LOAN DOCUMENT

PHOTOGRAPH THIS SHEET

AD-A265 340

DTIC ACCESSION NUMBER

LEVEL

1

INVENTORY

CALS Phase 2 Architecture
Vol. 19

DOCUMENT IDENTIFICATION

3 Jul 89

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

DISTRIBUTION STATEMENT

ACCESSION FOR		
NTIS	GRA&I	☑
DTIC	TRAC	☐
UNANNOUNCED		☐
JUSTIFICATION		
BY Per Ltr		
DISTRIBUTION/		
AVAILABILITY CODES		
DISTRIBUTION	AVAILABILITY AND/OR SPECIAL	
A-1		

DISTRIBUTION STAMP

DTIC
ELECTE
MAY 28 1993
S C D

DATE ACCESSIONED

DATE RETURNED

93 5 27 0 E

DATE RECEIVED IN DTIC

93-12167

REGISTERED OR CERTIFIED NUMBER

PHOTOGRAPH THIS SHEET AND RETURN TO DTIC-FDAC

DTIC FORM JUN 90 70A

DOCUMENT PROCESSING SHEET

PREVIOUS EDITIONS MAY BE USED UNTIL STOCK IS EXHAUSTED.

LOAN DOCUMENT

VOLUME 19
CALS Phase 2
Architecture

Preliminary

CALS

Phase II Architecture

D-779-89-01.2
July 3, 1989

OSD CALS Office
Office of the Assistant Secretary of Defense
Washington, D.C.

D. Appleton Company, Inc.
222 Los Colinas Blvd, Suite 1141
Irving, TX 75039

ACKNOWLEDGEMENTS

The project team is grateful for the valuable input received from a number of participants in CALS industry committees and U. S. Air Force development programs representing Boeing, General Dynamics, Martin Marietta, McDonnell Douglas, Northrop, and Rockwell. Special acknowledgement must go to John Zachman of IBM for sharing his insightful work in information system architectures.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ACRONYMS

ADP	Automated Data Processing
ANSI	American National Standards Institute
CAD	Computer-Aided Design
CALS	Computer Aided Acquisition and Logistics Support *(DoD Program)*
CIEM	Computer Integrated Engineering and Manufacturing
CIM	Computer Integrated Manufacturing
CITIS	Contractor Integrated Technical Information System
DLA	Defense Logistics Agency
DoD	Department of Defense
EIS	Engineering Information System *(Air Force Program)*
GFE	Government Furnished Equipment
GFI	Government Furnished Information
ICAM	Integrated Computer Aided Manufacturing *(Air Force Program)*
ICIM	Interorganizational Computer Integrated Manufacturing
IDEF	ICAM Definition Languages
IDEF0	ICAM Definition Language: Activity Modeling
IDEF1X	ICAM Definition Language: Data Modeling
IDS	Integrated Design Support System *(Air Force Program)*
IGES	Initial Graphics Exchange Specification
ILS	Integrated Logistics Support
IRDS	Information Resource Dictionary System
IRM	Information Resource Management
IWSDB	Integrated Weapon System Database
LSA	Logistics Support Analysis
MIL STD	Military Standard
NASA	National Aeronautics and Space Administration
OSD	Office of the Secretary of Defense
PDD	Product Definition Data
PDES	Product Data Exchange Specification
RAMP	Rapid Acquisition of Manufactured Parts
SQL	Structured Query Language
TMIS	Technical and Management Information System *(NASA Program)*

Section 1.
Executive Overview

A January 1988 survey of forty top defense contractors by the CALS Industry Steering Group resulted in the following observations about the current environment for managing weapon system technical data:

- Significant investments have been made in computer-aided technology; however,
  - less than 20% of the companies are more than 50% automated,
  - complete automation is projected to be ten years away,
  - most companies use multiple vendor systems.

- Virtually all contractors have some capability to deliver digital data to the government, but only a small percentage is actually delivered digitally.

- Most companies have or plan to have at least partial digital interface with suppliers.

- Most companies are considering the development of a common shared database using relational technology.

- Although a high percentage of the contractors support current standards, advances are needed in most of the major technical areas.

As this survey points out, the Aerospace/Defense industry is headed in a direction of full automation based on common shared databases, and industry will generally be capable of providing on-line access to weapon system specific data as a service to DoD during the 1990s. The challenge for DoD is to provide an incentive to support and, hopefully, accelerate these trends. It is also important, however, that DoD clearly define its expectations in terms of the emerging shared data environment. This is the overriding purpose of CALS Phase II.

Recent policy guidance issued by the Deputy Secretary of Defense already requires that weapon systems now in full-scale development or production be reviewed for opportunities to improve quality or reduce costs by providing digital delivery or direct access to contractor technical information. This policy guidance not only makes current CALS Phase I technical data exchange standards an important consideration for current and future weapon system acquisition programs, it sets the stage for CALS Phase II: controlled on-line access by appropriate elements of the DoD to integrated digital technical information stored in shared databases maintained throughout the weapon system life cycle by various members of the weapon system contractor team.

The concept of CALS Phase II is based on the premise that a carefully defined strategy to automate, integrate, and manage product definition and support technical data will significantly reduce weapon systems life cycle costs, while simultaneously improving the quality of the weapon system and its support processes and, therefore, defense readiness. Such an automation strategy will also improve the competitiveness of the U.S. defense industry and ultimately the U.S. manufacturing industrial base. The result will be a win-win environment for both government and industry. For this result to occur, however, defense prime contractors, subcontractors, vendors, and computer technology suppliers must cooperate closely with the government and with each other to establish a common vision and framework for industrial networking and database sharing. The CALS Phase II initiative is intended to serve as a catalyst to establish that common vision.

Critical Success Factors

Although most, if not all, of the concepts and technology required for CALS Phase II have been demonstrated by various individual contractors, a number of managerial and technical issues must be addressed in order to establish a broadly accepted common approach. The critical success factors for industry establishment of a CALS Phase II environment include:

- Clear delineation of responsibilities and authorities for technical data creation, maintenance, and access;
- Functional integration of design, manufacturing, and support processes;
- Logical data integration strategies necessary to ensure configuration control, security, currency, accuracy, and comprehensive representation of weapon system technical information;
- Integration of Government Furnished Information (GFI) with contractor-generated technical information;
- Open system interconnection and data portability between contractor and government technical information systems;
- Evolutionary integration strategies that accommodate near-term usage of legacy systems.

Summary of Key Architectural Constructs and Their Relationships

This "Preliminary CALS Phase II Architecture" report is the first step toward defining the basic DoD expectations regarding the types and methods of information services that will be provided to the DoD by contractors from a shared data environment. The set of capabilities and resulting technical information services provided to the DoD by a weapon system contracting team is referred to as "Contractor Integrated Technical Information Service (CITIS)." Contractor Integrated Technical Information Service is provided for a specific weapon system through the use of "integrated" ADP systems from many different suppliers cooperating in the various stages of the weapon system life cycle. Therefore, the CALS Phase II Architecture must address inter-enterprise integration, linking primes, co-primes, and subcontractors with each other and with various government organizations, as well as intra-enterprise integration, linking program management, engineering, manufacturing, and support functions within an enterprise.

The CALS Phase II Architecture has been defined from three different perspectives. The description of the life cycle activities and end-user information services constitutes the External Architecture. The description of the computer hardware and software systems that inter-connect to provide the delivery mechanism for information services constitutes the Internal Architecture. The description of the logical rules which guide the integration of functions, data, and automation technology constitute the Control Architecture. The focus of this report is to outline a strategy for establishment of a generic Control Architecture that will serve as a reference model for providing Contractor Integrated Technical Information Service for specific weapon system programs. Recommendations for specific development action items are described in Section 6.

Consistent application of the Control Architecture integration rules will result in an Integrated Weapon System Database (IWSDB) which facilitates shared technical data throughout the weapon system life cycle. This is not one huge, centralized database, but a distributed database, which is managed consistent with a special set of rules called Data Standards. The complete set of Data Standards defining an IWSDB constitute what is called the CALS Data Dictionary, which is to be created by methodically extracting data element definitions from existing and planned mil specs and standards, and integrating them. The CALS Data Dictionary is to be captured and maintained in a CALS Data Dictionary System, using established CALS Data Dictionary Management Procedures.

Other dimensions of integration in the CALS Phase II environment, besides the data management dimension, are to be defined by Functional Standards, which define how data are to be generated and used throughout the weapon system life cycle, and Technical Standards, which define how data are to be automated and systems inter-connected. Functional and Technical Standards, as well as Data Standards, make up the Control Architecture for CALS Phase II. Most of today's existing standards do not clearly separate the functional, technical, and data dimensions. MIL-STD-1388 for logistic support analysis records, for example, defines functional requirements for logistic support record keeping, defines required data elements, and implies a specific approach to automation. Many of MIL-STD-1388 data elements, however, are redundant with other standards, and the implied approach to automation has not kept pace with commonplace advancements in computing technology. The long-term objective of the CALS Phase II Control Architecture is to establish independent Data Standards that support many different Functional Standards and that can be automated according to various Technical Standards that evolve over time.

Background

In addition to the review of numerous industry surveys and weapon system program plans, informal information was gathered regarding existing programs from a number of leading aerospace companies in order to help validate the Preliminary CALS Phase II Architecture. The contractors and programs reviewed include:

- Douglas Aircraft C17 Program
- General Dynamics F16 Program
- Lockheed ATF Program
- McDonnell Aircraft F18 and ATA Programs
- Northrop ATF and B2 Programs
- Rockwell B1B Program

Although all of the existing contractor systems reviewed provide technical information services within the proposed scope of CALS Phase II CITIS, none of them covers the total scope; that is, none provides complete weapon system life cycle technical information services. Existing systems generally fit into three categories: integrated design systems, integrated manufacturing systems, or integrated logistics support systems.

Integrated design systems to date have focused primarily on Preliminary Design and Full-Scale Development activities and are oriented toward engineering requirements. Fully automated support for concurrent engineering is not generally available, but several contractors have internal development programs to create a shared product definition database for all disciplines. At least one example of significant use of on-line access to a shared product definition database was found between a Prime and Co-Designer, each with a different internal CAD system. A special agreement on data structures and implementation technology was required along with the development of custom software in order to link the contractors together.

A number of integrated manufacturing systems can be found that support direct access to internal design systems. The primary focus of these systems is on the Production life cycle phase. The ability to automatically plan and manufacture a part that was designed by another company is not generally supported. However, significant efforts to demonstrate this capability are being supported by private industry, as well as DoD.

Integrated logistics support systems are in common use among major defense contractors. Although these systems are directly accessible by government organizations, custom interfaces or special equipment is required and access is generally limited to "read only." Most of the existing systems are implemented using traditional hierarchical database management systems. However, one contractor has made significant progress toward an integrated relational database development. The primary focus is on the Product Support Phase of the life cycle and redundant information is usually maintained by the contractor's internal engineering and manufacturing organizations, as well as by government organizations and subcontractors.

CITIS - Contractor Integrated Technical Information Service

The challenge being undertaken in CALS Phase II is to provide a conceptual framework and a set of standards and guidelines that will not only encourage integration among these separate "islands of automation" within an enterprise, but among enterprises participating on a weapon system program.

Contractor Integrated Technical Information Service (CITIS) for a specific weapon system must be provided by inter-connected computing networks and application software that are utilized by members of the weapon system development team to enter, update, manage, and retrieve data from their own internal technical databases. In addition to requiring integration of the prime contractor's internal data and processes supporting a specific weapon system, the CALS Phase II Architecture for CITIS must further specify integration of prime contractor data and processes with subcontractor and vendor data and processes and with government-furnished information (GFI). The logical integration of prime, subcontractor, vendor, and government information for a specific weapon system creates an Integrated Weapon System Database (IWSDB).

An IWSDB is intended to provide availability of accurate technical information to DoD components and industry throughout the lifetime of the weapon system. A principal objective for establishment of an IWSDB is to *"create data once - use it many times."* Establishment of an IWSDB requires support by both government and contractor technical information systems, where contractors provide access to and maintenance of the IWSDB as a service to the government. Physically, the IWSDB will be distributed over numerous locations and computing systems, and it may ultimately be transferred from the contractor team to the government for maintenance and control.

Summary

The purpose of this document is to establish a framework for developing a common CALS Phase II Architecture and to present a preliminary architecture that identifies the overall scope, objectives, and critical issues for development and implementation of Contractor Integrated Technical Information Service (CITIS). The objective of the Preliminary CALS Phase II Architecture is to establish a composite "best practice" baseline for use in further development of a common government and industry vision of an Integrated Weapon System Database and the supporting technical information services. The following sections of this report discuss integration concepts and trends, the overall requirements for future Contractor Integrated Technical Information Service, and development strategies and issues.

Section 2.
Background and Introduction

2.1 Purpose

This report has been produced under the direction of the DoD CALS Office under one of a number of projects intended to articulate strategies for the use of current and emerging computer-based technologies to improve both defense readiness and the productivity and competitiveness of the U.S. defense industry. The purpose of this report is to provide a "strawman" architecture for CALS Phase II Contractor Integrated Technical Information Service (CITIS) that significantly improves the capture, management, and use of technical product and support data throughout the life cycle of a weapon system. The target audience for this document includes managers of weapon system programs, technical systems managers employed by defense prime contractors, subcontractors, and vendors, and product managers employed by computer technology suppliers. The report is intended to help establish an industry and government consensus of the requirements for Contractor Integrated Technical Information Service (CITIS) and its associated Integrated Weapon System Database (IWSDB), and to stimulate discussion of critical management and technical issues related to these concepts.

2.2 Overview of the CALS Program

CALS is a DoD and industry initiative to enable and accelerate the integration and use of digital technical information for weapon system acquisition, design, manufacture, and support. The CALS program is intended to facilitate the transition of current paper-intensive processes to a highly automated and integrated mode of operation, thereby substantially improving productivity and quality of the weapon system acquisition and logistic support processes. The Deputy Secretary of Defense initiated the DoD CALS program in September 1985 with the goal that by 1990 new weapon system acquisitions would require technical data in digital form or obtain government access to contractor integrated databases in lieu of paper deliverables. The benefits expected from CALS implementation, as stated in the 1988 CALS Report to the Committee on Appropriations of the United States House of Representatives, include the following:

- Reduced acquisition and support costs for weapon system programs through elimination of duplicative, manual, error-prone processes.

- Improved quality and timeliness of technical information for support planning, reprocurement, training, and maintenance, as well as improved reliability and maintainability of weapon system designs through direct coupling to computer-aided design and engineering processes and databases.

- Improved responsiveness of the industrial base by development of integrated design and manufacturing capabilities and by industry networking among prime contractors and subcontractors to build and support weapon systems based on digital product descriptions.

Both DoD and industry are currently investing substantially in the automation of a variety of functional areas to improve productivity and quality. However, these investments, because of the historical lack of a CALS-like structure, have resulted in a multitude of independent and inconsistent ADP systems, often called "islands of automation," that cannot economically exchange or share information. Accordingly, an exorbitant amount of time and money is currently wasted in converting information in one system to formats that can be used in another system, and in attempting - and often failing - to resolve the numerous inconsistencies in information managed by them. When one considers that a single major weapon system program, such as the F-16, SSN-21, B-1B, or the B-2, may involve 5,000 or more contractors and vendors throughout its life cycle, and that each of these entities employs multiple information systems to support its activities on the program, this means that the data needed to accomplish the design, delivery, and support of the weapon system exists in thousands of ADP systems, not including those managed by the government logistics infrastructure or the acquisition establishment. The problem of data management is severely compounded by the fact that these ADP systems are based on uniquely designed software programs operating on computers manufactured by a wide variety of computer technology suppliers, and that these software programs and computers are designed such they cannot communicate easily with one another.

Through the CALS Initiative, the existing islands of technical data automation within DoD and industry are being integrated to facilitate data exchange and access, as well as to reduce duplication of data preparation efforts. Industry has endorsed the action DoD has taken in CALS, and the transition to an automated integrated environment has begun.

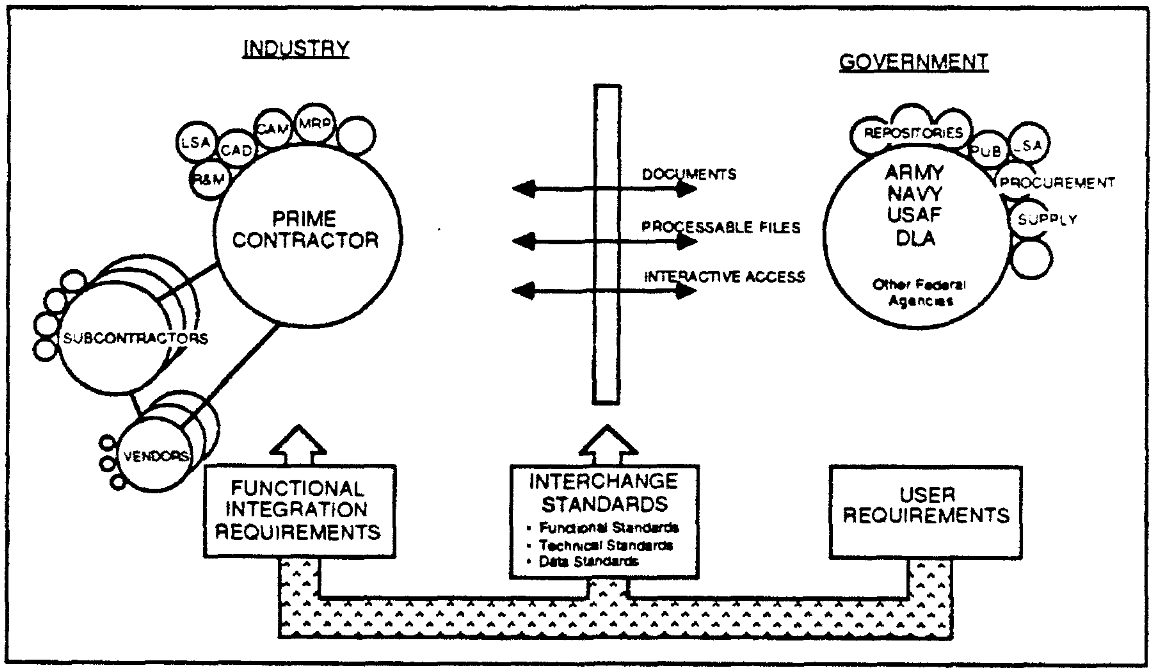


Figure 2-1 CALS Environment

CALS Strategy

The CALS environment is illustrated in Figure 2-1. CALS encompasses the generation, access, management, use, and distribution of technical data in digital form for the acquisition, design, manufacture, and support processes. These data and processes are supported by numerous information systems that reside within the prime contractor, subcontractor, and vendor environments. Although the interaction between the contractor team members alone is complex, technical data must also be supplied to government repositories that support numerous information systems within various government organizations. Within CALS, the common thread is technical product and support data, which includes engineering drawings, product definition and logistic support analysis data, technical manuals, training materials, technical plans, reports, and operational feedback data associated with weapon systems, support equipment and supplies. Large volumes of technical data must be shared between the members of a contractor team in order to successfully design and manufacture a complex weapon system. As the owner and operator of the weapon system, the government also has user requirements for technical data. The technical data requirements of both the internal contractor team and government organizations requires functional integration of life cycle activities for a weapon system and sharing of technical information through common interchange standards.

To achieve CALS benefits, a phased CALS strategy has been established by a team consisting of the Office of the Secretary of Defense (OSD), the Services, Defense Logistics Agency (DLA) and Industry. Phase I is focused on establishing standards to facilitate the replacement of paper document transfers with digital file exchanges as a beginning of the integration process, with implementation between now and the early 1990s. In parallel, technology is being developed for Phase II that involves substantial integration and redesign of current processes to take advantage of a shared database environment in the early 1990s and beyond. The main elements in both phases are as follows:

- Standards. Accelerate the development and testing of standards for digital technical data interchange and integrated database access.

- Technology Development and Demonstration. Sponsor the development and demonstration of the necessary technology for integration of technical data and processes in high-risk areas.

- Weapon System Contracts and Incentives. Implement CALS standards and integration requirements in weapon system contracts and encourage industry modernization and integration.

- DoD Systems. Implement CALS standards and integration requirements in DoD planning and infrastructure modernization programs. Infrastructure is the underlying framework of the organizations, systems, and processes within which DoD operates.

Technical Information Systems

As CALS capabilities evolve, technical data required by the government for a single weapon system will be logically integrated (not necessarily physically integrated) into tightly coupled, controlled, and secure weapon system technical databases, allowing access and transfer of data to those parties with proper authorization and need to know. ***The capabilities for a contractor team, including government organizations, to enter, update, manage, retrieve, and distribute data from technical databases for a specific weapon system is called Contractor Integrated Technical Information Service (CITIS).*** This service is provided by a collection of automated data processing systems and applications utilized by the contractor team. The required functional integration of those contractor processes necessary to ensure the security, currency, and accuracy of the technical information resident in the weapon system technical databases will be articulated and contractually specified as requirements for a weapon system program's CITIS. In addition to the required integration of the contractor's internal data and processes themselves, further integration of internal contractor data and processes with the government-furnished information for each weapon system is essential.

The collection of automated data processing systems and applications that are utilized by the government to enter, update, manage, retrieve, and distribute data from technical databases for a specific weapon system exist on multiple distributed automated data processing systems. These government information systems cross functional boundaries and may require access to a combination of data from more than one source to support information requests from a single weapon system's user community. This degree of interchange and integration will require tight control and coordination of the separate physical databases to allow transparent support to the user. The needed control and coordination of shared data within and among the contractor technical information systems and government systems supporting a weapon system will be provided by a logical data structure called the CALS Integrated Weapon System Database (IWSDB).

Integrated Weapon System Database (IWSDB)

The logical collection of shared data for a specific weapon system that is used throughout the weapon system life cycle is called an IWSDB. The physical location of the data may be distributed among contractor or government automated data processing systems. The required IWSDB structure is evolving and will be the basis for the CALS Phase II integrated, shared data environment. The CALS IWSDB requirements will provide a logical (not physical) collection of shared data to support both contractor team and government users throughout the complete life cycle of a specific weapon system. The IWSDB will be governed by groups of Data Standards, which together make up the CALS Data Dictionary. The CALS Data Dictionary will ultimately be maintained in a CALS Data Dictionary System developed in accordance with emerging standards such as the Information Resource Dictionary System (IRDS) The overall CALS Data Dictionary will be composed of component data dictionaries consistent with emerging CALS Data Standards, including PDES as well as Data Standards for various types of support data. The Data Standards will provide data element definitions, together with the data relationships and rules for data integrity and data consistency required to accommodate the changes in user requirements and computer technologies that are inevitable throughout the 20-to-40-year life of the weapon system.

CALS Benefits

The CALS Report to the Committee on Appropriations of the United States House of Representatives, dated July 1988, identified the following anticipated benefits for IWSDB implementation.

"Industry will eliminate development of duplicative data that drives separate processes in design, manufacturing, support planning, and development of technical manuals, spares provisioning, test equipment, training materials, and other support products. Technical data networks among primes/subcontractors and DoD access to industry databases will streamline weapon system acquisition and shorten lead times for data delivery and spares procurement. DoD will reduce paper deliverables in contracts and reduce government expenditures for manual processes involving paper handling. Design changes will be consistently promulgated throughout DoD's support structure with assurance that the required technical data will be correctly matched to weapon system configuration. Most importantly, the design of the weapon system and its support systems will have high quality by virtue of integrated design processes and a consistent technical database. This last benefit will ultimately lead to increased weapon system effectiveness and combat readiness."

2.3 Architectural Terms

The purpose of the Preliminary CALS Phase II Architecture is to establish a set of terms, concepts, and strategies that can be mutually employed by the defense acquisition establishment, DoD contractors, and technology vendors to achieve the primary CALS Phase II objective of controlled on-line access to an Integrated Weapon System Database as standard Contractor Integrated Technical Information Service. Delivery of CITIS requires a complex information system comprised of other (supporting) information systems. Since CITIS requires establishment of a complex information system, it is important that a well-defined architecture be established and used as the basis for managing near-term improvements, as well as to guide future development efforts.

To facilitate dialog on the Preliminary CALS Phase II Architecture, a commonly accepted framework for defining information systems has been adopted. This framework was originally developed by John Zachman of IBM, and over the last five years it has become accepted within the information systems community as a simple but effective way of describing complex information systems environments. (A copy of the full explanation of the Zachman Framework is contained in Appendix A.)

The Zachman Framework identifies six unique levels of abstraction for system definition:

- Scope/Mission Description

  This is the highest level of abstraction and defines the overall purpose and objective of the system from the end-user's point of view.

- Business Description

  This architectural level describes how the business will function with the system in place. It is a more detailed definition of the overall system and its interactions, still from an end-user's point of view.

- Conceptual Description

  The conceptual description defines the system as a set of integrated information assets that can be implemented with a variety of computing technology in order to satisfy the identified business objectives. The conceptual description is defined from the system integration point of view and is often not formally documented in traditional system development procedures.

- Technology Constrained/System Design Description

  Based on the capabilities of a selected set of implementation technology, the overall design of the system is described at this level. This description is from the system builder's view.

- Detail Description

  The Detail Description is the precise definition of the actual hardware and software components from the system builder/implementor point of view.

- Actual System

  The lowest level of system definition is the actual hardware and software used to process the data.

The six levels of abstraction only represent one dimension of the Zachman Framework. A series of architectural representations are used to describe different architectural views of the system at each level of abstraction. The following three unique types of representation have been identified by Zachman.

- Architectural Representations for Describing Functions

  At the highest level of abstraction, functions are described in terms of business objectives and strategies. Activity models, such as those produced using the IDEF0 modeling technique, provide a functional representation at the Business Description level. At the Conceptual

Description level, the functional view may be described as functional information services which provide end-user information as a result of applying inferencing rules, e.g., the method for calculating mean-time-between-failure, against data values. The System Design level defines functions in terms of the internal software design and reporting formats, based on the selected implementation technology. The Detail Design level defines functions in terms of executable programming languages.

- Architectural Representations for Describing Data

At the highest level, data is described in terms of overall domains of knowledge required to operate the business, or in the case an IWSDB, the domain of knowledge required to support a weapon system life cycle. Data may be described as information products used to operate the business at the Business Description level. At the Conceptual Description level, data are defined by an integrated semantic data model, such as those produced using the IDEF1X modeling technique. At the System Design level, a decision is made on which data management technology to use and data are described in terms of logical database structures, using a database definition language such as SQL. At the Detail Design level, data are described in terms of physical records and pointers.

- Architectural Representations for Describing Network

The system network is defined by basic organizational units and geographical locations at the highest level. At the Business Description level the network is defined by functional groups within the organization with assigned responsibilities. The Conceptual Description of the network includes definition of user types, data and function distribution strategies, and expected user/data interactions. Logical communications network design and hardware and software configurations are defined at the Design Description level. The Detail Description specifies actual computing nodes, data storage nodes, system software, communications linkages, and protocols.

