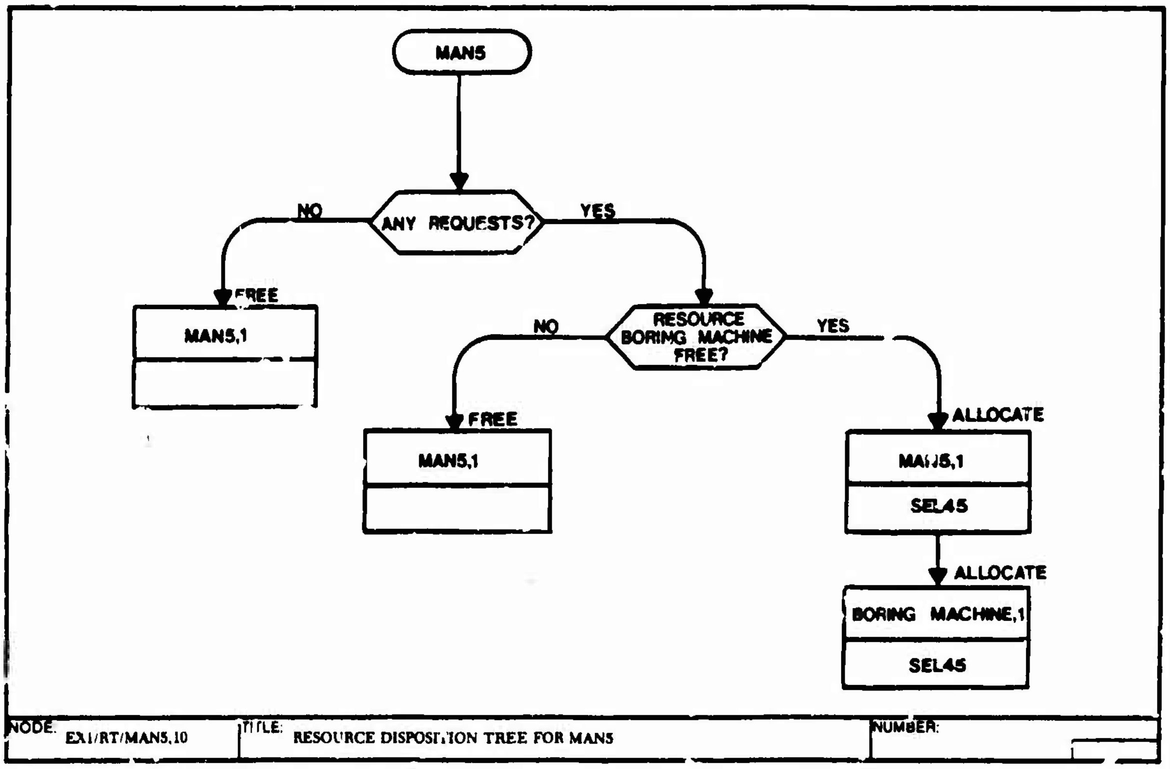


Figure 6-21. Resource Disposition Tree for MAN5

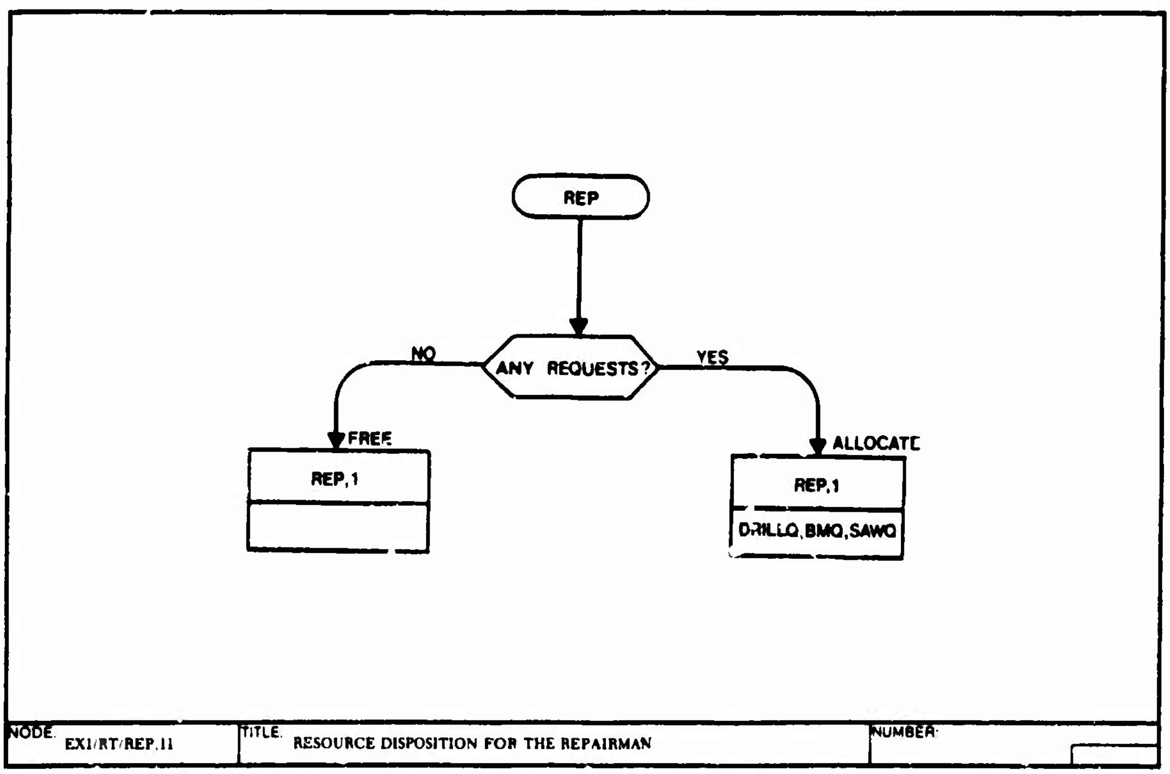


Figure 6-22. Resource Disposition Tree for the Repairman

As with entities the initial disposition of resources must be specified. In addition the total number of resources of each type must be specified. Resources may be initially allocated to a QUEUE node or they may initially be designated free. The Initial Resource Disposition form for the resources present at this point in the ABC model is shown in Figure 6-23.

INITIAL RESOURCE DISPOSITION

TYPE	ATTRIBUTES: IDENTIFIER	ATTRIBUTES: VALUE AT CREATION	NUMBER OF UNITS AVAILABLE	INITIAL DISPOSITION (QUEUE, ACTIVITY, FREE)
SAW1	-	-	1	FREE
SAW2	-	-	1	FREE
DRILL1	-	-	1	FREE
DRILL2	-	-	1	FREE
BORING MACHINE	-	-	1	FREE
MAN1	-	-	1	FREE
MAN2	-	-	1	FREE
MAN3	-	-	1	FREE
MAN4	-	-	1	FREE
MAN5	-	-	1	FREE
REP	-	-	2	FREE

MODE:	TITLE:	NUMBER:
EX1/RT/1	INITIAL DISPOSITION OF ALL RESOURCES	

Figure 6-23. Initial Disposition of all Resources

6.6 Constructing System Control Networks

Each of the machines in the ABC manufacturing system are subject to periodic failure due to breakdowns. Machine failure is an example of a controlling condition that is modeled with System Control Networks. System Control Networks are used to model controlling conditions which influence the flow of entities through manufacturing systems.

Consider the breakdown of a drilling machine that is used in the ABC manufacturing system. When the machine breaks down, it can no longer service parts. In effect, it is no longer available for use in processing the entities that flow through the Entity Flow Network. To make the drill unavailable for further part processing, we preempt it from the Entity Flow Network with a PREEMPT node. Conceptually, we can view this as though we are taking the drill out of the Entity Flow Network and inserting it into the System Control Network. In reality, the drill remains stationary and a call is put in to a repairman to fix the drill. We model this ACTIVITY by inserting the drill into a queue for a resource REP which represents the repairman. In doing this, we have transformed the resource into an entity. That is, the drill is entered into a QUEUE node and must await processing by the resource, REP.

This is modeled with an Entity Flow Network for the drill entity. A dotted line represents the disposition of the preempted drill to a new location in an Entity Flow Network. This new location is specified by naming the BOUNDARY node through which the drill will enter the Entity Flow Network. In addition, we must specify the disposition of any part that may have been in process on the drill at the time of its preemption. Therefore, we also name the location in the PART1 Entity Flow Network to which parts will be routed while the drill is being repaired. Figures 6-24 and 6-25 illustrate the use of a PREEMPT node to model the breakdowns of the drills. The Entity Flow Networks which model the repair of the drills are shown in Figures 6-26 and 6-27.

Continuing with the System Control Network, we represent the time for which the drill will be unoperational with an ACTIVITY. However, we cannot specify with certainty the length of time which the drill will be in repair because we do not, at this point, know whether or not a repairman is available to begin working on the machine. We will specify this ACTIVITY to be ended upon the release of a node which appears in the Entity Flow Network which models the repair of the machine. After

