IDEF1 Information Modeling

A Reconstruction of the Original Air Force
Wright Aeronautical Laboratory
Technical Report AFWAL-TR-81-4023

Dr. Richard J. Mayer, Editor

Knowledge Based Systems, Inc.

IDEF1 Information Modeling

A Reconstruction of the Original Air Force
Wright Aeronautical Laboratory
Technical Report AFWAL-TR-81-4023

Dr. Richard J. Mayer, Editor

Knowledge Based Systems, Inc.

Knowledge Based Systems, Inc.
One KBSI Place
1408 University Drive East
College Station, Texas 77840-2335
(409) 260-5274

One KBSI Place
1408 University Drive East
College Station, Texas 77840-2335
(409) 696-7979

Table of Contents

List of Figures

Foreword

Historical Perspective

IDEF1 can be viewed as a method for both analysis and communication in establishing CIM requirements. However, IDEF1 is primarily focused on support of the task of establishing the requirements for what information is or should be managed by an enterprise. In CIM applications, IDEF1 is generally used to: 1) identify what information is currently managed in the organization, 2) identify which of the problems identified during the needs analysis are caused by lack of management of appropriate information, and 3) specify what information will be managed in the "TO-BE" CIM implementation.

The IDEF1, Information Modeling Method, derives its foundations from three primary sources: The Entity-Link-Key-Attribute (ELKA) method developed by Hughes Aircraft Co., the Entity-Relationship (ER) method proposed by Peter Chen, and Codd's Relational Model. The original intent of IDEF1 was to capture what information exists or should be managed about objects falling within the scope of an enterprise. Thus, the IDEF1 perspective of an information system is one which includes not only the automated component, or the computer, but also includes humans, filing cabinets, telephones, etc. A design goal for IDEF1 was that it not be a database design method. At the time of the IDEF1 development, it was the opinion of the database community that what was needed was a way for organizations to analyze and clearly state their information resource management needs and requirements. This was the motivation for the development of IDEF1. Rather than a design method, IDEF1 is an analysis method used to identify:

1. What information is collected, stored, and managed by the enterprise.

2. The rules governing the management of information.

3. Logical relationships within the enterprise reflected in the information.

4. Problems resulting from the lack of good information management.

The results of information analysis can be used by strategic and tactical planners within the enterprise to leverage their information assets to achieve competitive advantage. Part of their plans may include the design and implementation of automated systems which can more efficiently take advantage of the information available to the enterprise. IDEF1 models provide the basis for those design decisions. IDEF1, then, is not used to design a database; rather, it is used to provide managers with the insight and knowledge required to establish good information management policy.

The popularity of the IDEF1 method is principally due to its focus on enhancing human-to-human communication. Over the years, a variety of automated tools have emerged that support the application of this method. As these tools become integrated with traditional Computer Aided Software Engineering (CASE) environments, a new world of opportunities is emerging. In this new order, Frameworks of Systems architecture methods including IDEF1 as a component will feed enterprise repositories. These repositories (or knowledge bases) will enable the realization of integrated systems of a scale presently unattainable.

To date, one of the small but important missing pieces in this picture has been the availability of the original descriptions of the IDEF methods. The original IDEF1 document, painstakingly constructed by Dr. Robert R. Brown, Tim Ramey, and Reuben Jones under the direction of Dr. Steven LeClair and myself, was published as a volume in an Air Force Technical Report.[1] Unfortunately, this report has been copied and recopied to the point where it is unreadable. It is also difficult to obtain. The purpose of this reconstruction is to provide in quality form the official reference manual for IDEF1.

With the exception of this foreword, the entirety of the original IDEF1 modeling manual has been reproduced in this document. The only changes incorporated were those spelling, grammatical, and typographical errors which escaped the watchful eyes of the original team, but which yielded to the power of today's word processing and text preparation systems.

Unlike many commercial methods, new IDEF methods are continuing to evolve. Initiatives in new IDEF method developments have been taken up by the Air Force Armstrong Laboratory, Logistics Research Division AL/HRGA at Wright-Patterson Air Force Base, Ohio through their Information Integration for Concurrent Engineering (IICE) program. In addition, an IDEF Users Group (IUG) has been formed as "An Association for Enterprise Systems Integration Methods." This association provides a forum for exchange of experiences relative to the application of IDEF methods.

---

[1] Integrated Computer-Aided Manufacturing (ICAM) Architecture Part II, Volume IV - Function Modeling Manual (IDEF0) AFWAL-TR-81-4023.

The success of the IDEF technology to date has been the results of efforts by many individuals. A complete anthology would have to include: Stu Coleman, Robert R. Brown, Tim Ramey, Peter Chen, Gerald Shumaker, Frank Borasz, Ken Melhope, Richard Preston, Paul Condit and Mike Painter–and would still be incomplete. It is hoped that no offense is taken for any errors or omissions, certainly none was intended. I intend for the availability of this book to spur even more widespread use of the IDEF methods.

The Evolving IDEF Family of Methods for Enterprise Integration

IDEF (Integrated Computer-Aided Manufacturing (ICAM) DEFinition) methods are used to perform modeling activities in support of enterprise integration. The original IDEFs were developed for the purpose of enhancing communication among people who needed to decide how their existing systems were to be integrated. IDEFØ (Function Modeling Method) was designed to allow a graceful expansion of the description of a system's functions through the process of function decomposition and categorization of the relations between functions (i.e., in terms of the Input, Output, Control, and Mechanism classification). IDEF1 (Information Modeling Method) was designed to allow the description of the information that an organization deems important to manage to accomplish its objectives. IDEF2 (Simulation Modeling Method) was originally intended as a user interface modeling method. However, since the ICAM Program needed a simulation modeling tool, the resulting IDEF2 was a method for representing the time varying behavior of resources in a manufacturing system providing a framework for specification of math-model-based simulations. As a result, the lack of a method which would support the structuring of descriptions of the user view of a system has been a major shortcoming of the IDEF system of languages. The IDEF3 (Process Description Capture Method) has been developed to fill this void. At this time there are two additional description capture IDEF methods under development. IDEF5 (Ontology Description Capture Method) will be a method for fact collection and knowledge acquisition. IDEF6 (Design Rationale Capture Method) will be a method for capture of information system design rationale.

The second class of IDEF methods that have been developed are focused on the design portion of the system development process. That is, they encapsulate the best known method for design with a particular technology (or class of technology.) Currently there are two IDEF design methods; IDEF1X (Data Modeling Method) and IDEF4 (Object-oriented Design Method). IDEF1X was developed to assist in the design of semantic data models. IDEF4 was developed to address the need for a design method to assist in the production of quality designs for object-oriented implementations. IDEF4, like IDEF1X, is intended to serve the needs of the systems designers and programmer analysts who are building and evolving large information systems. Unlike IDEF1X, which encapsulates a design method for relational database design, IDEF4 encapsulates the principles for design of object-oriented applications and databases. The target areas of application for the IDEF methods have been classified relative to an updated Zachman framework for information system architectures [Zachman 86, IDEF Users Group 90, and Mayer 91]. Figure 1 shows the additional IDEF methods that are planned for development over the next three years under the Air Force IICE program. The IICE program is a four-year research and development effort sponsored by AL/HRGA. In cooperation with a number of Department of Defense (DoD) and industry partners, the IICE program maintains a long-term commitment to improve the logistics supportability of Air Force weapon systems through the advancement of both CALS and Concurrent Engineering technology. The objective of the IICE program is to provide the foundations, methods, and tools to effectively implement and evolve towards an information-integrated Concurrent Engineering environment. The IICE effort is divided into eight distinct thrust areas:

1. Integrated Systems Theory Thrust
2. Ontology Thrust
3. Methods Engineering Thrust
4. Experimental Tools Thrust
5. Three-Schema Architecture Thrust
6. Application Thrust
7. Frameworks Thrust
8. Technology Transfer/Transition Thrust

These methods will provide a rich complement of method capabilities for enterprise integration efforts.