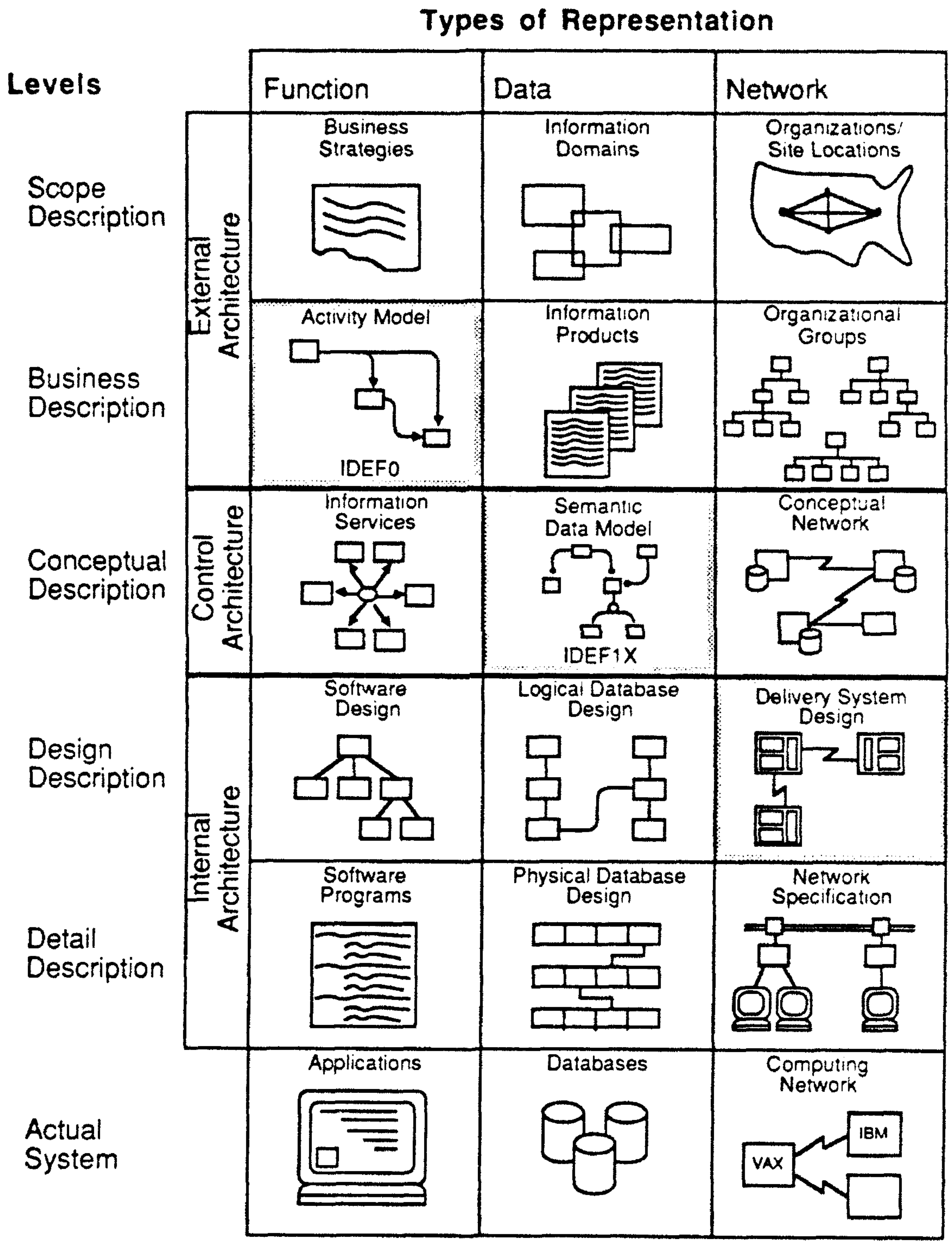


Figure 2-2 Architectural Framework

Together, the levels of abstraction and types of architectural representation form the framework shown in Figure 2-2. This framework will serve to establish a context in which to focus the discussion of the CALS Phase II environment. The scope and business descriptions will be addressed by an External Architecture with the principal focus on weapon system life cycle activities. The External Architecture covers the top two rows of the Zachman Framework using IDEF0 activity models as a principle means of architectural representation. The Conceptual Description, the third row of the Zachman Framework, will be addressed by a CALS Control Architecture with the dominate focus on the data dictionary for the IWSDB, represented by an IDEF1X semantic data model. It is at this level that CALS Phase II standardization efforts must be focused. Establishing a well-defined Control Architecture will be the key to achieving the desired broad-scope integration while allowing for technology migration from existing to future systems. Internal Architectures will be established by each contractor team to addresses the technology constrained and detail design descriptions for implementing the required technology to deliver CITIS, the fourth and fifth rows of the Zachman Framework. The architecture for the system hardware and software network will be the primary focus at this level.

2.4 Industry Trends toward Integration

The Zachman Framework provides a convenient way of describing the basic trends in industry that underlie the CALS Phase II concept, and that lay the foundation for the Preliminary CALS Phase II Architecture described in the next section of this report. The simplest way to describe these trends is to view them from the framework's "horizontal perspective," that is, by looking at them in terms of the rows in the framework.

The first two rows of the Zachman Framework describe business values and strategies that are intended to be supported by an information system. The third row describes the control structures that are moving into place in order to facilitate the mapping between business requirements and operational information systems. And the last two rows address the information systems technology that is being employed to implement information systems in support of the business strategy, and within the context of emerging control structures.

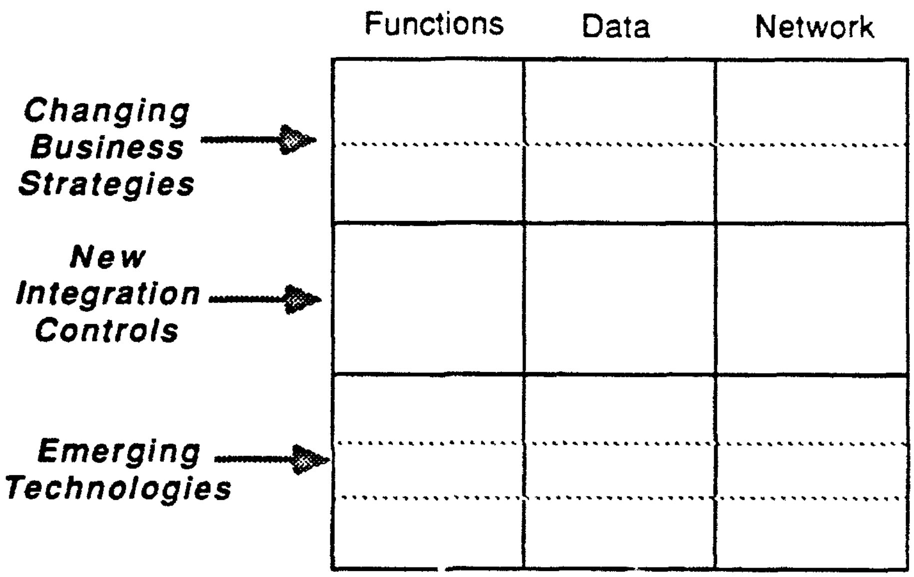


Figure 2-3 Impact of Industry Trends

Changing Business Values and Strategies

The objectives of CALS have been characterized as a natural evolution of Computer Integrated Manufacturing (CIM). CIM — also known as Computer Integrated Engineering and Manufacturing (CIEM) —is the best known strategy for industrial modernization. Forty percent (40%) of U.S. manufacturers have reported on-going CIM programs. CIM has superseded familiar functional automation strategies of the 1970s and early 1980s such as Material Requirements Planning (MRP) and CAD/CAM (Computer-Aided Design/Computer-Aided Manufacturing). Many manufacturing companies are adopting a CIM strategy because of its focus on the integration of manufacturing information systems throughout the enterprise. CIM is perceived by business managers as a consolidated information management strategy for reducing overhead and production costs, reducing product lead times, and increasing product quality.

According to a survey of 46 major aerospace/defense firms conducted by the Aerospace Industry Association, 39 of the firms surveyed reported that they had formal plans for "100% CIM implementations," and 34 firms reported that they expected to achieve their 100% CIM goal within the next ten years.

As CIM and CIEM have unfolded over the last five years, the result has been that more critical manufacturing information continues to be distributed across computers, of all types and sizes. At the same time, the U.S. manufacturing industry has been evolving from a paradigm wherein individual manufacturers approach the marketplace alone to one wherein the dominant form of supply is defined by "trading partnerships." A 1985 Manufacturing Futures Survey reported an increasing trend in which 60% of a typical manufacturer's costs went into the general category of "purchased material." Purchased material, in this context, means goods and services provided by trading partners.

The convergence of CIM and trading partner teams has led to the emergence of the concept of Inter-organizational Computer Integrated Manufacturing (ICIM).

Trading partner teams are finding it increasingly important to exchange information digitally, that is, directly among their computer systems. General Motors, for example, is establishing a "data pipeline" that will be used to link GM engineering and production facilities with the majority of its 30,000 suppliers.

ICIM got its first significant public exposure in April 1988 at the Enterprise Networking Event (ENE '88) held in Washington, DC. Sponsored by the MAP/TOP User's Group and the Corporation for Open Systems, and conducted by the Society of Manufacturing Engineers (SME), this event was attended by over 7,500 managers and technical professionals, and it dramatically demonstrated the cooperative commitment of manufacturers to the increased use of computer and communications technology.

The information of most concern to manufacturers is that which they develop to define their products and production processes. Originally, this information was contained in engineering blueprints and technical specifications. Initial Computer-Aided Design (CAD) and Computer-Aided Manufacturing (CAM) systems focused on automating the graphical representation part designs. But graphical representation and their underlying geometry is only a small subset of the total information needed to describe a product and its production processes. Digital models of product characteristics are also evolving to address versioning and configuration control, assembly structures, features and tolerances, material specifications and processing instructions, functional characteristics, reliability factors, and maintenance procedures. All of this information must be carefully controlled, efficiently employed internally, carefully preserved and protected, and effectively communicated to trading partners.

This realization of the importance of product definition data has led the manufacturing industry to focus its information management attention on the specific subject of Product Data Definition (PDD), which, in turn, in 1986 led to a major effort to standardize product data definitions. This standardization effort — voluntarily supported by over 260 U.S. and European manufacturers and coordinated by the National Institute of Standards and Technology (NIST) — is called the Product Data Exchange Specification (PDES).

Increasingly, PDES is being viewed as "the critical technology needed to accelerate both CIM and ICIM," and PDES development support has been actively funded by the DoD under the CALS program. In 1988, over ten major manufacturers and computer equipment vendors joined together to establish a privately funded cooperative called PDES, Inc. The stated purpose of the cooperative is "to rapidly accelerate the development of PDES." PDES, Inc. is working closely with the PDES volunteer group coordinated by NIST.

CIM, ICIM, and PDD are specific manifestations of an overall trend in manufacturing, as well as other industries, toward "integrated" — sometimes called, "asset-based" or "data driven" — information resource management (IRM). The trend is most often depicted as the final phase of an evolution from "insular" automation, where computer systems stand alone; to "interfaced" automation, where computer systems are interlinked with digital communications; to "integrated" automation, where computer systems share common databases.

IRM, in general, is a thrust toward improving the capabilities of businesses to manage and effectively employ computerized information, "using the information management technologies of the 1980s and 1990s." These technologies — which include personal computers, workstations, artificial intelligence languages and processors, relational and "object-oriented" database management systems, and telecommunications devices and software — dramatically improve the power of information technology beyond the technologies of the 1960s and 1970s. To effectively implement these new technologies, many Information Systems organizations are adopting planning, development, and support tools and methodologies focused on managing "information assets," that result in shared integrated databases, rather than traditional systems development approaches focused on independent functional applications. The establishment of data administration responsibilities is an important element of the "information asset" oriented approach.

CIM, ICIM, and PDD converge with IRM in the manufacturing environment. They all share a single common ground: the need for integration of "islands of automation" by means of "integrated" or "shared" databases. Data management has become the central focus of automation as it supports the emerging business strategies of CIM, ICIM, PDD, and IRM.

Integration Control Structures

The basic trends in business strategy toward CIM, ICIM, PDD, and IRM are being supported by an evolution of "consensus standards." These consensus standards, while they may or may not be officially sanctioned by a formal standards group such as the American National Standards Institute (ANSI) or the International Standards Organization (ISO) represent a felt need on the part of consumers for increased control over the evolution of both information systems functions and information technology.

Standards thrusts have impacted all four of the areas of strategic interest to manufacturers. In the CIM/CIEM arena, most corporations are setting internal company standards - often borrowed from standards organizations or from technology vendors - to control the evolution of both functions and technology. On the functional front, these standards often take the form of a company Information Architecture, such as the one depicted in the CASA/SME Wheel. On the technology front, most of the CIM/CIEM technical standards are intended to control internal communications, hardware/systems software, and data management, what is traditionally called the Delivery Systems Architecture. Additional company standards are being set for the control of automation planning, software development, and software maintenance.

In the ICIM arena, multiple sets of standards are being developed. For the control of inter-organizational functions, standards are being set for the control of document interchange in areas such as purchasing, invoicing, and inventory management. These standards are, for the most part, being set by major suppliers; however, many of them are beginning to find themselves formalized by industry standards groups such as the Automotive Industry Association. On the technical side, ICIM standards are being developed by organizations such as MAP/TOP for the control of inter-organizational communications. These groups focus primarily on telecommunications standards and on data management standards.

For Product Data Definition, standards and guidelines such as PDES and IGES are emerging for the control of technical data describing products. In addition, work is being done on engineering standards as they affect processes such as concurrent engineering and simultaneous design.

Most of the standards that affect Information Resource Management have to do with information technology. Of specific interest, because of the overriding concern for data management, are standards such as SQL and IRDS which affect database management systems and data dictionary systems.

Overall, consensus standards are experiencing an upsurge in popularity within the manufacturing community. However, because there are so many of them, most manufacturers are opting to select a specific set of standards that they believe to be appropriate to their business, and to assemble them into their own internal control structure. It is this control structure that will have an increasing effect on the evolution of technology and on the evolution of business capabilities in the future.

Emerging Technology

As business strategies and integration controls are heading rapidly in the direction of integration of processes and data, emerging technologies are facilitating the change. Though technology evolves in many forms, of most interest is the evolution of data management technology. The fundamental technology trends here can be observed in three major areas: 1) database management systems, 2) distributed, heterogeneous data management, and 3) data dictionaries/directories.

- Database Management

  With the strong emergence of relational database management technology, supported by the wide acceptance by major hardware vendors of the American National Standards Institute's Structured

Query Language (SQL) standard, there is little doubt that, over the next decade, relational database management will become the single strongest technical force in providing future applications solutions.

There are many important reasons why this is true. The facts of the SQL standard and the strengthening support of major hardware vendors (particularly IBM) are clearly important. But more important, relational database management is the key to accelerate many other trends, such as distributed data processing and user-controlled application solutions. Both IBM and DEC have announced that relational database management systems are pivotal to their future product architectures. Further, because of the widespread acceptance by customers and vendors alike of the SQL standard, relational database management is rapidly becoming insensitive to its hardware platform. It is currently being offered by PC vendors, workstation vendors, and mainframe vendors based on the SQL standard. As a result, relational database management is becoming an integral part of the "open system solution" being touted by all major suppliers.

Even more important than its emerging acceptance as "the" database management facility of the future for "all" hardware platforms, relational database management technology is much more extensible than predecessor database management technologies. Relational database management is seen throughout the industry as the natural bridge to even more advanced forms of database management such as "object-oriented" database management, an important technology underlying the emergence and growth of artificial intelligence.

- Distributed, Heterogeneous Data Management

One of the most aggravating situations faced by manufacturers is the simple fact that their databases are scattered all over the place, on different computers, from different vendors, in different application systems, and controlled by different database management systems.

Since few believe that all of these databases can be integrated into a homogeneous environment, most are committed to developing technology that will integrate them where they stand. This technology is generally called distributed, heterogeneous data management technology, or more simply, "integration technology."

For the last decade, researchers have been working on the development of practical integration technology. Significant breakthroughs have been made by research programs such as the Integrated Design Support System (IDS) and the Integrated Information Support System (IISS), both sponsored by the Air Force; Multibase, sponsored by the Navy; IPAD, sponsored by NASA; and IMDAS, sponsored by the NIST Advanced Manufacturing Research Facility (AMRF). Some commercial vendors have moved to produce integration technology products such as TRW's CIM Engine. Integration technology presents an extremely challenging problem to technology vendors, and it has received intense evaluation by both IBM and DEC.

- <u>Data Dictionaries/Repositories</u>

  At the heart of the data management solution is what is called the data dictionary. This technology is basically an automated library control system used to manage and control data stored in databases. Recently, ANSI approved an Information Resource Dictionary System (IRDS) standard, and this standard has set the stage for a more efficient evolution of data dictionary technology. Major technology vendors are moving aggressively in the data dictionary direction, particularly as data dictionaries affect distributed, heterogeneous database management.

These trends in shifting business strategies, approaches to enterprise integration, and standardization of enabling technologies play an important role in the establishment of an architecture for a CALS Phase II. Although the requirements of the government, as the customer and as a major consumer of technical data, will influence the architecture, the best solutions can only be found within the context of a natural migration of industry capabilities toward a shared vision of a CALS Phase II environment.

Example of Trends

Though there are numerous examples from Aerospace/Defense of how these four critical trends are being assimilated into automation strategies, the example shown in Figure 2-4 is one of the most descriptive. The example comes from Martin Marietta's Electronics and Missile Division, and represents a three-year strategy for integrating CIEM, PDD, ICIM, and IRM into one consolidated thrust.

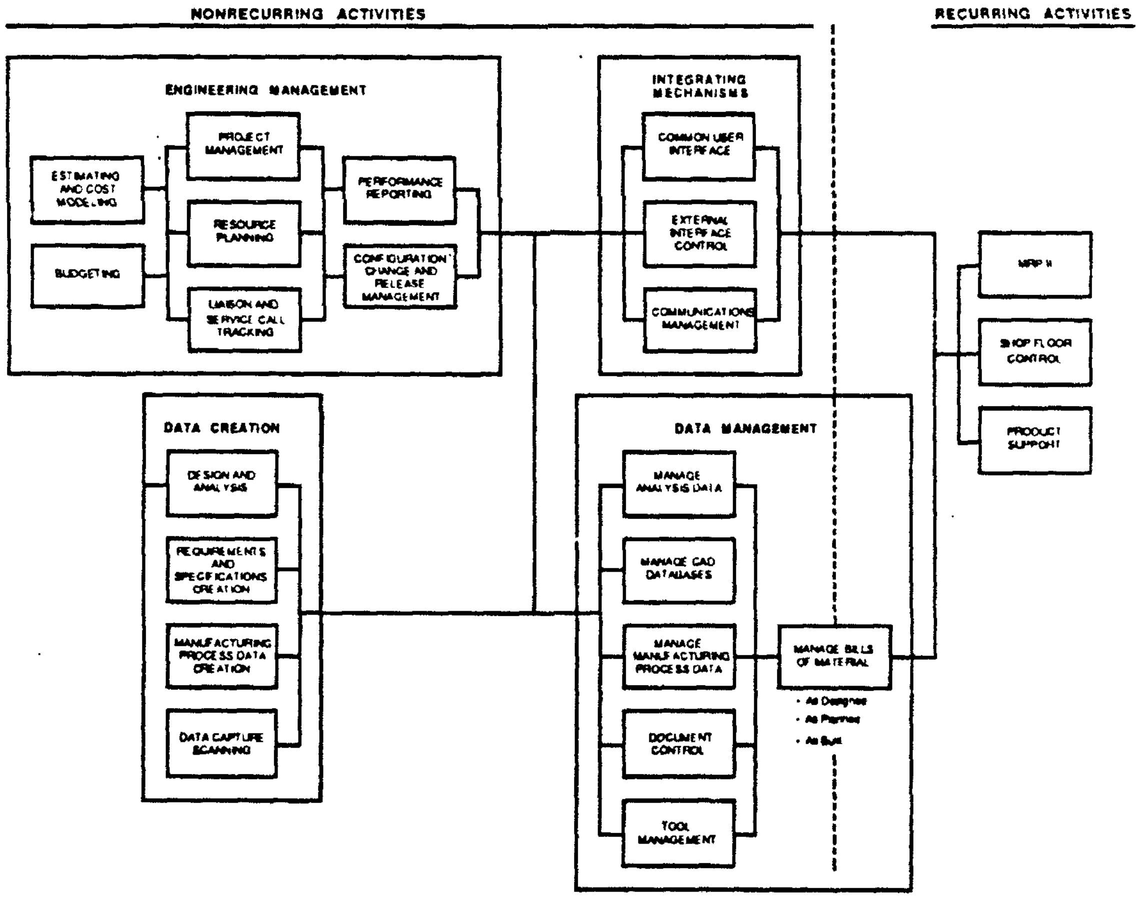


Figure 2-4 Martin Marietta Strategy

Martin's architecture was developed specifically in response to its business strategy of reducing costs and increasing flexibility in response to customer needs by integrating its internal product configuration management, design, manufacturing, and support systems, and by linking those systems to supplier and customer systems. Its next step is to formalize an internal control structure for this architecture, and subsequently to modernize its delivery systems architecture.

Of critical importance in the Martin Marietta architecture is the central role of data management in controlling the definition prod ict data; internal integration of program management, engineering, production, and logistics; and integration with subcontractors and customers. Martin's basic strategy is to use the data management "node" as the primary access point and integration mechanism for all internal and external application systems.

Section 3.
Concepts for a CALS Phase II Architecture

3.1 Objectives of the CALS Phase II Architecture

The CALS Phase II Architecture is intended to create a common industry and government vision for a future environment that supports integrated and shared technical data throughout the weapon system life cycle. This vision, developed from a top-down perspective, will serve as a guide for both managerial and technological changes with a primary focus on the development of standard interfaces for the required interaction between contractors and government organizations.

Although few organizations have yet mastered enterprise-wide integration, the CALS Phase II Architecture must take on the even larger issue of inter-organizational integration throughout a weapon system life cycle. The Architecture must, therefore, recognize and support the value chains that result from the interrelationships of autonomous organizations. To create the ultimate win-win strategy for government and industry, the overall Architecture must be consistent with the internal integration strategies of both government organizations and defense contractors. It must also deal with the legacy of today's systems and the need to evolve rather than start over.

The primary goal of CALS Phase II is to provide an environment in which the technical data of a specific weapon system is logically integrated and shared among contractors and government organizations throughout the weapon system life cycle. This environment will result in an Integrated Weapon System Database (IWSDB) that manages common digital data defining configuration, design, manufacturing, and logistic support information for each specific weapon system. Although an IWSDB is logically integrated, thus providing uniform on-line access by all users, the actual data will be physically distributed among contracting elements. Each IWSDB will provide rapid and secure availability of technical information to both DoD and industry contracting elements throughout the lifetime of the weapon system. The ability for elements of the DoD to have on-line access to an IWSDB will ultimately replace the need for technical data deliverables in paper/microfiche form (CALS Phase 0) and even in digital document form as supported by CALS Phase I standards.

Access to an IWSDB is provided by a "Contractor Integrated Technical Information Service" (CITIS). This standardized service is delivered through inter-connected computing networks and application software programs that are utilized by the contractor team to enter, update, manage, and retrieve data from various internal technical databases to support a specific weapon system. Overall responsibilities to manage access to an IWSDB may migrate between contractors and government organizations during the weapon system life cycle.

A common set of generic requirements for an IWSDB and its associated CITIS will serve as a reference model for specific individual implementations. Together, the IWSDB and its associated CITIS comprise the implementable elements of the CALS Phase II Architecture that are managed and controlled for each specific weapon system. Individual weapon system programs may modify or extend the reference architecture. However, because each IWSDB will be implemented consistent with the Data Standards defined in the CALS Data Dictionary - and maintained in a CALS Data Dictionary System - and each weapon system's CITIS will be provided according to CALS Functional and Technical Standards, over time, a high degree of consistency and inter-relatability across weapon system programs will be achieved.

Although substantial improvements are being made regularly, in today's environment many existing contractor technical information systems are, at best, only partially integrated. That is, only a portion of the data stored in them and the functions performed by them are consistent or compatible, requiring little or no translation. In order to achieve an "integrated" environment, Functional and Technical Standards defined in the context of a standard reference architecture are needed for building or retrofitting contractor technical information systems to support a specific weapon system program. Technical Standards will facilitate the inter-connectability of computing networks and Functional Standards will define Contractor Integrated Technical Information Service (CITIS) from the end-user perspective.

In addition to requiring integration of the prime contractor's internal data and processes, CALS Phase II requires integration of prime contractor data and processes with subcontractor and vendor data and processes, and with GFI, as appropriate for each weapon system.

Traditionally, technical data automation efforts have focused primarily on support for functions internal to prime or major subcontractors. Thus, most architectures for CIM (Computer Integrated Manufacturing) and CIEM (Computer Integrated Engineering and Manufacturing) do not address requirements to integrate multi-organization design and manufacturing teams, much less support the concepts of an IWSDB. Only recently has the manufacturing industry begun to recognize the need to address "inter-organizational" CIM (ICIM). Some leading manufacturers, particularly in the automotive industry, have established technical data networks with their suppliers. These initial trading partner networks, however, generally lack the flexibility and rigor required to support the complex technology of a major DoD weapon system throughout its entire life cycle.

The IWSDB identifies the logical interrelationships and semantic definitions of the configuration, engineering, and support data that must be shared by contracting team members to manage specific weapon system data required to sustain a weapon system throughout its life cycle. The requirements for an IWSDB will ultimately be reflected in a CITIS procurement specification by means of CALS Data Standards. The goal of the Preliminary CALS Phase II Architecture is to identify the overall scope and objectives of the IWSDB and its associated CITIS and to outline a strategy to develop the procurement specification. Rather than automate redundant data in document form, the IWSDB specification defines an integrated data structure that supports multi-use digital models of weapon system characteristics. The proposed data architectures of PDES and MIL-STD 1388-2B provide a baseline for the IWSDB specification, which must be validated against the requirements of existing weapon system programs and contracting practices.

The functional application of integrated weapon system technical information to support various contractor, subcontractor, and government life cycle activities will also ultimately be defined by a CITIS procurement specification. The functional architecture of this procurement specification will be implementable as CALS standards for application development and will generate information consistent with CALS Functional Standards. Current CALS Phase I functionally-oriented standards, along with emerging functional requirements for ILS and Concurrent Engineering, provide the basis for future CALS Phase II Functional Standards consistent with the concepts of an IWSDB and associated CITIS.

The functional requirements and role of an IWSDB and its associated CITIS change throughout the entire weapon system life cycle. The CALS Phase II Architecture must address all phases of the weapon system life cycle, including the following:

- Concept Exploration Phase
- Concept Demonstration/Validation Phase
- Full-Scale Development Phase
- Production Phase
- Operation and Support Phase

The concept of an IWSDB is based on logical and not necessarily physical integration of a weapon system's technical data. The definition of an IWSDB's logical structure provides a set of rules for specifying how technical and support data can be fully integrated across physically distributed databases. These physically distributed databases include subcontractor and vendor databases, as well as prime contractor and government databases; therefore, a basic architecture for networking the prime contractor with both government and subcontractor databases must ultimately be defined. This network architecture will be implementable according to CALS Technical Standards for systems interconnection and communications protocols, referenced by the CITIS procurement specification. An ancillary objective of the Preliminary CALS Phase II Architecture is to identify the overall scope and objectives of a contractor's Internal Architecture and to understand the technical dynamics of prime, subcontractor, vendor, and government interaction within the context of an IWSDB. The Preliminary CALS Phase II Architecture supports a strategy to migrate from information exchange based on the transmission of documents and stand-alone files, to data sharing based on transactions against shared integrated databases, which are consistent because they have been validated against the data integrity rules contained in the IWSDB.

The automated support of a weapon system's technical information must rely on multiple, heterogeneous computer systems managed and controlled by independent organizations, each with its own internal technical standards and business objectives. This results in dissimilar and incompatible hardware and software systems operating on a concurrent basis, even within individual organizations. While these systems may meet the objectives for which each was designed, their heterogeneity presents a major obstacle to ready access and assimilation of the technical information they contain.

The problem of managing distributed heterogeneous information systems is common to many industries, ranging from manufacturing to banking to retail distribution. These "complex information systems" must be geared to span applications, functional areas, organizational boundaries, and geographic separations in order to present a unified picture to the user. When designing complex information systems, it is necessary to look at a number of interrelated strategic, technical, and organizational issues.

Strategic issues include inducing cooperation between multiple, diverse organizations, each with its own goals, priorities, and security needs. One critical success factor for such cooperation is participant consensus on the issue of access to each other's technical information. There is a critical need to clearly define the domains of shared information, to agree on a set of rules that will be used to control the data from which shared information is derived, the potential benefit of data sharing and information sharing to each participant, and the role and the responsibility of each with regard to specific technical implementations that support the environment.

Under technical issues, the evolution and physical interconnection and management of distributed heterogeneous information systems and databases must be addressed. In addition to physical connectivity issues, it becomes essential to establish new techniques for incorporating logical connectivity across systems. Such techniques employ ideas from the fields of database technology, communication technology, and knowledge engineering.