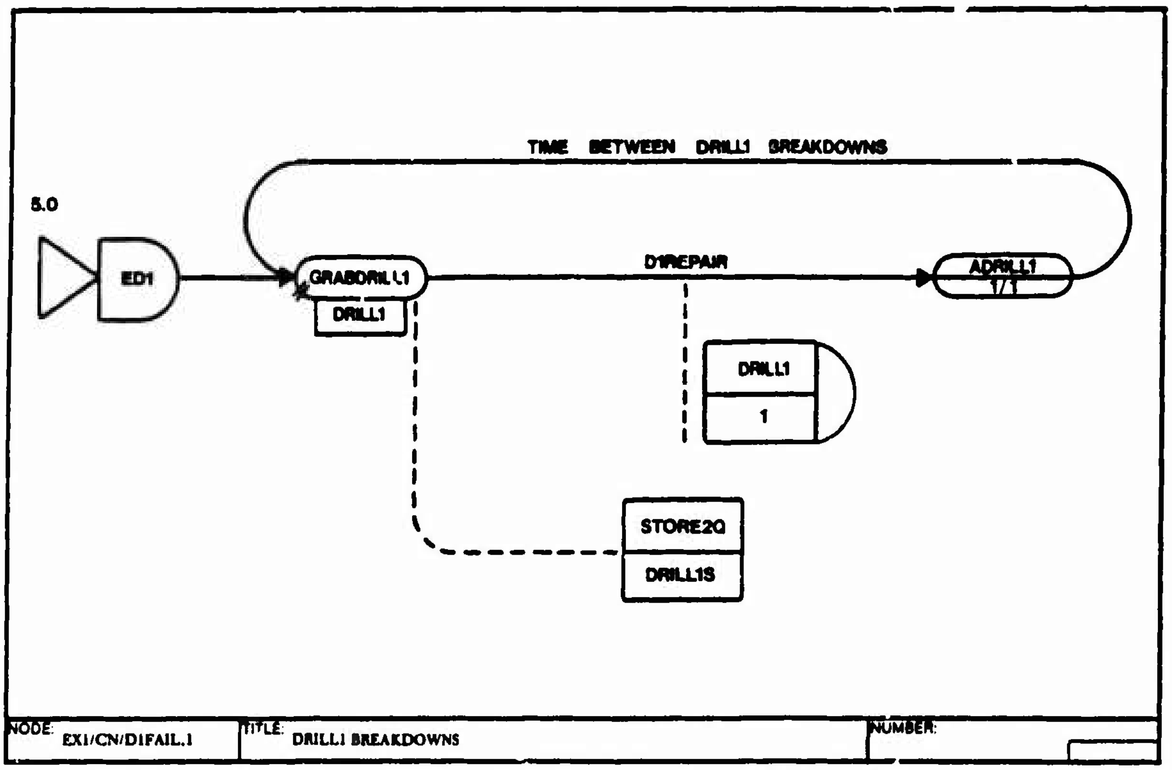


Figure 6-24. System Control Network for DRILL1 Breakdowns

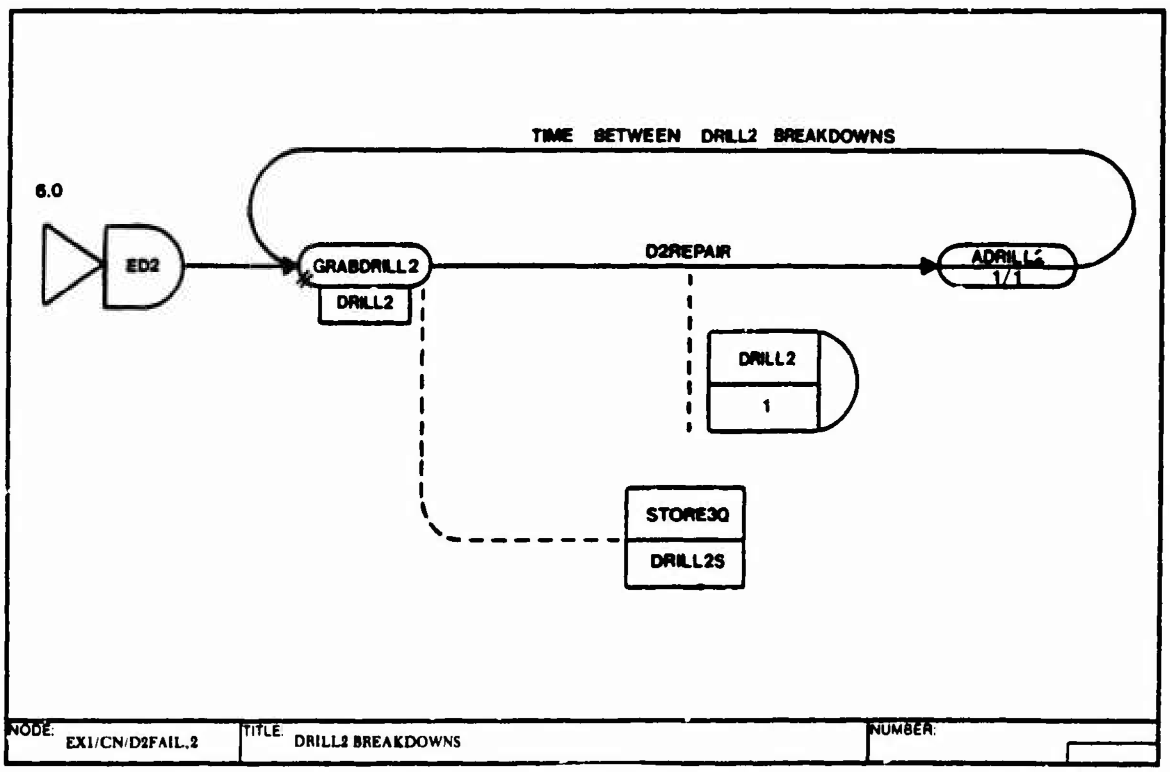


Figure 6-25. System Control Network for DRILL2 Breakdowns

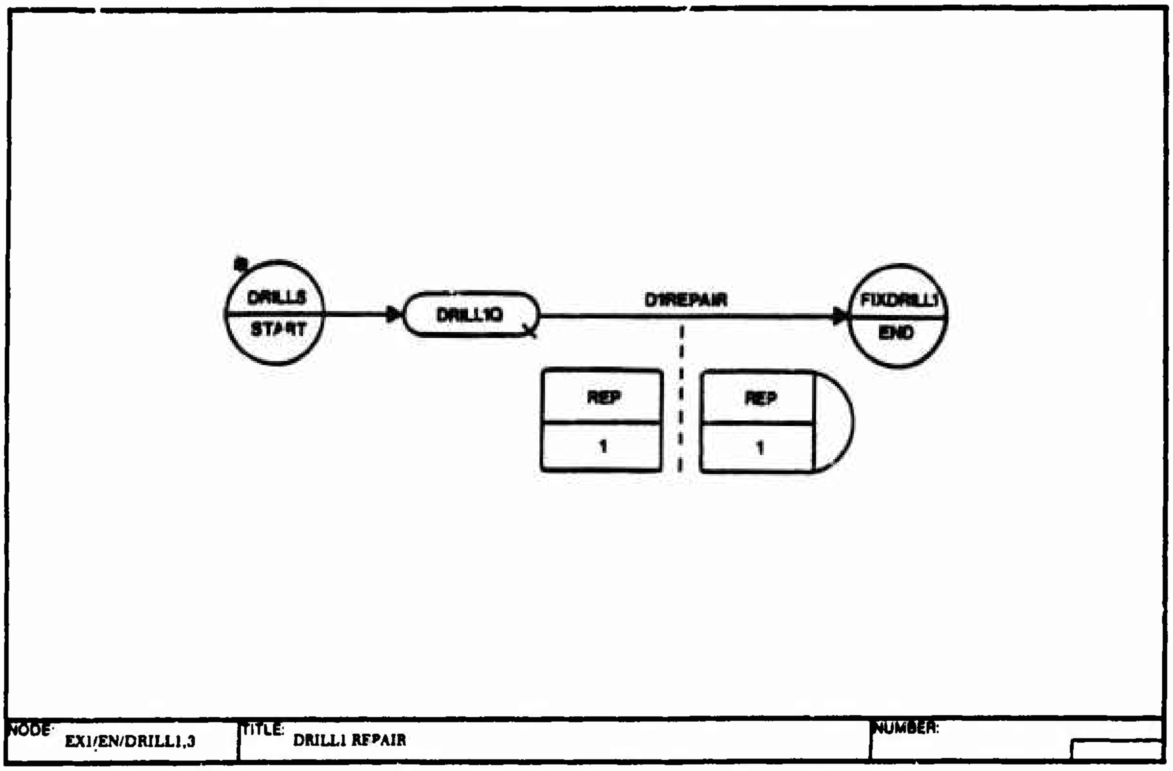


Figure 6-26. Entity Flow Network for DRILL1 Repair

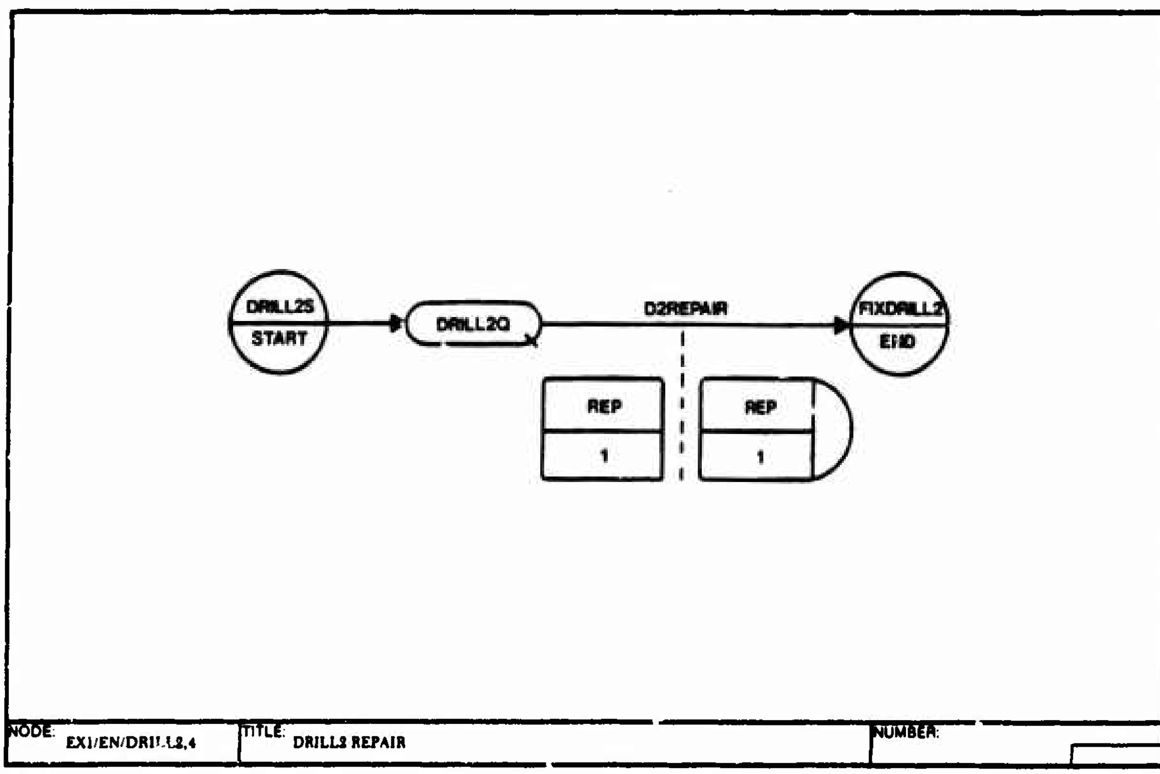


Figure 6-27. Entity Flow Network for DRILL2 Repair

completion of the repair ACTIVITY in the network shown in Figure 6-26, the drill will release a BOUNDARY NODE named FIXDRILL1. The release of this node will signal the end of the ACTIVITY that appears in System Control Network. In this way, activities whose durations are indeterminate because of situation dependence can be modeled.

Returning to the System Control Network, a RESOURCE RELEASE node once again makes the drill available for use in the PART1 Entity Flow Network. At this point the ACCUMULATE node named ADRILL1 is released. This begins an ACTIVITY which represents the time between drill breakdowns. In other words, this activity denotes the delay in time between subsequent breakdowns of a drill. After this time has passed, the PREEMPT node will again seize a drill resource from the PART1 Entity Flow Network, and the process will begin again.

The network is self regenerating, i.e., once the failure of a drill occurs, subsequent failures will be generated within the network. The first occurrence of a drill breakdown occurs at the time specified above the ENTER node. The ENTER node is used to initially activate the System Control Network.

System Control Networks represent the most abstract aspects of $IDEF_2$ modeling. The successful construction of System Control Networks requires a thorough understanding of the modeling concepts embodied in $IDEF_2$. However, because of this degree of abstraction, System Control Networks provide the flexibility to model a wide variety of manufacturing influences. Events that occur at irregular intervals, or that occur as a result of the status of the system are easily modeled with System Control Networks.

The system Control Network that models the breakdown of drilling machines applies several unique concepts that merit review. The key concepts in the initial ABC system control network include:

1. The preemption of resources from Entity Flow Networks via PREEMPT nodes. Conceptually, we are moving resources back and forth between Entity Flow Networks and System Control Networks.

2. The conversion of resources into entities. When resources are preempted and subsequently require service by other resources, they become, in effect, entities. The transfer is temporary and when the resource returns to the Entity Flow Network from which it was preempted, it is again a resource. This transfer represents an important interface between Entity Flow Networks and System Control Networks.

3. The specification of activities with indefinite durations. The duration of an activity may be specified by identifying a node elsewhere in the model whose release will signal the end of the activity. This feature is designed to model activities whose durations are situation dependent. It permits the $IDEF_2$ modeler to relax assumptions on resource availability, and accurately represent real situations.

These concepts are worth studying and represent some of the most powerful features of $IDEF_2$. They offer a great deal of flexibility to $IDEF_2$ modelers and require a thorough understanding and creative application. It is suggested that the example just presented be reviewed to assure understanding before continuing on with this text.