Suite of IDEF Methods	
IDEFØ	Function Modeling
IDEF1	Information Modeling
IDEF2	Simulation Modeling
IDEF1X	Data Modeling
IDEF3	Process Description Capture
IDEF4	Object-oriented Design
IDEF5	Ontology Description Capture
IDEF6	Design Rationale Capture
IDEF7	Information System Audit Method
IDEF8	User Interface Modeling
IDEF9	Scenario-driven Info Sys Design Spec
IDEF10	Implementation Architecture Modeling
IDEF11	Information Artifact Modeling
IDEF12	Organization Modeling
IDEF13	Three Schema Mapping Design
IDEF14	Network Design

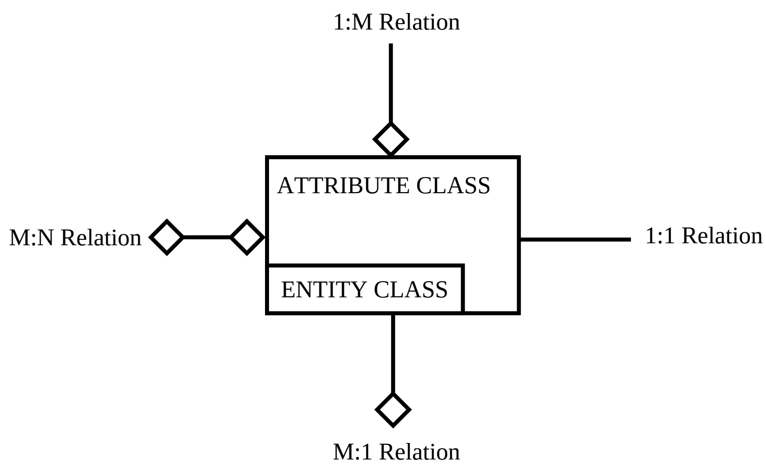
Figure 1. The IDEF Family of Methods

All technical content in this manual has been retained. The text was scanned from an original Integrated Computer-Aided Manufacturing (ICAM) Information Modeling Manual (IDEF1). Typographical errors were corrected, and figures were redrawn for clarity. This Foreword; Appendix B, KBSI Profile; and Appendix A, The IDEF Family of Methods Paper were added for the interested reader.

College Station, Texas — Richard J. Mayer

May 1992

Section 1.0 Introduction



1.0 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 technology 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:

1. IDEFØ is used to produce *a function* model which is a structured representation of the functions of a manufacturing system or environment and of the information and objects which interrelate those functions.

2. IDEF1 is used to produce *an information* model which represents the structure of information needed to support the functions of a manufacturing system or environment.

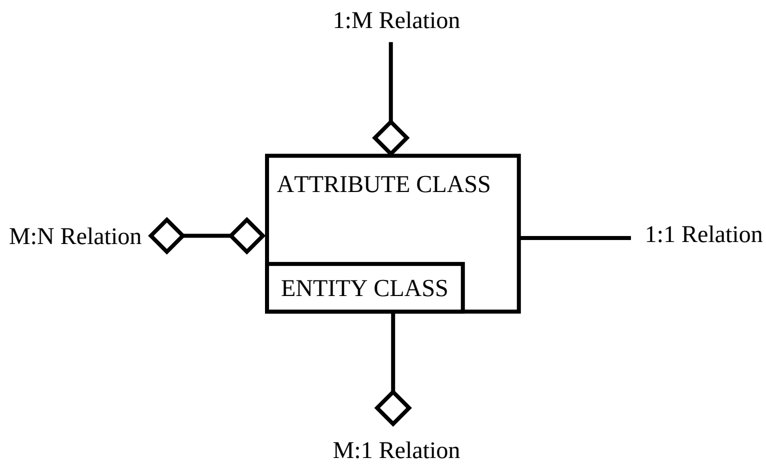
3. IDEF2 is used to produce *a dynamics* model which represents the time varying behavior of functions, information, and resources of a manufacturing system or environment.

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 also 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 IDEF1 to produce an information model.

Section 2.0 IDEF1 Concepts



2.0 IDEF1 Concepts

2.1 Introduction

This manual is designed to serve as an introduction to and reference guide for the IDEF1 information modeling methodology. The conceptual and tangible aspects of the methodology are described, displayed, and depicted in various examples throughout this manual. The IDEF1 modeling methodology incorporates basic principles into a specified process to produce an information model. This is accomplished through the efforts of selected individuals who serve specific capacities, or roles. Each of these roles has a specific set of functions which ensure the constant an continual evolution of the model. Each evolutionary phase is designed to produce certain products along the way. The development of these products is done in accordance with the prescribed procedures, through the efforts being conducted by the roles being served, and eventually equates to a detailed information model.

This reference manual will focus its attention on the role of the modeler and the conduct of the modeler's responsibilities. It will define what an information model is and how one is built, primarily from the modeler's perspective.

The need for the IDEF1 method became clear as the difficulty in designing integrated manufacturing systems became more and more apparent. Controlling and coordinating integration of manufacturing information often appears to be virtually impossible. Manufacturing operations tend to be so diverse that the real requirements to be placed on an integration effort are simply "buried" in all the complexity.

From an extensive and detailed examination of available manufacturing and engineering practices and the kinds of problems cited above, it was concluded that the most practical way to approach the problem of integration of manufacturing information was to develop an information requirements model prior to designing and building the corresponding information system. The IDEF1 method for developing such models was designed to reflect the integration of manufacturing information within the overall manufacturing enterprise. The IDEF1 approach is to:

1. Build an integrated information model.

2. Design database(s) from the information model.

3. Implement and install the data base(s) and associated functional and procedural components.

The IDEF1 method offers a set of rules and procedures for creating information models. It incorporates the necessary graphics, text, and forms to inject an organized discipline into the process. It provides for the measurement and control of the progressive development of the model through the routine of the modeling discipline.

Because the modeling discipline involves an evolutionary process, the IDEF1 method is organized into stages with measurable results and specific products. It develops toward a more exact definition with each iteration. It provides a modularity, in both practice and product, that cannot be found in other methods, and that protects against the incompleteness, imprecision, inconsistencies, and inaccuracies so often encountered.

There are two fundamental components of an information model:

1. Diagrams – The structural characteristics of the information model, displayed in accordance with a set of rules and procedures that construct a meaningful representation of information.

2. Dictionary– The meaning of each element of the model reflected through the compendium of text and indices that clearly define the information reflected in the model.

An IDEF1 model involves the entire manufacturing organization. There are several roles that have to be fulfilled to conduct a successful modeling effort. This manual is principally geared toward the benefit of the Modeler, or “recorder” of the model. The Modelers teammates are: the Project Manager, the Source(s); the Reviewer(s); and the Review Committee.

The Modeler is a modeling expert. The employment of the techniques, the maintenance of the momentum, the organization and publication of data, and, in general, the production of the model are the responsibility of the Modeler. The shape of the information structure of a manufacturing activity (its architecture) as represented in the model, is the primary responsibility of the other team members.

An IDEF1 information model is a reflection of the total manufacturing enterprise and provides a baseline definition of that organization’s informational needs. It ensures that the information can be shared and that the information system of the total enterprise is integrated.

IDEF1 is a new methodology which addresses the many problems cited above with a structured, broad-based, fresh approach. An information model is an attempt to determine "what is needed" in terms of information, for a manufacturing enterprise, and to represent this graphically as modular units of detail. An information model provides a precise, accurate, and concise description of the information needed by a manufacturing enterprise. It has a formal character, which provides for a precise understanding of the information it portrays and it is a tool which has practical value whether or not the manufacturing enterprise is heavily committed to the use of computers. It is of optimum value to the enterprise struggling with the problem of integrated system design.

2.2 Roles

The participants in an information modeling effort are grouped into five roles. Individuals who are involved in the modeling may each fulfill several roles, but each role is dealt with distinctly and should be clearly separated in the minds of the participants. The Project Manager guides the project. The Modeler, or author, is the recorder of the model. Sources provide the information for the model. Experts are individuals who validate the model. The Review Committee acts as an arbitrator in times of dispute and determines the final acceptability of the end product.

Each role may be filled by several individuals. In most cases, the workload of a role may be distributed across several participants, but in the cases of the project manager and the modeler, there must be a lead, or principal individual, who fulfills the role. This allows for the distribution of responsibility and the resolution of lines of authority throughout the modeling effort. Further, while it is the modeler's ultimate goal to have the model approved by the review committee, the modeler reports to the project manager, not the review committee.

In this way, the otherwise conflicting interests of the modeler, review committee, and project manager, are somewhat disentangled. The project manager is always placed in a position of control, while the various technical discussions and approvals are automatically delegated to the qualified agency under that control. Figure 2-1 illustrates the relationship of the various roles.

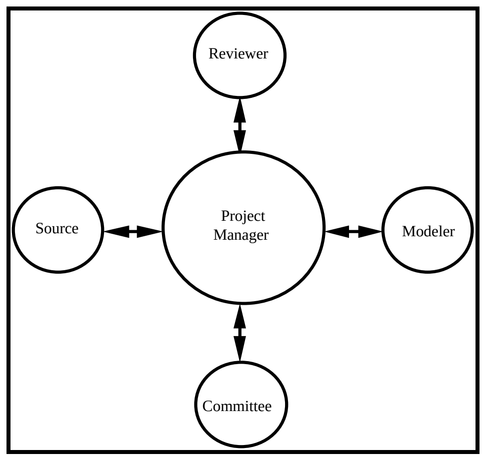


Figure 2-1. Functional Organization

2.3 Multi-Phase Development

The development process of the information modeling technique is composed of five phases. Each of these phases is described below:

1. Phase Zero – Phase Zero is the context-setting phase. During this phase, the scope of the model is defined and its objectives are stated.