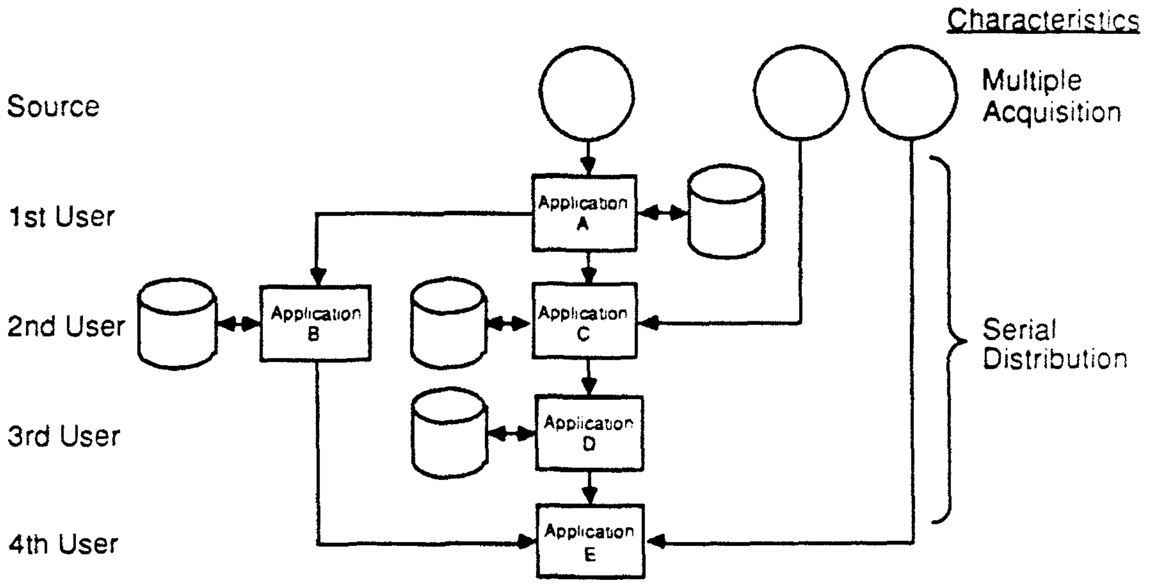
The CALS Phase II Architecture must serve as the foundation for automating the inter-organizational efficiency in all stages of a weapon system's life cycle. The resulting organizational issues centered on the process of making controlled changes in complex organizational environments must be addressed before an IWSDB and associated CITIS implementation plan will be successful.

3.2 The Central Role of Information Asset Management

The long-term objective of CALS is to have an environment in which all the technical information required to design, manufacture, operate, and maintain a weapon system is available electronically through on-line access to an integrated database that ensures the currency and integrity of these data. Instead of producing documents in hardcopy form or even in electronic document form, such a system would generate the required information from the database as needed and in the form needed. Ultimately, the integrated database must be capable of supporting automated interpretation of weapon system characteristics in order to drive advanced application systems without the need for human interpretation of the data. This is a long-term goal that will require resolution of a number of managerial, as well as technical issues; however, some very significant capabilities can be implemented in the near-term and, with the proper architectural planning, these near-term improvements can contribute toward the long-term solution. Without proper consideration for the long-term architecture, however, some near-term enhancements may actually inhibit future integration.

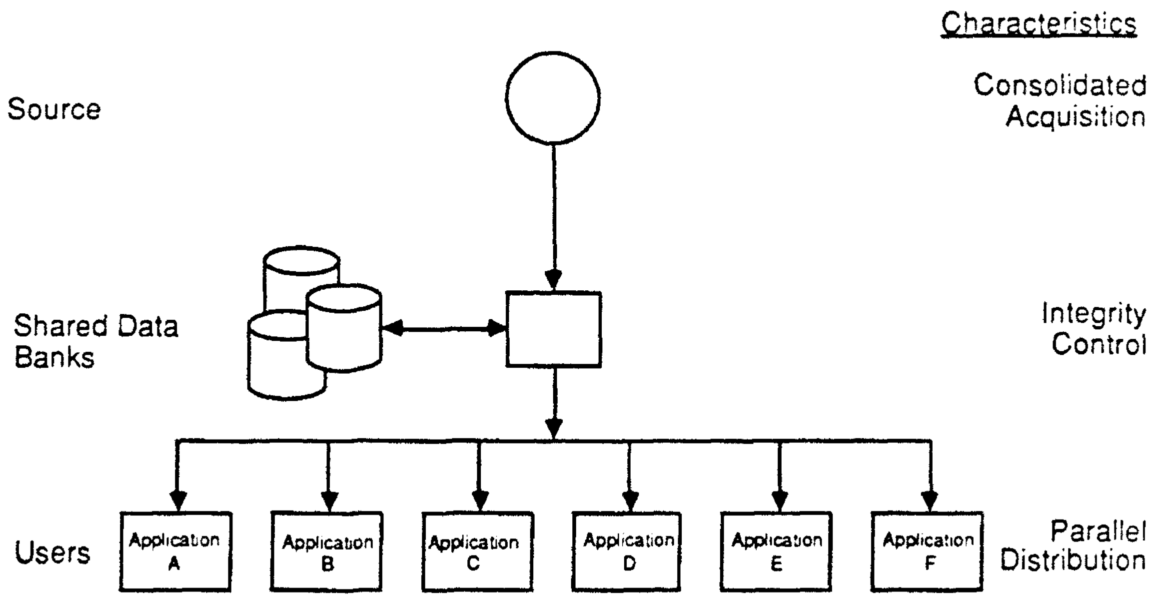
Although considerable investment has been made in automation of all aspects of design, manufacturing, and product support over the last thirty years, most of this automation has been put into place without the benefit of a plan for enterprise-wide integration, much less a plan for integrating government components, prime contractors, co-designers, subcontractors, suppliers, second-source producers, and other organizations that must interact throughout the life cycle of a major weapon system. The predominate approach has been one of automating specific functions within a particular organization. (See Figure 3-1.) The result of applying this approach over time is the serial distribution of data, where the output of one system is modified to serve as an input to another system. Multiple acquisition of the same data often results when the data from one system is not in a suitable form for use by another system. Currency and integrity control of data is extremely difficult, if not impossible, in this environment.

The need for integration has grown from the bottom up. As the overlap in functionally-oriented systems was recognized, new systems have been identified with broader and broader scopes. Basic CAD systems, for example, are being replaced with more comprehensive systems that use a common database for design, engineering analysis, and manufacturing planning. Ultimately, it must be recognized that broad scale integrated systems that cut across organizational boundaries cannot be developed using traditional functional automation system design concepts and techniques. Thus, the need for an information asset management approach that builds reusable shared databases from an enterprise perspective is gaining widespread recognition. Under this approach, data are identified from the common perspective of the environment being integrated, and a single source of



Applications A, B, C, D, & E are "Islands of Automation."
Currency and integrity control is difficult and expensive.

Figure 3-1 "Applications-Oriented" Systems Design



Shared pool of reusable data is provided
to applications "on demand"

Figure 3-2 "Data-Oriented" Systems Design

acquisition is established for each unique type of data. The result is a pool of reusable data with controlled integrity and currency. Parallel distribution of the data can then be made to various functional applications as needed. (See Figure 3-2.) This approach supports a basic government objective to "buy data once and reuse it many times."

The information asset management concept offers a viable approach for the establishment of a CALS Phase II environment that can satisfy both contractor internal integration needs and support the requirements of subcontractors, vendors, and government components. A number of problems, however, must be solved in order to fully implement the approach. First of all, a common understanding and acceptance of the long-term architecture must be achieved between government and industry. Not only must the ability to satisfy government requirements for information access be demonstrated, but also the architecture must address contractor internal integration needs in order to be accepted by industry. The major investment in existing systems will most likely be an inhibitor to this acceptance. An evolutionary strategy will be required to either retrofit existing systems to the new architecture or replace them. The current environment of multiple source data acquisition makes the job of retrofitting more difficult.

A system architecture that supports a wide area computing network also presents some unique technical challenges. The potentially vast amounts of data will dictate a distributed database management support system, even though the data must be defined and managed as one logically integrated database. The establishment of standard data dictionaries that rigorously define shared weapon system technical data will be important to managing the distributed databases.

In addition to retrofitting internal contractor functional application systems and databases, government requirements for the delivery of technical data must be restated in terms that are consistent with the CALS Phase II architectural concepts. Many CDRLs in today's environment are heavily oriented toward documents in either hardcopy or electronic form. These requirements must be restated to identify non-redundant data requirements for an integrated database environment.

3.3 The Three Architecture Concept

The central theme of Information Asset Management in a complex information system is an infrastructure that allows alternative, self-sufficient component subsystems to be utilized as long as they conform to standard rules defined for the overall environment. Multiple, self sufficient subsystems that conform to a consistent set of standard rules are deemed to be "integrated."

Although the idea of standard rules is most often applied through standard functional interfaces and protocols at the communication and operating system level to create an environment of "cooperating" computing systems, the CALS Phase II Architecture makes a major advancement by applying the standard rule concept to information itself. It addresses the issues of standard data rules, which are used for controlling the quality and integrity of information, not just the connectivity of computer hardware or the interoperability of systems software. The importance of standard data rules in controlling a complex information system is embodied in the "three architecture concept" presented in the 1987 CALS Framework.

The three architecture concept is an extension to the 1977 ANSI/X3/SPARC study on the standardization of data management systems. Expounding the "Three Schema Architecture," the ANSI report observed that traditional data management systems manage data in two separate structures or "schemas," an "internal schema" as seen by the system and an "external schema" as seen by the user. The report noted that to manage data effectively requires a third structure, called the "conceptual schema." This structure must be independent of, but transformable into, the other two structures. It represents a neutral, independent set of data rules that are used to control multiple, derivative data structures.

The role of the conceptual schema in the three schema concept is to provide a single, unambiguous, internally consistent, and minimally redundant set of internal rules that control the data resource independently of both functional applications and computer implementations. Application and implementation views are mapped to the conceptual view to provide flexibility and integrity control, as well as data integration.

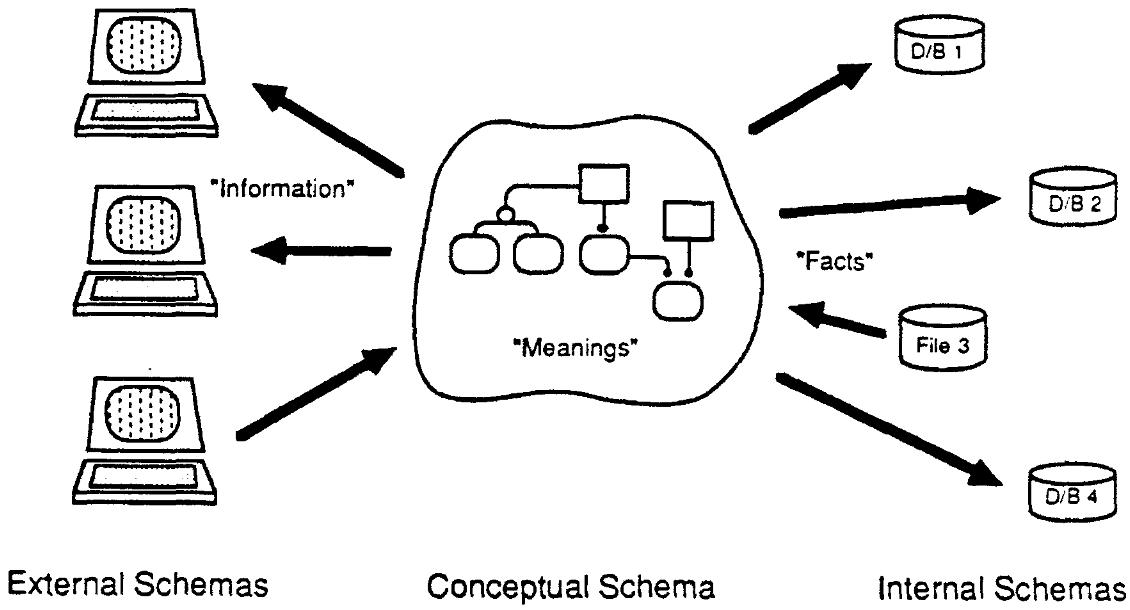


Figure 3-3 Three Schema Architecture

The Three CALS Architectures

Considerable effort has already gone into the definition of a CALS Framework, and this framework provides the point of departure for the Preliminary CALS Phase II Architecture. The draft CALS Framework document, developed in March of 1987, defines three elemental architectures for the CALS environment based on the Information Asset Management concepts. The three elemental architectures, as shown in Figure 3-4, include the following:

- External Architecture, which defines the user view of the information required to support various functional applications.

- Internal Architecture, which defines the computer systems technology necessary to automate the required information.

- Control Architecture, which defines functional, technical, and data standards and procedures for maintaining alignment between the External and Internal Architectures.

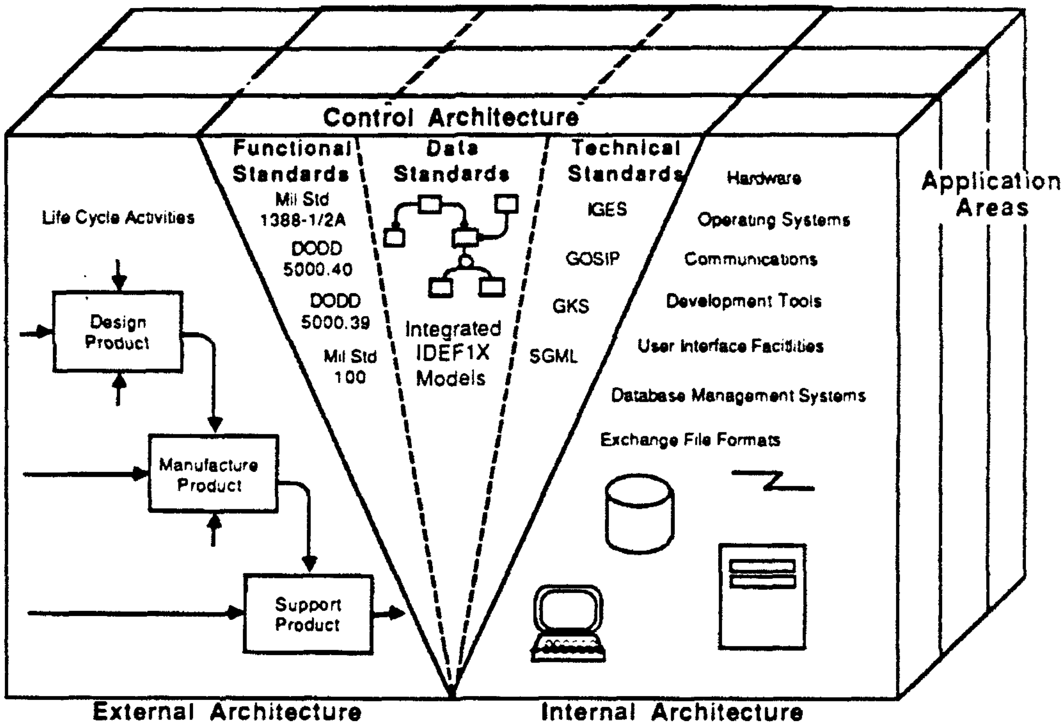


Figure 3-4 CALS Architectures

The purpose of the Control Architecture is to specify functional, technical, and (for CALS Phase II) data standards that are independent of specific user requirements (defined by the External Architecture) and of specific applications, databases, hardware, and software implementations (defined by the Internal Architecture).

The Internal Architecture defines the specific hardware and software technology that will be used to implement the shared data resource for application across all life cycle functions. In order to develop a CITIS delivery system and to demonstrate an implementation of an IWSDB, the Internal Architecture should be based on available technology with mappings to an integrated data structure and required functional views. The Internal Architecture must address issues such as geographical location of users, data volumes and access patterns, and user interface requirements. A number of Internal Architecture strategies for distributed heterogeneous database access have been studied, including the work of the Air Force IISS and IDS programs. In addition, a number of major defense contractors, hardware and software vendors, and telecommunication suppliers are working to establish Internal Architectures for management and use of technical data.

The three architectures map to the Zachman Framework as shown in Figure 3-5.

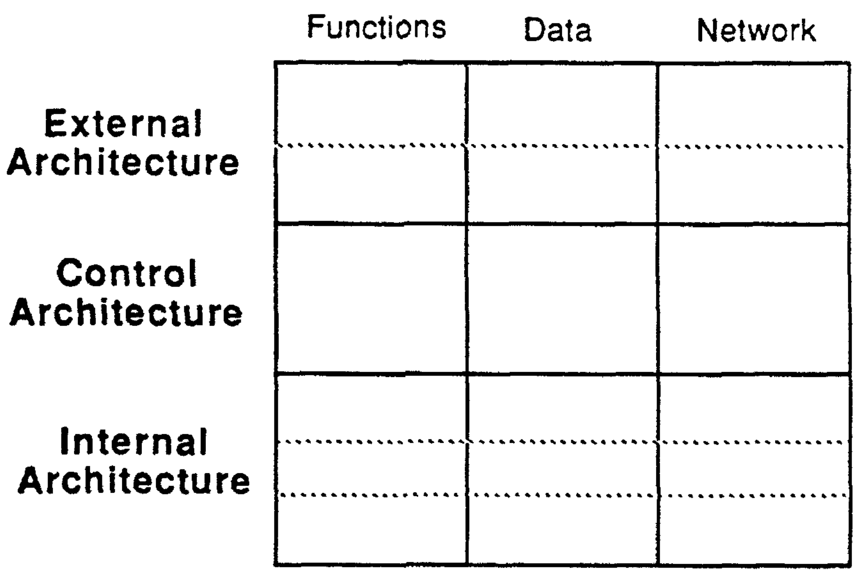


Figure 3-5 Three Architecture Framework Mappings

3.4 The CALS Control Architecture

The CALS Control Architecture, as defined in the CALS Framework, identifies three types of standards:

- Functional Standards - e.g., MIL-STD-1388, DOD-STD-100, which define the information requirements to support various life cycle processes and functions,

- Technical Standards - e.g., SGML and GOSIP, which define the hardware, system software, application software, data management software, and communications technology used to define a specific delivery system,

- Data Standards - which define the underlying integrated data structure independent of any functional application or computer implementation format.

The three forms of CALS standards map to the Zachman Framework as shown in Figure 3-6.

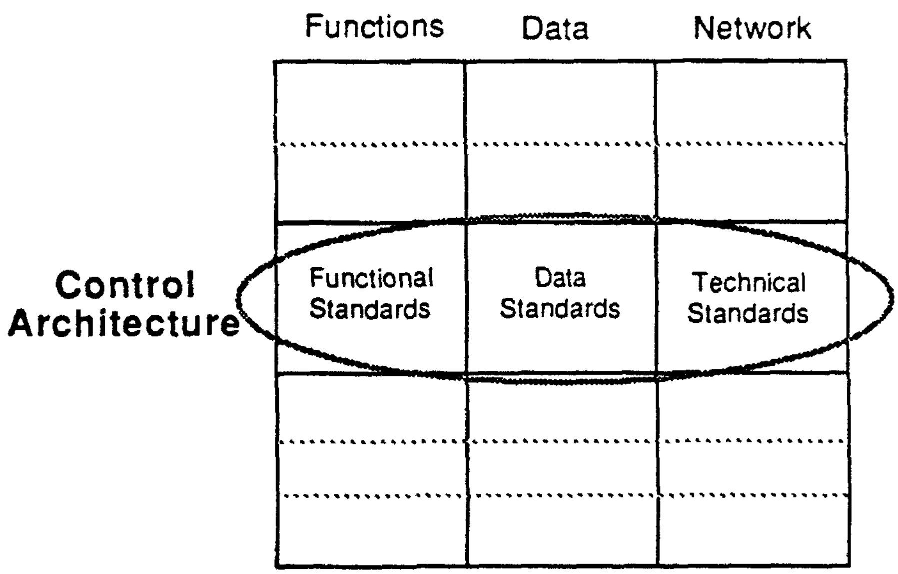


Figure 3-6 Control Architecture Framework

CALS Phase II versus CALS Phase I

Phase I of the CALS Program is focused on near-term objectives which make use of existing standards. A subset of available standards was selected to create a draft CALS Phase 1.0 Core Requirements document. The initial set of standards included MIL-STD 1840A, SGML, the CCITT Group 4 raster scan standard, IGES, GOSIP, and DDN. A "CALS Implementation Guide for Weapon System Acquisition" was also developed as a military handbook for applying the CALS Phase 1.0 Core Requirements. The initial standards and guidelines will be continually refined through subsequent releases.

Standardization efforts to date have focused on Functional and Technical Standards, rather than Data Standards. Data Standards that are truly independent of specific applications and implementation technologies do not yet exist. The data modeling efforts of the Product Data Exchange Specification (PDES) project and the MIL-STD 1388-2B development project, however, provide the initial basis for independent Data Standards for industry. Industry-wide Data Standards will be an important part of the long-range CALS goal to create an integrated environment.

CALS Phase I is focused on standards that support the building of interfaces or bridges between independent systems. Most of today's engineering, manufacturing, and logistics systems have been built without formally defining a conceptual schema; therefore, the standards for interfacing have focused on functional standards that provide for document exchange at the external schema level or technical standards that provide for file exchange at the internal schema level.

Document-based exchange often requires human interpretation and reentry of the data for processing by the receiving system. This is even true when a digitized version of the document is used. The use of digitized engineering drawings is an example of redundant digital information requiring human interpretation.

File exchange generally requires translators to be built to move data from one system to another. Since both the sending and receiving systems only support their own internal views of the data, the proper meanings of the data may be lost because of a misalignment of the views or improper semantic interpretations of the data. Many of the reported problems with IGES are a prime example of the difficulties that may be encountered.

The three schema architecture provides an improvement opportunity over the typical document and file exchange scenarios. Even though two systems may be developed and operated independently, if they share a common conceptual schema (a common set of internal rules for control of their data) they should be able to properly interpret and use exchanged or shared data.

The functional, technical, and data standards defined by the Phase II CALS Control Architecture should provide for a consistent mapping from External Architecture requirements to Internal Architecture implementation as shown in Figure 3-7. Functional Standards control how functional requirements are satisfied by application software; Data Standards control how information requirements are satisfied by shared databases; and Technical Standards control how networking requirements (organizational interactions) are satisfied by computing hardware and software networks.

Current "data standards" exist only at the lower "technology constrained" levels of the Zachman matrix. MIL-STD 1388-2B, for example, identifies Logistic Support Analysis data using SQL table definitions based on current relational database technology. The use of a language such as IDEF1X, a commonly used semantic data modeling language developed by the Air Force, provides a standard definition language for describing data standards that are not technologically constrained, and which can therefore become part of the CALS Control Architecture. The IDEF1X representations of PDES and the future IDEF1X representation of MIL-STD 1388-2B are future reference models for CALS Data Standards.

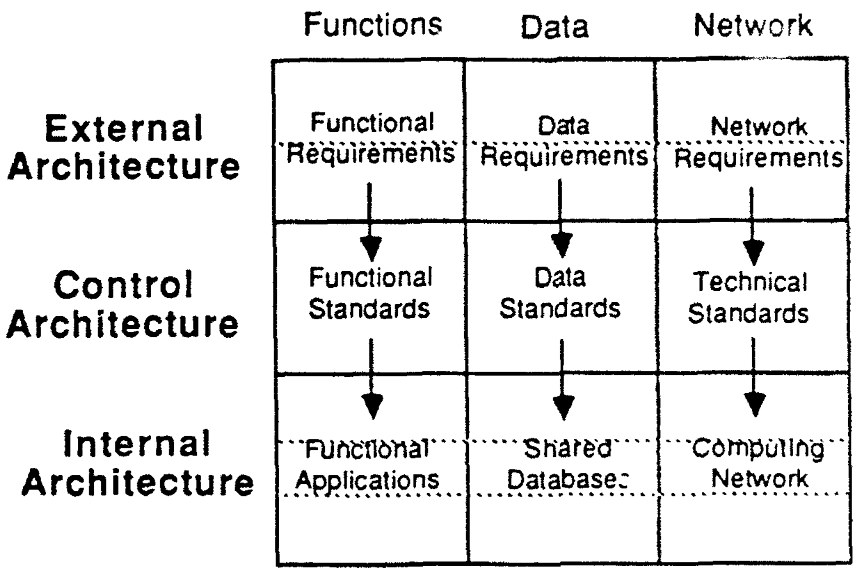


Figure 3-7 Role of Control Architecture Standards

Current Phase I "functional standards" are also described at the "technology constrained" levels. DOD-STD 100, for example, describes engineering drawing information based on a paper/microfiche document format. Functional Standards for CALS Phase II must ultimately be independent of the implementation technology.

"Technical standards" generally deal with only lower level "technology constrained" definitions of computer systems. Networking topologies in the CALS Phase II environment should be defined independent of specific operating systems, communications protocols and other technical standards. However, sufficient standards must be in place to allow for the interconnection of prime, sub, and government internal networks. Database portability, software portability, and standard user interfaces are also issues that must be addressed by Technical Standards.

Section 4
CALS Phase II Architecture Requirements

4.1 External Architecture Requirements

According to the CALS architectural framework, the External Architecture describes the target system from the user viewpoint in terms of its scope and the business model which it will support. Defining the user's view of a CALS Phase II environment is a complex problem since no single user or organization has a full appreciation of all the requirements for the total system. The DoD logistics infrastructure, as the customer and weapon system user, has specific requirements for the IWSDB and associated CITIS which must be identified and evaluated. These requirements define an External

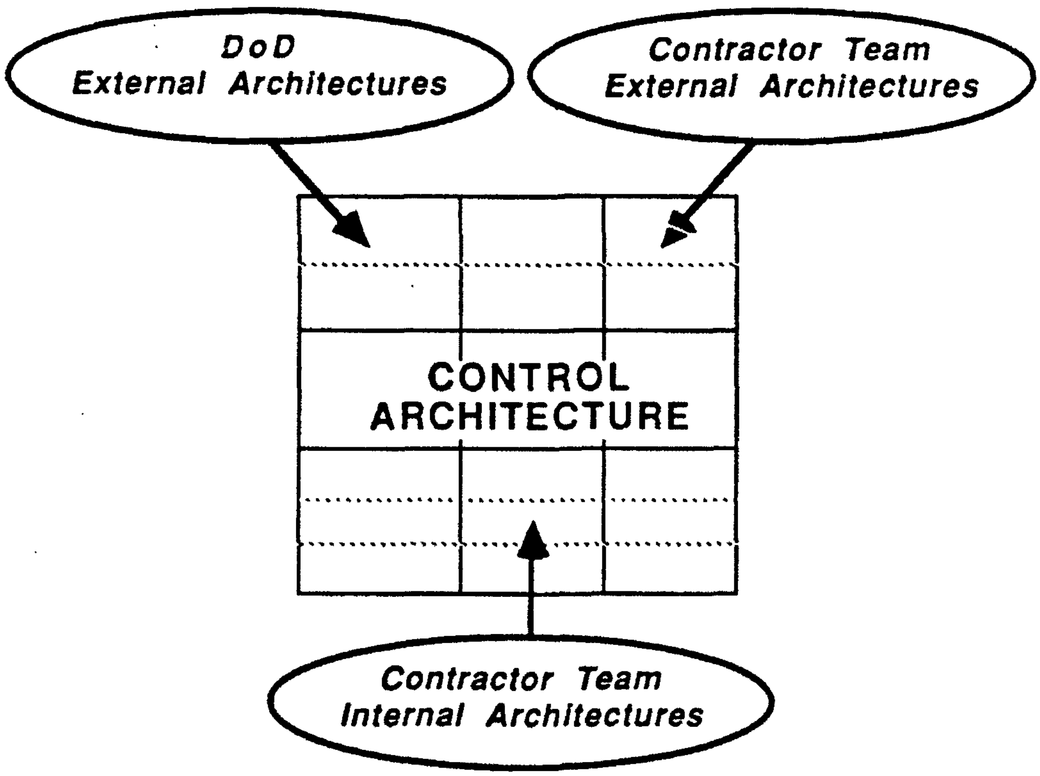


Figure 4-1 DoD and Contractor External Architectures

Architecture from a DoD view that must be satisfied by the contractor team's Internal Architecture. The functional, data, and technical standards defined by a common CALS Control Architecture must support the mapping of DoD External Architectures to the Contractor Team Internal Architectures, as illustrated in Figure 4-1.

The contractor team will also have its own External Architecture for the CITIS delivery system. Although the prime contractor will generally have the responsibility of putting the CITIS delivery system in place, the weapon system technical data it manages may migrate from contractor to contractor over the life cycle of the weapon system. Therefore, a clear definition of the overall user view of an IWSDB and its associated CITIS is critical and requires agreement from all organizations that support the weapon system life cycle.