Although there are several new concepts to be mastered in the application of System Control Networks, many System Control Networks can be built upon the same concepts. The example just presented illustrated one method of modeling machine breakdowns. This approach to modeling breakdowns can be used for each of the machines in a system, presuming that each machine must be repaired by a repairman. Figures 6-28 through 6-30 show the System Control Networks that model the occurrence of breakdowns for the boring machine and the saws. The only differences between the networks are the time specified for first occurrences, and the nodal references within the networks. Figures 6-31 through 6-33 represent the Entity Flow Networks associated with the networks shown in the previous three figures.

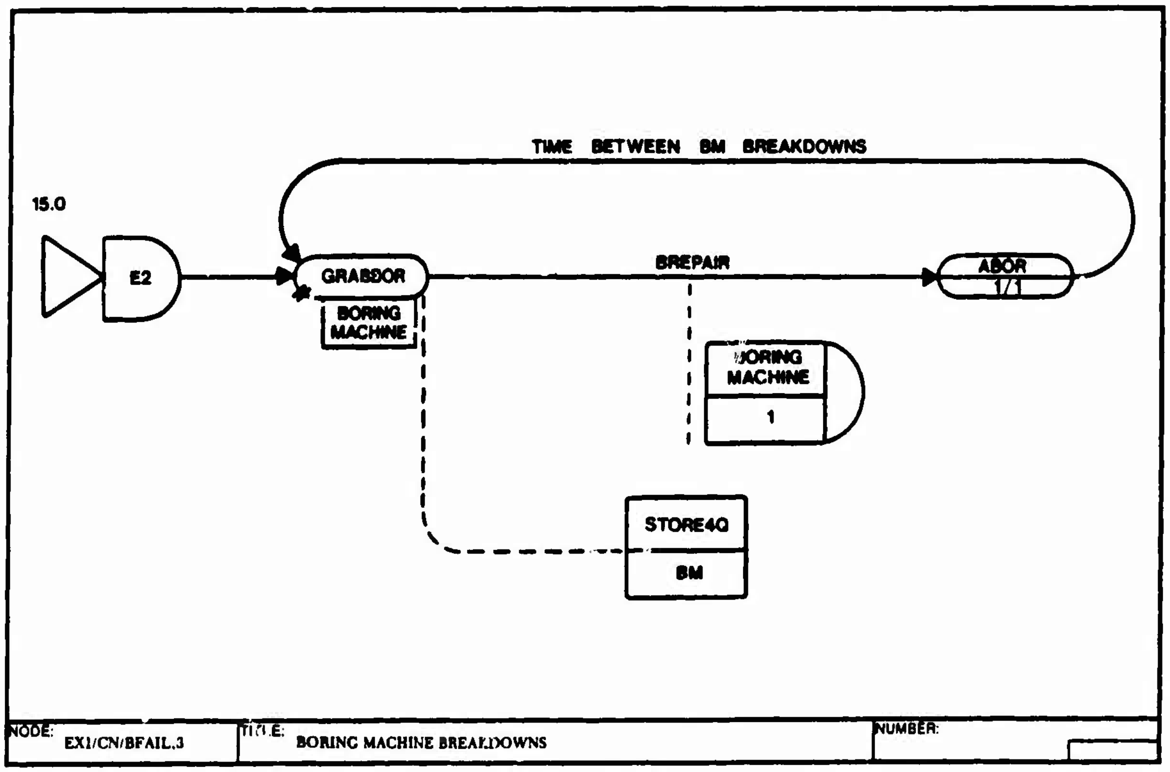


Figure 6-28. System Control Networks for Boring Machine Breakdowns

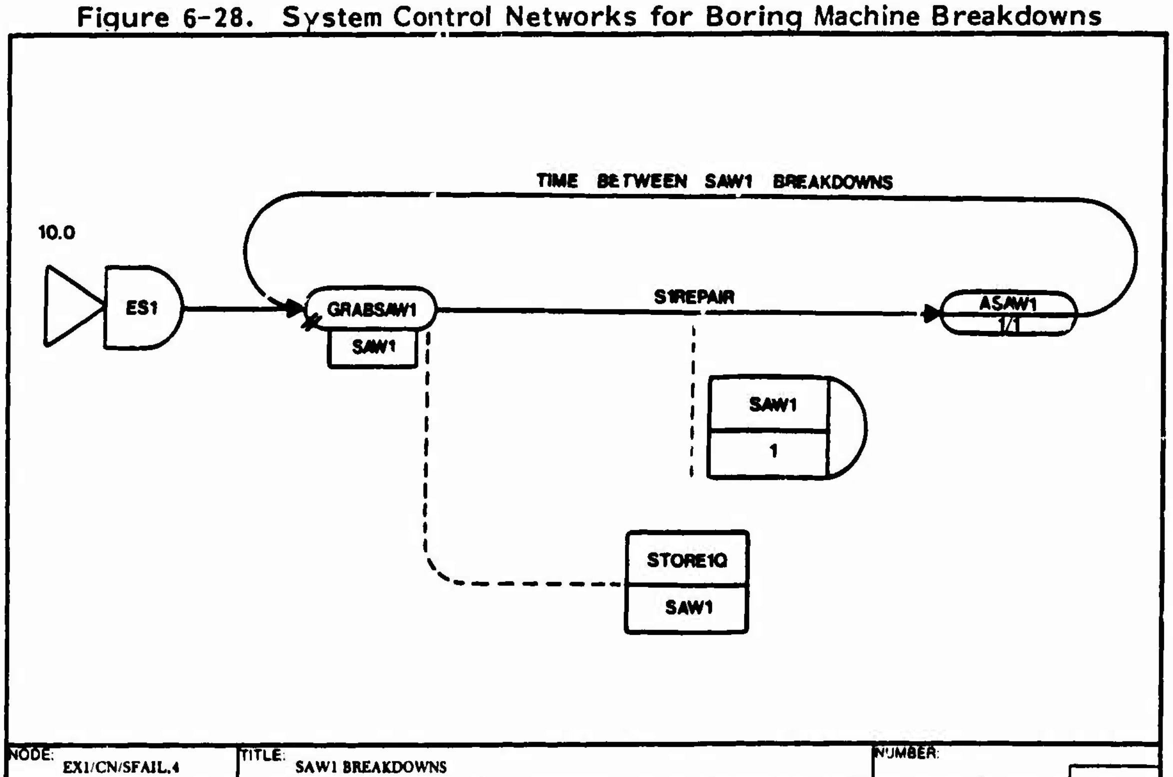


Figure 6-29. System Control Network for SAW1 Breakdowns

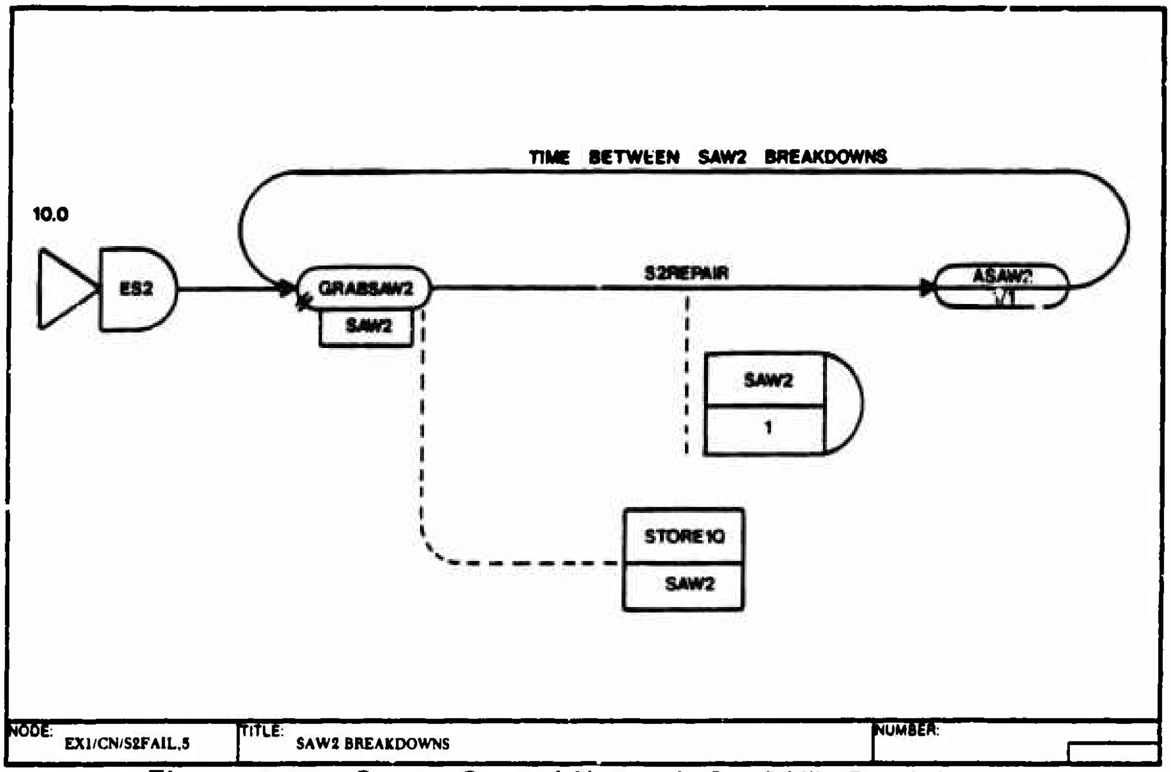


Figure 6-30. System Control Network for SAW2 Breakdowns

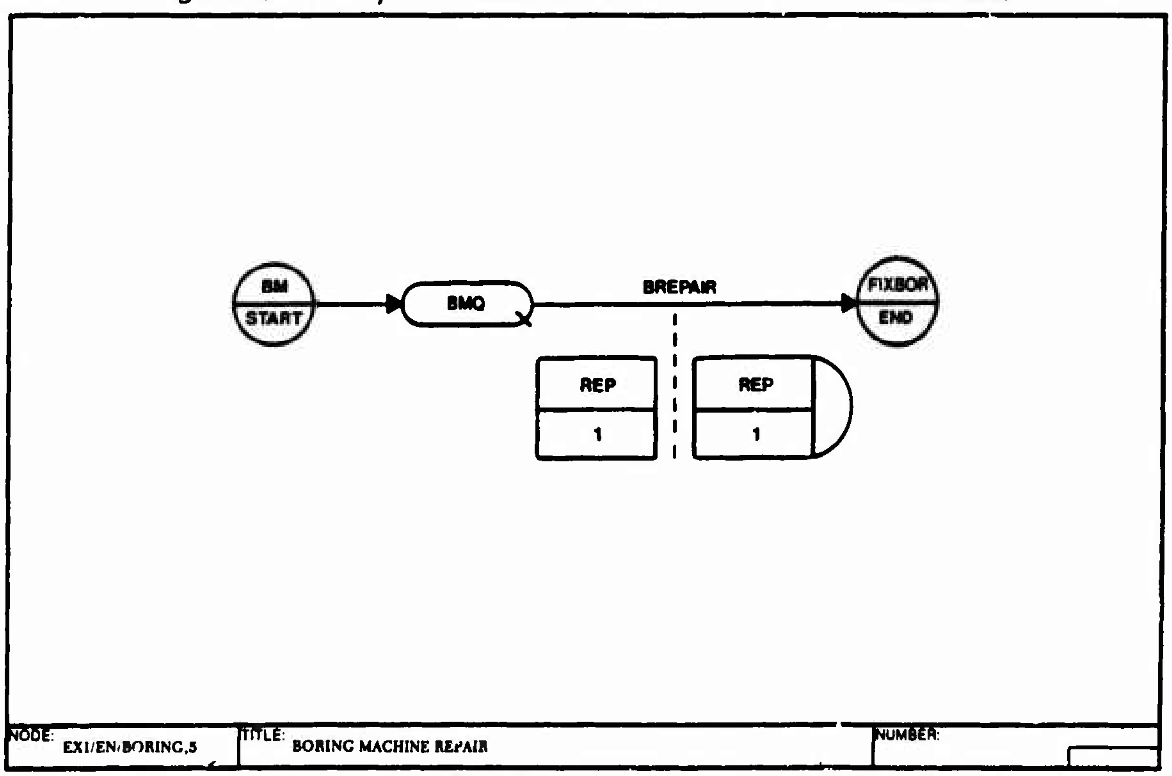


Figure 6-31. Entity Flow Network for Boring Machine Repair

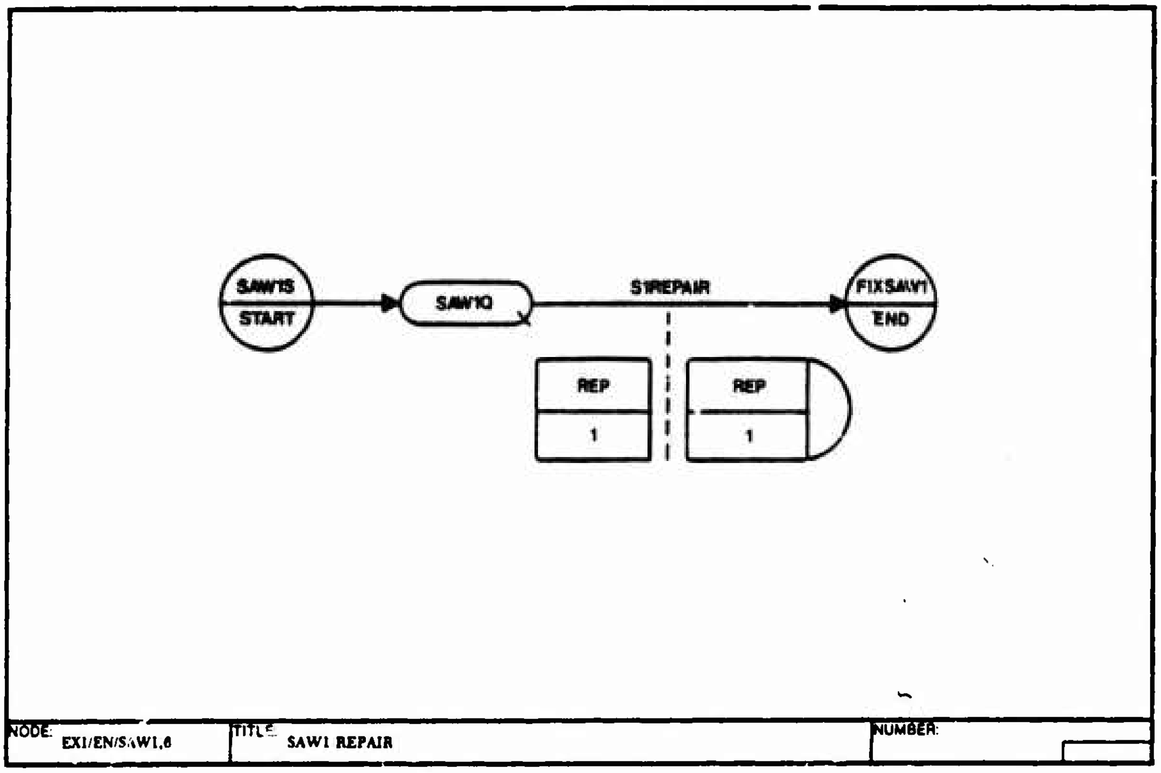


Figure 6-32. Entity Flow Network for SAW1 Repair

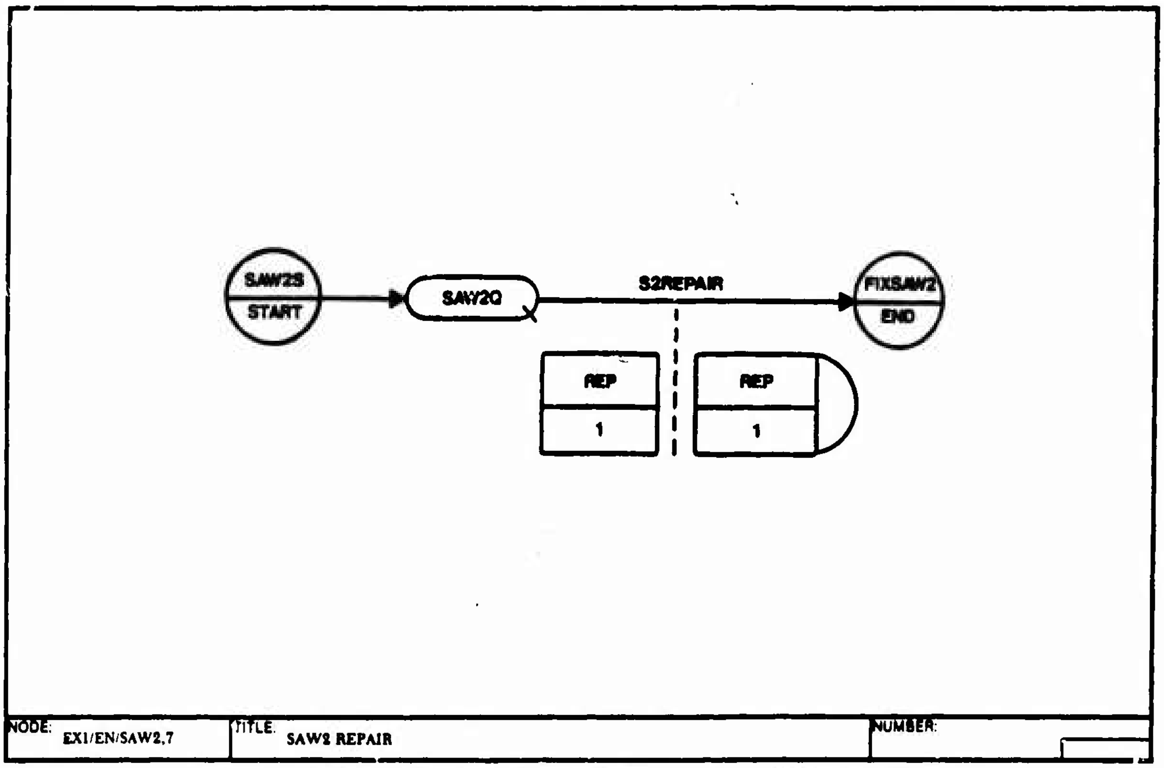


Figure 6-33. Entity Flow Network for SAW2 Repair

Each of the components of the System Control Networks must be characterized on System Control Node Attribute Definition forms. Figures 6-34, 6-35 and 6-38 through 6-40 show the System Control Node Attribute Definition forms for each of the respective System Control Networks. Figures 6-36, 6-37, and 6-41 through 6-43 contain the Entity Flow Node Attribute Definition Forms which supplement Figures 6-26, 6-27, and 6-31 through 6-33.

Let us now consider the concept of shift changes for the ABC manufacturing system. Shift changes are easily modeled with the use of ACTIVATE and DEACTIVATE nodes. Figure 6-44 illustrates the System Control Network for shift changes. The ENTER node at the left of the network represents the entry to the network at time zero. At this time, all of the resources in the network are activated. The SHIFT label on the ACTIVATE node denotes the fact that all resources in the model are to be activated each time this node is released. The ACTIVITY emanating from the ACTIVATE node has a duration of eight hours. At this time the DEACTIVATE node also containing the SHIFT specification is released deactivating all of the resources in the model. Emanating from the DEACTIVATE node is another ACTIVITY of duration sixteen hours which returns to the ACTIVATE node. This represents the cyclic activation and deactivation of all of the resources in the ABC manufacturing system corresponding to a single shift operation. Figure 6-45 shows the System Control Node Attribute Definition form for each of the components of the shift change System Control Network.