2. Phase One – The objective of Phase One is to define the Entity Classes which are readily apparent at this stage of the model development.

3. Phase Two – The objective of Phase Two is to define the Relation Classes which exist between the entity classes of which the model is comprised at this level.

4. Phase Three – The objective of this phase is to identify the Key Classes for each Entity Class of which the model is comprised at this time and to define each Attribute Class which is used in a Key Class.

5. Phase Four – The objectives of this phase are to identify which Non-Key Attribute Classes should be associated with which entity classes in the model and to fully define each of these Non-Key Attribute Classes.

It is necessary to re-emphasize that the process of developing an information model is iterative in nature; that is, the model evolves from one stage to another. It is not until completion of Phase Four that the basic structural characteristics of the information resident within the scope of the model, as defined in Phase Zero, are complete.

The construction of an information model requires that a discipline and coordinated teamwork be employed daily. Teamwork means constant and effective communication among all participants in the modeling project. A regular process of critical reviews, with written comments from readers of the material, is the single most important activity in early detection of errors and the evolution of a sound model. Decisions can be made in the context of the discovered need, recorded as they unfold, and challenged while alternatives are still available. Oversights can be spotted before they cause a major disruption or critical misinterpretation. However, for the review process to work, it must not wait until the document is formally published or approved. The review process must be an everyday working procedure, conducted throughout all phases of model development.

2.4 Cyclical Activities

There are three kinds of interchange cycles evident in the IDEF1 technique:

1. Data collection cycle
2. Validation cycle
3. Acceptance review cycle

Each of these cycles can occur multiple times throughout the life of the modeling project.

The Data Collection Cycle is initiated in Phase Zero. Its purpose is to establish a baseline of documentation from which to extract the fundamental nature of the information represented in the model. It is not unusual for the modeler, during later phases, to return to the sources of this documentation to clarify certain aspects of it. This is why the Data Collection Cycle is viewed as a “recurrent,” rather than a “one time,” activity.

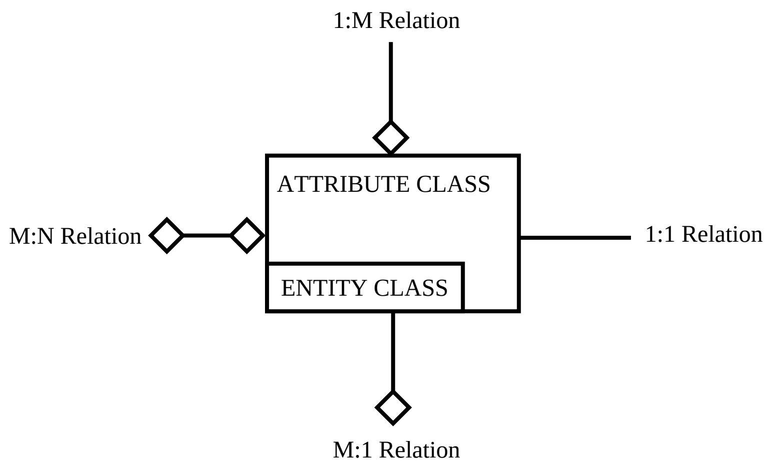
The validation cycle is sometimes called the IDEF Kit Cycle. The author (modeler) of the information model should have several readers review the evolving model at various stages before it is presented for acceptance. The written comments made by these reviewers should be incorporated into the model and the process should be repeated until the desired results are achieved. This is the substance of the IDEF Kit Cycle. The modeler is the draftsman; the reviewers are the architects.

The modeling effort is a continuous cycle of the synthesis of comments and findings. Certain analytical efforts must be performed at various stages along the way, but the final model description and structure is the result of a series of synthesized reviewer comments and observations.

The Acceptance Review Cycle is where a panel of experts evaluates the evolving (or fully evolved) information model to determine its acceptability for the purpose intended. This panel may require that specific areas of the model be re-validated by the experts. It may also find the model acceptable to a level (phase) of evolution reflected and simply recommend continued development. If the panel finds the model to be unacceptable, the Project Manager is required to actively resolve the points at issue to assure the development of a sound model.

Acceptance review typically occurs more than one time during a modeling project. Often, it will occur at the end of a phase, although infrequently for every phase. In all cases, a final acceptance review cycle should be conducted at the end of Phase Four.

Section 3.0 Understanding IDEF1 Diagrams



3.0 Understanding IDEF1 Diagrams

3.1 Phase Zero

As stated previously, the model is developed continually through predetermined phases each comprised of specific activities, objectives, and results. The first of these is Phase Zero, the Context Definition phase.

During this phase, the interpretation of the objectives, as communicated through the Project Manager, unfolds. The project definition is written, published, and the data collection techniques and contingencies are spelled out. The author establishes “conventions” that are intended for use when such latitude may be exercised within the bounds of the technique.

The results of this phase are a clearly established set of products which include the following:

1. Project Plan
   - A. Statement of Strategic Objectives
   - B. Strategic Plan
   - C. Functional Organization Plan
   - D. Resource Allocation Plan
2. Source Material
   - A. Data Collection Plan
   - B. Source Material Log
   - C. Source Data List
3. Author Conventions

This is also where the first data collection cycle occurs. Its results represent the very foundation of the information model. The results are documented in Phase Zero using the forms reflected in Figures 3-1 and 3-2. The Source Material Log and the Source Data List serve to organize the source material in a way that can easily be used to reduce future concerns as to the relevance or relationships in the information model.

3.2 Phase One

Phase One serves the purpose of identifying and defining entity classes in the model. First, "entity classes" must be identified. A perusal of various documents, forms, records, and other files of an organization can serve to get the process started. It is here that the modeler's understanding of and skill with the modeling technique are first tested. The basic requirement is to be cognizant of the way the terms "entity" and "Entity Class" are used in the technique.

An entity may be thought of as an object, either real (i.e., physical; something we could pick up and handle), or abstract (i.e., not within our physical grasp) that has properties. It is something about which specific characteristics are known. For example, a person is a physical entity. Each person has identifiable properties. These properties can be used to describe the person. One person (entity) that might be discussed is Jerry, the manager of production control. It is the properties known about the person–the name Jerry, the job position Manager of Production Control–that enable an exact identification of this entity. Conversely, the phoned–in complaint received this afternoon about the wrong product being delivered to one of our customers is a good example of the non-physical entity–the conceptual entity. The complaint exists, as an entity, whether it is represented on a complaint form or not. It has properties by which it is identifiable, such as the customer, its subject, the time received, etc.

The properties of an entity, its individual characteristics, are what differentiate it from another. For example, Jerry is only one individual (one entity) out of many who might be discussed. Jerry can be thought of as a member of a class of entities which might be labeled "person" or "employee."

USED AT:	AUTHOR: I.M. Modeler PROJECT: IDEF1 Workstation NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 30 OCT 90 REV:	X WORKING DRAFT RECOMMENDED PUBLICATION	READER: DATE:	CONTEXT

Source Material #	Source Material Name/Description	Received From	Comments
SM#1	Purchase Requisition/Form PI-R6 4-72	U.R. Buyer	
SM#2	Procedure #079-003 /Rev. 00 " Preparation of the Requisition"	U.R. Buyer	
SM#3	Procedure #079-001/ Rev. 00 " Preparation of the Purchase Order"	Policy and Procedures Manual	
SM#4	Procedure #101-506 " Purchasing Codes"	Policy and Procedures Manual	
SM#5	B.J. Commodity Code List	U.R. Buyer	
SM#6	B.J. Product Code List	U.R. Buyer	
SM			
SM			
SM			
SM			
SM			

NODE: P /X1	TITLE: Source Material Log	NUMBER: IMM5

Figure 3-1. Source Material Log

USED AT: | AUTHO I.M. Modeler | DATE 30 OCT 90 | X WORKIN | READE | CONTEX
PROJECT IDEF1 Workstation | DRAF | DATE
NOTES 1 2 3 4 5 6 7 8 9 | REV | RECOMMEND | PUBLICATI

Source Data #	Source Data Name	Source Material Cross-Reference	Comments
SD#1	Requisition Number	SM#1,#27,#36	Pre-Printed on Form
SD#2	Buyer Code	SM#1,#17	
SD#3	Vendor Number	SM#1,#21,#27	
SD#4	Order Code	SM#1	Only for Orders NOT Delivered t Plant 1 or 3 See 079-001
SD#5	Chg. No. (Change Number)	SM#1	
SD#6	Ship To (Location)	SM#1	
SD#7	Purchase Requisition	SM#1	
SD#8	Vendor Name and Address	SM#1	
SD#9	Non-Confirming/Confirming To	SM#1	Name of Person Contacted is entered in space provided
SD#10	Extra Copies	SM#1	For extra purchase order copies
SD#11	Requester (Name)	SM#1	

NODE P /X2 | TITLE Source Data List | NUMBE IMM6

Figure 3-2. Source Data List

Each entity is an individual member of some Entity Class. It has specific properties of its own and relates to an individual set of circumstances. It is an individual thing.