Business Strategies

The CALS Phase II Architecture must recognize and support basic business strategies that are common in the defense industry. Defense industry business strategies must address the following areas:

- Multi-phase/multi-contract development programs
- Multi-corporation development teams
- Government-furnished information and equipment
- Government-owned and contractor-managed technical data

Multi-phase, multi-contract weapon systems development programs require support for migration of technical data from contractor to contractor or from contractor to government. Ultimately, CITIS may need to be contracted for separately from weapon system development contracts in order to guarantee continuity in the management of the technical data. The NASA Space Station Program, for example, has an independent contractor for its Technical and Management Information System (TMIS) that will support all contractors and NASA centers throughout the life cycle of the Space Station.

Today's complex weapon systems cannot be designed and built by a single contractor. This means the IWSDB and associated CITIS must support the requirements of a multi-corporation development team. Each team member has its own internal standards and preferred methods of operation, which adds greatly to the problem of integration in a heterogeneous environment. The provided CITIS must support access and use of the IWSDB by development team members, as well as government agencies. Furthermore, in order to create a win-win environment for government and industry, establishment of CITIS capabilities must add to the competitive advantage of the development team in order to promote development and use of an IWSDB.

In commercial manufacturing industries, such as the automotive industry, prime contractors have been able to create technical information system networks with their suppliers by dictating specific hardware and software. This approach is not practical, however, for the defense industry since a high percentage of subcontractors are common between many weapon system programs. In fact, even at the prime contractor level, any two contractors are often partners on one weapon system program while competitors on another program.

Ultimately, subcontractors will benefit the most from a standard "open system" architecture for CALS Phase II, since it will allow them to work efficiently with a variety of prime contractors without having to install and maintain separate systems for each prime contractor. Currently, a subcontractor would need CADD, NCAD, and CADAM to have full digital interchange with McDonnell Douglas, Northrop, and Lockheed.

The government itself must be considered as part of the overall development team that is supported by CITIS capabilities. In addition to review and approval of technical information, the government may also perform design engineering, manufacturing, and product support functions. In some cases, such as overhaul and modification, a government organization may actually compete with commercial contractors. The government may also contract directly for some components that are common across several programs, such as jet engines, rather than subcontracting through a prime contractor. The government must then be responsible, either directly or indirectly, through the associate prime, for supplying the necessary technical information to the prime and for furnishing the actual component.

The primary lines of responsibilities for weapons system development and support gradually change over the life cycle. This concept is illustrated in Figure 4-2. In the initial acquisition phase the government primarily deals with a few contractors. Once the initial design and development are completed, second-source suppliers and replacement contractors are introduced. Ultimately the government, as the owner of the weapon system, must assume a greater role of coordinating the technical activities of even second and third tier subcontractors.

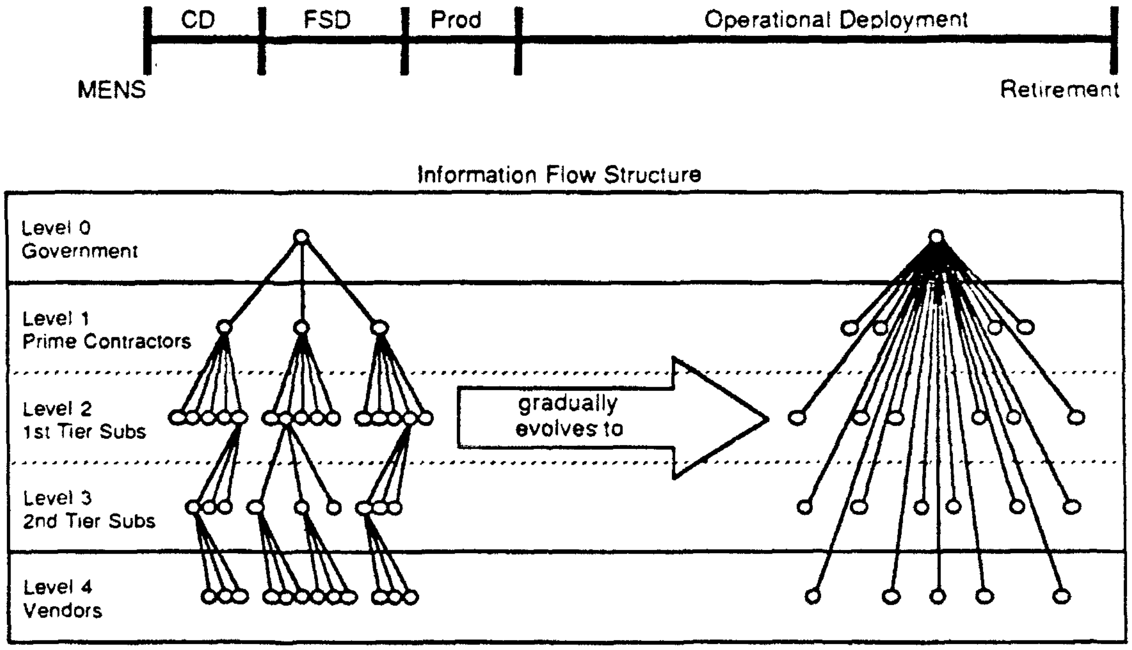


Figure 4-2 Life Cycle Responsibility Changes

Since the government pays for the engineering costs as well as the manufacturing costs, most of the new technical data created for a weapon system is legally owned by the government and is typically supplied by the contractor as a contract delivery item. The prime contractor and the associated subcontractors, however, are better able to maintain the data since they are the creators. A new business strategy that will be supported by CALS Phase II is to allow government-owned technical data to be contractor managed and maintained. This strategy is already being used on a limited basis throughout the services, at least during the acquisition phase of the life cycle. In order to effectively use contractor-maintained technical data, modernization of the government logistics infrastructure must be consistent with the contractor's CITIS capabilities.

Data security is obviously an important issue for any weapon system program. The creation of an integrated database means access security is even more critical since a breakdown in security could result in loss of the entire weapon system specification. Since most contractors have proprietary knowledge which gives them a competitive advantage, security of contractor private information is also a major concern that must be addressed before government and other development team members are allowed direct access to a contractor's database. Several major contractors have solved this issue within limited functional areas.

Organizational Support

The goal of CITIS should be to support all users of technical information throughout the weapon system life cycle. Some weapon system programs have reported involvement of more than 20,000 organizations. The general categories of organizations include government organizations, prime and co-prime contractors, subcontractors, and suppliers. Figure 4-3 provides a brief list of the types of organizational roles that must be supported by CITIS.

Government

- Program Office
- Logistics Centers
- Command Centers

Contractors

- Design Prime Contractor
- Co-Designer
- GFE Contractors
- Production Prime Contractor
- Co-Producer (Dual Source)
- Second Source Producers
- Support Contractor

Subcontractors

- Design Subcontractors
- Manufacturing Subcontractors
- Suppliers (std parts)

Figure 4-3 CALS Phase II Organizational Support

Functional Discipline Support

Each organization requires support for multiple disciplines within the organization, as well as intercommunications between organizations. The basic disciplines that require support within the various organizations include management, engineering, manufacturing, and logistics.

Traditionally, the engineering, manufacturing, and logistics disciplines have been thought of as having serial responsibility as the weapon system progressed through its life cycle. However, the concept of concurrent engineering is highlighting the need to have active involvement of all three disciplines throughout the life cycle. The requirement to support concurrent engineering is further compounded when considering the distribution of various life cycle disciplines across multiple organizations.

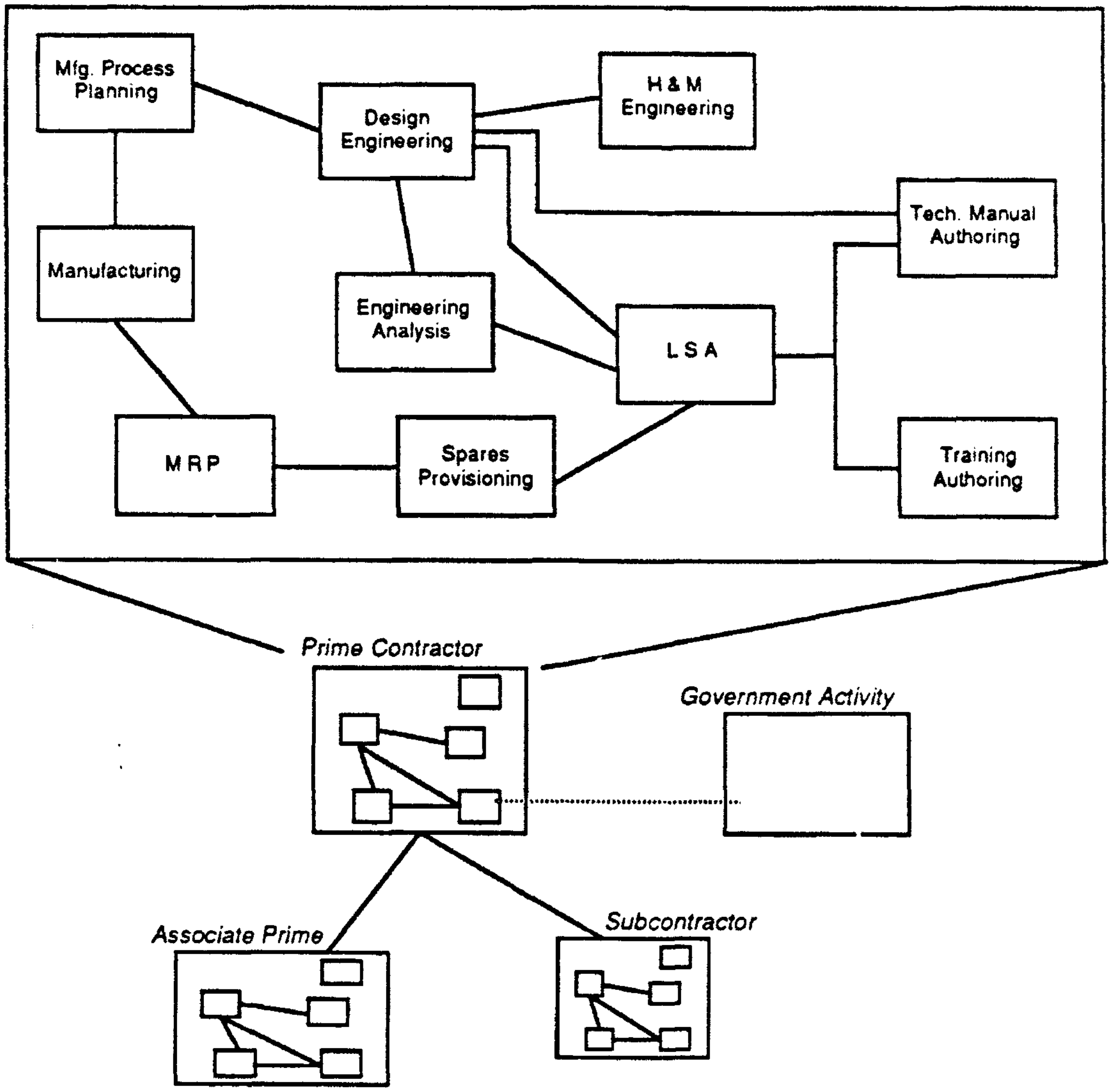


Figure 4-4 CALS Phase II Functional Support

Life Cycle Support

Future CITIS and the resulting IWSDB must support all users through all phases of a weapon system life cycle. This includes support for the typical life cycle phases:

- Conceptual Exploration
- Concept Demonstration/Validation
- Full-Scale Development
- Production
- Operation and Support

The technical data created in each phase is used and expanded in the next phase. A number of activity models have been developed to describe the functions of an aerospace enterprise, including the USAF Factory of the Future Framework. However, an integrated activity model of the total weapon system life cycle is needed to rigorously define an activity-based business model for CALS Phase II. Figure 4-5 provides a simplified view of life cycle activities for a weapon system. The following paragraphs discuss the role of CITIS in each life cycle phase.

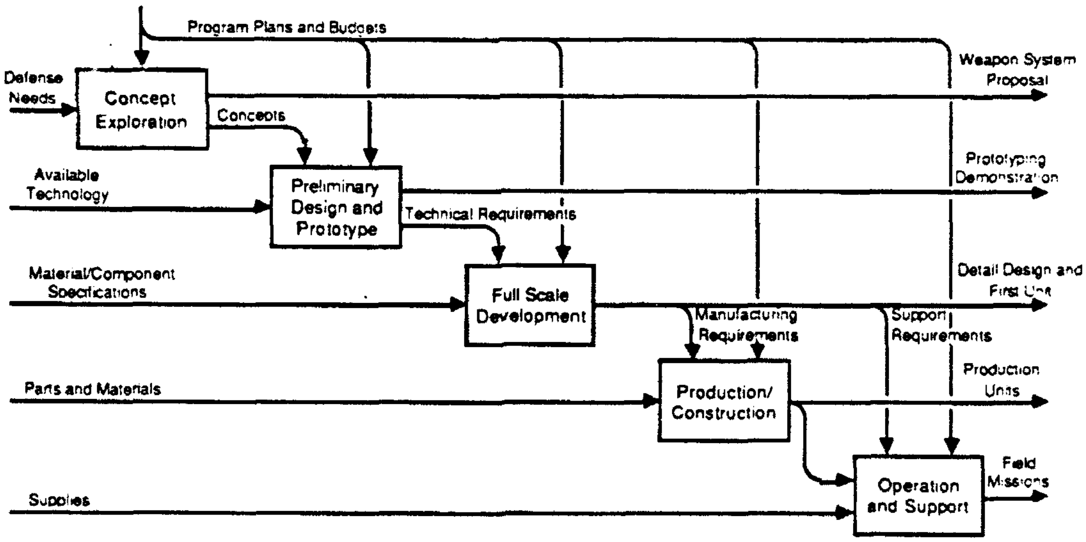


Figure 4-5 Life Cycle Activities

Role of CITIS in the Concept Exploration Phase

In this phase of the life cycle, the contractor will begin to set up the delivery system and database structures for CITIS. GFI will include performance schedules and cost parameters for trade-off analysis. Multiple contractors may be competing to explore system design concepts through relatively short-term planning contracts. The GFI will include operational test conditions, mission performance criteria, and life cycle cost factors.

As the IWSDB and associated CITIS is initially established, it is important to recognize different types of data status. An IWSDB must have the capability of distinguishing among, and providing visibility and accessibility of the following data status:

> Working Data - The government may be provided a read-only capability for in-process review of selected initial or change data/information (using partitioned databases or other appropriate techniques).
>
> Submitted Data - The IWSDB stored data released for review and approval must have a method for incorporation of government-proposed changes and feedback to working data files, while maintaining version control and protection against unauthorized changes.
>
> Approved Data - Data that have been reviewed and approved by the government are archived and require additional controls against unauthorized changes.

Role of CITIS in the Concept Demonstration/Validation Phase

At this stage of the life cycle, a CITIS must support development, fabrication and testing of prototype systems. The IWSDB is oriented primarily around functional subsystems that are required to define the total weapon system. Most weapon system functional requirements will be translated to technical specifications. GFI will expand to include production planning data such as broad cost, schedule, operational effectiveness, operational suitability goals, and thresholds. Design changes still occur frequently and positive control must be maintained over the designation of working, submitted, and approved data.

Another important requirement for CITIS in this phase is support for concurrent engineering. As stated in the CALS Handbook (MIL-HDBK-54), concurrent engineering is a systematic approach to creating a product design that considers all elements of the product life cycle from conception through disposal. In so doing, concurrent engineering simultaneously defines the product, its manufacturing processes, and all other required life cycle processes, such as logistic support. Concurrent engineering is not the arbitrary elimination of a phase of the existing, sequential, feed-forward engineering process, but rather the co-design of all downstream processes toward a more all-encompassing, cost-effective optimum. Concurrent engineering is an integrated design approach that takes into account all desired downstream characteristics during upstream phases to produce a more robust design that is tolerant of manufacturing and use variation, at less cost than sequential design. It affects all system procurement activities from Milestone 0 (concept definition and exploration) to the start of Milestone III (the end of full-scale development).

Role of CITIS in the Full-Scale Development Phase

The scope of IWSDB expands now to fill out the production schedule and provisioning requirements. The engineering and manufacturing specification must be extended to include a complete definition of support specifications prior to production commitment. The primary orientation of the IWSDB is extended from subsystems to detail part numbers with specific production effectivity. The first and second tier of subcontractors and their supporting vendors will be in place and the stream of technical data will flow between subcontractors, the prime contractor, and the government. Engineering design and analysis data is extended with manufacturing specifications. As the volume of changes requiring government approval increases, the importance of on-line access intensifies. The continued positive control of data status and versioning are an implicit part of the required CITIS capabilities in the full-scale development phase.

Role of CITIS in the Production Phase

At this stage of the weapon system life cycle, the IWSDB is relatively complete and is in place to support delivery of the weapon system. A key event is delivery of the weapon system to the government and the need for configuration management by version (tail number in the case of aircraft or hull number in the case of ships) becomes important. Once delivery occurs and the ship or aircraft enters the operation support phase, the program's CITIS must be capable of supporting a larger population of users.

Historically, most of the data associated with a weapon system were (and in fact still are) delivered in various hardcopy formats. Although required in all phases, the on-line government access capability requires the acquisition manager to make some new choices concerning the data processing categories required to support the weapon system during the operation and support phase.

Role of CITIS in the Operation and Support Phase

The configuration management aspect of an IWSDB takes on critical importance in the operation and support phase . The IWSDB must accommodate integration of old data, generated in the development phase, and new data resulting from field operations. CITIS capabilities must provide support for logistics functions, depot and field modifications, personnel training for system operation and maintenance, and field experience feedback requiring design modifications. In essence, the requirements for CITIS capabilities now include all the requirements of the previous phases plus the requirement to provide closed loop communications between the development contractors and depot and field support organizations.

The CITIS delivery system in this phase of operation must be flexible and capable of expansion as more and more units are produced and enter service in the field. World-wide gathering of field performance data and distribution of support data presents a significant technical challenge.

Data Sharing between Weapon Systems Programs

Although each major weapon system is unique, even at the unit level, a high-degree of commonalty exists between many weapon systems. Using the concepts of group technology to exploit this commonalty in the support function could produce significant benefits, as it has within many manufacturing operations. Once an IWSDB is established with well-documented standards and procedures for use of the data, sharing between individual IWSDBs is conceptually an attractive option. Assuming the adherence to compatible control architectures, the ability to use common CITIS capabilities and shared databases across weapon system programs becomes feasible from a technical viewpoint. The key to this data sharing is the development of a comprehensive data dictionary. The factors associated with data dictionary development are discussed in Section 6.

Data Requirements

An IWSDB and its associated CITIS address the technical data associated with a weapon system. Technical data includes all aspects of product definition including product configuration, shape/size characteristics, functional characteristics, physical properties, and operational characteristics. The product, in the case of an IWSDB, is a weapon system such as an armored tank, battleship, aircraft, or missile, including arms, munitions, and ground support equipment. Product definition is not limited to engineering design and analysis information; it also includes manufacturing and support specifications.

Although not generally considered technical data, product definition data is closely related to other types of information, such as cost estimates, program tasks and schedules, materials management information, and field operations data. Therefore, an IWSDB and its associated CITIS should also provide support for some aspects of acquisition management and field operations management. The primary focus must be on data that is shared between contractors and/or government organizations.

A common set of basic technical data can be used to generate a variety of information reports or screens for functional users. These reports or screens are referred to as information products. For the most part, these information products are currently being generated by today's application systems with some manual intervention. Ultimately, these information products should be generated on demand as part of the CITIS. Furthermore, by providing an integrated database of all technical information, new information products, such as required by concurrent engineering, may be generated. Figure 4-6 provides a list of information products by functional area as identified by the Digital Information Exchange Task Group of the CALS Task Force.

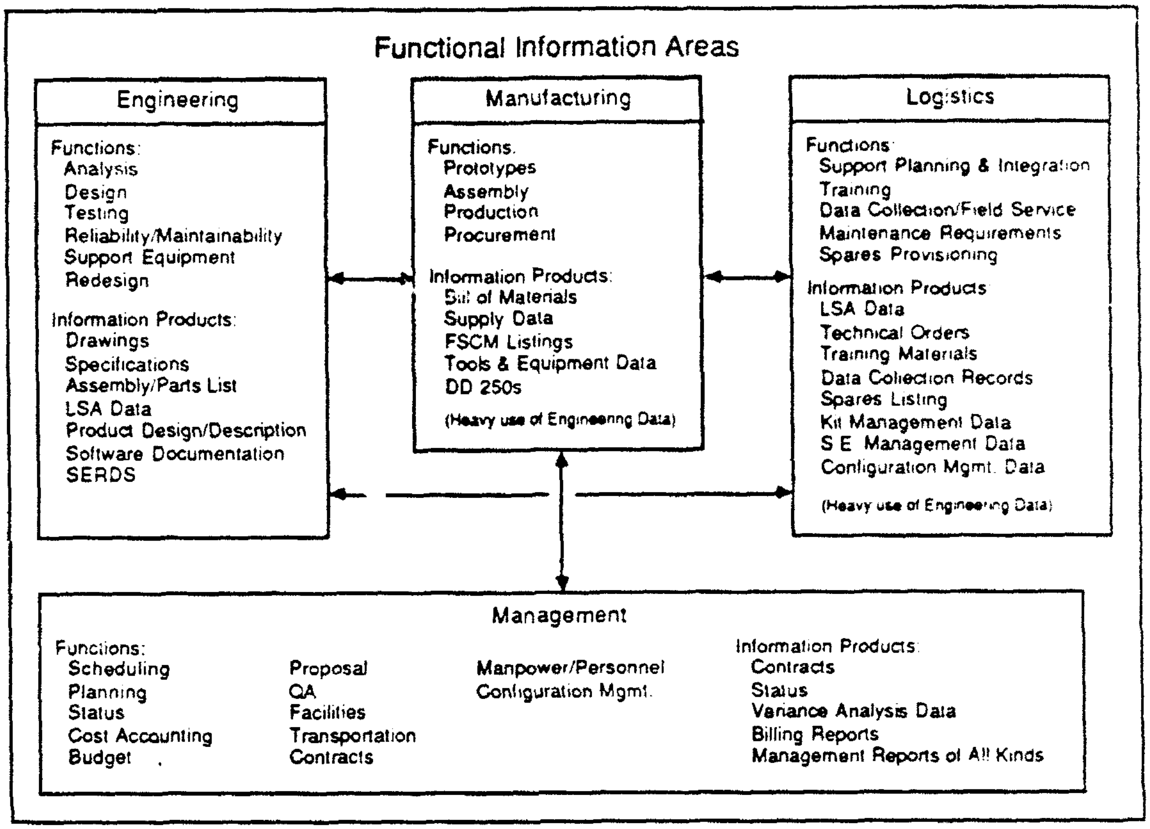


Figure 4-6 Information Products Requirements

4.2 Control Architecture Requirements

The Control Architecture is intended to formally and rigorously define a CALS Phase II environment that is integrated across all life cycle functional requirements and independent of any specific implementation technology. Establishment of a common Control Architecture as a reference model for establishment of an Integrated Weapon System Database and uniform Con ractor Integrated Technical Information Service for each individual weapon system program is critical to achieving CALS Phase II objectives. The Control Architecture must allow contractors to retrofit their existing technical data systems to the greatest extent possible, while establishing an integrated environment with substantial benefits and maintaining a clear migration path to exploit rapidly emerging computing technology.

The dependence on and enormous financial and emotional investment in legacy systems has been sighted by numerous contractors as the number one inhibitor to the establishment of an integrated environment, even internally to a single organization. Legacy systems are also a dominant constraining factor in the government's ability to receive and use digital technical data.

Neither the government nor individual contractors can afford the full-scale replacement of existing systems. Therefore, the development strategy to establish a CITIS delivery system must be evolutionary. Interfaces must be established and maintained with existing systems, but as systems are changed or replaced, the new developments should conform to the integration architecture.

Function Standards

From a conceptual integration viewpoint, a CALS Phase II environment should provide a logically integrated database, which will directly support a comprehensive set of advanced life cycle support applications. This concept is illustrated in Figure 4-7. The key objective is to integrate the functional processes that support the weapon system life cycle through the sharing of technical data, both within individual contractor organizations and between various contractors and government organizations. The incorporation of logistics support functions as an integral element in a contractor team's design process is a critical requirement for the overall CALS Program. Ultimately, a weapon system's CITIS should have the capability to support all elements of concurrent engineering, allowing parallel analysis of design, manufacturing, and support characteristics, as well as identification of schedule, cost, and quality drivers, by multiple contractor team members and government organizations.

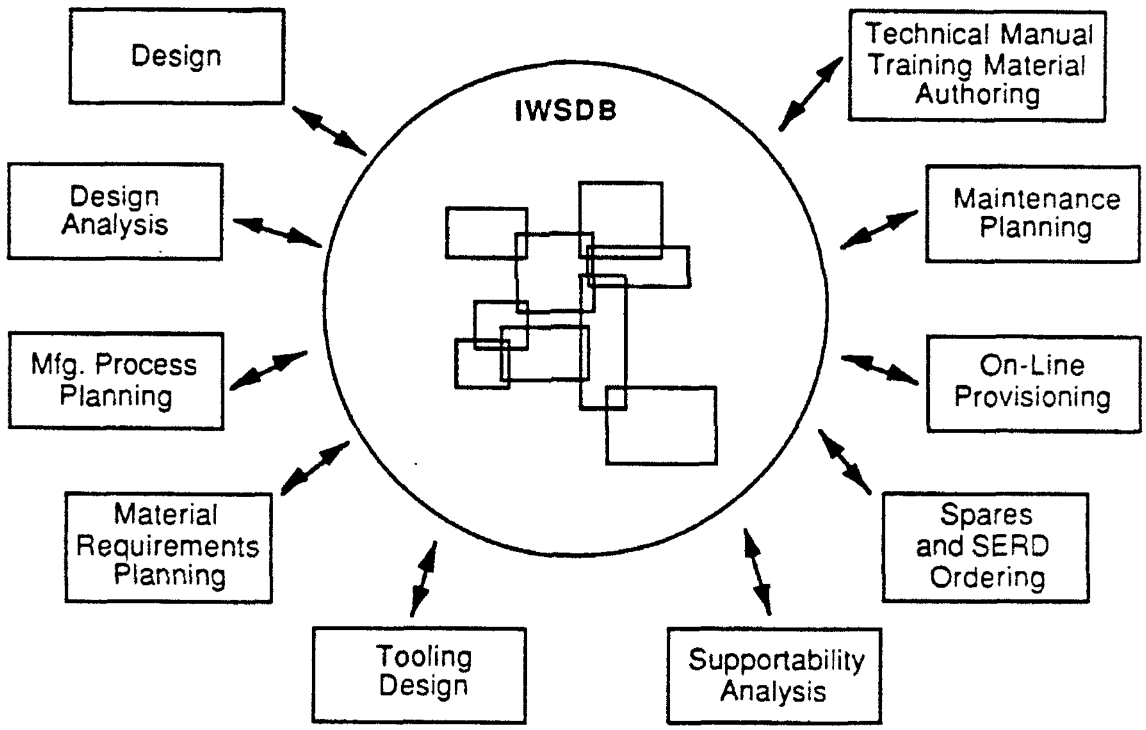


Figure 4-7 Integrated Functional Processes

The requirement for improved automation can be viewed somewhat independently from the requirement for process integration. However, functional processes that depend on human interpretation and processing of data are more difficult to integrate, since the interpretations are generally private rather than shared. The establishment of a shared comprehensive and robust definition of weapon system characteristics that minimizes the need for human interpretation will not only allow for improved automation of individual functions, but will also facilitate the integration of functional processes.

The objective of creating data once and using it many times implies that no single user or application system has complete responsibility for the shared data. Therefore, functional responsibility must be established for the management and control of shared data. At least one major defense contractor is in the processes of establishing a totally independent functional organization for the maintenance of product definition data. The function of data administration has been recognized for some time to support business applications and is now beginning to receive recognition for support of engineering systems.

The basic functions that any information system performs include capture, storage, retrieval, processing, and distribution of data. Stand-alone applications perform all these functions on their own. A shared database management system provides storage, retrieval and sometimes distribution as a common service to all applications, thus avoiding the need to create and maintain duplicate data, since a single system utility maintains the data and provides it in the desired form to the applications that need it.

Logical processing functions are also often redundantly implemented across various application systems. The definition and use of standard logical processes can also be implemented as part of a shared system utility that supports multiple functional applications. Not only are new functional processes easier to automate with such a utility, but data integrity is also improved since a single consistent interpretation is used. This capability is already in wide use for lower level system functions, such as standard scientific subroutine libraries, and is consistent with the emerging concepts for object-oriented programming and rule-based inference engines.