SYSTEM CONTROL NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES: IDENTIFIER	ATTRIBUTES: VALUE
PREEMPT	GRABDRILL1	REMAINING PROCESSING TIME	RESTART FROM STOP
		LIST OF ACTIVITIES	DRILLING1
		PRIORITY	1
ACTIVITY	D1 REPAIR	DURATION	REL(FIXDRILL1)
ACTIVITY	TIME BETWEEN DRILL1 BREAKDOWNS	DURATION	EXPONENTIAL(50)

NODE: EX1/CA/D1FAIL,1,1 | TITLE: DRILL1 BREAKDOWNS | NUMBER:

Figure 6-34. System Control Form for DRILL1 Breakdowns

SYSTEM CONTROL NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES: IDENTIFIER	ATTRIBUTES: VALUE
PREEMPT	GRABDRILL2	REMAINING ACTIVITY TIME	RESTART FROM STOP
		LIST OF ACTIVITIES	DRILLING 2
		PRIORITY	1
ACTIVITY	D2REPAIR	DURATION	EXPONENTIAL([illegible]0)
ACTIVITY	TIME BETWEEN DRILL2 BREAKDOWNS	DURATION	REL(FIXDRILL2)

NODE: EX1/CA/D2FAIL,2,1 | TITLE: DRILL2 BREAKDOWNS | NUMBER:

Figure 6-35. System Control Form for DRILL2 Breakdowns

ENTITY FLOW NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES IDENTIFIER	ATTRIBUTES VALUE
QUEUE	DRILL1Q	RANKING RULE	FIRST-IN, FIRST-OUT
		MAXIMUM NUMBER IN QUEUE	∞
		INITIAL NUMBER IN QUEUE	0
ACTIVITY	D1REPAIR	DURATION	EXPONENTIAL(25)

NODE: EX1/EA/DRILL1.3.1 TITLE: DRILL1 REPAIR NUMBER:

Figure 6-36. Entity Flow Form for DRILL1 Repair

ENTITY FLOW NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES IDENTIFIER	ATTRIBUTES VALUE
QUEUE	DRILL2Q	RANKING RULE	FIRST-IN, FIRST-OUT
		MAXIMUM NUMBER IN QUEUE	∞
		INITIAL NUMBER IN QUEUE	0
ACTIVITY	D2REPAIR	DURATION	EXPONENTIAL(25)

NODE: EX1/EA/DRILL2.4.1 TITLE: DRILL2 REPAIR NUMBER:

Figure 6-37. Entity Flow Form for DRILL2 Repair

SYSTEM CONTROL NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES: IDENTIFIER	ATTRIBUTES: VALUE
PREEMPT	GRABBOR	REMAINING ACTIVITY TIME	RESTART FROM STOP
		LIST OF ACTIVITIES	BORING
		PRIORITY	1
ACTIVITY	BREPAIR	DURATION	REL(FIXBOR)
ACTIVITY	TIME BETWEEN BM BREAKDOWNS	DURATION	EXPONENTIAL(100)

NODE: EX1/CA/BFAIL,3,1 TITLE: BORING MACHINE BREAKDOWNS NUMBER:

Figure 6-38. System Control Form for Boring Machine Breakdowns

SYSTEM CONTROL NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES: IDENTIFIER	ATTRIBUTES: VALUE
PREEMPT	GRABSAW1	REMAINING ACTIVITY TIME	RESTART FROM STOP
		LIST OF ACTIVITIES	SAWING 1
		PRIORITY	1
ACTIVITY	S1REPAIR	DURATION	REL(FIXSAW1)
ACTIVITY	TIME BETWEEN SAW1 BREAKDOWNS	DURATION	EXPONENTIAL(45)

NODE: EX1/CA/SFAIL,4,1 TITLE: SAW1 BREAKDOWNS NUMBER:

Figure 6-39. System Control Form for SAW1 Breakdowns

SYSTEM CONTROL NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES: IDENTIFIER	ATTRIBUTES: VALUE
PREEMPT	GRABSAW2	REMAINING ACTIVITY TIME	RESTART FROM STOP
		LIST OF ACTIVITIES	SAWING2
		PRIORITY	1
ACTIVITY	S2REPAIR	DURATION	REL(FIXSAW2)
ACTIVITY	TIME BETWEEN SAW2 BREAKDOWNS	DURATION	EXPONENTIAL(45)

NODE: EX1/CA/S2FAIL.5.1 | TITLE: SAW2 BREAKDOWNS | NUMBER:

Figure 6-40. System Control Form for SAW2 Breakdowns

ENTITY FLOW NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES: IDENTIFIER	ATTRIBUTES: VALUE
QUEUE	BMQ	RANKING RULE	FIRST-IN, FIRST-OUT
		MAXIMUM NUMBER IN QUEUE	∞
		INITIAL NUMBER IN QUEUE	0
ACTIVITY	BREPAIR	DURATION	NORMAL(25, 5.2)

NODE: EX1/EA/BORING.5.1 | TITLE: BORING MACHINE REPAIR | NUMBER

Figure 6-41. Entity Flow Form for Boring Machine Repair

ENTITY FLOW NODE ATTRIBUTE DEFINITION			
SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES	
		IDENTIFIER	VALUE
QUEUE	SAW1Q	RANKING RULE	FIRST-IN, FIRST-OUT
		MAXIMUM NUMBER IN QUEUE	∞
		INITIAL NUMBER IN QUEUE	0
ACTIVITY	S1REPAIR	DURATION	UNIFORM(5,30)
NODE: EX1/EA/SAW1,6,1	TITLE: SAW1 REPAIR		NUMBER:

Figure 6-42. Entity Flow Form for SAW Repair

ENTITY FLOW NODE ATTRIBUTE DEFINITION			
SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES	
		IDENTIFIER	VALUE
QUEUE	SAW2Q	RANKING RULE	FIRST-IN, FIRST-OUT
		MAXIMUM NUMBER IN QUEUE	∞
		INITIAL NUMBER IN QUEUE	0
ACTIVITY	S2REPAIR	DURATION	UNIFORM(5,30)
NODE: EX1/EA/SAW2,7,1	TITLE: SAW2 REPAIR		NUMBER:

Figure 6-43. Entity Flow Form for SAW2 Repair

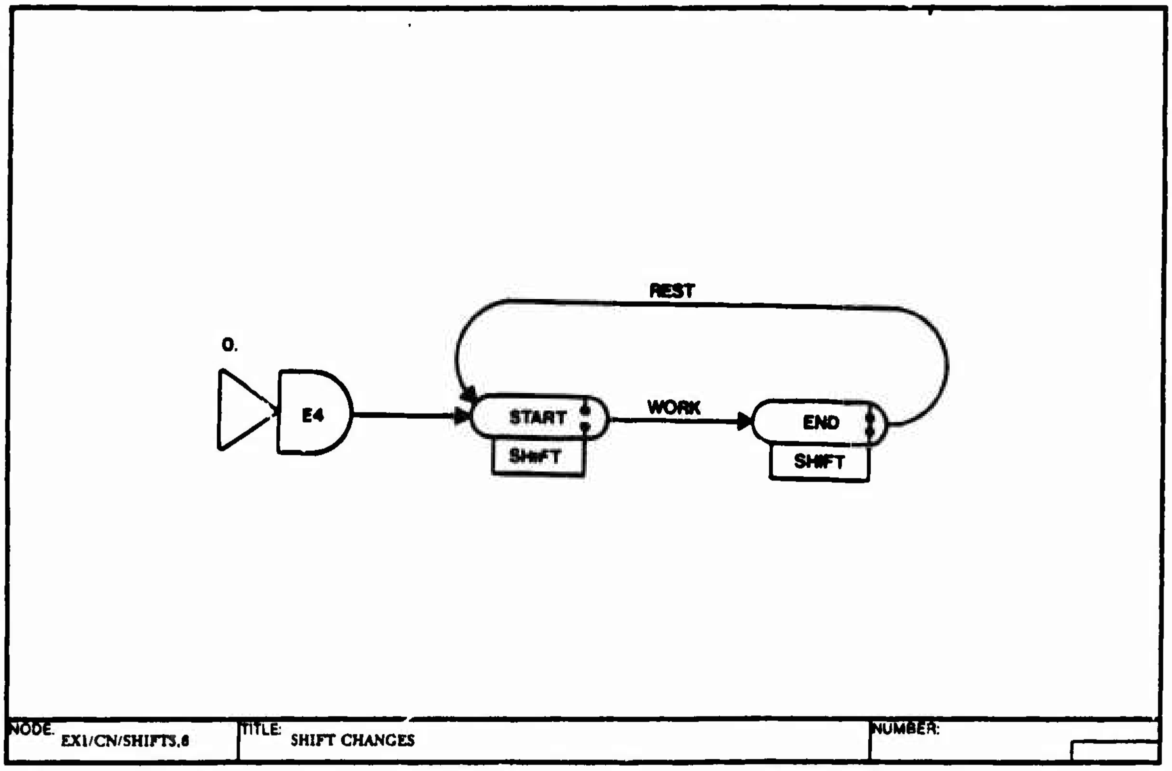


Figure 6-44. System Control Network for Shift Changes

ENTITY FLOW NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES: IDENTIFIER	ATTRIBUTES: VALUE
ACTIVITY	WORK	DURATION	8.0
ACTIVITY	REST	DURATION	16.0

NODE: EX1/CA/SHIFTS.6.1 | TITLE: SHIFT CHANGES | NUMBER:

Figure 6-45. System Control Form for Shift Changes

The human resour n the model take lunch breaks four hours into the working shift. Figure 6-46 illustrates the System Control Network for lunch breaks in the ABC system. The 4.0 above the ENTER node at the left of the network denotes the fact that the first entry to this network occurs at time 4.0. At this time a DEACTIVATE node, with the resource SHIFT specification, makes each of the resources in the system unavailable for use. Notice that although only the humans in the system take lunch breaks, since each of the machines requires a human operator, all of the resources are deactivated. The emanating ACTIVITY, EAT from the DEACTIVATE node represents the lunch break. This ACTIVITY releases the ACTIVATE node, also with the resource type SHIFT specified, which again activates each of the resources in the system. The ACTIVITY emanating from the ACTIVATE node which returns to the DEACTIVATE node is labeled WORK and represents the time until the next lunch break occurs. Figure 6-47 shows the System Control Node Attribute Definition form for each of the components in the lunch break System Control Network.

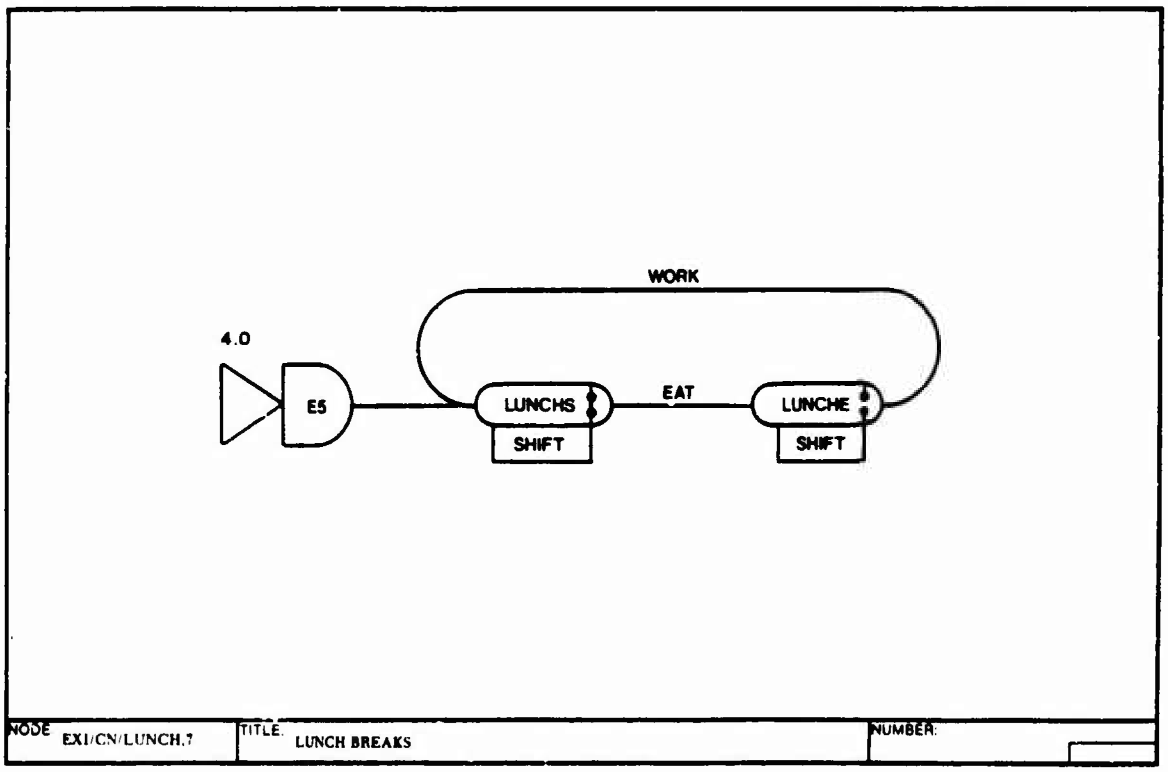


Figure 6-46. System Control Network for Lunch Breaks

SYSTEM CONTROL NODE ATTRIBUTE DEFINITION			
SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES	
		IDENTIFIER	VALUE
ACTIVITY	EAT	DURATION	1
ACTIVITY	WORK	DURATION	23
NODE: EX1/CA/LUNCH,7,1	TITLE: LUNCH BREAKS		NUMBER:

Figure 6-47. System Control Form for Lunch Breaks

Finally, we must model the arrival of parts to the system. This is accomplished with a CREATE node labeled PART, shown in Figure 6-48. The specification of the node location, PART1S, denotes the fact that parts created by this node are immediately inserted into the Entity Flow Network which begins with the BOUNDARY node labeled PART1S. The ACTIVITY emanating from the CREATE node is completed when the release of the node PART1X located in the part Entity Flow Network occurs. The System Control Node Attribute Definition form for this network is shown in Figure 6-49.