An entity class represents the kinds of things that are known in common about a collection of individual entities which have similar properties. For example, Jerry is one member of an Entity Class called employee, but so are Bob, Helen, JoAnn, etc. Each of these individual entities, which are members of the Entity Class employee, have their own unique characteristics, but the characteristics are similar in that they apply to each employee in the same way. They represent information commonly used to describe an employee. An Entity Class represents the information which is known about a collection of individual entities which have similar properties. Entity classes are one of the primary building blocks of the information model. An example of how a number of entities (members of an Entity Class) are represented as an Entity Class is shown in Figure 3-3.

An analysis of the source material from Phase Zero results in the identification of some number of candidate entity classes which can be used
to represent the entities observed. These are selected and transferred onto a document that is called the Entity Class Pool. Figure 3-4 shows the standard form for recording the Entity Class Pool.

Once they have been identified, entity classes must be defined. This is the first step in constructing the Entity Class Glossary. The basis of this glossary is the Entity Class pool. It provides for the full definition of each Entity Class in the pool. An example of the Entity Class definition page used in the Entity Class Glossary is reflected in Figure 3-5. The Entity Class glossary is a formalized way of capturing the meanings people attach to information represented in the model.

As indicated before, a combination of graphic and textual representations is used in information modeling to convey the information about each Entity Class. All of the information compiled about an Entity Class, though resident on multiple pages referred to as the Entity Class Set. The specific content of an Entity Class set, that is, the amount of information available about an Entity Class, will vary with the development phase. In Phase One, the modeler begins construction of the Entity Class Sets by development of the Entity Class definition page for each Entity Class.

3.3 Phase Two

Phase Two takes the modeling effort to the next level of detail in the search for the most specific definitions available/determinable. This phase begins to identify the relationships that give meaning to associations between entities. It results in the construction of Entity Class Diagrams, appropriately displaying the syntax that communicates the meaning of the relationships represented as "relation classes" in the model. The modeler is now required to understand what is meant by the term "Relation" or "Relationship" and "Relation Class."

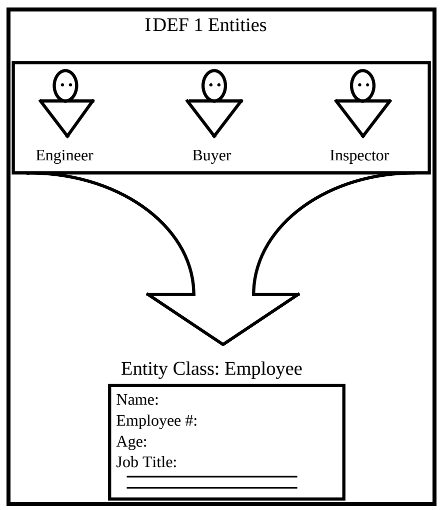


Figure 3-3. Synthesizing an Entity Class

USED AT:	AUTHO I.M. Modeler PROJECT IDEF1 Workstation NOTES 1 2 3 4 5 6 7 8 9	DATE 30 OCT 90 REV	X WORKIN DRAF RECOMMEND PUBLICATI	READE DATE	CONTEX

Entity Class Node No.	Entity Class Name	Source Data ID #	Entity Class Node No.	Entity Class Name	Source Data ID #
E1	Purchase Requisition Number	SD1	E18	Bill Of Material	SD38
E2	Buyer	SD2	E19	Route Sheet	SD40
E3	Vendor	SD3	E20	Destination	SD41
E4	Purchase Order	SD15	E21	Approver	SD43
E5	Ship To Location	SD6	E		SD
E6	Requester	SD11	E		SD
E7	Department	SD12	E		SD
E8	Pattern	SD21	E		SD
E9	Part	SD26	E		SD
E10	Purchase Req. Item	SD23	E		SD
E11	Commodity	SD30	E		SD
E12	Purchase Req. Line	SD31	E		SD
E13	Job	SD34	E		SD
E14	Account	SD36	E		SD
E15	Product	SD37	E		SD
E16	B.M. Page	SD38	E		SD
E17	B.M. Line	SD39	E		SD

NODE P1/X1	TITLE Entity Class Pool	NUMBE IMM11

Figure 3-4. Entity Class Pool

USED AT:	AUTHO I.M. Modeler PROJECT IDEF1 Workstation NOTES 1 2 3 4 5 6 7 8 9	DATE 30 OCT 90 REV	X WORKIN DRAF RECOMMEND PUBLICATI	READE DATE	CONTEX

Entity Class Name: Purchase Requisition

Entity Class Label: Purch.Req.

Entity Class Definition: A Purchase Requisition reflects information which is used by the Inventory Control Department to request the Purchasing Department to buy one or more parts which are needed for either a specific customer order or to replenish stock inventory.

Entity Class Synonyms:

NODE P1/E1 (G1)	TITLE Entity Class Definitions Purchase Requisition	NUMBE IMM12

Figure 3-5. Entity Class Definition

Relationship is an association between two entities. The entity known by the name Jerry is related to the entity known by the name Production Control in a way that may be defined as "manages." Some meaningful sense can be made out of the relationship between these two entities if we express them in sentence form: "Jerry manages Production Control."

It is not unusual for one entity to relate to many other entities. A good example of this is the relationship between the buyer, JoAnn, and the many purchase orders which she releases. JoAnn then has some relationship with Purchase Order 123, Purchase Order 457, Purchase Order 972, etc. Each of these purchase order entities is a member of the Entity Class that can be called Purchase Order, just as JoAnn is a member of the Entity Class called Buyer. The entity JoAnn (a member of the Entity Class Buyer) has some relationship with many entities (members of the Entity Class) called Purchase Order.

The information model expresses these relationships as a relation class. For example, the relation class describing the relationships that are possible between a Buyer and various individual purchase orders might be identified as "Issues." A meaningful sentence can now be constructed which describes this relationship in the following way: "Buyer issues purchase order." A relation class describes the manner in which members of an Entity Class relate to members of another (or other members of the same) Entity Class.

The first thing that must be done to successfully identify the relationships between entities and to build the diagrams which represent them is to create a Relation Matrix. This matrix is the preliminary indicator that some relationship may in fact exist between two entity classes. An example of a Relation Matrix is shown in Figure 3-6.

From the Relation Matrix, the first set of model diagrams can be built. These diagrams are called the Entity Class Diagrams.

An Entity Class diagram focuses attention on a single Entity Class, which is called the subject. The subject Entity Class is approximately in the center of the diagram. Surrounding the subject Entity Class are other entity classes which share some relation class with the subject Entity Class. Other than the subject Entity Class, the only other entity classes which appear on an Entity Class diagram are those which have a direct relation class linking them to the subject Entity Class. An example of an Entity Class diagram is reflected in Figure 3-7.

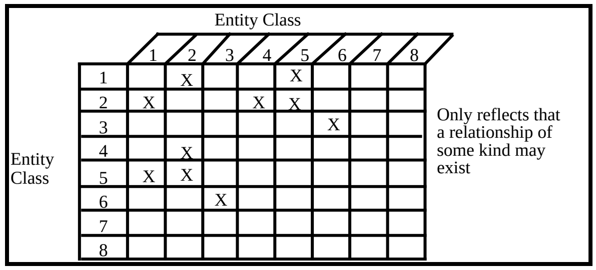


Figure 3-6. Relation Matrix

IDEF1 diagrams consist of some number of entity classes connected by lines and symbols to represent the relationships between entities being represented. The combination of lines and symbols represents the basic relation class syntax employed.

A half-diamond is used to represent a “zero or one” relationship, while a full diamond is used to represent a “zero, one, or many” relationship capability, between members of the related entity classes.

Since the objective of the information model is to construct a meaningful image of the information used in an enterprise, a relation class label is assigned to the relation class to convey specific meaning to the relationship represented. With the relation class label affixed, a meaningful sentence can be constructed, which conveys the basic meaning of the relationship between entities being represented. Remember, the information model graphically depicts entity classes. This representation is used to define entities and relationships between entities. The model depicts the structure of information in the enterprise in a two-dimensional form, necessitating visualization of many occurrences of entities “within” each Entity Class and many occurrences of relation classes “between” entity classes.

The relation class labels are of “verb-like” words, and, occasionally, “preposition–like” words, which describe the meaning of the relationship represented. Relation classes must be defined in detail as are entity classes. The relation class definition sheets become a part of the Entity Class Glossary. An example of a relation class definition sheet is contained in Figure 3-8.

3.4 Phase Three

The objective of Phase Three is to identify how members of one Entity Class are identified among members of the same Entity Class. This involves the identification of what are called "Key Classes." A Key Class is composed of some number of "Attribute Classes" by which each member of an Entity Class is uniquely identified. Now the modeler must know what is meant by the terms "Attribute," "Attribute Class," and "Key Class."