Shared logical processing functions can also be applied at a higher level as a basic CITIS capability. The CAM-I Geometric Modeling Program, for example, has developed a standard set of commands for application access to a geometric modeling system. Through the standardized interface a shared geometric modeler serves as a specialized inference engine for multiple functional application systems, such as generative process planning, which require part geometry information. A similar concept has been demonstrated by the USAF PDDI and GMAP Programs. The technology from these programs is being used on a limited basis by a few defense contractors.

The establishment of reusable functions that support the total weapon system life cycle, as well as the maintenance of shared reusable technical data, is a principal objective of the Functional Standards of the CALS Phase II Control Architecture.

Data Standards

An integrated view of the information topics that should be formally defined by Data Standards contained in the CALS Data Dictionary is shown in Figure 4-7. The complete set of Data Standards, defined according to a formal semantic modeling language such as IDEF1X, will likely contain several thousand interrelated entities and associated attributes. As a practical matter, multiple dictionaries will most likely be developed and maintained in parallel. However, achievement of a totally integrated environment will require that each data element (attribute) be unique and non-redundant. The interrelationships between entities must also be non-contradicting and logically consistent with the established overall practice for weapon system management. A hierarchy of data dictionary control structure that provides consistency across multiple data dictionaries is discussed in Section 5. Each weapon system program may elect to use only a portion the Data Standards contained in the CALS Data Dictionary in order to control its IWSDB. Each contractor will likely map the standard CALS data definitions selected for the IWSDB to his own internal data standards. This mapping would provide the necessary roadmap for internal integration, as well as external integration, via

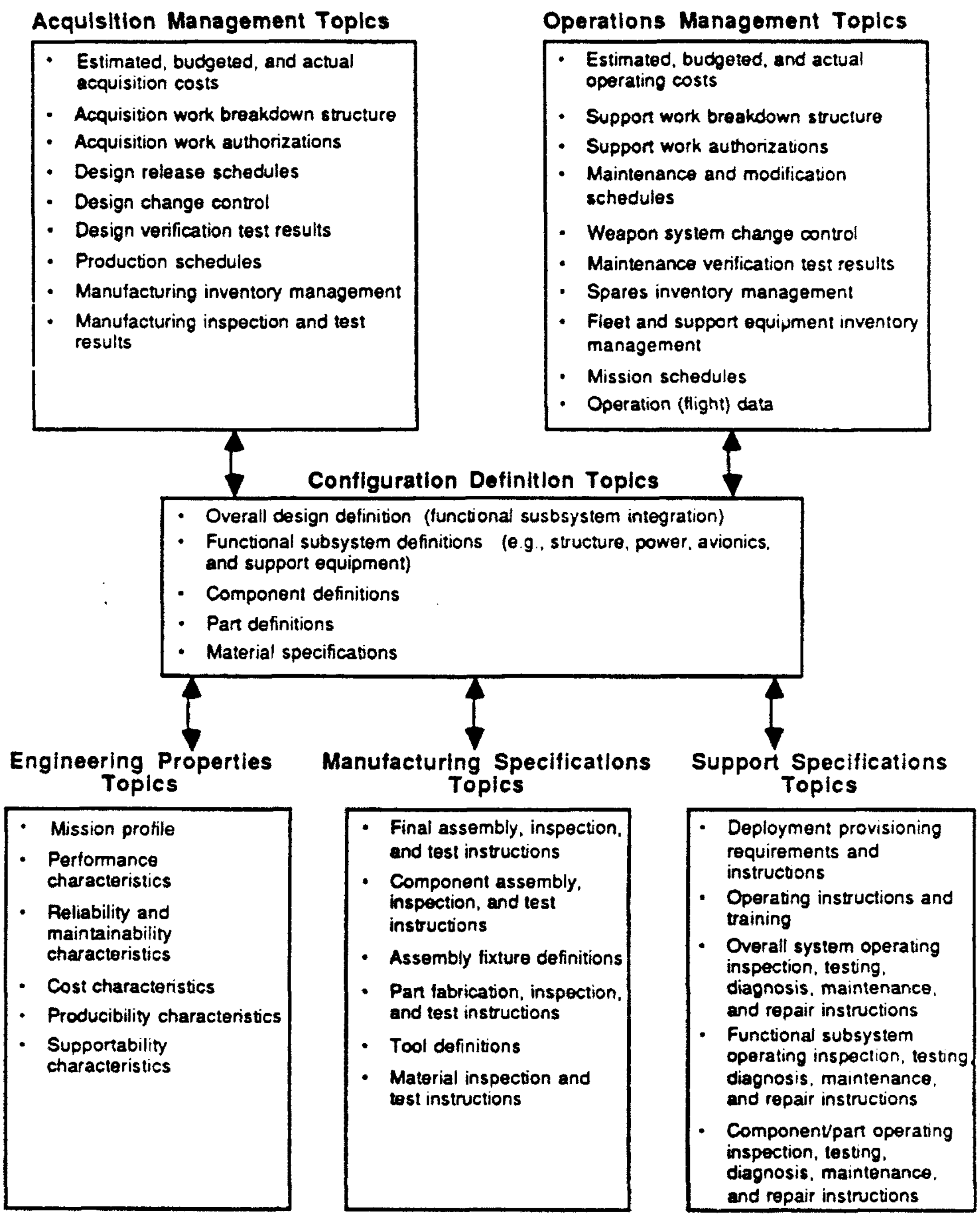


Figure 4-8 Integrated CALS Data Standards

CITIS capabilities. A contractor's internal data standards, for example, would generally include business and management data types not addressed by the CALS Data Standards.

The overall scope of the required technical data is set by the External Architecture data requirements. In order to support advanced life cycle applications and to facilitate integration by eliminating human interpretations of weapon system characteristics, the target CALS Data Standards must provide a robust representation of weapon system characteristics. Many existing technical data management systems manage technical data in files that are treated as undefined "bit buckets" which must be interpreted by a specific application. Furthermore, part characteristics are often only defined by raster images or two dimensional drawing files that can be computer displayed, but require human interpretation. Several major defense contractors have already committed to the establishment of full three-dimensional part models that support computer interpretation of tolerances, features, and other characteristics. Unfortunately, the most robust schemes for representation of part characteristics are often proprietary to a specific CAD/CAM system vendor. In order to achieve the desired level of integration in a distributed heterogeneous multi-organizational environment, the product definition representation scheme must be fully shared and accessible by all users. The long-term goal of the PDES project is to identify a standard robust definition of product characteristics. This requires state-of-the-art advancements in non-proprietary digital representation schemes in parallel with standardization efforts - a non-trivial challenge indeed.

The set of data definitions associated with the physical definition of the weapon system itself is central to the integrated structure. This definition set must support all aspects of configuration management and eliminate the requirement for separate as-designed, as-built, and as-maintained data structures. Several defense contractors have adopted a mono-detail approach to configuration management, which allows multiple views of the configuration to be generated from the same data structure.

No single geometric representation has proven to be adequate for all applications. Therefore, a scheme to interrelate multiple representation models for a single part is being investigated by the PDES project. The ability to adequately model part features and assembly structures in three-dimensional systems is still emerging. Furthermore, representation of functional characteristics is available on a limited basis, but requires further development.

The multi-layer weapon system definition is further extended by engineering properties including reliability and supportability characteristics, manufacturing specifications including the definition of tooling and fabrication and assembly processes, and support specifications including the definition of operating, maintenance, and repair processes.

Weapon system definition data is also interrelated with acquisition and operations management information domains, which include life cycle tasks, schedules, cost, change control, and inventory management of work-in-process, field units, and spares.

The potential role of emerging PDES and LSAR (MIL-STD 1388-2B) data definitions is discussed in Section 5.

Technical Standards

The conceptual network description allocates data generation, management, access, and distribution responsibilities to specific functional groups that participate in the weapon system life cycle. Figure 4-9 illustrates the interrelationship between functions, data types, and user groups. The basic decision on how the system will be distributed is defined by the External Architecture.

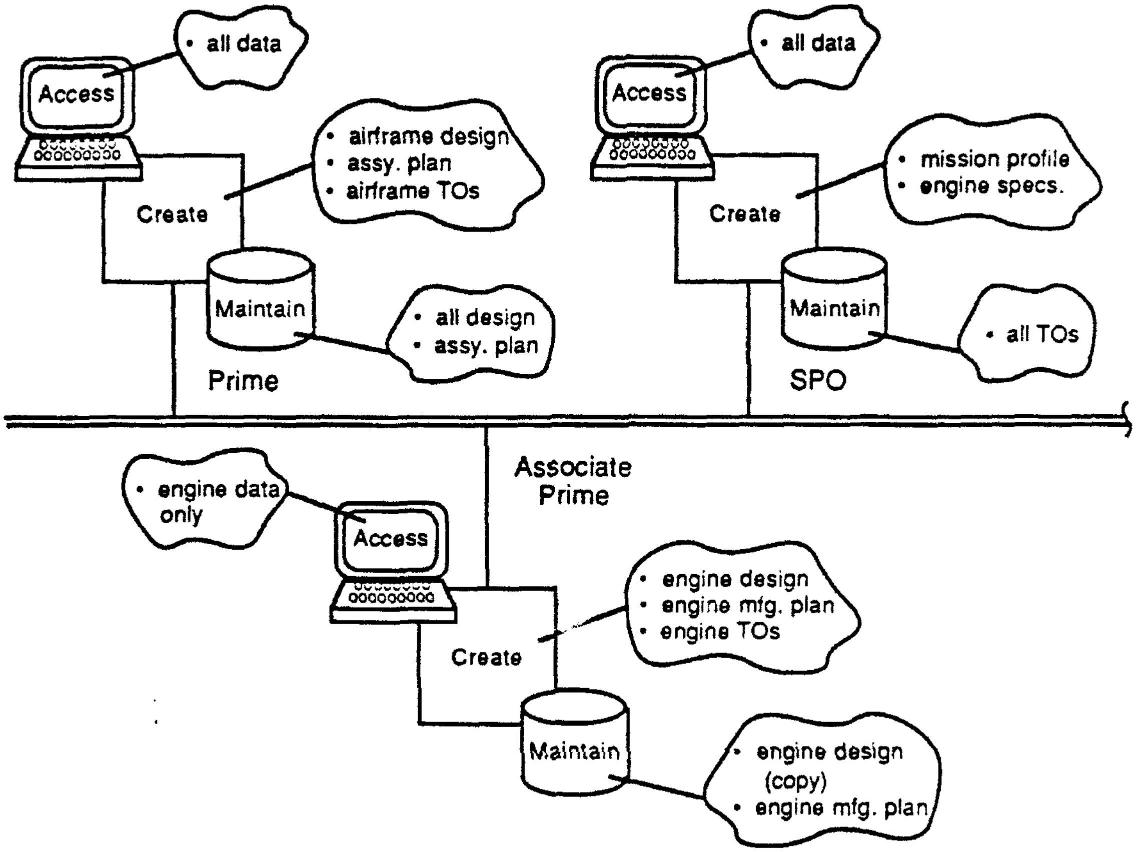


Figure 4-9 IWSDB Data Distribution Strategy

Data management responsibilities may be partitioned by subsystem or assembly, as well as by data type and function. One subcontractor, for example, may be responsible for maintaining landing gear design and manufacturing data, while another subcontractor has read-only access to the landing gear tire specifications.

The actual network distribution strategy will probably vary from weapon system to weapon system. However, basic roles and responsibilities should be fully discussed as a part of the Preliminary CALS Phase II Architecture review. Prime Contractors generally see their role as having centralized control over all technical data since they are responsible for the total weapon system. Subcontractors, however, generally maintain their own technical data and provide the data to the prime. A subcontractor may use common technology and even common parts to satisfy the requirements of several primes, in which case the subcontractor could be better able to manage the technical data. An overall trade-off study is needed to establish the basic decision criteria.

4.3 Internal Architecture Requirements

The Internal Architecture is technology constrained, which means the actual description depends on the computing technology to be used. A wide variety of computing technology and system designs may be used to implement CITIS capabilities. However, the delivery system must adhere to the established Control Architecture definitions and employ the proper set of CALS Technical Standards. Two important areas of standardization for which CALS preferences have not been identified are Database Management Systems (DBMS) and Data Dictionaries. Near-term implementations of a CITIS delivery system will most likely require standardization on a single database access language since available distributed heterogeneous database management capabilities are limited. These issues are discussed further in Section 5.

A simplified view of the ideal CITIS delivery system architecture required to support CALS Phase II IWSDB concepts from a technology implementation design viewpoint is presented in Figure 4-10. Basically the strategy is to provide a multi-layer open system architecture that allows for extensibility at any layer.

At the bottom layer, users of various disciplines within various organizations must have parallel access to the system on a national, if not world-wide, basis. System access is limited by the user's role in the life cycle support of the weapon system. Old users can be removed and new users added at any time.

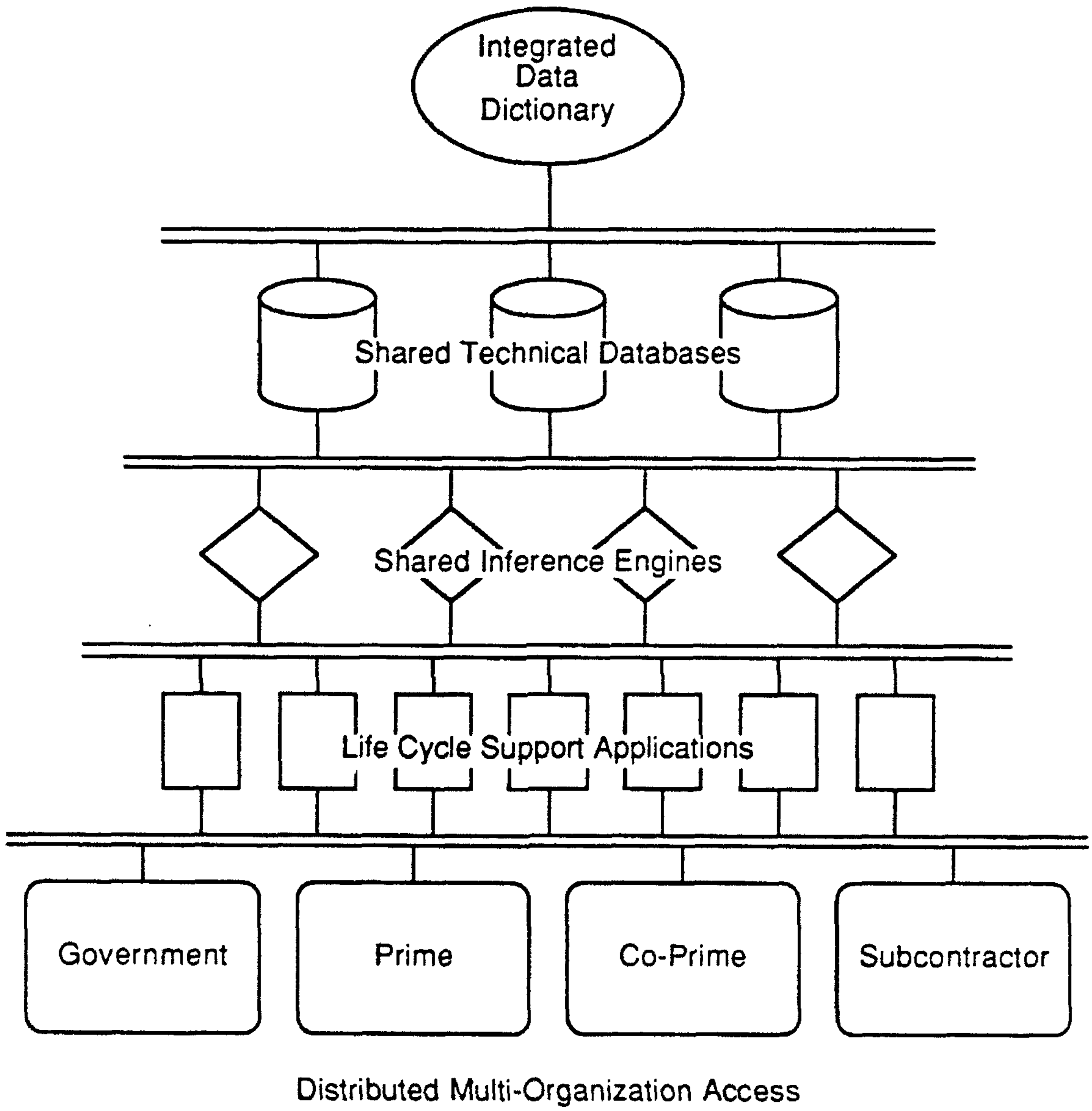


Figure 4-10 Ideal CITIS Delivery System Architecture

A user's information requirements are satisfied by one or more life cycle support applications defined at the next layer. Applications can be modified and new applications added without disruption to other applications.

Various life cycle applications may rely upon common tools for information analysis or inferencing, such as a three-dimensional solid modeler, finite element analysis processor, wind tunnel simulator, etc. Many of these common "inferencing engines" are used internally by defense contractors, and in some cases, defacto standards exist for using the tools, such as NASTRAN for Finite Element Analysis. Ultimately, CALS Functional Standards may be specified at this level, such that common tools may be used across many applications by multiple contractor team members and government agencies.

The applications are integrated through distributed but shared databases defined at the next layer. New shared databases may be added without disrupting existing databases or applications. The shared databases must be developed and maintained in accordance with the data integrity rules defined by an integrated data dictionary. The data dictionary definitions are defined at the semantic level and fully normalized to allow for extensibility. The data dictionary logically defines the IWSDB for specific weapon system programs.

The strategy for data and function distribution defined by the Network View of the Control Architecture provides guidance for a more detailed network specification as illustrated by Figure 4-11. Although the CALS Control Architecture will not define a network structure to this level of detail, the Control Architecture must identify Technical Standards that facilitate the interconnection of government organizations, prime contractor, associate prime contractors, subcontractors, and eventually even suppliers. The CALS Telecommunication Plan identified an overall strategy for applying CALS Technical Standards for networking between organizations.

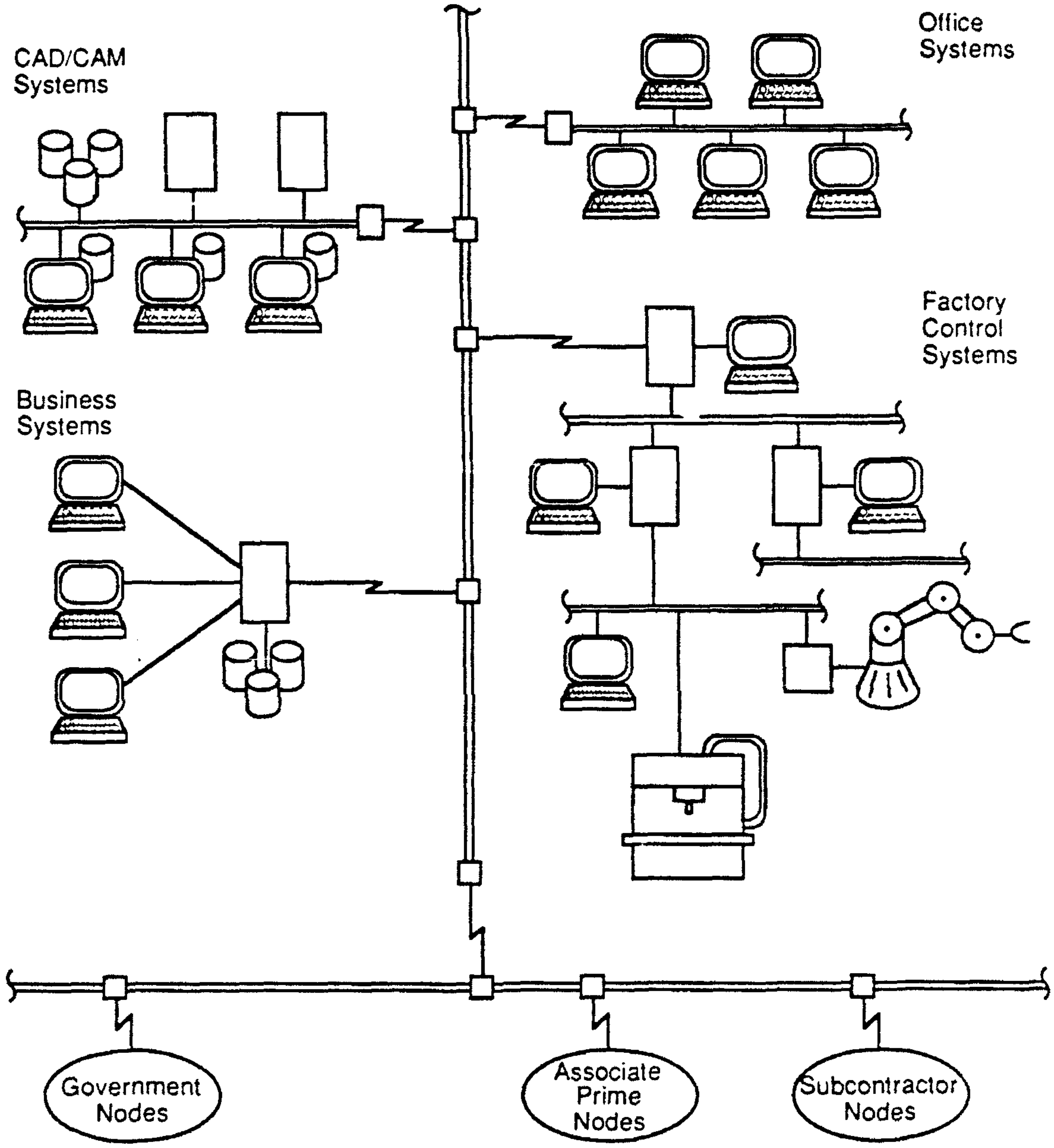


Figure 4-11 CALS Phase II Networking Requirements

The data dictionary of the IWSDB must be generally available throughout the network and may itself require a distributed data dictionary system. Ultimately, the IWSDB data dictionary system should support intelligent gateway capabilities as described in the CALS Telecommunications Plan.

A series of subnetworks will often need to be interconnected within a contractor node in order to provide direct accessibility to weapon system technical data. These subnetworks generally fit within four broad categories: office systems, CAD/CAM systems, business systems, and factory control systems. Each category of subnetwork may have its own technical requirements that must be considered in establishing the Technical Standards for CITIS.

Section 5.
Architecture Development Strategies

5.1 Strategy for Data Dictionary Development

The CALS Phase II Architecture is focused on the establishment of an integrated weapon system life cycle database that supports the technical information needs of both government organizations and the weapon system contractor team. In order to establish a shared weapon system database, a common view of product definition data must be established. As discussed in the CALS Handbook, "One of the most difficult missions of the CALS (Phase II) program is to review each item of data required by DoD or federal acquisition policy or by a government functional specialist and then determine areas of duplication, overlap, functional equivalency, and ultimately, unique requirements. Only then will it be possible to intelligently discuss the file structures and database schemas that will permit CALS to function. Simply automating the existing reams of custom reports and other deliverables may improve delivery schedules, but will provide little long-term gain."

The problem of non-integrated data is further compounded by the loss of information resulting from inadequate representation of product characteristics. Studies of the use of the engineering drawing, which has been the principal means of communicating design intent for decades, have revealed that engineering drawings are often ambiguous and sometimes self-contradicting. Early attempts to automate product definition, still in wide use today, resulted in digitized drawings, using either raster scanning or vector graphics technology. In these environments, human interpretation of the drawing is still required in order to determine the characteristics of a part.

The need to functionally replace engineering drawings with digital models of product characteristics that support automated design interpretation and drive advanced CAD/CAM and other product life cycle support applications has been recognized since the introduction of CIM concepts in the late 1970s. Most leading CAD/CAM vendors use three-dimensional geometric modeling techniques to provide automated interpretation for advanced applications. Unfortunately, many of these CAD/CAM systems are incompatible with one another and are built around closed system architectures. As a result, the exchange of technical data describing product characteristics must often resort to the lowest common denominator between the systems, human-oriented graphical representations, rather than exchange computer-interpretable digital part models.

In addition to improved integration and robustness of product definition data, there is a need to expand the recognized scope of product definition data. Complete product definition requires more than merely geometric coordinates describing product shape. Product definition data also include configuration characteristics, functional characteristics, physical characteristics relating to materials and manufacturing processes, and operational characteristics relating to reliability and maintainability. To provide comprehensive and consistent life cycle information, the basic product definition data must be interrelated with task definitions for engineering, manufacturing, and logistics activities that control the cost, schedule, and quality of the product. Complete life cycle support also requires the basic product definition information to be interrelated to information about the physical inventory of weapon systems, component parts, and materials both in production and field operations. The scope of the required contents of an IWSDB is defined not only by the type of data needed but also by the level of definition. The required product definition data and its related information must be able to describe the weapon system at the total system level, the functional subsystem level, the physical assembly level, and the detail piece part level. Although current logistics support data standards often address the system and subsystem level descriptions, most of the work on product definition standards has focused on piece part descriptions.

Potential Role of PDES in a CALS Phase II

The Product Data Exchange Specification (PDES), being jointly developed by volunteer industrial organizations under the coordination of the National Institute for Standards and Technology and the PDES, Inc. cooperative R&D organization, is recognized as a critical element of the CALS Phase II objectives and IWSDB implementation strategies. The establishment of the first version of PDES, submitted to ISO in December 1988, has proven to be a difficult and tedious task. The difficulty can be attributed to the requirement to both extend the formal representation of product characteristics and to obtain industry consensus on the data specification.

The PDES project mission from a CALS perspective should be to establish an industry-accepted definition of the data that constitute a complete product definition. This definition must support the sharing of data throughout the product life cycle, provide for automated interpretation of product characteristics, and allow for efficient implementation in a variety of computing environments.

Although the publishing of PDES Version 1.0 is only the beginning of a long-term goal, implementations of PDES Version 1.0 will likely provide little or no functional improvement over current implementations of IGES and internally-developed neutral formats. This is due to the limited scope of PDES Version 1.0 and the lack of a comprehensive validation and testing plan for shared database implementations of PDES versus the traditional file exchange implementations. PDES Version 1.0 primarily addresses geometric definition of detail parts. Although configuration definition and some functional characteristics are being developed, considerable work is still required. Furthermore, current PDES development efforts have not addressed system, subsystem, and assembly-level definitions, except for some preliminary work by the Architectural, Engineering, and Construction Committee. A top-down development strategy based on the overall requirements and improvement objectives of CALS Phase II is needed to help structure and prioritize current and future PDES-related development efforts.

Weapon system requirements may well be different from the general requirements of the current PDES community.

Potential Role of MIL-STD 1388-2B in CALS Phase II

The two primary components of an Integrated Weapon System Database, as identified in the 1988 CALS Report to Congress, are product data and support data. The evolutionary path for standard definition of support data began with the definition of procedures to provide on-line access to existing contractor Logistic Support Analysis Record (LSAR) databases. Standard inputs and reports from these databases are defined by MIL-STD 1388-2A, which is oriented toward sequential record management systems. However, a number of contractors have implemented the standard using traditional hierarchical database management systems. This standard is currently being updated for relational database implementations by MIL-STD 1388-2B.

Although relational implementation of LSAR data will improve the accessibility and flexibility of the database, a more important objective is to understand and define the conceptual information structure independent of any particular implementation strategy. A number of defense contractors have developed their own semantic data models for LSAR and are actively participating in the development of an IDEF1X model of MIL-STD 1388-2B. The recognition of the need for an integrated semantic data model in order to properly define a LSAR database is similar to the evolution from IGES to the current PDES development approach.

A Logistics Support Committee has been established for PDES development activities. The scope of the data addressed by this committee will include the data identified in MIL-STD 1388-2B, as well as additional data required to provide integrated logistic support. Although the scope of this committee's activities appears to align with the overall requirements for a standard CALS data architecture, the committee work is still in early development and has not yet been integrated with other PDES development activities.

Evolutionary Data Definition Strategy

Despite the enormous efforts that have already been put forth to standardize data definitions for product definition and support, a significant multi-year development effort still lies ahead. However, the size of the task should not discourage attacking the problem head-on, given the fact that millions of man-hours and dollars will be spent by DoD, contractors, and technology vendors to improve technical information systems over the next several years as the normal course of business. What is needed is an overall strategy that will organize these development efforts such that data standards evolve from the natural process of improving or replacing existing systems. Agreement on the overall architecture of a CALS Phase II could provide the necessary framework and catalyst to structure future development activities. The current development efforts of PDES lack an overall business perspective to guide the development of technical solutions. The proper business perspective should be provided by the CALS initiative.