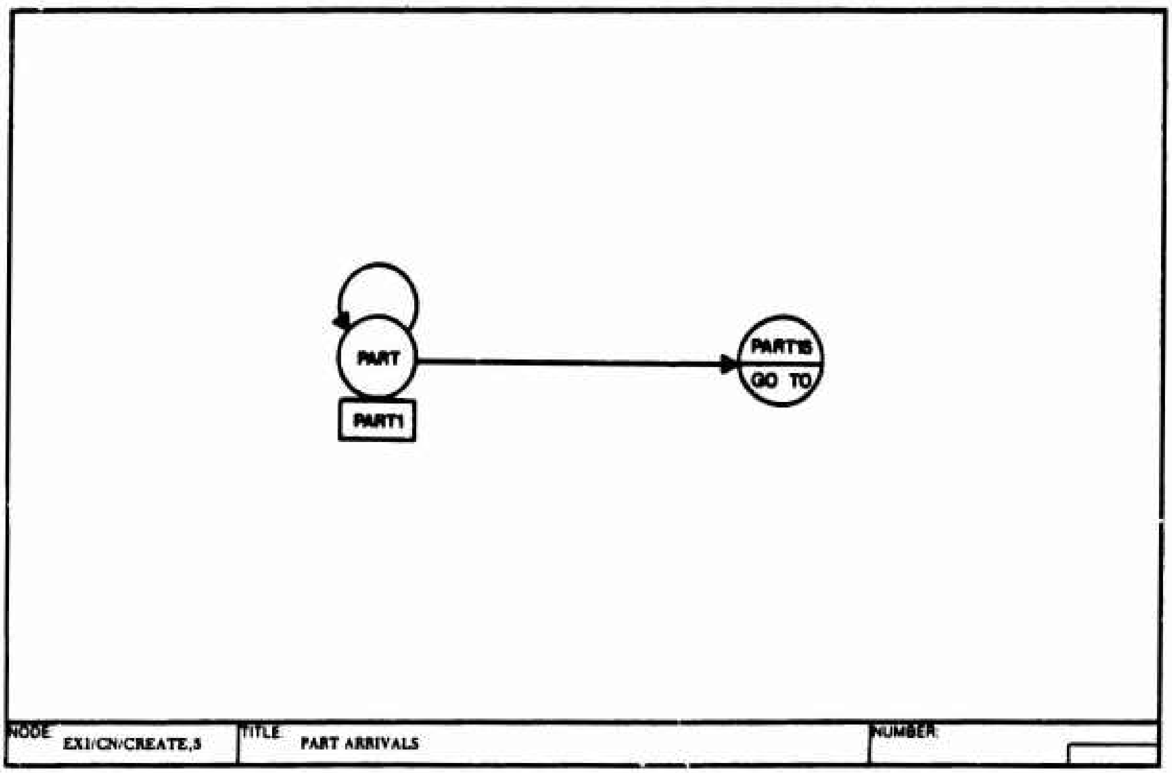


Figure 6-48. System Control Network for Part Arrivals

SYSTEM CONTROL NODE ATTRIBUTE DEFINITION

SYMBOL TYPE	IDENTIFYING NAME	ATTRIBUTES	
		IDENTIFIER	VALUE
CREATE	PART	TIME BETWEEN CREATIONS TIME OF FIRST CREATION	EXPONENTIAL(8.0) -

NODE: EX1/CA/CREATE.5.1	TITLE: PART ARRIVALS	NUMBER:

Figure 6-49. System Control Form for Part Arrivals

6.7 Model Initiation

Three additional forms must be prepared in order to complete the construction of an $IDEF_2$ model. The first of these forms defines the attributes for entities in the model. Figure 6-50 illustrates the Entity Definition form for the entity PART1 which was the primary entity type in the ABC manufacturing system. On the form, the identifier for each of the attributes is specified along with the initial value for each of those attributes.

The second form which must be prepared describes the initial disposition of the entities in the model. This is actually a means of specifying the initial conditions of the system. The way in which the system is initialized is a function of the disposition of entities through the system. By specifying the initial entity disposition within the model, we can effectively describe the initial status of all the activities and queues in the system. An example of the Initial Entity Disposition form for the entity PART1 is shown in Figure 6-51.

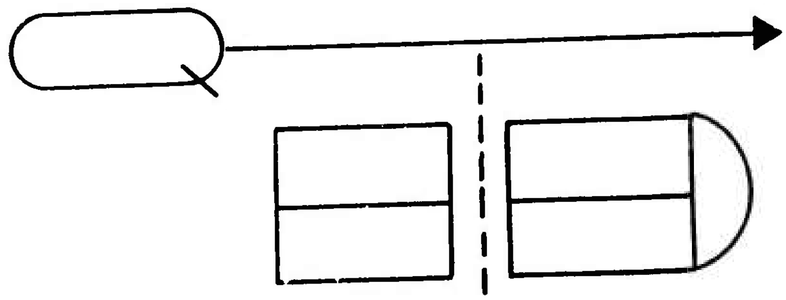
The third type of form which must be completed is the Initial Resource Disposition Form which specifies the quantity of each resource available and the initial disposition of the resources.

ENTITY DEFINITION		
	ATTRIBUTES	
ENTITY TYPE	IDENTIFIER	VALUE AT CREATION
PART1	TYPE BATCHNUM BATCHSIZE CTIME COST	1 1 5 TIME .08
NODE: EX1/ED/PART1.1	TITLE: ENTITY DEFINITION FOR PART1	NUMBER:

Figure 6-50. Entity Definition Form for PART1

INITIAL ENTITY DISPOSITION							
	ATTRIBUTES			RESOURCES			
ENTITY TYPE	IDENTIFIER	VALUE AT CREATION	NUMBER OF UNITS	TYPE	NUMBER	ACCUMULATE, QUEUE OR ACTIVITY NAME	REMAINING ACTIVITY TIME
PART1	TYPE BATCHNUM BATCHSIZE TIME COST	1 1 5 TIME 0.08	1	-	-	STORE1Q	-
NODE: EX1/EI/PART1.1	TITLE: INITIAL ENTITY DISPOSITION					NUMBER:	

Figure 6-51. Initial Entity Disposition Form



SECTION 7

DATA COLLECTION FOR IDEF MODELING

DATA COLLECTION FOR IDEF MODELING

7.1 Introduction

When analyzing or designing any system, it may be necessary to obtain or verify facts about the system or subject matter at hand. There are many sources of factual information. One might:

- read existing documents, using each table of contents and index to locate needed information.
- observe the system in operation, if it already exists.
- survey a large group of people, through questionnaires or other such means.
- talk to one or more "experts" who possess the desired knowledge.
- use whatever is already known by the author.
- guess or invent a hypothetical description, and ask readers to help bring it closer to reality.

Of all these methods, the most important is face-to-face interaction with an expert. Seldom will all existing information be written. Preconceived notions that are reflected in questionnaires are often faulty.

Obtaining information from an expert has been formalized in an interview process. This step by step process allows an interview to be conducted without unduly influencing the expert with information already obtained by the interviewers.

7.2 The Interview Process

The purpose of an interview is to gather information from an individual who possesses an expertise considered important to the analytical effort. There are four types of interviews that might be conducted during the course of performing the analysis phase of an IDEF project.

Four Types of Interviews

a) Fact finding for understanding current operations. This type of interview is used to establish the content of a Current Operations Model, or to help understand the existing environment.

b) Problem Identification to assist in the establishment of future requirements.

c) Solution Discussion regarding future system capabilities.

d) IDEF Author/Commenter Talk Session. This type of interview is used to resolve problems which have surfaced during the construction of an IDEF model.

The reason for identifying types of interview is that during the course of performing an actual interview, ingredients of each type appear. The respondent might tell the interviewer facts about a given system in terms of problems. Also, the respondent might identify problems in terms of solutions to the problems. The interviewer, by constantly classifying the respondents remarks, can obtain the maximum useful information from the interview.

7.3 The Interview Kit

It is recommended that a "standard" Interview Kit be used for recording the interview. It may be stored in an Interview File and it may be distributed to appropriate individuals. This distribution might include other members on the analysis team or even the interview respondent for corrections, additions, and deletions. The interview kit would contain:

1. Cover Page (Kit cover)
2. Interview and Record Follow-up
   a. Interviewer Name (IDEF Author Name)
   b. Interview Date (IDEF Diagram Date)
   c) Interview Duration (Start time, End time)
   d) Respondent Name
   e) Respondent Title and Organizational Responsibility
   f) Respondent Telephone Number and Extension
   g) Additional Sources of Information Identified
      1) Documents - Title and Location
      2) Other Interviewees - Name, Title, Organizational Responsibility, Address, Telephone number
   h) Essential Elements of Information - A Summary of the key points covered in the interview.
   i) Follow-up questions and/or areas of concern either not covered during the interview, or postponed
   j) New Terms for Project Glossary
3. Data Collection Sheets
4. Interview Agenda (Developed in preparation of Interview - This is covered in following section)
5. Interview Notes and Rough Diagrams

7.4 Interview Guidelines

There are five stages to the successful interview; each must be performed in order to assure that the most information is obtained and recorded in the least amount of time.

- Preparation
- Initialization
- Interview
- Termination
- Finalization

In each stage of an interview there are certain basic activities which must be performed. Additionally, associated with each stage, there exist psychological aids which will help the interviewer establish an atmosphere of professionalism and trust with the respondent.

7.4.1 Interview Preparation

By thinking through certain key interview needs before the interview, a more organized and efficient dialogue can be achieved. Preparation for an interview should contain, but is not limited to, the following activities:

1. Select Interviewee
   a) From areas of responsibility
   b) From recommendations of others
   c) From various levels of the organizational hierarchy - upper levels useful for "big picture," lower levels for detail information, and middle levels for bridging the gap
2. Make Appointment
   a) Short duration - ½ to 1 hour
   b) Not immediately before lunch, nor late afternoon
   c) Identify purpose of interview
   d) Explain interviewer role

3. Establish Tentative Agenda

   a) Topical areas - used as a foundation for interview (this helps prepare "broad general questions")

   b) Specific questions

4. Review applicable background information

5. Review appropriate terminology

6. Insure coordination with other interviews

   Check interview file to ascertain that the respondent has or has not been previously interviewed. If the interview is a follow-up interview, then examine the results of previous interviews.

7. Fill out Interview Record and Follow-up with pertinent information

8. Make out Interview Agenda

7.4.2 Interview Initialization

This stage of the interview is directed at establishing a rapport between the interviewer and respondent. The courtesy permitted by the respondent at the start of an interview is usually short. This time is important in motivating the respondent to help the interviewer. This stage of the interview should contain the following topics:

1. Provide respondent with a tangible means of introduction, e.g., a business card (this removes doubt on the part of the respondent as to how to pronounce or spell the interviewer's name and can therefore remove a frequent cause of respondent embarrassment)

2. Establish purpose of interview

   a) Expand on information provided in initial contact

   b) Establish point of view for the interview. Use interview type 1, 2, 3 or 4 as a basis.

   c) Establish purpose of the interview - even if the interview is a follow-up interview.

3. Establish the acceptability of note taking. The respondent may require assurance of confidentiality.

4. Establish the Expert/Author relationship - alleviate the fear that the interview will be used to tell the respondent how to do his job, or that the respondent's job is in jeopardy.

5. Start with broad general questions which will get the respondent talking - these should be based upon the topical areas identified in the agenda.

6. Assess the respondent's ability to provide pertinent information - if the information is too general or too detailed for the stage of the IDEF model being prepared, evaluate respondent's ability to contribute. Terminate the interview if necessary - it may be a waste of both the interviewer's and respondent's time.

7. Begin to formulate specific questions which complement the agenda.

8. Write, don't talk.

7.4.3 Conducting the Interview

While it is not useful to define questions to ask during an interview, it is possible to identify guidelines that should be considered during the interview. The first set of guidelines deals with the qualification of the information being obtained. The second set of guidelines relate to the stimulation of information flow.

Information Qualification: The human mind can comprehend at double the rate at which people speak. The danger in interviewing is that this rate difference is typically used by the listener to think about what should be said in response instead of about what is being said.

To assist the interviewer in thinking about what is being said, there is a series of questions which may be used to help the interviewer keep his mind on the information being provided:

- What supporting facts are being provided for the main points being discussed?

- How recent is the information?

- How complete is the information?
- Do I really understand what is being said?
- Is the level of detail being presented appropriate for my purpose?
- Are there areas being omitted?
- Has this information been discussed with someone else?
- How important is this information?
- Are side-topics being discussed?
- Has the interview viewpoint changed?

Information Flow Stimulation: The following set of guidelines can be used to stimulate the respondent into providing maximum reliable information:

- Keep extraneous comments and conversations to a minimum. The interview is used to obtain information, not make friends, or sell ideas.
- Be aware of the respondent's failure to identify problem areas in the environment. This may indicate that the respondent is not at ease with the interviewer.
- Provide the respondent time to think. Do not suggest answers or ask another question. A pause in the interview is useful to allow the respondent to recall vital pieces of information.
- Avoid outside distractions which tend to "uncouple" the train of thought. If at all possible, conduct interviews outside of the respondent's normal habitat.
- Be aware of internal distractions, signs that the respondent is not comfortable or at ease with the interview.
- Try to determine if the information being obtained is fact or opinion.
- Encourage elaboration by asking for a rephrasing or a summary of the information presented.