An attribute is what we call an individual property of an entity. An attribute has both a name and a value. A value alone, such as "123," has no meaning in and of itself, until we associate it with a name, but, as soon as we say "length in centimeters equals 123," the characters "123" now take on some meaning. In this example, "length in centimeters" is the name of the attribute and "123" is the value of the attribute. It is by some combination of individual attributes, i.e., properties that individual entities are described. Entities which are described as a group by the use of the same attribute names are represented as entity classes. Each member of that Entity Class is uniquely identified, one from the other, by some unique combination of values associated with the attribute names, which are themselves common to all members of the Entity Class. The attribute names which are common to all members of an Entity Class are referred to as Attribute Classes.

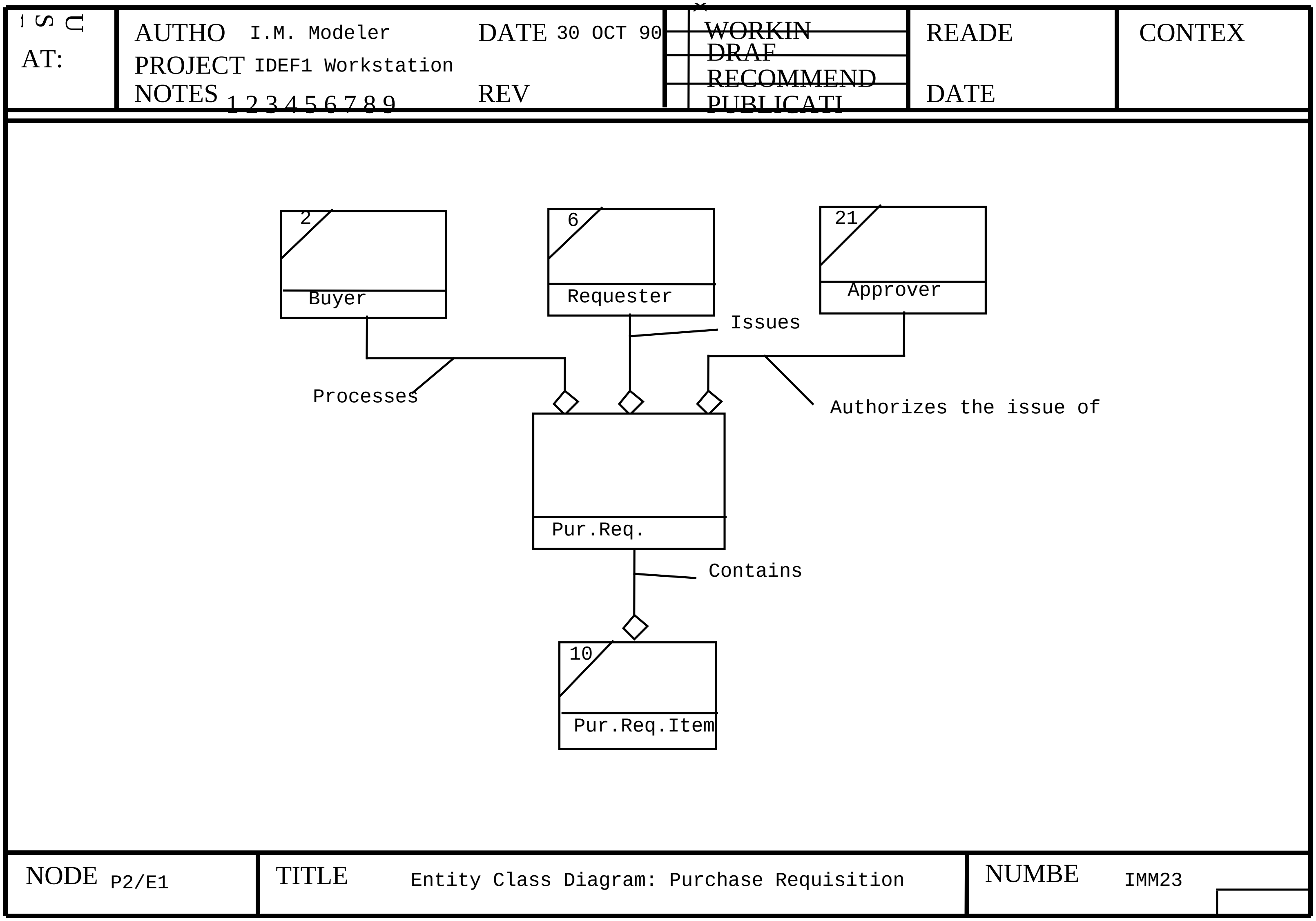


Figure 3-7. Entity Class Diagram

Each member of an Entity Class (an entity) is uniquely distinguishable from all other members of the same Entity Class by some unique combination of attributes (values) which apply only to it. For example, each member of the Entity Class called Purchase Order is identified by some value represented by an Attribute Class called Purchase Order Number. In this context, the Attribute Class Purchase Order number is used as the "Key Class" for the Entity Class Purchase Order. What determines this as the name of the attribute by which a single purchase order can be uniquely identified from all other purchase orders: Purchase Order Number. Purchase order number 12345 is different than purchase order number 12578. These attributes identify two separate and distinct purchase orders.

When an Attribute Class represents the attributes by which an entity is uniquely identifiable, it is referred to as a Key Attribute Class. It is not infrequent to find that several key attribute classes must be used in conjunction with one another to uniquely identify an individual member of an Entity Class. Together, these concatenated key attribute classes are referred to as the Key Class of the Entity Class. This means that a Key Class is a collection of one or more attribute classes used as a group to identify one member of an Entity Class from another.

A collection of attribute classes for the entire model constitutes the Attribute Class pool. Most of these emerge from the original source material. A sample Attribute Class pool is reflected in Figure 3-9.

Attribute classes, just like entity classes and relation classes, must be fully defined. Phase Three focuses on the development of definitions for attribute classes which are used in a Key Class. Once the key classes have been identified and Attribute Class definitions developed, the modeler can proceed to the development of Attribute Class diagrams.

USED AT:	AUTHO: I.M. Modeler PROJECT: IDEF1 Workstation NOTES 1 2 3 4 5 6 7 8 9	DATE 31 OCT 90 REV	WORKIN DRAF RECOMMEND PUBLICATI	READE DATE	CONTEX

Name					
A1					
A2					
A3					
A4					
A5					
A6					
A7					
A8					
A9					
A10					
A11					
A12					
A13					
A14					
A15					
A16					
A17					

NODE P3/X1	TITLE Attribute Class Pool	NUMBE IMM55

Figure 3-9. Attribute Class Pool

Like the Entity Class diagram, the Attribute Class diagram deals with a single subject Entity Class, which is located in the approximate center of the diagram page. The Attribute Class