Although the near-term development of a single integrated CALS Data Dictionary seems extremely difficult, if not impossible, without common data definitions across all life cycle functions, integration through shared data access will not be achieved. To be practical, the total CALS Data Dictionary must be evolved a chunk at a time. Focusing on an individual functional area, such as LSA, however, will result in redundancy and inconsistency with other functional areas with common data requirements. Therefore, a layered approach to constructing the CALS Data Dictionary, as illustrated in Figure 5-1, is recommended. A CALS Kernel Data Dictionary which standardizes on the data defining a weapon system's configuration should be established first and used as the starting basis of all other data dictionary developments. The second layer of data dictionaries should standardize on the data defining basic weapon characteristics which are used in more than one functional application area, such as shape/geometry definition, material specifications, functional characteristics, and operational characteristics. Finally, the last layer of dictionary definitions would standardize on data associated with a particular functional area, such as spares reprocurement. The actual data dictionary used to support a functional area would include the kernel definition, required shared characteristics data definitions, plus function-specific data definitions. Any changes to the kernel or shared characteristics data definitions would be validated across all functional areas. Standardization efforts operating in parallel could be established for each component of each layer.

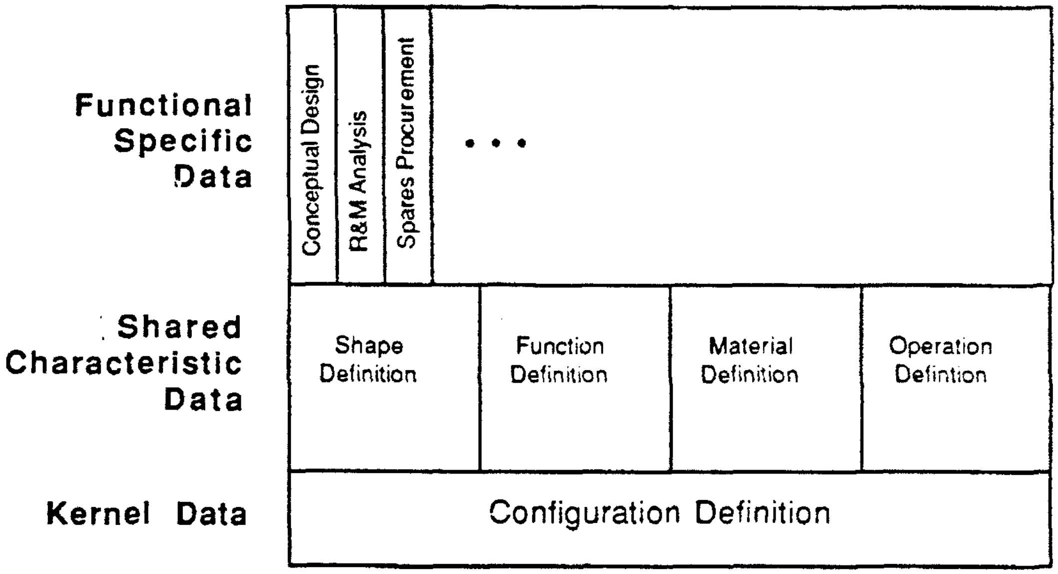


Figure 5-1 Multi-Layer CALS Data Dictionary

5.2 Applications Development and Legacy Integration Strategy

The CALS target for the the 1990s is for digital data interchange and database access to support a wide range of design, manufacturing, and support applications within both DoD and industry. Cited examples include:

- Computer-aided Design
- Design Analysis
- Manufacturing Process Planning

- Tool Design
- Materials Requirements Planning
- Flexible Manufacturing Automation
- Supportability Analysis
- Maintenance Planning
- Technical Manual/Training Material Authoring
- Paperless Maintenance Aids
- On-line Provisioning
- Automated Spares Procurement/Reprocurement

The CALS approach is to impose interface and access standards, but to leave development of the applications to the users. Considerable investment has already been made in existing application software that must continue to be supported even with the establishment of an IWSDB supported by a CITIS. Any information required by a specific application, which is not available directly from the IWSDB, must be provided by a human user or some other external source. If the same additional information is required by more than one application system, the possibility for redundancy and loss of integrity is introduced. Therefore, IWSDB extensions should lead application development. For example, if part tolerance is required by both a standard generative process planning system and by an automatic maintenance inspection program, tolerance data should be added to the IWSDB and made available to both.

Generally speaking, it is more feasible to build a standard interface from a more robust database structure to a less robust application interface. A two-dimensional part definition required for a part sketch can be generated from a three-dimensional part definition stored in the database. The reverse interface from a two-dimensional database to a three-dimensional interface is obviously considerably more difficult. Maintaining a more robust database definition, however, creates an additional burden for data capture and database maintenance.

An overall strategy for database development and application interface support must be carefully thought out in the process of developing a detailed CITIS delivery system architecture. Several contractors already have aggressive strategies to maintain complete three-dimensional databases of product definition. In addition, the work of cooperative development programs such as IDS and GMAP should help establish a state-of-the-art baseline for product definition and application interfaces.

5.3 Inter-organization Networking Strategy

A nation-wide and, ultimately, world-wide digital communications linkage must be established that supports the movement of massive amounts of technical data between heterogeneous computing systems to fully implement on-line access to contractor-managed weapon system databases. The CALS Telecommunications Plan identifies and discusses many of the issues associated with this objective, including establishment of communication protocol standards, network configuration management, access control, and data security.

In addition to standardizing the application of communications technology, an inter-organizational networking strategy must also standardize data management services throughout the network. The long-term objective identified by the CALS Telecommunications Plan is to provide an Intelligent Gateway that integrates current data management and communications capabilities. The Department of Energy is currently working with Lawrence Livermore Laboratories to demonstrate such capabilities.

The required services of an Intelligent Gateway have been identified to include the following:

- User/application interface services provide for the specification of requests for integrated data, as well as the presentation of results. Though they are not a part of the gateway, they provide the requester with uniform access to the gateway by supporting a "global request language" that embodies the semantics of the applications.

- View management services keep track of the objects in the integrated view, including relationships among such objects and associated operations, in order to interpret requests for operations on objects that are accessible through the gateway. They include directory services to locate the referenced objects.

- Decomposition and routing services map requests specified in terms of objects and operations in the integrated view into a set of specific requests targeted at the underlying databases and access services, and route the resulting queries to specific target facilities.

- Invocation and execution control services provide facilities for initiating and tracking requests in the distributed, heterogeneous computing environment. In this they provide services similar to a distributed operating system. They execute the access plan produced by the decomposition and routing services and manage the submission of the individual queries and the return of responses.

5.4 Standards Development Requirements

A primary objective of CALS is to improve efficiency and effectiveness in the management of digital data. It is crucial that shared weapon system data be standardized; that is, that rules be established to govern just what the data are. Clearly, those rules must govern data definition. However, they must also govern data integrity and data consistency. In addition, the rules must be defined in such a way as to ensure that consistency and integrity are maintained as new data are added to support new activities. To accomplish this objective, data standards established for an IWSDB must define "data integrity rules."

Because technologies will undoubtedly change through time, it is critical that the data integrity rules be independent of technical standards used for implementation; that is, they cannot be linked to any specific storage or manipulation technology. However, data integrity rules must be transformable, algorithmically, into various technical representation structures for computer-based implementation. The data integrity rules, therefore, must be neutral.

Currently, some Data Standards are embedded in CALS Functional Standards. However, these Data Standards are limited to data element descriptions, and defined separately for individual Functional Standards, leading to redundancy and inconsistency in data definitions among Functional Standards. The CALS Control Architecture must ensure that Data Standards are consistent among the various Functional Standards. The most practical way to accomplish this objective is to maintain all Data Standards as a whole through the use of a common CALS Data Dictionary System. Unfortunately, most current data dictionary systems are focused on maintaining data definition in an implementation-based format.

Figures 5-2 and 5-3, presented by Boeing to an IDS TAG group, illustrate the evolving role of a data dictionary system. Initial usage may be limited to simply a glossary that is used by system developers. An active data dictionary system, however, may support development activities directly by generating data definition sections of applications programs and controlling the design of shared databases. At maturity, the data dictionary may be involved directly in the run-time processing of user requests by providing the basic capability of an intelligent gateway. The current IRDS standard only addresses the basic glossary functions of a data dictionary. Furthermore, without user extensions, the data definitions are oriented to implementation views of the data (internal schema). Additional capabilities will be required to support CALS integration. Most significantly, data definitions must be captured and maintained at the semantic data modeling level.

Development of rigorous definitions for the proper semantic interpretation of shared data requires the use of a formal semantic modeling language, such as the IDEF1X syntax developed for the USAF and widely used by a number of defense contractors. However, the modeling language is only a tool to define the definitions. Consensus must be reached on a common semantic interpretation by all parties wishing to share data. This requires a comprehensive strategy and well-established procedures for development of CALS Data Standards for the IWSDB. The dictionary system should serve as a common tool to help facilitate the development of Data Standards and as a common tool to use the Data Standard in implementing and using an IWSDB.

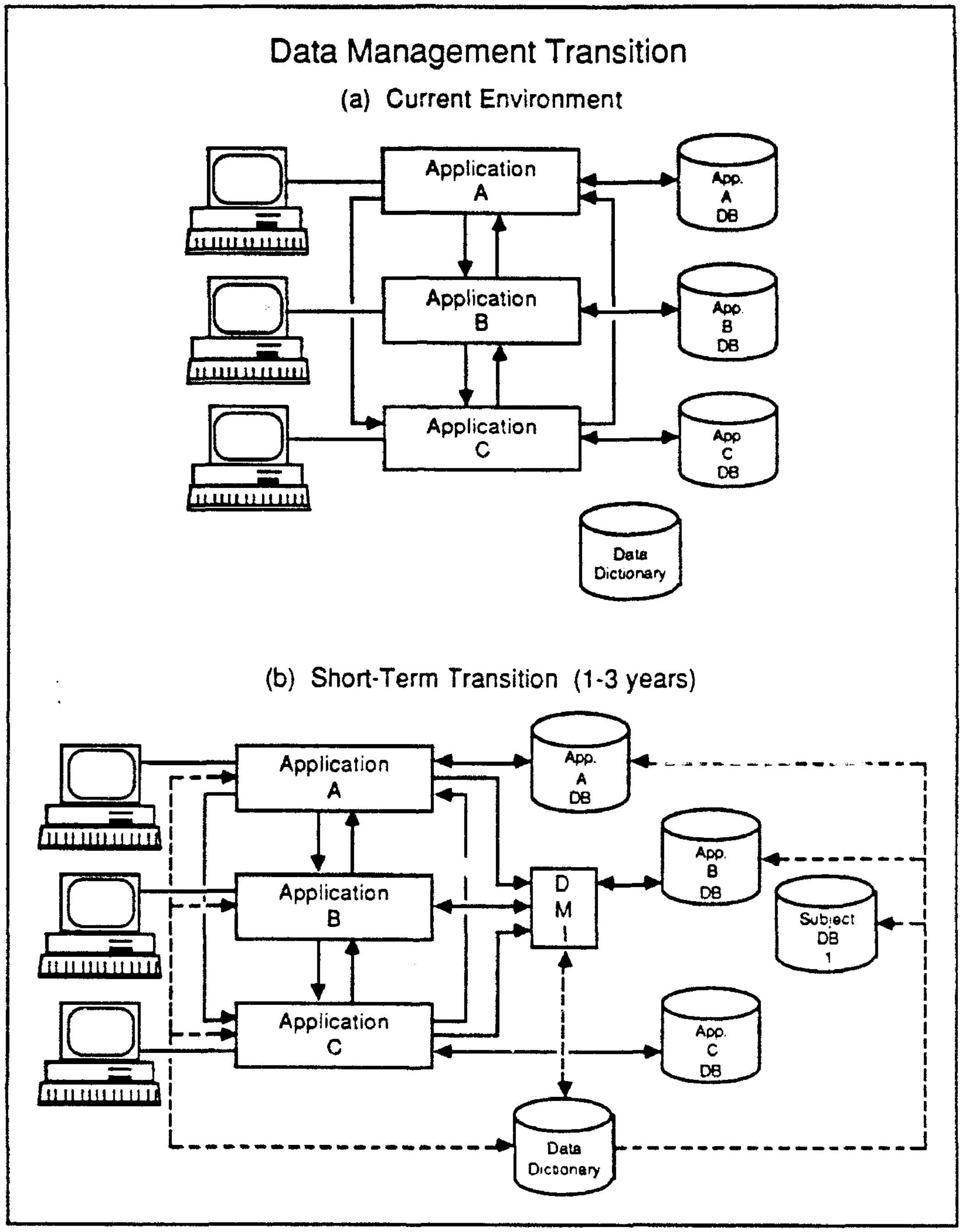


Figure 5-2

Source: Boeing IDS Presentation

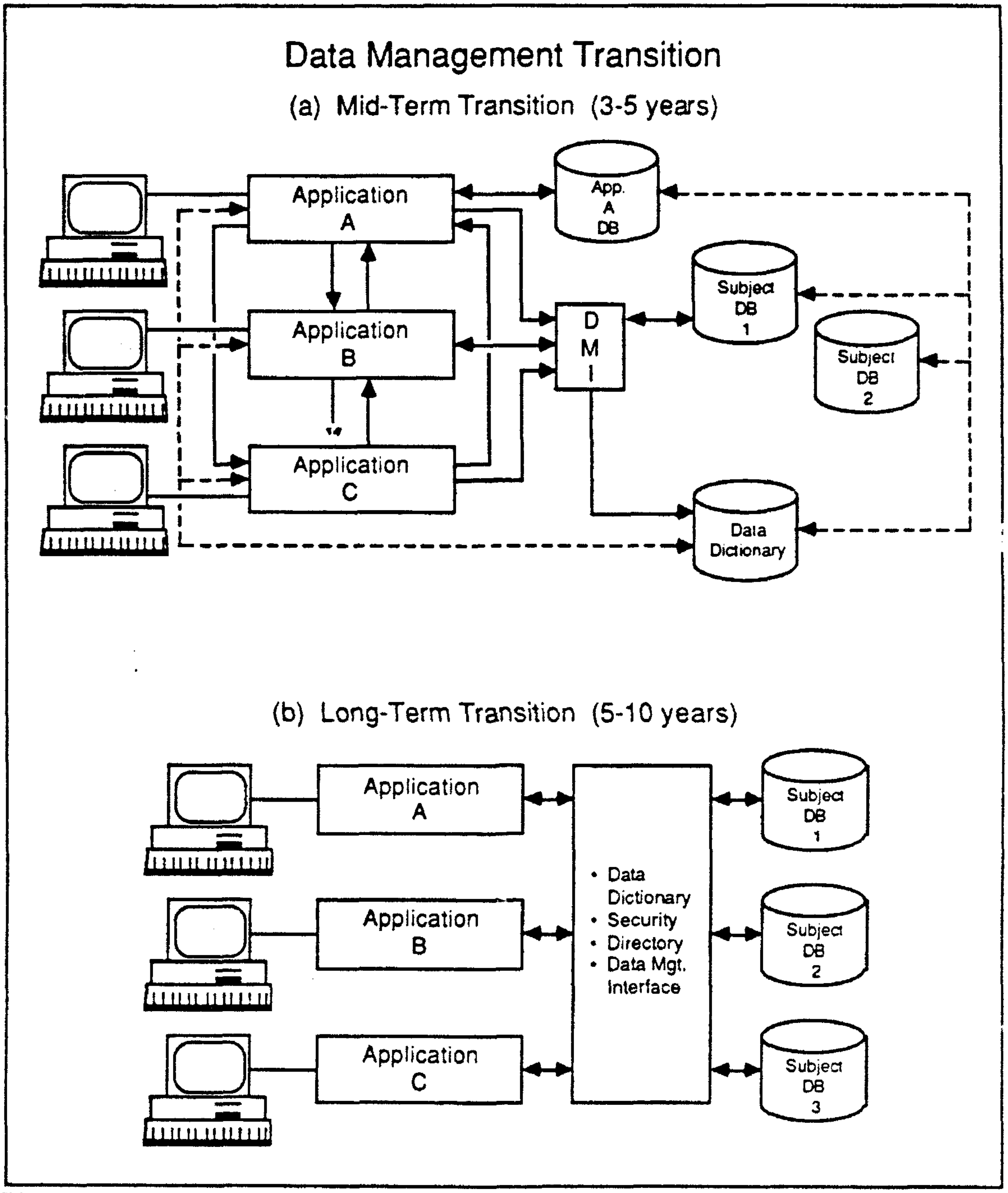


Figure 5-3
Source: Boeing IDS Presentation

Section 6.
Issues and Recommendations

Establishment of Contractor Integrated Technical Information Service and the resulting Integrated Weapon System Database are not unique CALS concepts. The basic concepts, as well as the technology to support them, are natural outcomes of fundamental trends in information technology, supporting even more fundamental trends toward electronically-based industry trading partnerships. The unique contribution of CALS Phase II is an overall architecture and set of standards that will be used to guide industry and government investments in automation to ensure their interoperability, to reduce costs attendant to information management (for both government and industry), and to increase the quality of information overall.

Even though the CALS Phase II Architecture is not based on revolutionary concepts, there are numerous barriers to effective implementation. Most of these barriers are cultural; however, it would be misrepresenting the state of integration technology to imply that there are no technical issues. The following section provides recommendations based on some of the key issues that are slowing the meaningful implementation of CALS Phase II. The recommendations offered are based on the following critical assumptions:

- That while CALS Phase I and CALS Phase II options are not mutually exclusive in that they share some common functional and technical standards, CALS Phase II defines a distinct set of services that DoD wishes to buy and industry wishes to provide.

- To achieve these mutual objectives, DoD and industry are willing, over time, to make significant adjustments to their operating relationships.

- That the technical capabilities required to establish a rudimentary, but nevertheless definitive, form of CALS Phase II exist today and will continue to evolve as the demand for CALS Phase II services increases.

- That specific actions, especially as related to existing and future standards, must be taken to definitize CALS Phase II, especially with regard to Data Standards and standards affecting data management technology.

- That standards actions spearheaded by the CALS Policy Office will generally be consistent with those currently being contemplated by industry.

The following recommendations have been divided into the categories of general recommendations and standards recommendations.

6.1 General Recommendations

Development of Final CALS Phase II Architecture

This document has been developed as a "Preliminary" Architecture. It is important that it be reviewed by appropriate personnel in the DoD and aerospace/defense industry, and that comments and recommendations be considered for incorporation into the document before its final publication.

Recommendation #1 - Within the month following submission of this report, the CALS Industry Task Group and the DoD CALS Steering Committee should select a few (no more than 10 each) specific individuals to review this Preliminary CALS Phase II Architecture document and provide substantive comments to the OSD Policy Office. Within three months following that review, the OSD Policy Office should complete modifications to this report, and, working with the contractor, define and publish the Final CALS Phase II Architecture.

Industry Technical Advisory Group for CALS Phase II

The CALS Industry Task Force (CITF) organization has constituted a number of "task groups," including Acquisition, Concurrent Engineering, Data Protection and Integrity, and Digital Information Exchange. Currently, all of these task groups with their subcommittees discuss various issues that have implications for CALS Phase II implementation; yet, these discussions are uncoordinated as regards CALS Phase II. It is important that the CALS Industry Task Force support the eventual functional specifications for CITIS procurement. The creation of a CALS Phase II Technical Advisory Group is an approach that would focus CITF attention on CALS Phase II.

Recommendation #2 - Immediately upon issuance of the Final CALS Phase II Architecture, the OSD Policy Office should formally request the CALS Industry Task Force to establish the CALS Phase II Technical Advisory Group. Within the first six months of its existence, this group should provide detailed recommendations for modifications to existing standards and for implementation strategies. It should also provide guidance to other formal CITF Task Groups.

Coordination with Computer Technology Vendors

Ultimately, the primary path toward implementation of CALS Phase II capabilities will come through commercial products that support CALS Phase II concepts and goals. It is important, therefore, that a means be developed for involving computer technology vendors in the development of CITIS and IWSDB specifications.

Recommendation #3 - It is recommended that, immediately upon publication of the Final CALS Phase II Architecture, the CALS Policy Office move to establish liaison with one or more organizations that can provide a forum for review and comment by technology vendors on the CALS Phase II documentation, after that documentation has been reviewed by industry and government.

6.2 Standards Recommendations

The CALS Control Architecture calls for three distinct types of standards: Functional Standards, Technical Standards, and Data Standards. Many of the existing standards could be placed into more than one of these categories; for example, many of today's Functional Standards provide data element descriptions, which could be classified as Data Standards. However, in the discussion below, there is an effort to classify existing standards into one and only one category based on the primary purpose of the standard.

Functional Standards

Review Data Element Descriptions

Appendix B contains a list of existing functional standards mapped against individual weapon system life cycle stages. The majority of these standards contain data element descriptions, and there are redundancies and inconsistencies in those data element descriptions across Functional Standards. These redundancies and inconsistencies create serious problems in reporting.

Recommendation #4 - Within the next six months, the CALS Policy Office should establish a schedule for the review of data element descriptions in existing CALS critical path standards, i.e., standards dealing directly with configuration, product, and support data deliverables. This schedule should be synchronized with any planned republishing of existing standards. Data element descriptions for each selected functional standard should be modeled, using the modeling technique and procedure recommended below, and coordinated with other functional data models by means of the CALS Data Dictionary.

Revise Standard for Developing Functional Standards

Though there are many existing Functional Standards, new standards that affect the CALS agenda are in various stages of preparation. It is important that these new Functional Standards be developed in accordance with CALS Phase II requirements as they affect data definitions.

Recommendation #5 - Within the next six months, the CALS Policy Office should propose modifications to the DoD standard for developing Functional Standards (MIL-STD-962A), particularly in the area of data element definitions. The proposed changes should be consistent with the CALS Phase II data management strategy.

Concurrent Engineering

Because of its focus on technical data in a production team environment, CALS Phase II has a lot in common with DoD activities focused on Concurrent Engineering.

Recommendation #6 - Within the first six months following publication of the Final CALS Phase II Architecture, the CALS Policy Office should establish formal relationships with the various Concurrent Engineering projects currently on-going in DoD. It should seek to influence these projects to be compatible with CALS Phase II strategies and objectives.

Technical Standards

The bulk of the Technical Standards that affect CALS Phase II are contained in MIL-STD-1840A. These standards must be reviewed to determine what, if any, modifications are required to support CALS Phase II. The primary area of new Technical Standards development to support CALS Phase II is data management. The following recommendations concentrate on that area.

Data Dictionary Systems

Data dictionary systems are a crucial enabler to CALS Phase II. The basic Information Resource Directory System (IRDS) standard is insufficient for the purposes of CALS Phase II. Significant extensions must be made to this standard to introduce several important dimensions, including capability to deal with distributed, heterogeneous databases; increased capabilities for dealing with technical data; introduction of the three schema architecture; and capabilities for developing, managing, and deploying a conceptual schema.

Recommendation #7 - Upon release of the Final CALS Phase II Architecture, the CALS Policy Office should initiate a detailed review of the IRDS standard. Based on that review, the Policy Office should request that NIST recommend changes to that standard necessary to bring it into compliance with CALS Phase II requirements. In addition, once the Data Dictionary System requirements have been defined, the Policy Office should initiate action to build a CALS Data Dictionary System that will be used to support construction and maintenance of the CALS Data Dictionary, as described below. The capability to import, export, and compare data dictionaries and to use the data dictionary actively in a three schema architecture is of paramount importance.

Distributed, Heterogeneous Database Management

Currently, distributed, heterogeneous database management is being handled by "home grown" data dictionary systems and intelligent gateways. These home grown systems are the linchpins of CALS Phase II. It is important that a standards effort be initiated to deal with this emerging problem. Such an effort should be consistent with plans for the IRDS standard, as well as with the distributed database management strategies being considered by SQL standards groups. At minimum, a distributed database management strategy should be based on standard SQL interfaces to existing DBMSs. To accomplish this, the SQL standard, itself, may have to be modified to be consistent with the data modeling technique employed by CALS (especially as it relates to referential integrity) and to provide additional SQL capabilities for handling technical data. The latter types of extensions may require that SQL be modified to handle object-oriented database representations. If the SQL standards groups will not deal with object-oriented technology, then a new standards activity, specifically focused on object oriented database management standards, may be required.

Recommendation #8 - Immediately upon approval of the Final CALS Phase II Architecture, the CALS Policy Office should request NIST to conduct a review of the various standards activities that have an effect on distributed, heterogeneous database management, specifically including the IRDS and SQL activities. NIST should be asked to make recommendations to the Policy Office as to appropriate actions that would bring these efforts into compliance with CALS Phase II requirements.

Security and Data Access

One of the most sensitive issues associated with CALS Phase II implementation is unauthorized access by the government or by trading partners to data that is resident in "private" ADP systems. Because of uncertainty and confusion about what the term "controlled access" actually means, there is concern that CALS Phase II will result in the government or competitors gaining unauthorized access to corporate proprietary data. This, of course, is not the intent or objective of CALS Phase II. There are examples of successful government on-line access to data housed in contractor databases. For example, the Air Force Logistics Command has on-line access to F-16 data maintained by General Dynamics and the Navy logistics infrastructure has on-line access to F-18 data maintained by Northrop.

There are several technical issues associated with protecting classified DoD data and the proprietary data of contractors. Recent widespread unauthorized access to nationwide computer networks by creative "hackers," as well as the emergence of "viruses," have exacerbated the problem. There are many technical dimensions to the security problem, such as personnel security clearances, facility clearance, storage, databases, etc. From the viewpoint of a CALS Phase II implementation, it is important to address the security and access issue forthrightly in the functional specification for implementation.

Recommendation #9 - The joint DoD/industry CALS security task group should participate in the development of security and data access specifications to be included in CALS Phase II Control Architecture.

Data Standards

Since the primary end game for CALS Phase II is on-line access to contractor maintained technical and support databases, the area of data management is of primary importance. The following recommendations relate to the development of a new class of standards, called Data Standards, that have not heretofore existed. This is not to say that Data Standards as defined below are not under development within most of the corporations that comprise the defense industrial supply base. Indeed, they are. However, the bulk of these standards - with the notable exception of the PDES standard - are being developed to support internal data communications, not communications with trading partners or with the DoD. One of the primary CALS objectives is to provide a set of Data Standards that can be employed by contractor teams for controlling those data that will be delivered to, or accessed by, elements of the DoD. The immediate focus of these Data Standards activities is on product configuration, design, and support data, which is a subset of the Integrated Weapon System Database. These CALS Data Standards will comprise the CALS Data Dictionary, and they will be managed by the CALS Data Dictionary System. They will be derived, for the most part, from modeling data element descriptions contained in current and future DoD Functional Standards. The first effort in this direction has already been undertaken in the rework of the data element descriptions contained in MIL-STD-1388-2B.

Validate the Preliminary CALS Phase II Architecture Data Categories

The IWSDB data categories contained in this Preliminary Architecture were derived from input developed under a project sponsored by the Digital Information Exchange Task Group of the CALS Industry Task Force. The project defined a broad range of data categories and associated information products that are generated at various stages of the weapon system life cycle. This taxonomy of data categories reflects some industry consensus but, by no means has it been generally accepted. For instance, perhaps the data category "transportation" should be treated as a subset of the "logistics" data category instead of the "management" data category. The important point is: an industry generated template exists that should be validated and which should eventually become the reference point for the CALS Data Dictionary.

Recommendation #10 - Upon issuance of the Final CALS Phase II Architecture, the CALS Policy Office should formally request that the Digital Information Exchange Task Group of the CALS Industry Task Force should be encouraged to take the list of data categories, revise it in accordance with IDEF1X (the Entity-Relationship level), and officially recommend the revised categories as a template for the CALS Data Dictionary. This template should be validated against long-range PDES development plans.

CALS Data Dictionary Management Procedure

A major objective of CALS Phase II is to establish a CALS Data Dictionary that will be employed by the DoD and the Aerospace/Defense industry for the development of IWSDBs. CALS has determined that it is of critical importance that work be initiated to build a CALS Data Dictionary that will contain the CALS Data Standards. Elements of this data dictionary will be promulgated through the CALS Handbook as well as MIL STD-1840A. Building and maintaining the CALS Data Dictionary is not a project; it is a process, that will be on-going for many years. The CALS Data Dictionary Management Procedure should define how functional data dictionaries, i.e., data dictionaries that support individual functional standards, will be developed - see the above recommendation - and how they will be validated by means of the CALS Data Dictionary. The procedure will also involve the use of the CALS Data Dictionary System.