- Ascertain the respondent's background and association with the subject matter being discussed. Valuable insight into the respondent's remarks can be obtained knowing his relationship to the organization and existing systems.

- Do not enter into or encourage sarcasm and humor.

- Do not mention or discuss any interview with another person.

- Record all questions asked by the respondent. The interviewer should answer all questions except those dealing with user-organization management, plans, or personalities.

- Show interest in what the respondent is saying.

- Concentrate on the unfamiliar and difficult aspects of the subject being discussed. Avoid the obvious.

- Be alert for the inconsistent or incorrect use of words. Ask for definitions for any unfamiliar or questionable term. Record the definition for the project glossary.

- Do not contradict the respondent even if facts do not support what is being said. Use the Kit Cycle to resolve such conflicts.

- Be humble. The respondent is the expert, not the interviewer.

- Postpone subjects which cannot be fully covered within the agreed upon time frame. Do not extend the interview time, but rather make another appointment.

- Appreciate different opinions on the same subject. Use IDEF to show these opinions and to resolve conflicts.

- Stimulate the respondent with pertinent open ended questions.

7.4.4 Termination

The interview should be terminated for any of the four following reasons:

a) The information being obtained in the interview is not appropriate.

b) The time limit has been reached.

c) The interviewer has been saturated with information.

d) There is a clash of personalities between the interviewer and the respondent.

Depending upon the cause of termination, the following topics should be considered during the termination of the interview:

- The interview should not be closed abruptly, but rather should end with a few minutes of informal discussion.
- The main points of the interview should be summarized.
- Areas of concern which have been postponed or not covered should be identified.
- A follow-up interview, if necessary, should be arranged.
- The respondent should be asked to recommend other persons who should be interviewed.
- If the interview notes are to be reviewed by the respondent prior to distribution, this fact should be mentioned during the termination.
- The respondent should be thanked for his time and effort.

7.4.5 Finalization

This stage of the interview is directed at assuring that the information obtained during the interview is properly recorded and disseminated to the project team. The vehicle used to accomplish the finalization of an interview is the Interview Kit. If note taking was not permitted by the respondent, the interviewer should, upon termination of the interview, immediately write down the salient points discussed. Finalization of the interview includes the following:

a) Identify additional sources of information.

b) Summarize the Essential Elements of Information.

c) Identify new terms for the project glossary.

d) List the follow-up questions and areas of concern either postponed or not covered during the interview.

e) Complete Data Collection Sheets.

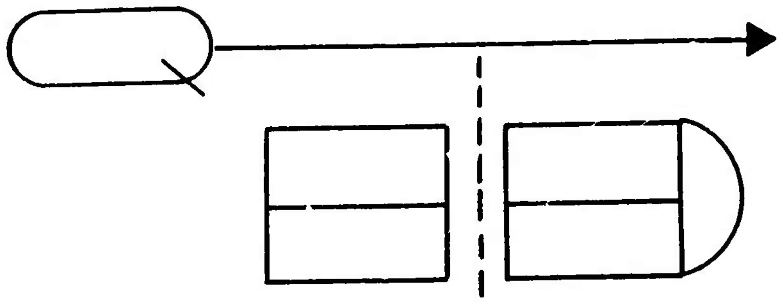
f) Expand on any notes with any information recalled during the review.

g) Prepare rough IDEF diagrams that reflect the information obtained.

h) Identify any assumptions being made or any items which are not clear.

i) Publish and distribute the Interview Kit.

j) Address the names, areas of expertise, phone numbers, and addresses of persons mentioned in the interview.



SECTION 8

$IDEF_2$ GLOSSARY

IDEF$_2$ GLOSSARY

ACCUMULATE NODE

A node used in IDEF$_2$ Entity Flow Networks or System Control Networks to model the grouping of entities and to separate activities.

ACTION (RESOURCE DISPOSITION TREES)

The component of an IDEF$_2$ Resource Disposition Tree which specifies what will be done with the resource whose disposition is being determined.

ACTIVATE NODE

A node used in IDEF$_2$ System Control Networks to make a resource type available for use.

ACTIVITY

An arrow used in IDEF$_2$ Entity Flow Networks and System Control Networks to represent time delay.

ACTIVITY SELECTION RULE

A rule which determines which activity an entity will perform when a selection is required.

ALTER NODE

A node used in IDEF$_2$ System Control Networks to make a discrete change in the capacity of a resource type.

ARROW

The symbol used to represent activities and branches in IDEF$_2$ models.

ASSEMBLY NODE

A node used in IDEF$_2$ Entity Flow Networks to model the physical joining of multiple entities.

ASSIGNMENT NODE

A node used in $IDEF_2$ Entity Flow Networks and System Control Networks to prescribe values to system variables or entity attributes.

ATTRIBUTE

A quantifiable characteristic of an $IDEF_2$ entity.

ATTRIBUTE SAVE CRITERION

A rule which determines which attributes will be assigned to entities which leave $IDEF_2$ accumulate nodes or assembly nodes.

BALKING

One of the two actions which may be specified for a full condition for an $IDEF_1$ queue node. The balking entity is routed to another $IDEF_2$ node.

BLOCKING

One of the two actions which may be specified for a full condition for an $IDEF_2$ queue node. The blocked entity is held in the previous activity and prevented from continuing until the queue is unblocked.

BOUNDARY NODE

An $IDEF_2$ node which marks the beginning or end of a segment of an $IDEF_2$ Entity Flow Network or System Control Network. Boundary nodes are START nodes, END nodes, or GO TO nodes.

BRANCH

Probabilistic or conditional. Branching is a direct transfer between nodes in $IDEF_2$ Entity Flow Networks and System Control Networks, with no passage of simulated time.

BRANCH SELECTION RULE

A rule which determines which branches and how many branches will be initiated when a selection is required.

CENTRAL PROCESSING UNIT (CPU) MODEL

The symbol used in an $IDEF_2$ Facility Diagram to represent the Central Processing Unit of a computer system.

COMMUNICATIONS LINK SYMBOL

The symbol used in an $IDEF_2$ Facility Diagram to represent any data line or path over which data may be transmitted.

COMPUTER SYMBOL

The symbol used in an $IDEF_2$ Facility Diagram to represent a computer system, including all units controlled by the system.

CONDITIONAL BRANCHING

A type of branching allowed in $IDEF_2$ Entity Flow Networks and System Control Networks where an entity takes a branch only if a specified condition is met.

CONTINUOUS VARIABLE

An $IDEF_2$ attribute or variable whose value changes continuously over time according to a prescribed equation.

CONVEYOR

A key word used with $IDEF_2$ activate/deactivate nodes to begin or end the movement of a conveyor.

CREATE NODE

A node used in $IDEF_2$ System Control Networks to generate the entities which flow through the system being modeled.

DEACTIVATE NODE

A node used in $IDEF_2$ System Control Networks to make a resource type unavailable for use.

DEFAULT VALUE

A value which an $IDEF_2$ variable or parameter will have if the modeler does not specify another.

DESTINATION SYMBOL

A symbol used in an $IDEF_2$ Facility Diagram to designate where entities exit the system.

DETECT NODE

A node used in $IDEF_2$ System Control Networks to detect crossings of threshold values by continuous variables.

DISCRETE VARIABLE

An $IDEF_2$ attribute or variable whose value changes at discrete points in time.

ENTER NODE

A node used in $IDEF_2$ System Control Networks to gain entry into the network by denoting the first occurrence of an event.

ENTITY

Represents a conceptual or physical transaction that moves through $IDEF_2$ Entity Flow Networks or System Control Networks.

ENTITY DEFINITION FORM

An $IDEF_2$ form used to identify an entity's attributes and their respective values for reference when an entity is created.

ENTITY FLOW NETWORK

A graphical representation of all possible paths that an entity can take through a system.

ENTITY FLOW NODE ATTRIBUTE DEFINITION FORM

An $IDEF_2$ form used to specify the unique characteristics of an Entity Flow Node.

ENTITY FLOW SUBMODEL

The means by which $IDEF_2$ describes the flow of products or information (entities) through a system, consisting of a graphical Entity Flow Network and supporting information contained on forms.

EVENT

An occurrence in time which may change the state of a system.

FACILITY

The combination of manpower, machines and computers working together to serve a specific function or perform a particular service.

FACILITY COMPONENT ATTRIBUTE DEFINITION FORM

An $IDEF_2$ form used to identify the attributes and their values of components of a facility.

FACILITY DIAGRAM

An $IDEF_2$ graphical representation of the spatial relationships between components of a system.

FACILITY SUBMODEL

The means by which $IDEF_2$ describes the operational arrangement of components of the system used to produce the final products or information.

$IDEF_2$ MODELING

A technique used to describe the time varying behavior of manufacturing systems in a way that can be analyzed via computer simulation.

INITIAL CONTINUOUS ATTRIBUTE DEFINITION FORM

An $IDEF_2$ form used to define initial values of continuous variables.

INITIAL ENTITY DISPOSITION FORM

An $IDEF_2$ form used to define the initial characteristics of entities and their locations in an Entity Flow Network or System Control Network.

INITIAL RESOURCE DISPOSITION FORM

An $IDEF_2$ form used to define the initial characteristics of resources and their initial allocations.

INPUT SYMBOL

The symbol used in an $IDEF_2$ Facility Diagram to represent any source of input to a computer system.

INSPECTION STATION SYMBOL

The symbol used in an $IDEF_2$ Facility Diagram to represent any location where entities are inspected.

MACHINE SYMBOL

The symbol used in an $IDEF_2$ Facility Diagram to represent any device which performs activities on entities.

MAGNETIC TAPE SYMBOL

The symbol used in an $IDEF_2$ Facility Diagram to represent any magnetic tape used by a computer system.

MANPOWER SYMBOL

The symbol used in an $IDEF_2$ Facility Diagram to represent any type of personnel including operators, repairmen, foremen, and managers.

MATCH NODE

A node used in $IDEF_2$ Entity Flow Networks and System Control Networks to match entities residing in specified queue nodes that have equal values of a specified attribute.

MATERIAL HANDLING EQUIPMENT SYMBOL

Any of the symbols used in an $IDEF_2$ Facility Diagram to represent devices used in the movement of materials whether stand alone devices, devices that move on tracks, or fixed path devices.

MEMORY SYMBOL

The symbol used in an $IDEF_2$ Facility Diagram to represent any mass storage device used by a computer system.

MODEL EXECUTION (OR SIMULATION)

The representation of the dynamic behavior of the system by moving it from state to state in accordance with well defined operating rules.

MODEL NUMBERING CONVENTIONS

Rules which define a numbering system for identification of elements of an $IDEF_2$ model.

NNACT (NAME)

An $IDEF_2$ function used to count the number of entities in activity "NAME" at the current simulated time.

NNCNT (NAME)

An $IDEF_2$ function used to count the number of entities that have completed activity "NAME."

NNQ (NAME)

An $IDEF_2$ function used to count the number of entities in queue "NAME" at the current simulated time.

NODE

Any of a variety of symbols used to construct $IDEF_2$ networks.

NODE RELEASE

The process whereby an entity is released from an $IDEF_2$ node after conditions for the release have been met.

NUMRES (NAME)

An $IDEF_2$ Function used to count the number of units of resource "NAME" available at the current simulated time.

PERFORMANCE MEASURE

Production rate, waiting time, utilization or any other statistic generated through model execution used to evaluate performance.

PREEMPT NODE

A node used in $IDEF_2$ System Control Networks to remove resources from the activity in which they are currently engaged, for employment in another activity.