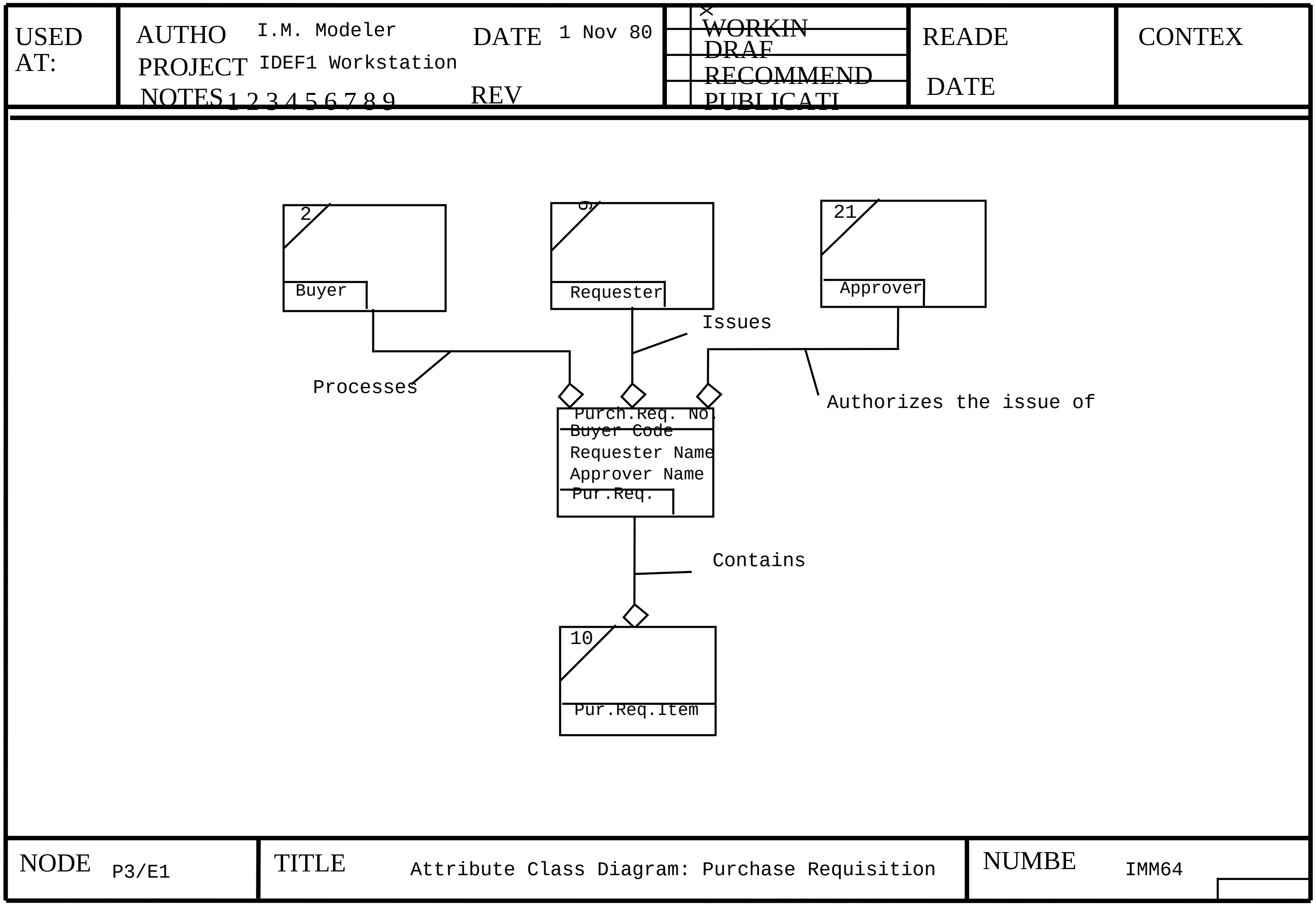


Figure 3-10. Attribute Class Diagram

diagram might be looked upon as an extension or expansion of the Entity Class diagram. This is because the essential difference between them is the information contained within the context of the Entity Class box. In the Attribute Class diagram, key classes, and other attribute classes used by the subject Entity Class, are reflected in the Entity Class box. Aside from that, there is little difference in the structure of an Attribute Class diagram and an Entity Class diagram. An example of an Attribute Class diagram is reflected in Figure 3-10.

3.5 Phase Four

Phase Four focuses attention on attribute classes which were not utilized as members of any Key Class in Phase Three. Each of these attribute classes must be defined and their use in the model determined. Phase Four is like the fine-tuning knob on a radio receiver. It incorporates all of the previous selection and tuning done in prior phases and homes in on the final information structure.

There is little difference between Phase Three and Phase Four where content is concerned. The primary differences are in the number of Attribute Class definitions and the full distribution of all non-key attribute classes in Phase Four. What is not readily apparent is the impact upon the structural characteristics of the model resulting from the application of certain tests for Entity Class validity during this phase. When applied, these tests cause appreciable structural modification to be made to the model.

The primary activity in Phase Four involves assignment of all non-key attribute classes to their appropriate entity classes. The objective is the distribution of attribute classes throughout the model in such a way that all members of every Entity Class can be individually and appropriately described. With disciplined application of the "validity" tests to each Entity Class, the basic structural nature of the information in the enterprise is evolved.

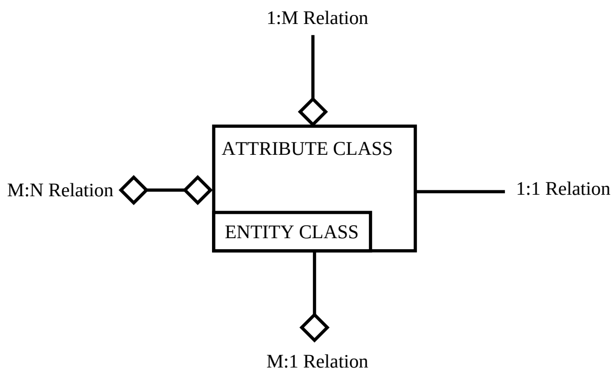
This process results in the generation of some amount of new material, but, by and large, the material is an extension of that produced in Phase Three taken to its final intended level of detail.

3.6 Conclusions

Upon completion of Phase Four, the modeler will have produced a structurally sound information model. If all of the methodology rules have been applied correctly up to this point, each Entity Class will represent a non-redundant collection of information and each Entity Class pair sharing a relation class will convey some non-redundant meaning in the model.

At this point, the information model is in a form which will facilitate basic translation into any data base management system currently on the market. This is not to say that the information model at the end of Phase Four is a data base design. It represents a stable information structure and a stable set of rules and definitions upon which a viable data base design can be constructed. From this, rationality and consistency are injected into the arena of integrated systems definition.

Section 4.0 Reading IDEF1 Diagrams



4.0 Reading IDEF1 Diagrams

4.1 Introduction

An IDEF1 model is made up of a series of diagrams and associated materials arranged by Entity Class number. A table of contents listing the entity classes by number is provided. It is helpful if an Entity Class index is also provided which lists the entity classes alphabetically. When published, a model is bound in page-pair format and in Entity Class number order. Page-pair format means that each diagram and the associated information appear on a pair of facing pages as shown in Figure 4-1.

Entity Class number order means that entity diagrams in an IDEF1 model are presented in the order in which they were developed. As each new entity is established in Phase One, it is given the next consecutive number, as shown below. If an Entity Class is dropped, the number assigned that Entity Class is dropped as well and is no longer used.

Node No.	Entity Class Name
1	Customer
2	Sales Order
3	Project
4	Cost Account
5	End Item
6	Part

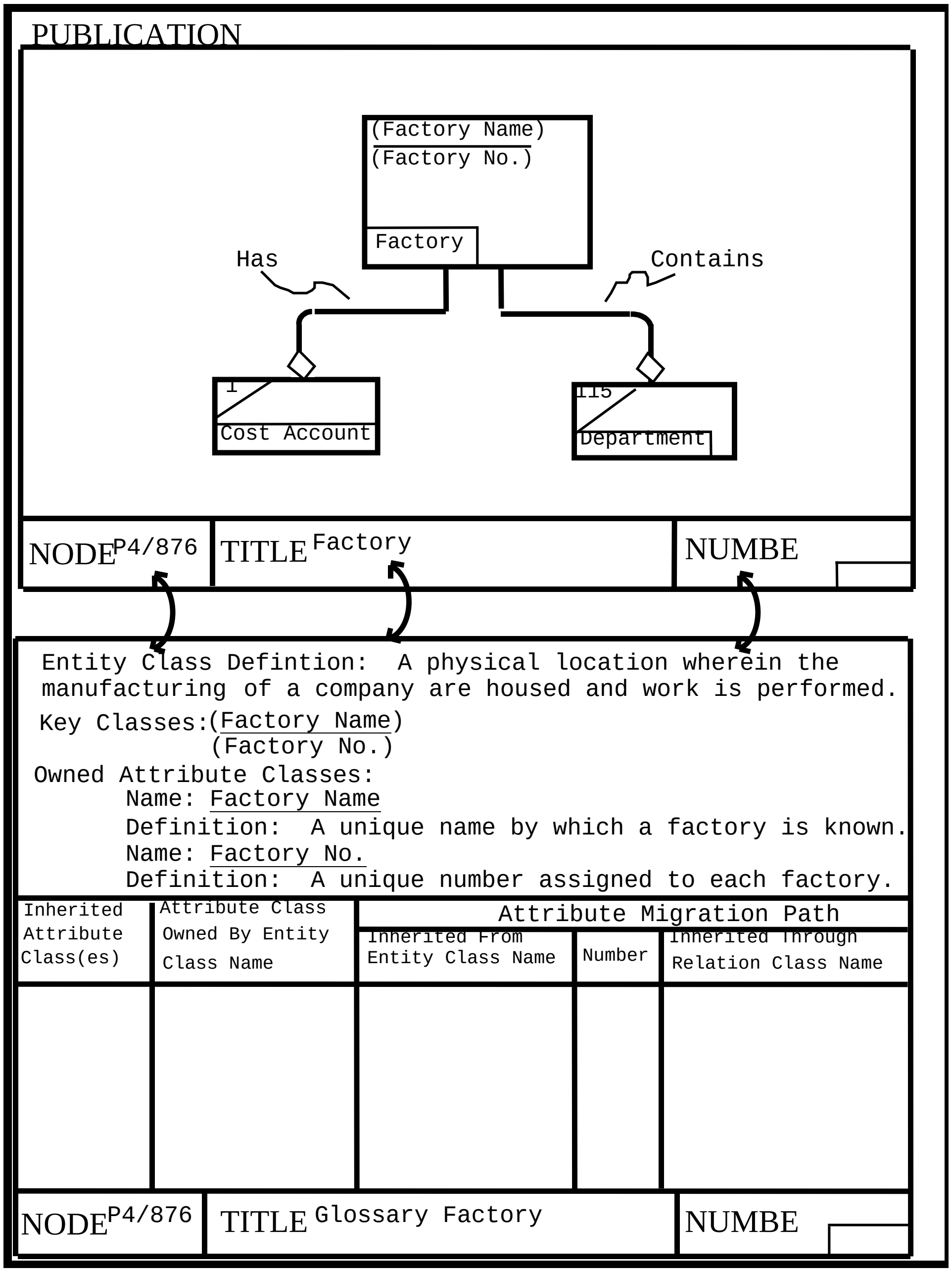


Figure 4-1. "Page-Pair" Format

4.2 Diagram Reading Steps

The precise information a diagram will contain is designated by the phase number on the diagram.