Recommendation #11 - Upon release of the Final CALS Phase II Architecture, the CALS Policy Office should initiate action to develop the CALS Data Dictionary Management Procedure using the guidelines established in this report. This procedure should be a source of modifications to the standard for developing Functional Standards, and it should be a source of requirements for development of the CALS Data Dictionary System.

Preliminary

CALS Phase II Architecture

APPENDIX A

"A Framework for Information Systems Architecture"
John A. Zachman
IBM

IBM Los Angeles Scientific Center
Report No. G320-2785

A Framework

for

Information Systems Architecture

John A. Zachman
International Business Machines Corporation
355 South Grand Avenue
Los Angeles, California 90071

FOREWORD

Long-range planning for computing has been a subject of great interest for a long time. The MIS/DP executive has long-range decisions to make about database and communications architectures, hardware strategies, and application portfolio priorities. The emergence of the personal computer and decentralization adds organizational strategies (and politics) to the already complex technical issues.

Enterprise-wide Information Management (EwIM)[1] addresses the issues of long-range planning for computing. The premise of EwIM is that it is possible to effectively plan for the use of computing and information technology in business. Furthermore, it is possible to conduct planning that effectively links business planning with technology planning. The EwIM approach is to develop the underlying intellectual framework to link the planning concepts and components to produce a useful and coherent result. EwIM expects to produce a result that can be used to drive planning and produce effective planning processes for enterprises.

A key component of EwIM is information system architecture. Residing in the technology domain, it plays the crucial role in the alignment issues and ultimately in the impact issues. John Zachman has, for many years, been a prominent figure in the area of information systems architecture. John's early attempt to define the term architecture, has led him to develop a rather unique conceptualization of information systems architecture. Drawing from the field of classical architecture, he suggests that, in the course of constructing a building, there exists several levels of architectural representations each level having a different perspective. He further suggests that there exists analogous sets of architectural representations in the course of building any complex engineering product, including an information system. John has presented his work at many conferences held by the information systems community including the first EwIM workshop. It has received wide acceptance. This document describes his work.

M. M. Parker
Los Angeles Scientific Center

March 1986

[1]Parker, M. M., Benson, R. J., Enterprise-wide Information Management - An Introduction to the Concepts, IBM Los Angeles Scientific Center Report Number G320-2768, May 1985.

ABSTRACT

The subject of Information Systems Architecture is currently receiving considerable attention. The increased design scopes and levels of complexity of information systems implementation necessitate the use of some logical construct (or architecture) for defining and controlling the interfaces and the integration of all of the system's components. The amount of capital involved and the increasing dependency of a business success on its information systems preclude undisciplined approaches to management of those systems.

On the assumption that an understanding of information systems architecture is important to the development of a disciplined approach, the question that naturally arises is "What, in fact, is Information Systems Architecture"? This paper is an attempt to establish an independent definition of architecture and to map that definition on to the area of information systems. Some preliminary conclusions as to the implications of the resultant descriptive framework are drawn.

Preface

The subject of Information Systems Architecture is receiving increasing attention among information professionals. IBM has taken considerable interest in the subject and recently convened a task force at Corporate I/S to study the subject, establish corporate directions and guidelines for architecture, and select a set of tools and methodologies for implementation. Having been afforded the opportunity to participate in that task force, my initial reaction was that the subject of architecture needed more definition before guidelines could be established or tools selected.

Since I have been a part of a rather small community of people interested in the subjects of Enterprise Analysis and Architecture, etc. for many years, I am painfully aware that we have had great difficulty communicating with one another and understanding how we related to each other. It has always been clear to me that "architecture" meant different things to different people and it was equally clear that the subject's definition would continue to be elusive as long as we were attempting to define it out of the context of our own experiences and biases. Therefore, it would likely require a totally independent definition, outside the realm of Information Systems, to establish a basic understanding or definition, and then it would require an effort to build an analogue in the information systems area in order to get a rational, objective specification of the subject.

This paper is an attempt to establish that independent definition of architecture and to map that definition into the area of information systems; then, to draw some preliminary conclusions as to the implications of the resultant descriptive framework.

Introduction

The subject of information systems architecture is beginning to receive considerable attention. The increased design scopes and levels of complexity of information systems implementations are forcing the use of some logical construct (or architecture) for defining and controlling the interfaces and the integration of all of the system's components. A mere 30 or so years ago this was not a significant issue at all because the technology itself did not provide for either breadth in scope or depth in complexity in information systems. The inherent limitations of 4K machines, for example, constrained design and necessitated sub-optimal approaches for automating a business.

Current technology is rapidly removing both conceptual and financial constraints. It is not hard to speculate about, if not realize, very large, very complex systems implementations, extending in scope and complexity to encompass an entire enterprise. One can readily delineate the merits of the large, complex, enterprise-oriented approaches. Such systems allow flexibility in managing business changes and coherency in the management of business resources. However, there also is merit in the more traditional, smaller, sub-optimal systems design approach as well. Such systems are relatively economical, quickly implemented, and easier to design and manage.

In either case, as the technology permits "distributing" very large amounts of computing facilities in very small packages to very remote locations, some kind of structure (or architecture) is imperative because decentralization without structure is chaos. Therefore, to keep from dis-integrating the business, the concept of information systems architecture is becoming less and less of an option for establishing some order and control in the investment of information systems resources. The amount of capital involved, and the increasing dependency of the business' success on its information systems preclude undisciplined approaches to management of those systems.

On the assumption that an understanding of information systems architecture is important to the development of a disciplined approach, the question that naturally arises is "What, in fact, is Information Systems Architecture"? Unfortunately, among the proponents of I/S architecture, there seems to be little consistency in concepts or in specifications of "architecture" to the extent that the words "information systems architecture" are already losing their meaning!

It is necessary to develop some kind of framework in order to rationalize these varying architectural concepts and specifications in order to provide for clarity of professional communication, to allow for improving and integrating development methodologies and tools, and to establish credibility and confidence in the investment of systems resources.

In searching for an objective, independent pattern on which to base a framework for information systems architecture, it seems only logical to look to the field of (classical) architecture itself. In so doing, it could be possible to learn from the thousand or so years of experience that has been accumulated in that field. Definition of the deliverables of the classical architect could lead to the specification of analogous information systems architectural products and in so doing, classify our concepts and specifications.

With this objective in mind, that is, of describing the analogous information systems architectural representations, the following is an examination of the classical architect's deliverables produced in the process of building a building.[1]

A. "Bubble Charts"

The first architectural deliverable created by the architect is a conceptual representation, a "bubble chart", which depicts, in gross terms, the size, shape, spatial relationships and basic intent of the final structure. This "bubble chart" results from the initial conversations between the architect and prospective owner. A sample of such an initial conversation might be:

"I'd like to build a building."
"What kind of building did you have in mind? Do you plan to sleep in it? Eat in it? Work in it?"
"Well, I'd like to sleep in it."
"Oh, you want to build a house?"
"Yes, I'd like a house."
"How large a house did you have in mind?"
"Well, my lot size is 200' x 300'."
"Then, you want a house about 50' x 100'?"
"Yes, that's about right."
"How many bedrooms do you need?"
"Well, I have two children, so I'd like three bedrooms."
etc.

Note that each question serves to pose a constraint (the lot size) or identify a requirement (the number of bedrooms) in order to establish the "ballpark" within which any design will take place. From the above dialog, the architect can depict what the owner has in mind in the form of a series of "bubbles", each bubble representing a room, its gross size, shape, spatial relationships, etc.

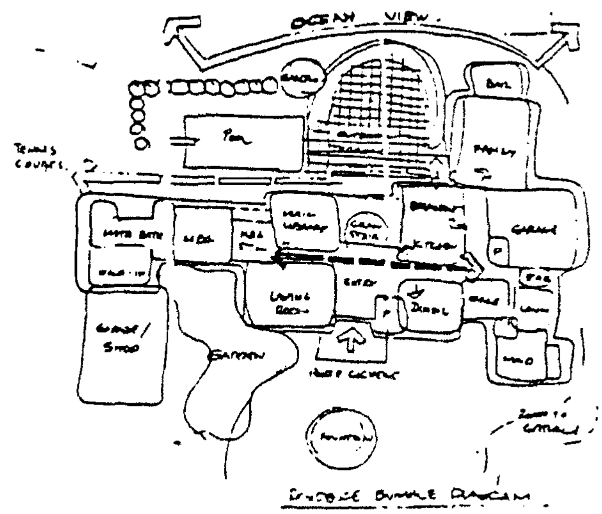


Figure 1. Architect's Bubble Chart.

The architect prepares this bubble chart for two reasons. First, he/she must elicit from the prospective owner what they have in mind in order to serve as a foundation or basis for the architect's actual design work. Second, the architect must convince the owner that he/she understands the owner's desires well enough that the owner will <u>pay</u> for the creative work to follow, and in effect, initiate the project.

PRODUCT	NATURE/PURPOSE
"BUBBLE CHARTS"	• BASIC CONCEPTS FOR BUILDING • GROSS SIZING, SHAPE, SPATIAL RELATIONSHIPS • ARCHITECT/OWNER MUTUAL UNDERSTANDING • INITIATE PROJECT

Figure 2. "Bubble Charts"

Having established a basic understanding with the prospective owner, the architect produces the next set of architectural deliverables which are called architect's drawings.

B. Architect's Drawings

The architect's drawings are a transcription of the owner's perceptual requirements, a depiction of the final product from the owner's perspective.

The drawings include horizontal sections (floor plans), vertical sections (cut-aways), and pictorial representations depicting the artistic motif of the final structure. The purpose of these drawings is to enable the owner to relate to them and to agree or disagree: "That is exactly what I had in mind!" or "Make the following modifications."

The drawings can be very detailed, however, they are normally developed only to the level of detail required for the prospective owner to understand and approve the design.

PRODUCT	NATURE/PURPOSE
ARCHITECT'S DRAWING	• FINAL BUILDING AS SEEN BY THE OWNER • FLOOR PLANS, CUT-AWAYS, PICTURES • ARCHITECT/OWNER AGREEMENT ON BUILDING • ESTABLISH CONTRACT

Figure 3. Architect's Drawings

Once the owner agrees that the architect has captured what he or she had in mind, and further agrees to pay the price for continuing the project, the architect produces the next set of architectural deliverables which is called the architect's plans.

C. Architect's Plans

Architect's plans are the translation of the owner's perceptions/requirements into a producible product. The plans are the designer's representation of the final product (as opposed to the owner's representation which is embodied in the drawings). The designer's representation (Plans) specify the material composition of the final product.

Plans are composed of 16 categories of detailed representations including site work, electrical system, masonry, wood structure, etc. They describe material relationships in the form of diagrams ("drawings") as well as bills of material. These plans are the final deliverables prepared by the architect and ultimately become the official "record" of the finished structure.

PRODUCT	NATURE/PURPOSE
ARCHITECT'S PLANS	• FINAL BUILDING AS SEEN BY THE DESIGNER • TRANSLATION OF OWNER'S VIEW INTO A PRODUCT • DETAILED DRAWINGS -- 16 CATEGORIES • BASIS FOR NEGOTIATION W/GEN. CONTRACTOR

Figure 4. Architect's Plans.

The architect's plans are prepared to serve as a basis for negotiation with a general contractor. The owner takes the plans to a contractor and says "Build me one of these." If the contractor builds "one of these," which is represented in the architect's plans, the owner has a high probability of getting what he/she wants, which is depicted in the architect's drawings.

As a result of the negotiations between the owner and general contractor, the plans may be modified because of cost/price considerations, but finally serve to represent what is committed to construction.

D. Contractor's Plans

At this point, the contractor re-draws the architect's plans to represent the builder's perspective. This is due to the fact that complex engineering products are not normally built in a day. Some phased approach is required which, in the case of a building, may be comprised of first, some site work; next, the foundation; next the first floor, etc. Furthermore, the contractor may have technology constraints. Either the tool technology or process technology may constrain his ability to produce precisely what the architect has designed. In either case, the contractor will have to design a reasonable facsimile which is buildable and satisfies the requirements. These technology constraints, plus the natural constraints requiring phased construction, are reflected in the contractor's plans which represent the builder's perspective and serve to direct the actual construction activity.

PRODUCT	NATURE/PURPOSE
CONTRACTOR'S PLANS	• FINAL BUILDING AS SEEN BY THE BUILDER • ARCHITECT'S PLANS CONSTRAINED BY LAWS OF NATURE AND AVAILABLE TECHNOLOGY • "HOW TO BUILD IT" DESCRIPTION • DIRECTS CONSTRUCTION ACTIVITIES

Figure 5. Contractor's Plans.

E. Shop Plans

Other representations, short of the final structure itself, are prepared by sub-contractors. These representations are called shop plans and are drawings of parts or subsections which are an out-of-context specification of what actually will be fabricated or assembled. The drawings, architect's plans and contractor's plans are all in-context because the owner, architect and contractor are all concerned with the entirety of the structure whereas the sub-contractor's representations are out-of-context because they are concerned with components or parts of the total structure. These shop plans might even serve as patterns for a quantity of identical parts to be fabricated for the project.

PRODUCT	NATURE/PURPOSE
SHOP PLANS	• SUB-CONTRACTOR'S DESIGN OF A PART/SECTION • DETAILED STAND-ALONE MODEL • SPECIFICATION OF WHAT IS TO BE CONSTRUCTED • PATTERN

Figure 6. Shop Plans.

F. The Building

In the case of buildings, the final representation is the physical building itself.

In summary, there are a set of "architectural" representations that are produced in the process of constructing a building.

PRODUCT	NATURE/PURPOSE
"BUBBLE CHARTS"	• BASIC CONCEPTS FOR BUILDING • GROSS SIZING, SHAPE, SPATIAL RELATIONSHIPS • ARCHITECT/OWNER MUTUAL UNDERSTANDING • INITIATE PROJECT
ARCHITECT'S DRAWING	• FINAL BUILDING AS SEEN BY THE OWNER • FLOOR PLANS, CUT-AWAYS, PICTURES • ARCHITECT/OWNER AGREEMENT ON BUILDING • ESTABLISH CONTRACT
ARCHITECT'S PLANS	• FINAL BUILDING AS SEEN BY THE DESIGNER • TRANSLATION OF OWNER'S VIEW INTO A PRODUCT • DETAILED DRAWINGS -- 16 CATEGORIES • BASIS FOR NEGOTIATION W/GEN. CONTRACTOR
CONTRACTOR'S PLANS	• FINAL BUILDING AS SEEN BY THE BUILDER • ARCHITECT'S PLANS CONSTRAINED BY LAWS OF NATURE AND AVAILABLE TECHNOLOGY • "HOW TO BUILD IT" DESCRIPTION • DIRECTS CONSTRUCTION ACTIVITIES
SHOP PLANS	• SUB-CONTRACTOR'S DESIGN OF A PART/SECTION • DETAILED STAND-ALONE MODEL • SPECIFICATION OF WHAT IS TO BE CONSTRUCTED • PATTERN
BUILDING	• PHYSICAL BUILDING

Figure 7. The set of architectural representations prepared over the process of building a building.

A Generic Set of Architectural Representations

Having specified the set of architectural representations produced in the process of building a building, it becomes apparent that this may be a generic set of "architectures" produced in the process of building any complex engineering product. A cursory examination of air frame manufacturing appears to validate this hypothesis as follows:

A. "Concepts" Equals "Bubble Charts"

The air frame manufacturers begin with some "concepts" specification of the "ballpark" in which they intend to manufacture. For example, the final product will fly so high, so fast, so far, for such and such purpose, so many people, etc. etc. to establish the gross size, shape, performance of the intended product.

B. Work Breakdown Structure Equals Architect's Drawings

The work breakdown structure is the "owner's perspective." The government requires that the manufacturer specify the work to be accomplished in terms of the components/systems against which costs are accrued and schedules are managed. In this fashion, the government controls the manufacturer in the production of the product.

C. Engineering Design Equals Architect's Plans

Engineering, the designers, translates the work breakdown structure into a physical product. The resultant "engineering design" is composed of drawings and bills of material.

D. Manufacturing Engineering Bill of Materials Equals Contractor's Plans

Manufacturing Engineering, the builders, apply the laws of nature and technology constraints to the engineering design to describe how to build the product (inside-out, bottom-up) and insure everything designed is actually producible.

E. Assembly and Fabrication Drawings Equals Shop Plans

Assembly and Fabrication drawings are the instructions to the shop floor personnel on how they are to assemble/fabricate the pieces or parts as stand-alone entities.

F. Machine Tool Representation

Because manufacturing uses computer-controlled equipment to produce some parts, they insert an additional representation of the final piece or part, short of the physical part itself. This representation is a "program" ("numerical code program"), a machine language representation.

G. Airplane Equals Building

The final representation is not really a representation (architecture) but the actual, physical thing itself.

In any case, there appears to be conceptual equivalents in the manufacturing industry for the architectural representations of the construction industry. This would strengthen the argument that an analogous set of architectural representations are likely to be produced over the process of building any complex engineering product, including an Information System.

Before identifying the information systems analogues, it is useful to make some general observations with regard to architecture.

First, there appear to be three fundamental architectural representations, one for each "player in the game," that is: the owner, the designer and the builder. The owner has in mind some product that will serve some purpose. The architect transcribes this product for the owner, the owner's perspective. Then the architect translates this representation into a physical product, the designer's perspective. Then the builder applies the constraints of the laws of nature and available technology to make the product producible, the builder's perspective.

Preceding these three fundamental representations, a gross sizing, shape, scope representation is created to establish the "ballpark" within which all of the ensuing architectural activities will take place.

Succeeding the three fundamental representations are the detailed, out-of-context representations which technically could be considered architectures because they are representations short of being the final physical product. However, they are somewhat less interesting "architecturally" since they do not depict the final product in total, and are more oriented to the actual implementation activities. Nonetheless, they are included in this discussion for the purpose of insuring a comprehensive framework.

A significant observation regarding these architectural representations is that each is of a different <u>nature</u> than the others. They are not merely a set of representations, each of <u>which</u> is an increasing level of detail than the previous one. Level of detail is an independent variable, varying <u>within</u> any one architectural representation. For example, the designer's representation (i.e., architect's plans) is different than the owner's representation (i.e., architect's drawings). It is not a succeeding level of detail, it is different in <u>nature</u>, representing a different perspective. The level of detail of the designer's representation (i.e., Plans) is variable, and quite independent from the level of detail for the owner's representations (i.e. Drawings). <u>Et Cetera</u>.

Given this description of the levels of architectural representation produced over the process of building a complex engineering product, it is relatively straight-forward to identify the analogues in the information systems area, since information systems are also "complex engineering products." See Figure 8.

<u>Different Ways to Describe the Same Thing</u>

In the process of examining the field of architecture to discover the generic architectural "products" that are produced in the construction of a complex engineering product, a second important idea emerges with regard to descriptive representations (or architectures). In addition to the different perspectives which have to be represented (e.g. the owner, the designer and the builder), there also are different types of descriptions. For example, there are functional descriptions, and there are material descriptions. It is

Levels of Architectural Representations

GENERIC	BUILDINGS	AIRPLANES	INFORMATION SYSTEMS
"Ballpark"	Bubble Charts	Concepts	Objectives/Scope
Owner's Representation	Architect's Drawings	Work Breakdown Structure	Business Description
Designer's Representation	Architect's Plans	Engineering Design/Bill of Materials	Information System Description (Conceptual Model)
Builder's Representation	Contractor's Plans	Manufacturing Engineering Design/ Bill of Materials	Technology Constrained Description (Physical Model)
Out-of-Context Representation	Shop Plans	Assembly/Fab-rication Drawings	Detailed Description
Machine Tool Representation	-	Numerical Code Programs	Machine Language Description (Object Code)
Product	Building	Airplane	Information System

Figure 8. The levels of architectural representations produced over the process of building a complex engineering product along with the analogues in the building, airplane, and information system communities.

common for any physical product to have functional specifications as differentiated from material specifications.

In developing functional descriptions, the person who is preparing the description is looking at the product from the perspective of how it works or what it does. The focus is on the "transform" that is taking place. Typically, the generic descriptive model that is used to describe transform is "input - process - output" where "process" represents the transform and the "inputs" and "outputs" are resource flows that link the transforms together in some sequential fashion. An example of a functional description might be the process description of an oil refinery.

In contrast, a description of the material components of a product focuses on structure as opposed to transform. The describer's perspective is "what the product is made out of." The generic descriptive model typically used is "thing - relationship - thing" where "thing" is some material component and "relationship" specifies the structural relationship between one component (thing) and another component (thing). An example of the material description of a product is a bill-of-materials.

When an architect is describing multiple units in, for example, a development project, the geographical relationship between the components becomes significant as provisions must be made for the flow of traffic, people, electricity, gas, etc. from unit to unit. In this event a flow description, composed generally of "site - link - site" is appropriate for describing the product.

In any case, a complex engineering product may have a variety of types of descriptive models depending on the use of the description. Clearly, there are, for example:

a. Functional descriptions - How it works.
b. Material descriptions - What it's made of.
c. Geographical description - Where flows exist.

There may be other descriptions including:

d. Organization descriptions - Who is involved.
e. Dynamics descriptions - When things happen.
f. Objectives descriptions - Why things happen.

etc.

For 1985 purposes, it is complex enough to focus on the Functional, Material and Geographical descriptions which address transform, structure and flow. Consideration of the other descriptions can be postponed at least temporarily until the architectural implications of the first three are appropriately assimilated.

As in the case of the levels of architectural representations, the Information Systems analogues for the different descriptive models are also readily identifiable.

The functional description is obvious, in fact, the information systems terminology is identical, "input-process-output."

The material description, describing the what the product is made of equates to the data description, data being the "stuff" the information systems products are made of.

The geographical description (flow model) equates to the network or communications description. See Figure 9.

DESCRIPTIVE MODEL:	MATERIAL (Structure)	FUNCTIONAL (Transform)	GEOGRAPHIC (Flow)
Generic Formula	-Thing- -Relationship- -Thing	-Input- -Process- -Output-	-Site- -Link- -Site-
I/S NAME:	DATA MODEL (Structure)	FUNCTIONAL MODEL (Transform)	NETWORK MODEL (Flow)
Specific Formula	-Entity- -Relationship- -Entity-	-Input- -Process- -Output-	-Node- -Line- -Node-

Figure 9. Various descriptive models for describing objects (products) along with the Information Systems analogues.

The Framework

Combining the two ideas that:

a. There are a set of architectural representations produced to represent different perspectives involved during the process of building complex engineering products, and

b. There are different types of descriptive models for a product, developed for different purposes,

results in specifying the relationship between them, that is, for every different descriptive model (functional, data, network, etc.), there is a set of architectural representations, representing the different perspectives of the different people involved (owner, designer, builder, etc.).

The single factor that makes this relationship significant is that each element on either axis of the resultant matrix is explicitly differentiable from all other elements on the same axis. That is, the data model (entity - relationship - entity) is different from the functional model (input - process - output). The functional model (input-process-output) is different from the network model (node-line-node). Et cetera.

ARCHITECTURAL PERSPECTIVE / DESCRIPTIVE MODEL	DATA ENT.-REL.-ENT.	FUNCTION IN-PROC.-OUT	NETWORK NODE-LINK-NODE
"SCOPE" DESCRIPTION			
BUSINESS DESCRIPTION			
INFORMATION SYSTEMS DESCRIPTION			
TECHNOLOGY CONSTRAINED DESCRIPTION			
DETAIL DESCRIPTION			
MACHINE LANGUAGE DESCRIPTION			
ACTUAL SYSTEM			

Figure 10. A Framework for Information Systems Architecture.

By the same token, the business description (owner's perspective) is different from the Information Systems Description (designer's perspective). The Information Systems Description (designer's perspective) is different from the Technology Description (builder's perspective). Et cetera. Note once again, for example, that the Information Systems Description is not merely a lower level of detail than the Business Description. It is different. It would be analytically convenient if it was one-for-one, more detail - and at times it just happens to work out that way, but not as a general rule. Level of detail is an independent variable. That is, the level of detail in any one description can vary independently from the level of detail in any other description.

In any event, each of the elements on the "perspective" axis are different in nature just like the elements on the "descriptive model axis" are different in nature.

The explicit differentiation of elements on either axis is significant because, since it is possible to generically characterize every element on both axes, it is then possible to explicitly characterize the contents of every cell in the matrix. Such a characterization constitutes the specification of a Framework for Information Systems Architecture, as follows.

Architectural Representations for Describing Data

First, focusing on the Data (entity-relationship-entity) column and further, looking at the Scope Description level architectural representation, one might expect to find a list of things of significance to the business under consideration.[2]

This representation would be a list of <u>things</u> (i.e. material; grammatically, nouns) as opposed to a list of <u>actions</u> (i.e. processes; grammatically, verbs). A list of actions (verbs) could be expected in the next column, the Function column. The list of things (material) in the Data column would be called "entities" in the data modeling vernacular.

Since this architectural representation is at the Scope Description level, one would also expect that the entities (things) would likely be entity "classes," higher levels of aggregation, because the decision being made as a result of this level of description would be one of scope, not one of design. That is, a selection would be being made of the entity class or classes in which to invest I/S resource for "inventory" management purposes.

Further, at this level, one might not expect to be definitive about the relationship between the entities. The scope decision would constitute overlaying the business values on the total range of possibilities to identify a subset of entity classes for implementation which is consistent with the resources available for investing in information systems, specifically, in this case, the management of the selected class (or classes) of data. (For example, see Figure 11.)

WWW ENTITIES	
Product	Policies & Procedures
Part	Legal Requirements
Supplies	G/L Accounts
Equipment	Accounts Payable
Employee	Accounts Receivable
Customer	Long Term Debt
Supplier	Marketplace
Competitor	Promotion
Bldg. & Real Estate	Purchase Order
Objectives	Customer Order
Job	Production Order
Organization Unit	Shipment

Figure 11. Data column. Scope Description row.
Example: List of Entities.[3]

Looking at the next lower level of architectural representation in the matrix, the owner's representation or Business Description, what could be anticipated, for example, is an "entity-relationship diagram."[4,5,6]

At this level, "entity" would mean "business entity" as opposed to "data entity," which would be found at the succeeding level. For example, when an owner, in describing the business, would specify an entity like "employee," what he/she would have in mind would be the real thing, that is, flesh and blood "employee." That meaning of employee is entirely different than an Information Systems Description (the designer's description) in which "employee" would refer to a record in a machine which also happens to be called "employee," conceptually, entirely different.

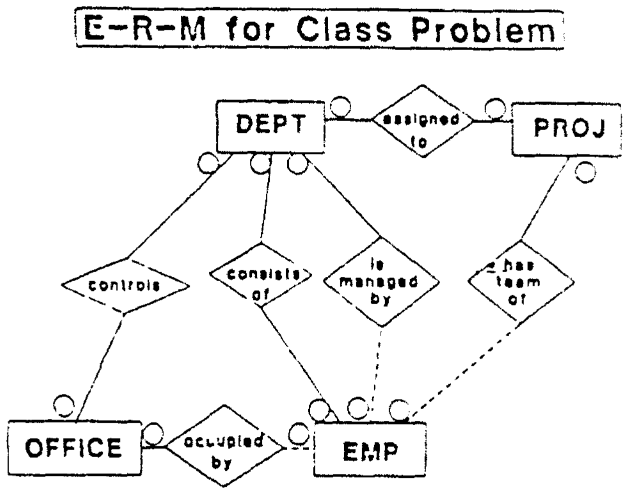


Figure 12. Data column. Business Description row. Example: Entity-Relationship Diagram.[3]

Further, when the owner, describing this business, would specify a relationship between the entities, what he/she would have in mind would be the business rule that relates one entity to another entity.[7] This is, for example, "one employee must have one (and only one) organization to which he or she belongs for payroll purposes." This is a business rule and not a data relationship as would be expected in the next lower architectural level, the Information Systems Description (designer's perspective).

In attempting to find "real life" examples of each of the architectural representations, it is interesting to note that finding good examples which crisply illustrate each representation is very difficult. There are two reasons for this. First, as the real life representations were being developed, no framework existed to clearly define and differentiate one representation from the others. Therefore, many real life illustrations are a mixture of representation, both conceptually (e.g. business entities and data entities get mixed together) and physically (e.g. entities and inputs/outputs, that is, user views, from the Function column, get mixed together). Secondly, real life examples are hard to understand because it is not always clear what level or model the author had in mind when developing the representation.