PRINTED OUTPUT SYMBOL

The symbol used in an $IDEF_2$ Facility Diagram to represent any hard copy report used or generated.

PROBABILISTIC BRANCHING

A type of branching allowed in $IDEF_2$ Entity Flow Networks and System Control Networks where an entity takes a branch a prescribed percentage of the time.

PROBABILITY DISTRIBUTION

Any rule which assigns a probability to each possible value of a random variable. $IDEF_2$ contains 10 such rules: Exponential, Uniform, Weibull, Triangular, Normal, Lognormal, Erlang, Gamma, Beta, Poisson.

PUNCHED TAPE SYMBOL

The symbol used in an $IDEF_2$ Facility Diagram to represent any punched tape used or produced by a computer system.

QUESTION (RESOURCE DISPOSITION TREES)

The component of an $IDEF_2$ Resource Disposition Tree which asks a question to determine what action will be taken or which question will be asked next.

QUEUE NODE

A node used in $IDEF_2$ Entity Flow Networks which represents a place where entities can wait for resources or other entities.

QUEUE RANKING RULE

An $IDEF_2$ rule which determines the relative order in which entities wait in a queue node. $IDEF_2$ contains the following rules: FIFO, LIFO, Low-Value first based on an entity attribute value, High-Value first based on an entity attribute value.

QUEUE SELECTION RULE

A rule which determines which queue an entity will enter following a selector or which queue in front of a selector it will choose from.

RESOURCE

Items such as equipment, manpower, documents, etc. necessary to perform an activity.

RESOURCE ALLOCATION NODE

A node used in $IDEF_2$ Entity Flow Networks to assign resources to entities in order to perform activities.

RESOURCE ALLOCATION RULE

A rule which determines which entity will be allocated a resource when the action of allocation is specified in a Resource Disposition Tree.

RESOURCE DISPOSITION SUBMODEL

The means by which $IDEF_2$ describes the disposition of resources when they become available.

RESOURCE DISPOSITION TREE

A graphical representation of the procedure by which the disposition of a freed resource is determined, constructed for each resource type.

RESOURCE PREEMPTION RULE

A rule which determines which entity will lose a resource when the action of preemption is specified in a Resource Disposition Tree.

RESOURCE RELEASE NODE

A node used in $IDEF_2$ Entity Flow Networks and System Control Networks to release a resource upon completion of an activity.

RESOURCE SELECTION RULE

A rule which determines which set of resources specified for an activity will be chosen when a selection is required.

SCRAP PRODUCTION SYMBOL

The symbol used in an $IDEF_2$ Facility Diagram to represent any area where scrap is produced.

SELECTOR NODE

A node used in $IDEF_2$ Entity Flow Networks and System Control Networks to make decisions concerning: queue selection, activity selection, resource selection, or branch selection.

SHIFT

A key word used with $IDEF_2$ Activate/Deactivate nodes to control resources at the beginning/end of a shift.

SOURCE SYMBOL

A symbol used in an $IDEF_2$ Facility Diagram to designate where material enters the system.

STORAGE SYMBOL

A symbol used in an $IDEF_2$ Facility Diagram to designate the location of racks, totes, bins, or any other device used to store material.

SUBMODEL

A portion of an $IDEF_2$ model: Facility, Entity Flow, Resource Disposition, or System Control.

SYSTEM CONTROL NETWORK

A graphical representation of activities or conditions which may affect the status of the system and flow of entities, but do not directly cause entity flow.

SYSTEM CONTROL NODE ATTRIBUTE DEFINITION FORM

An $IDEF_2$ form used to specify the unique characteristics of a System Control Node.

SYSTEM CONTROL SUBMODEL

The means by which $IDEF_2$ describes the occurrence of activities which may affect the status of the system but are not associated with entity flow.

TERMINAL SYMBOL

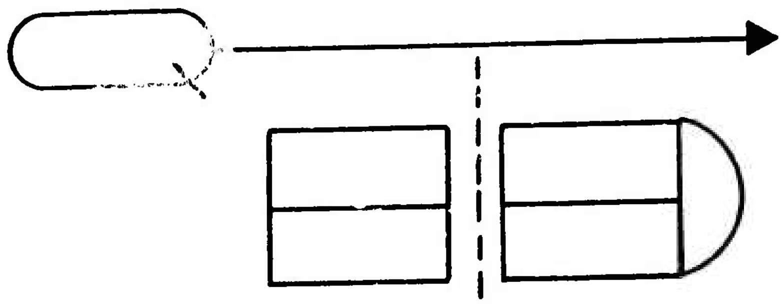
A symbol used in $IDEF_2$ Facility Diagrams to represent any computer access device such as CRTs, teletypes, etc.

TIME

$IDEF_2$ variable representing current simulated time.

UTILIZATION

A performance measure which provides information concerning the portion of time a resource or activity was actively engaged in service to an entity.



SECTION 9

$IDEF_2$ INDEX OF TERMS

IDEF$_2$ INDEX OF TERMS

Appendix A

ANALYSIS WITH $IDEF_2$ MODELS

What we have prepared in the preceding pages is an $IDEF_2$ model of a hypothetical manufacturing facility, which describes the dynamics of the system. If we had actually built the model for the purpose described in the problem statement, we would now have to select an analysis vehicle to obtain the measures of system performance defined earlier. The information that is required to build $IDEF_2$ models allows a characterization of dynamic systems which lends itself to simulation analysis. Returning to the purpose for the model, recall that the ABC Manufacturing Company's management wanted a tool which would help them to predict the capacity of their manufacturing system. With the help of appropriate software, the system that was modeled in this example could be simulated and measures of the system's performance could be obtained. The performance measures which the ABC management could expect from this model include machine utilizations, queue sizes, the production rate of the system, the amount of time parts spend in the system (makespan), the amount of time spent waiting for processing, and the amount of time actually spent being processed. By performing multiple runs of a simulation of the system, alternative system designs could be evaluated by comparing the values obtained for the performance measures just mentioned. Figure A-1 illustrates a typical output report that could be prepared by simulating the $IDEF_2$ model of the ABC manufacturing system.

Obtaining measures of system's performance is a typical purpose for constructing dynamic models. However, another benefit derived from building $IDEF_2$ models of manufacturing systems is an increased knowledge of the system's behavior. In order to construct an $IDEF_2$ model, the analyst must thoroughly understand the operation of the manufacturing system and must be able to relate this understanding to others via the $IDFF_2$ syntax.

The true value in modeling efforts comes not in the specific numbers that are obtained by analysis; rather, it is the increased knowledge and understanding of the behavior of the system which provides the real rewards of the modeling efforts. Without understanding, numbers are meaningless. With understanding, however, numbers can provide insight to the solution of complex problems. $IDEF_2$ makes the first step in problem solution possible by helping to increase understanding and by providing a structure for the analysis of dynamic systems' behavior.

Production Summary

Production Rate: 42 pieces per shift

No. Shifts Simulated: 30

Statistics for PART1

	Average	Minimum	Maximum
Time in System.	.45	.27	1.42
Time Waiting:	.22	.17	.97
Time Being Processed:	.23	.08	1.05
Time Between Departures:	1.28	.65	3.92

Machine Statistics

Machine Type	*Avg. No. Busy*	*Avg. Queue Size*
Saw	1.22	3.54
Drill	1.75	7.82
Boring Machine	.63	4.20

Figure A-1. Typical Output Report from $IDEF_2$ Model Simulation

Appendix B

SYNOPSES OF VOL. I THROUGH VOL. XI

Vol. I Architecture Part II Accomplishments

This volume presents an overview of the Project, individual task overviews and recommendations for future ICAM projects.

Vol. II Architecture - A Structured Approach to Manufacturing

The ICAM approach to better understanding, communicating and analyzing manufacturing through the development and use of the Architecture is explained in this volume. The reasoning for the development of Architecture, the components, application and benefits are described in detail.

Vol. III Integration Using Architecture

Integration of Manufacturing to improve productivity and reduce manufacturing costs is the goal of the ICAM program. This volume details the procedures for integrating systems and the benefits to be gained from integrating "AS IS" models prior to building "TO BE" models. Two subsystem function models integrated with the manufacturing function model under this contract are presented:

1. Integration of Manufacturing Control - Materials Management Subsystem $IDEF_0$ Model into the Manufacturing $IDEF_0$ Model. (MCMM0/MFG0)

2. Integration of Sheet Metal Center $IDEF_0$ Model into the Manufacturing $IDEF_0$ Model. (SMC0/MFG0)

Vol. IV Function Modeling Manual ($IDEF_0$)

This volume is the manual given to students learning the $IDEF_0$ function modeling methodology for describing manufacturing functions.

Vol. V Information Modeling Manual ($IDEF_1$)

This volume is the manual given to students learning the $IDEF_1$ Information Modeling Methodology for describing manufacturing information.

Vol. VI Dynamics Modeling Manual ($IDEF_2$)

This volume is the manual given to students learning the $IDEF_2$ Dynamics Modeling Methodology for describing the time varying behavior of functions and information.

Vol. VII Composite Function Model of "Manufacture Product" (MFG0)

This volume presents the composite view depicting manufacturing as it exists today in the form of an "AS IS" Function Model of Manufacturing.

Vol. VIII Composite Function Model of "Design Product" (DESIGN0)

This volume presents the composite view depicting the design process as it exists today in the form of an "AS IS" Function Model of Design.

VOL. IX Composite Information Model of "Manufacture Product" (MFG1)

This volume presents the composite view depicting manufacturing as it exists today in the form of an "AS IS" Information Model of Manufacturing. Because of its voluminous size, this model has been printed in several parts to facilitate ease of handling.

Vol. IX, Part 1 MFG Development

This part explains the process of development that the MFG1 model has undergone.

Vol. IX, Part 2 MFG1 Model

The MFG1 model diagrams and attribute class definitions comprise this part.

Vol. X Dynamics Model of a Sheet Metal Center Subsystem (SMC2)

This volume contains an $IDEF_2$ Dynamics Model of the sheet metal center at Northrop Corporation's Mariposa facility. It demonstrates the application of the $IDEF_2$ Dynamics Modeling Methodology.

Vol. XI ICAM Library Maintenance and Distribution Procedures

Contained in this volume are procedures developed to allow for the proper dissemination of the material generated under the ICAM program. They are the ICAM Program Library User's Guide and ICAM Program Library Maintenance Procedures.

Appendix C

ARCHITECTURE PART II - FINAL REPORT

DOCUMENT REQUEST ORDER FORM

VOLUME I - Architecture Part II Accomplishments

VOLUME II - Architecture - A Structured Approach to Manufacturing

VOLUME III - Integration Using Architecture

VOLUME IV - Function Modeling Manual ($IDEF_0$)

VOLUME V - Information Modeling Manual ($IDEF_1$)

VOLUME VI - Dynamics Modeling Manual ($IDEF_2$)

VOLUME VII - Composite Function Model of "Manufacture Product" (MFG0)

VOLUME VIII - Composite Function Model of "Design Product" (DESIGN0)

VOLUME IX - Composite Information Model of "Manufacture Product" (MFG1)

Part 1 - MFG1 Development

Part 2 - MFG1 Model

VOLUME X - Dynamics Model of a Sheet Metal Center Subsystem (SMC2)

VOLUME XI - ICAM Library Maintenance and Distribution Procedures

---

Submit document requests to: ICAM Program Library
AFWAL/MLTC
Wright-Patterson AFB, OH 45433

NAME ______________________________

TITLE ______________________________

COMPANY ______________________________

DEPARTMENT ______________________________

MAIL CODE ______________________________

STREET OR P.O. BOX ______________________________

STATE ____________________ ZIP ____________

PHONE # ______________________________

*U.S.Government Printing Office: 1981 — 559-006/5046