Phase One identifies and defines Entity Classes.

Phase Two identifies and defines Relation Classes.

Phase Three identifies and defines Attribute Classes.

Phase Four identifies and defines Non-Key Attribute Classes.

The following reading sequence is recommended:

1. Check the phase number of the diagram to determine the stage of development.

2. Check the title for the entity class on which the diagram is focused.

3. Mentally walk through the information given on the diagram and ascertain that it contains everything needed at that particular phase of development.

4. Read the supporting documentation accompanying the diagram.

The graphic notations on a diagram indicate the modeler's interpretation of the information. The message each conveys must expressly communicate to the reader the conditions which exist. Graphic notations used in IDEF are shown below. These symbols are used to specify relation classes between entity classes.

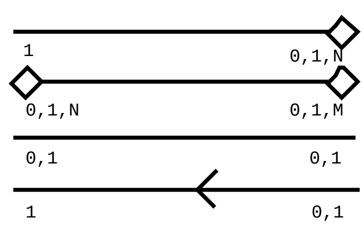


Figure 4-2 shows a simple entity diagram. The diagram is in Phase Two of development. It should contain entity classes and define the relation classes that exist between them. The title tells us that entity class number 32 is “Customer Representative.” The entity class name is confirmed since it appears in the unnumbered box on the diagram. Abbreviations of titles are allowed in the boxes when space limitations make use of the formal title difficult. Another entity class, “Customer” E26, is linked to the entity class “Customer Representative” by the relation class “Employs.”

The label on the relation class line is read from the “one” end to the “n” (or diamond) end. The relation class “Employs” used in Figure 4-2 says that:

1. Any “customer” entity may employ zero customer representative entities or any positive integer of “customer representative” entities (usually expressed as 0, 1, or n).

2. Each “customer representative” entity is employed by precisely one customer.

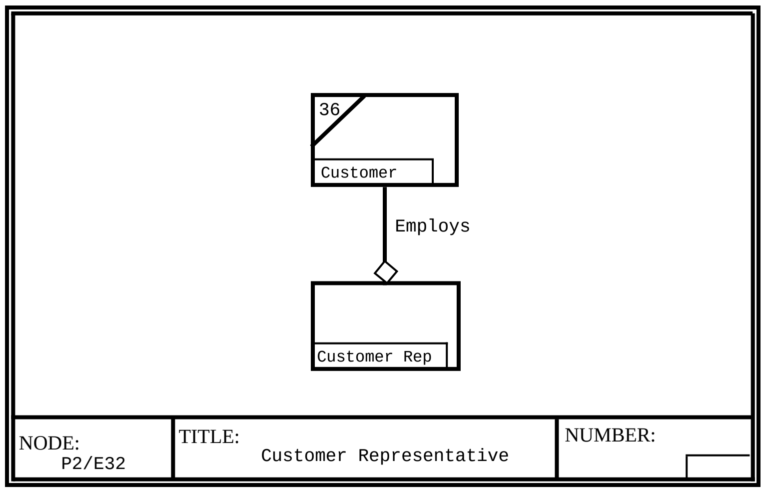


Figure 4-2. A Simple Entity Diagram

These statements indicate what may happen. For example, there may be no customers with zero representatives. The only reports what can exist. The statement that each representative is employed by one customer, and only one, is without exception. If there is an agent representing two customers, the diagram is in error.

The diagram implies two further facts:

1. No other entity class of interest relates to the “customer representative” entity class and

2. No other relation class of interest exists between the “customer” entity class and the “customer representative” entity class.

These assertions follow from the fact that no further entity classes are shown.

While reading the diagram, questions may arise. For example:

1. For what other “customers” did each “customer representative” work in the past?

2. For what members of the entity class “competitors” did each “customer representative” once work?

To answer these questions, additional relation classes and an additional entity class would be needed. Figure 4-3 shows how the diagram would look if both the issues raised were deemed important to include in the model.

The line with two diamonds between the entity class “competitor” and the entity class “customer rep” says that:

1. For every “competitor” entity, there may be zero or any positive integer “customer representatives” whom the competitor once employed.

2. For every “customer rep” entity, there may be zero or any positive integer “competitors” for whom the “customer rep” once worked.

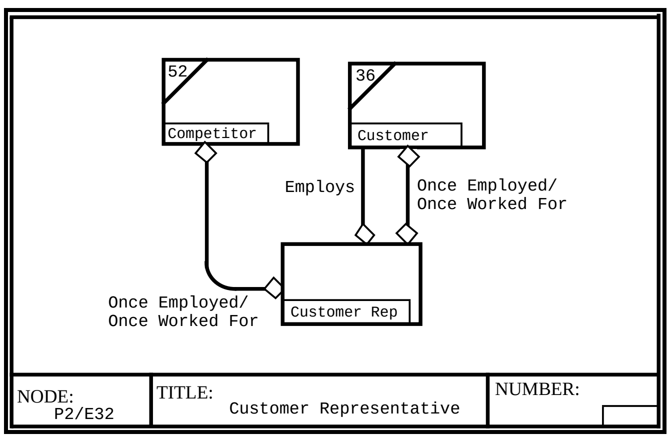


Figure 4-3. Customer Representative

This is referred to as an m:n relation class while the line with a single diamond is referred to as a 1:n relation class. The m:n relation class should have two labels, one for each direction. The 1:n relationship class needs only one label.

IDEF1 uses two other relation class symbols both of which deal with cases where any one entity can be related to at most one other entity.

The symbols shown in Figure 4-4 would apply between employees and machines in a shop where:

1. Any machine can be operated by only one employee.

2. One employee can operate any machine without help.

3. Any machine may have no employee assigned.

4. Any employee may have no currently assigned machine.

A simple line labeled "has assigned/is assigned" is used to show the relation class tying the "employee" entity class to the "machine" entity class. In this case, for any entity at either end, there may be zero or one entities at the other end.

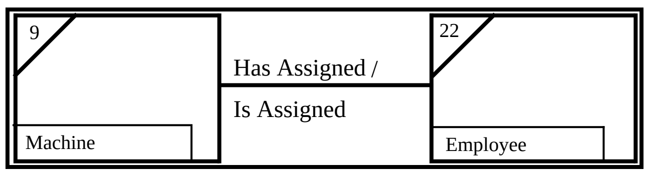


Figure 4-4. Other Relation Class Symbols

A refinement to Figure 4-4 can be made. A new entity class "assignment" can be introduced where the entity class "assignment" records information such as when employee 32 was made responsible for machine 19. Then every assignment entity refers to exactly one employee and one machine. This is shown by the Attribute Class symbol ↑, Figure 4-5.

The diagrams in Phases Three and Four should contain all Key Attribute Classes and some Non-Key Attribute Classes. The Key Attribute Classes are underlined in the focal entity box. Non-Key Attributes are not underlined.

Attribute diagrams of Phases Three and Four may contain only the relation classes shown in Figure 4-6.

Key attribute classes are shown for the focal entity class. The Key attributes may be:

1. Single – The one Attribute Class which identifies an entity such as the name of a country. The name of such a key Attribute Class is underlined.

2. Compound – Two or more attribute classes which together identify an entity such as the name of a city plus the name of a state. The names are underlined.

3. Alternate – Interchangeable attribute classes such as (employee number and social security number) either of which will uniquely identify an entity. Parts of the alternate key attribute classes may be single or compound. Each part is enclosed in parentheses and underlined.

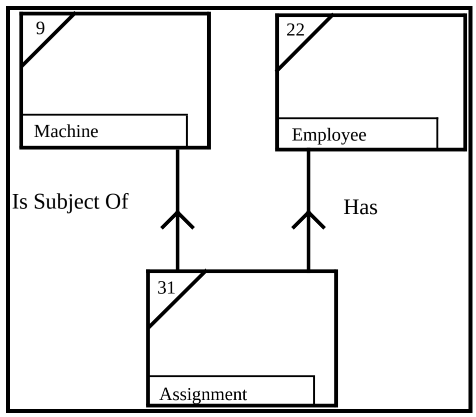


Figure 4-5. Attribute Class Symbol↑

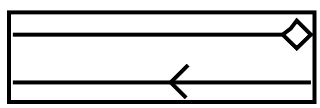


Figure 4-6. Relation Classes Allowed in Attribute Diagrams

One other requirement must be met by each attribute diagram. If the focal entity class is at the “n” end or the “O,1” end of a relation class, the key attribute classes of the entity class at the “1” end of the relation class must be shown in the focal entity class. The migrated attribute classes may appear as key (underlined) attribute classes or as non-key attribute classes. Figure 4-7A illustrates a migrated non-key Attribute Class (Vendor Name). Figure 4-7B illustrates a migrated key Attribute Class (P.O. No.).

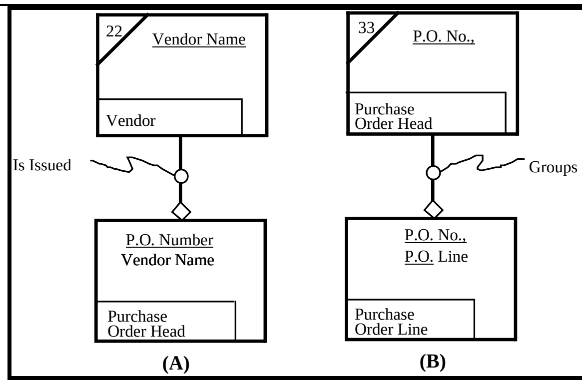


Figure 4-7. The Migrated Attribute Classes

4.3 Semantics of IDEF1

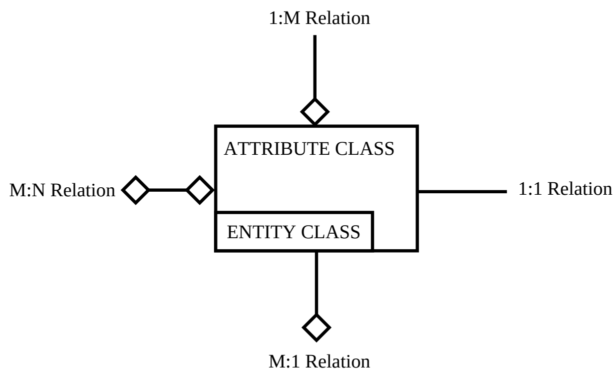
When interpreting Phase Two diagrams, check the following points:

1. Are all necessary entity classes shown?
2. Are any improper entity classes shown?
3. Are any relation classes missing?
4. Are any excess relation classes shown?
5. Are any relation classes of the wrong type shown?
6. Are the relation classes applied correctly in both directions?

When interpreting Phase Three and Four diagrams, check all points listed for Phase Two diagrams and also check:

1. Are the Key and Non-Key attribute classes sufficient?
2. Are the Key and Non-Key attribute classes necessary?
3. Have any alternate Key or Non-Key attribute classes been omitted?
4. Are only the permitted relation class symbols used?

Section 5.0 IDEF Forms and Procesures



5.0 IDEF Forms And Procedures

5.1 IDEF Teamwork Discipline

The development of any IDEF model (IDEFØ, IDEF1, and IDEF2) is a dynamic process which requires the participation of more than one person. Throughout a project the draft portions of a model are crafted by authors (modelers) 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 commentors. 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 commentor 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 evolution of a model is recorded by disseminating a model (with most recent changes) every 3 months in the form of a Kit which is 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.

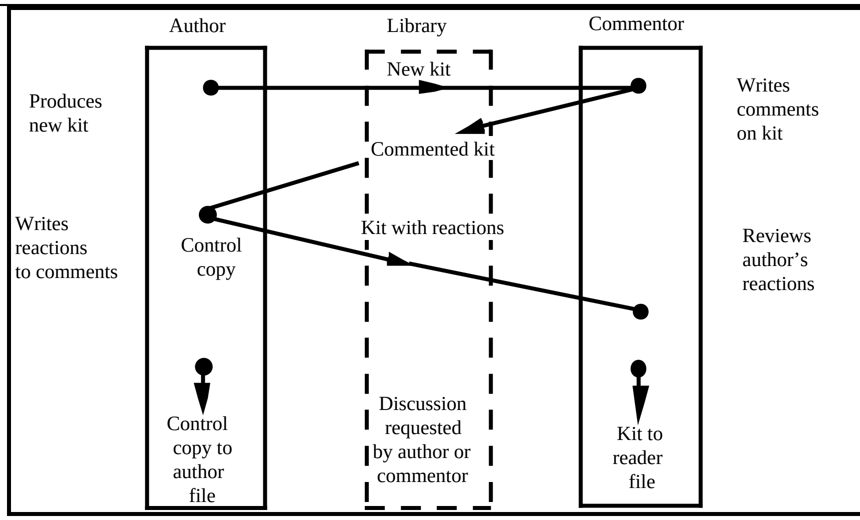


Figure 5-1. Kit Cycle

5.2 The IDEF Kit Cycle

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

1. 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 commentors and to the author. The original is filed for reference.

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

3. The author responds in writing directly on each commentor'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 commentor.

4. The commentor reads the author's responses and, if satisfied, files the kit. (Commented Kits are always retained by the Commentor.) If the

* Types of IDEF Kits are explained in Section 5.3.

commentor 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 commentors 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 commentors.

5.2.1 Personnel Roles

The roles and functions of people involved are:

1. Authors (Modelers) – People who prepare any IDEF model.

2. Commentors (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.*

3. 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.

4. Librarian – A person assigned the responsibility of maintaining a file of documents, making copies, distributing kits, and keeping records.

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

---

* Comments between commentator 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.

5.2.1.1

An author interviews experts and creates documents, but 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, and organizing the presentation of knowledge obtained from experts.

5.2.1.2

Commentors read material produced by authors and verify its technical accuracy. Commentors are responsible for finding errors and suggesting improvements. The role of a commentor is the key to producing high quality results. The commentor 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 Commentors

5.2.2.1 Commentor Guidelines

No set pattern of questions and rules can be adequate for commenting, since subject matter, style, and technique vary so widely, but 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:

1. 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.

2. Use the ⓝ notation to identify comments. To write an ⓝ-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)

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

4. 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/ commentor 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 commentor and author, etc. A kit returned to an author with no comments means that the commentor is in total agreement with the author. The commentor should realize that there is a shared responsibility with the author for the quality of the work.

5.2.2.2

When a commentor returns a kit, the author responds by putting a “~” or “X” by each ⓝ -note. “~” means the author agrees with the commentor 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 commentor to retain.

After reading the author’s responses, it is the commentor’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, commentors 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. Each Phase of an IDEF model requires specific material. Section 6.0 explains the contents of the kit in each phase of model building. 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:

1. Standard Kit – All kits to be distributed for comment. It is considered a "working paper" to assist the author in refining his total model.

2. 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. These 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 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 commentor and author.

5. Node Index/Contents:

   Node number, title and C-number of each page of the document (including the cover sheet) CONTENTS SHEET, Figure 5-3 (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. This space may also be used by the commentor for generalized comments on the total kit.

---

* 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.”

5.3.2 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.

Gather helpful materials and append these for the commentor's benefit. Never use this supplemental material to convey information which should properly be conveyed by the diagram itself. 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 Node Index/Contents Sheet and submit it to the Library.

5.4 Standard Diagram Form

The Diagram Form (Figure 5-3) has minimum structure and constraints. The sheet supports only the functions important to the discipline of structured analysis. They are:

1. Establishment of context
2. Cross-referencing between diagrams and support pages
3. 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:

1. Working Information (top)
2. Message Field (center)
3. 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 created during the modeling effort including preliminary notes.

5.4.1 Working Information

The Author/Date/Project

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 released without any change, 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.

IDEF COVER SHEET FORM

MODEL / DOCUMENT

TITLE: Entity Class Definition "AS IS"

LIFE CYCLE STEP: 1

IDEF METHOD — SYSTEM: IMCV

DISTRUBUTION TYPE

PRODUCT

AUTHOR: D. Stone — DATE

COMPANY: SofTech — TASK NO: 112

AF PROJECT NO

KIT

STANDARD KIT

SUMMARY KIT

SUPERSEDED OR REVISED DOCUMENT NUMBER

REVIEW

REVIEWER — DATE

AUTHOR — DATE

REVIEWER

COPY FOR	COMMENT	READ	NAME	COMPANY	PROJECT NUMBER

REVIEWER

COPY FOR	COMMENT	READ	NAME	COMPANY	PROJECT NUMBER

LOC. FILE

AUTHOR

KIT CYCLE

RECEIVED BY LIBRARY

KIT TO REVIEWER

COMMENTS DUE BACK TO LIBRARY

COMMENTS TO AUTHOR

AUTHOR RESPONSES DUE BACK TO LIBRARY

AUTHOR RESPONSE TO COMMENTOR

KIT CYCLE COMPLETE

COPYING

copies of pages = total

NODE INDEX /

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

COMMENTS / SPECIAL

Return Kit through ICAM Program Library

NOMENCLATURE

DOCUMENT/MODEL TITLE

DOCUMENT C-NUMBER: DSC1094 — PAGE 1

Figure 5-2   IDEF Cover Sheet