An illustration of this difficulty exists in Figure 12. It is clear that this model is describing Data and not Function (the middle column), but the question is, did the author have in mind a description of a business or a description of an information system? In this case, it is likely that this is a description of a business because of the existence of the "many-to-many" relationships including the one between the department and project entities. Many-to-many relationships cannot be implemented on a two dimensional machine. They have to be resolved into many-to-one and one-to-many relationships by creating an artificial entity through the concatenation of the keys of the two original entities' keys. That is, in order to make the business description into an information systems product, it has to be "normalized." Therefore, the person who built the model in Figure 12, probably had in mind describing a business as opposed to an information system. (Although, because a "framework" may not have existed at the time the model was built, which conceptually differentiated the two descriptions, the actual picture may be mixed conceptually, that is, not clearly a business description or an information systems description but a little of both. Be that as it may, the example in Figure 12 is more likely to be a business description.)

Looking at the next level down in the Data column, the Information Systems Description (designer's perspective), one might expect to find, for example, a "data model."[4,5,6]

In this case of an Information Systems Description as opposed to a Business Description, the meaning of "entity" would change to that of a record in a machine and a relationship would change to that of a data relationship. Clearly, the example in Figure 13 is a model of an information system and not a model of a business because of the existence of "artificial" entities, specifically the "DEPTPROJ" entity, the concatenation of department and project which clearly is not a real life thing, but an information system thing created in the process of translating the business description into an information systems "product."

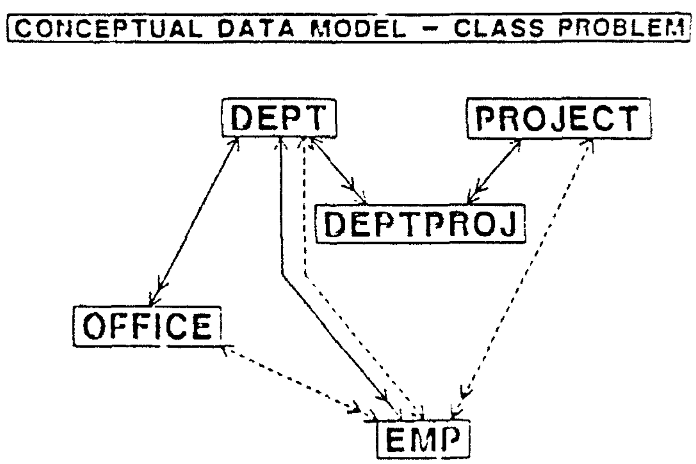


Figure 13. Data column. Information System Description row. Example: Data Model.[3]

Once again, shifting to the next level of architectural representation in the data column, the Technology Constrained Description, what could be expected would be the physical implementation or data design for the conceptual model of the information system above.

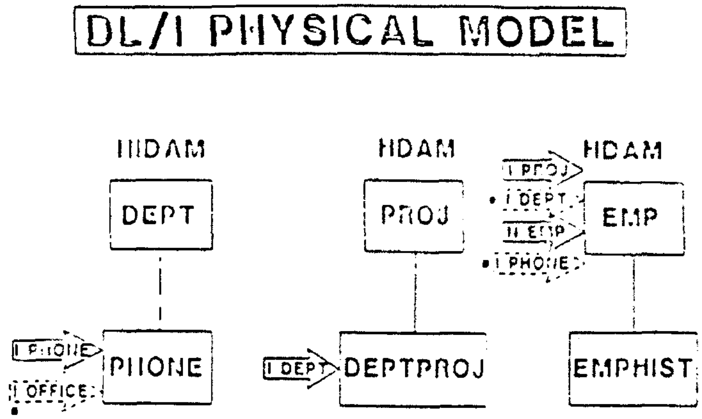
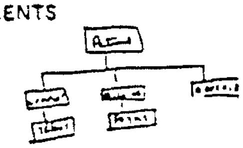


Figure 14. Data column. Technology Description row.
Example: Data Design

At the Technology Constrained Description level, the laws of nature and technology constraints are being applied. A decision is made to use IMS or DB2 or XYZ and depending on the choice, the meaning of entity and relationship change. In the case if IMS, entity means "segment" and relationship means "pointer." In the case of DB2, entity means "row" and relationship means "key," etc.[8].

Proceeding down the Data column to the Detail Description, or "out-of-context" level of description, the example would be some data description language like

DBDGEN — SAMPLE STATEMENTS



```
DBD NAME=CLINIC,ACCESS=HIDAM
DATASET DD1=HIDD,DEVICE=3340
SEGM NAME=PATIENT,PARENT=0,BYTES=100
     FIELD NAME=(NAME,SEQ,U),BYTES=40,START=1
SEGM NAME=COMPLNT,PARENT=PATIENT,BYTES=77,RULES=FIRST
     FIELD NAME=ILLNS,BYTES=35,START=1
SEGM NAME=TRTMNT,PARENT=COMPLNT,BYTES=140
     FIELD NAME=(DATE,SEQ,M),BYTES=8,START=1
     FIELD NAME=ACTN,BYTES=100,START=9
SEGM NAME=BILLING,PARENT=PATIENT,BYTES=60,RULES=LAST
SEGM NAME=PAYMT,PARENT=BILLING,BYTES=60
SEGM NAME=HOUSHLD,PARENT=PATIENT,BYTES=50
     FIELD NAME=RELATN,BYTES=20,START=31
DBDGEN
FINISH
END
```

Figure 15. Data column. Detail Description row.
Example: DBDGEN

a DBDGEN in which the entities are specifications of the "fields" and relationships are specifications of the "addresses."[8]

This description is "compiled" to produce the Machine Language representation (relative addressing, not shown in the figure) which is further "link edited" to produce the actual physical data residing in the machine.

It is clear that real life examples can be found to illustrate the levels of architectural representations, representing various viewpoints or perspectives that are created for the data (or material) description of the information system.

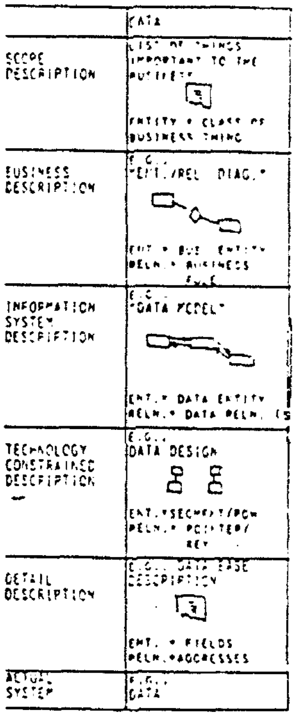


Figure 16. Total set of architectural representations describing data.

Architecture Representations for Describing Function

Similarly, examples can be found for describing Function (Input-Process-Output).

At the Scope Description level, a comprehensive list of the range of possibilities for functional automation could be expected. In describing Function, the elements of the descriptive model are input-process-output. Function is equivalent to 'process' and would likely be some process "class", a relatively high level of aggregation, as the decision being made at the Scope level is the selection of some subset of the business processes appropriate in which to invest some finite amount of information systems resources for automation purposes. Further, in making the scope decision, by overlaying the business values against the total range of automation possibilities, it is unnecessary to be definitive about the input and output linkages between the functions. Therefore, simply a list of business processes would appropriately be expected at this level of representation.[3]

SAMPLE MANUFACTURING PROCESSES

- Determine Product Requirements
- Plan Production
- Purchase Raw Materials
- Control Raw Materials Inventory
- Produce Product
- Assess Production Quality
- Control Product Inventory
- Distribute Product
- Market Product
- Process Order

Figure 17. Function column. Scope Description row.
Example: List of Processes[3]

Proceeding to the Business Description level, what could be expected, for example, is a functional flow diagram,[9,10] in which "process" would be a business process (not an information systems process) and input/output would be business resources like people, cash, material, product, etc.

Figure 18 is clearly a business model (as opposed to an information systems model) because, in the original, it can be seen that the inputs and outputs are business resources (not necessarily information). This particular example in Figure 18 is a very high level example, not putting much detail specification around either the inputs/outputs or the processes for that matter.

An example of the next level, the Information Systems Description, would be a data flow diagram[9,10] in which processes would be information systems (application) processes (not business processes) and the inputs/outputs would be "user views" (some aggregates of data elements that flow into and out of the applications processes, connecting them in some sequential fashion.

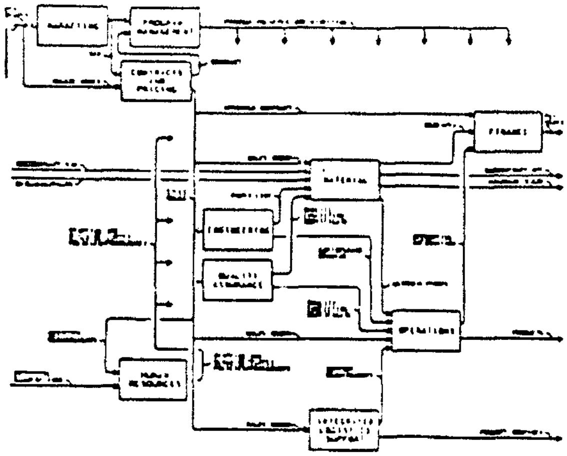


Figure 18. Function column.
Business Description.
Example: Functional Flow Diagram.

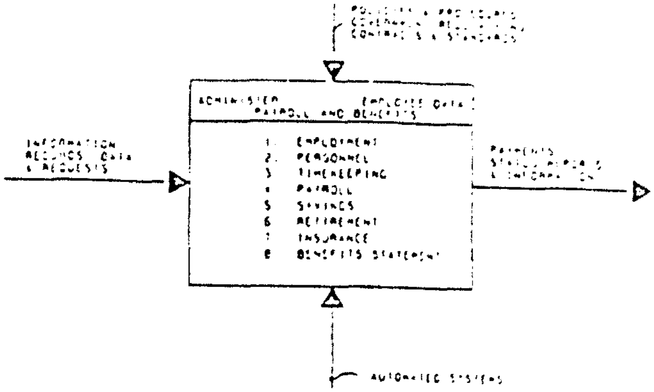


Figure 19. Function column.
Information Systems Description.
Example: Data Flow Diagram

Figure 19 is, once again, a very high level data flow diagram in the IDEF convention[4] which shows inputs and outputs (user views) as well as controls (the arrow from the top) and mechanization (the arrow from the bottom).

Applying the physical constraints of the technology chosen for implementation, for example; disks vs. tapes, IMS vs. CICS, COBOL vs. FORTRAN, video displays vs. typewriters, etc.; results in the Technology Constrained Description in which process is a computer function and inputs/outputs are device formats. The predictable representation would be a structure chart with screen/device formats (Figure 20).[11,12] (Note that this does not preclude depicting the manual functions that are introduced as a result of the employment of the technology.)

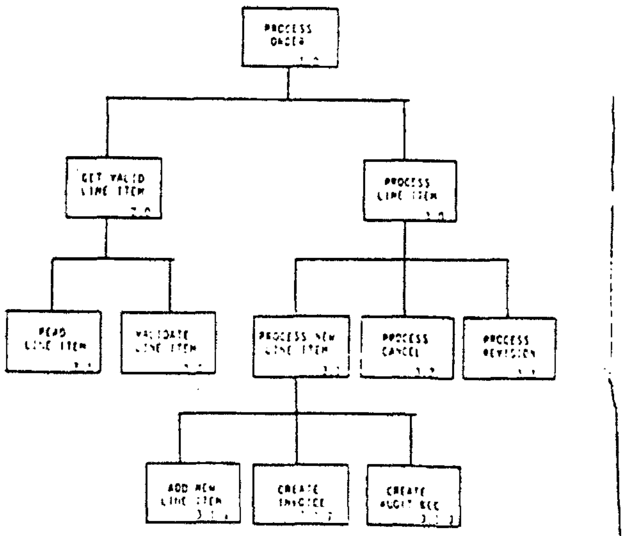


Figure 20. Function column.
Technology Description.
Example: Structure Chart.

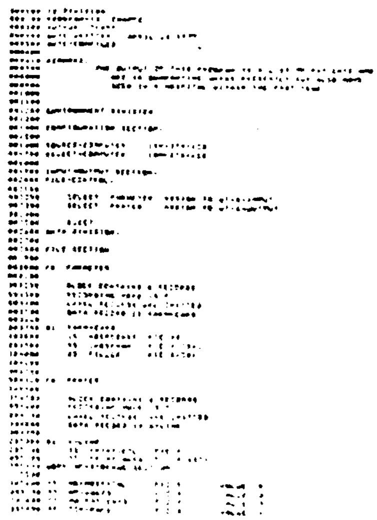


Figure 21. Function column.
Detail Description row.
Example: COBOL program.

At the Detailed Description level the example is a program in which the process is a language statement and the inputs/outputs, control blocks.

The program is compiled to produce object code, the Machine Language representation which in turn is assembled to produce running instructions, the actual, physical system.

Again, it is clear that examples can be found for every descriptive representation for the Functional Model as well as the Data Model.

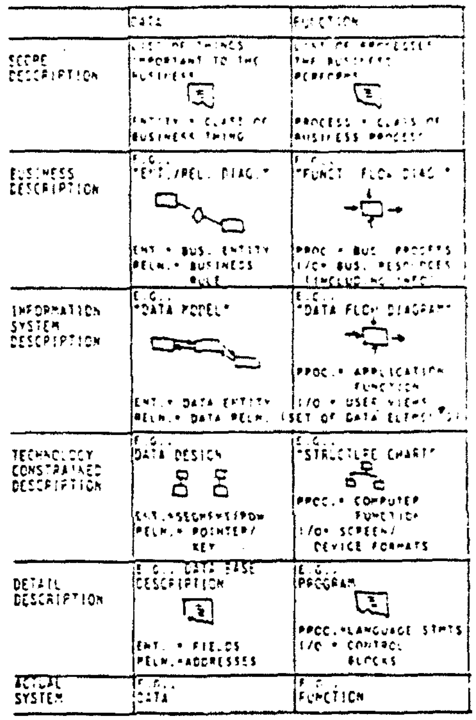


Figure 22. Total set of architectural representations representing Data and Function.

Architecture Representation for Describing Network

Examples of the architectural representations for describing Network (node - line - node) are as follows:

At the Scope Description level, a map. (See Figure 23.)

At the Business Description level, specification of the business unit locations for the nodes and the business relationships (e.g. organizational, product, informational, etc.) for the lines. (See Figure 24.)

At the Information Systems Description level, specification of the I/S function for the node (e.g. processor, storage, access, etc.) and line characteristics for the lines. (This is the "distributed systems" decision description.) (See Figure 25.)

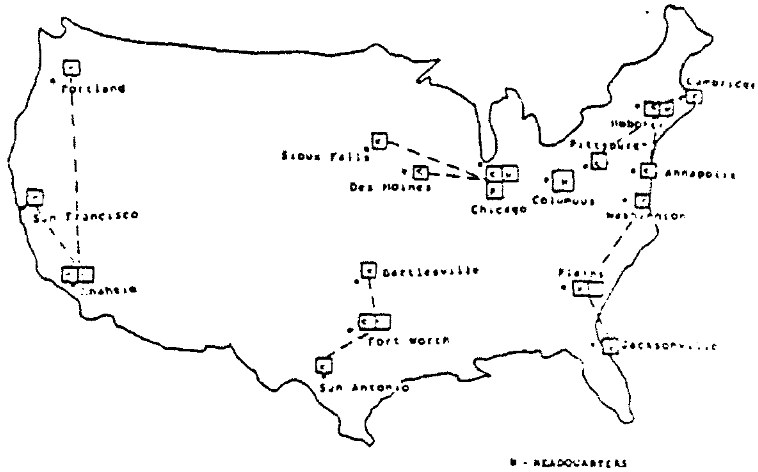


Figure 23. Network column. Scope Description row.
Example: A map.

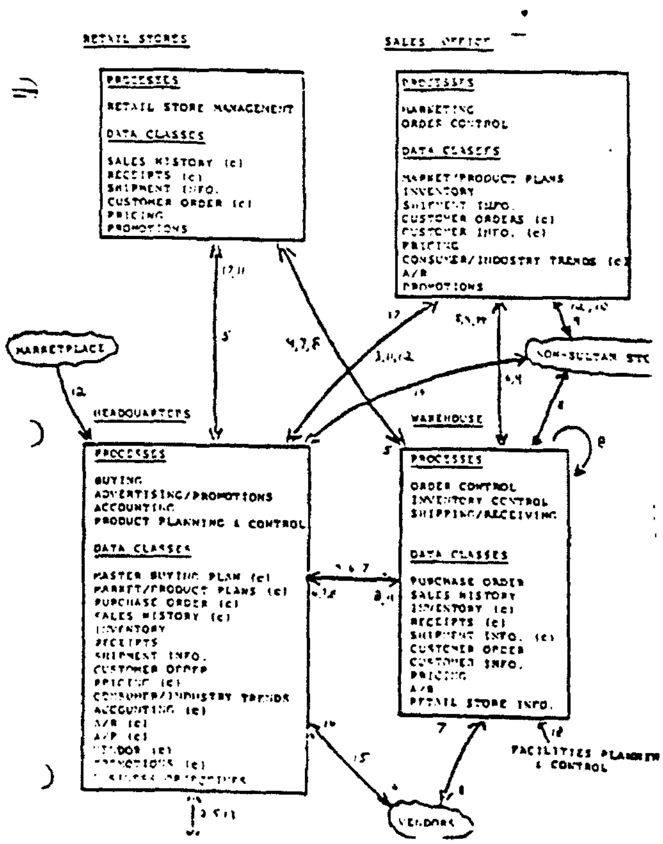


Figure 24. Network column. Business Description row.[3]

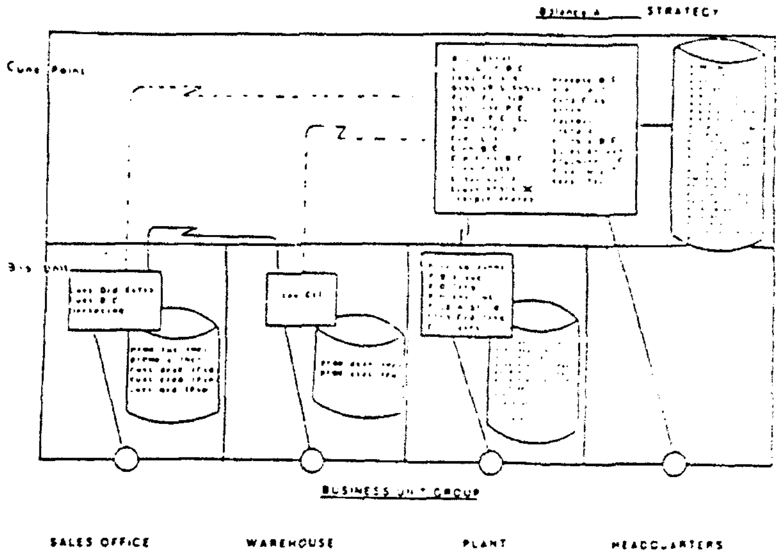


Figure 25. Network column. Information Systems Description row.[3]

At the Technology Constrained Description level, the nodes are specific hardware/software implementations (e.g. 4341, CICS, NCP VTAM, etc.) and the lines are line specifications. (See Figure 26.)

At the Detailed Description level, nodes are addresses and lines are protocols. (I don't know much about communications, but these are probably "compiled" to produce some object code equivalent which is then "link-edited" to produce the running network.) (See Figure 27.)

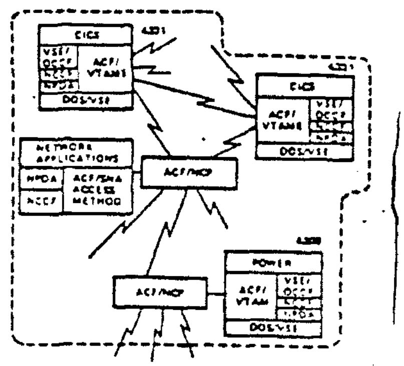


Figure 26. Network column.
Technology Description row.

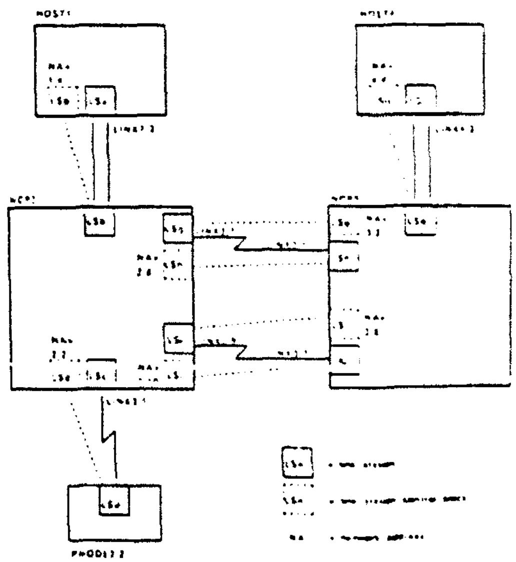


Figure 27. Network column.
Detail Description row.

In summary, examples can be found to illustrate every hypothetical architectural representation postulated by the relationship between the different descriptive models and the various levels of architectural perspective.

	DATA	FUNCTION	NETWORK
SCOPE DESCRIPTION	LIST OF THINGS IMPORTANT TO THE BUSINESS ENTITY = CLASS OF BUSINESS THING	LIST OF PROCESSES THE BUSINESS PERFORMS PROCESS = CLASS OF BUSINESS PROCESS	LIST OF LOCATIONS IN WHICH THE BUSINESS OPERATES NODE = BUSINESS LOCATION
BUSINESS DESCRIPTION	E.G., "ENT./REL. DIAG." ENT.= BUS. ENTITY RELN.= BUSINESS RULE	E.G., "FUNCT. FLOW DIAG." PROC.= BUS. PROCESS I/O= BUS. RESOURCES (INCLUDING INFO)	E.G., ? NODE=BUSINESS UNIT LINK=BUSINESS RELATIONSHIP (ORG.,PRODUCT,INFO.)
INFORMATION SYSTEM DESCRIPTION	E.G., "DATA MODEL" ENT.= DATA ENTITY RELN.= DATA RELN.	E.G., "DATA FLOW DIAGRAM" PROC.= APPLICATION FUNCTION I/O = USER VIEWS (SET OF DATA ELEMENTS)	E.G., ? NODE= I/S FUNCTION (PROCESSOR,STORAGE ACCESS, ETC.) LINK = LINE CHARACTERISTICS
TECHNOLOGY CONSTRAINED DESCRIPTION	E.G., DATA DESIGN ENT.=SEGMENT/ROW RELN.= POINTER/KEY	E.G., "STRUCTURE CHART" PROC.= COMPUTER FUNCTION I/O= SCREEN/DEVICE FORMATS	E.G., SYSTEM ARCHITECTURE NODE= HARDWARE/SYS SOFTWARE LINK= LINE SPECIFICATIONS
DETAIL DESCRIPTION	E.G., DATA BASE DESCRIPTION ENT. = FIELDS RELN.=ADDRESSES	E.G., PROGRAM PROC.=LANGUAGE STMTS I/O = CONTROL BLOCKS	E.G., ? NODE = ADDRESSES LINK = PROTOCOLS
ACTUAL SYSTEM	E.G., DATA	E.G., FUNCTION	E.G., COMMUNICAIONS

Figure 28. Framework for Information Systems Architecture.

Conclusions

When the question is asked, "what is Information Systems Architecture?" The answer is, "there is not <u>an</u> Information Systems Architecture, but a <u>set</u> of them!" Architecture is relative. What you think architecture is depends upon what you are doing.

- o If you are <u>programming</u>, you probably think 'architecture' is a structure chart.

- If you are the Data Base Administrator, you probably think 'architecture' is data design.

- If you are the Data Administrator, you probably think 'architecture' is a data model.

- If you an Analyst, you probably think 'architecture' is a data flow diagram.

- If you are a Planner, you probably think 'architecture' is some combination of entity/relationship diagram and functional flow diagram.

- If you are the Communications Manager, you probably think 'architecture' is the yet to be named communications representations.

- If you are the Operations Manager, you probably think 'architecture' is the "systems architecture."

- If you are the President, you probably think 'architecture' is the entity classes, process classes and a map.

- If you are the Program Support Representative, you probably think 'architecture' is the detailed descriptions.

- If you are the Computer Designer, you probably think 'architecture' is machine language. (The level not represented on the summary chart.)

It is little wonder we are having difficulties communicating with one another about architecture because there is not an architecture, but a set of architectural representations. One is not right and another wrong. The architectures are different. They are additive, complementary. There are reasons for electing to expend the resources for developing each architectural representation. And, there are risks associated with not developing any one of the architectural representations.

Research is being done to put some more explicit definitions around each of the architectural representations in this framework, to understand the design issues, the reasons for developing each representation, the risks associated with not developing any one, and the "tool" implications of each cell. This research and some of the management implications of the framework will be the subject of forthcoming articles in the "Framework" series.

Summary

In summary, by studying fields of endeavor external to the information systems community, specifically those professions involved in producing complex engineering products (e.g. architecture/construction, manufacturing, etc.), it is possible to hypothesize by analogy, a set of architectural representations for Information Systems.

The resultant "Framework for Information Systems Architecture" could prove quite valuable for:

- o Improving professional communications within in the information systems community;
- o Understanding the reasons for an risks of not developing any one architectural representation;
- o Placing a wide variety of tools and/or methodologies in relation to each other; and
- o Developing improved approaches (including methodologies and tools) to produce each of architectural representations as well as possibly rethinking the structure of the classic "application development process."

REFERENCES

1. G. D. Larson, private communication, G. D. Larson, Architects International Association, Woodland Hills, California (1985).

2. "Business Systems Planning - Information Systems Planning Guide," Application Manual GE20-0627, IBM Corporation. Available through IBM Branch Offices.

3. "Business Systems Planning," class material, IBM Corporation, Information Systems Management Institute, Los Angeles, California.

4. ICAM Architecture Part II, Volume V - Information Modeling Manual (IDEF1) AFWAL-TR-81-4023. Air Force Wright Aeronautical Labs, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio 45433 (June 1981).

5. P. P. Chen, "Entity-Relationship Approach to Systems Analysis and Design," UCLA, Los Angeles, California (December 1979).

6. M. Flavin, "Fundamental Concepts of Information Modeling," Yourdon Press, Inc., New York (1981).

7. D. S. Appleton, "Business Rules: The Missing Link," Datamation (October 15, 1984).

8. C. J. Date, "An Introduction to Database Systems." Addison-Wesley, Reading, Massachusetts (1981).

9. T. DeMarco, "Structured Analyses and Systems Specifications," Yourdon Press, Inc., New York (1978).

10. S. M. McMenamin and J. F. Palmer, "Essential Systems Analysis," Yourdon Press, Inc., New York (1984).

11. J. K. Hughes, J. I. Michtom, "A Structural Approach to Programming," Prentice-Hall, Englewood Cliffs, New Jersey (1977).

12. "HIPO - A Design Aid and Documentation Technique," IBM Manual GC20-1851, IBM Corporation. Available through IBM branch offices.

13. H. D. Leeds, G. M. Weinberg, "Computer Programming Fundamentals," McGraw-Hill, New York (1961).

14. "Systems Network Architecture - Technical Overview," Systems Manual GC30-3073, IBM Corporation. Available through IBM branch offices.

IBM LOS ANGELES SCIENTIFIC CENTER OUTSIDE PUBLICATIONS

FOR THE YEAR 1982

F. W. BALL, J. G. SAKAMOTO, "Supporting Business Systems Planning Studies with the DB/DC Data Dictionary", IBM Systems Journal, Vol. 21, No. 1 (1982).

F. W. BALL, J. G. SAKAMOTO, "Supporting Business Systems Planning Studies with the DB/DC Data Dictionary", Database Management, November-December issue, Auerbach Information Management Series (1982).

S. A. JUROVICS, "Daylight, Glazing, and Building Energy Minimization", ASHRAE Transactions, Vol. 88, Part 1 (1982).

S. A. JUROVICS, "The Impact of Daylight Utilization on Thermal and Lighting Loads: A Case Study", presented at ASHRAE Annual Meeting, Toronto, June 1982.

J. M. MARSHALL, "Systems Architect Apprentice System (SARA)", Proceedings of Corporate Symposium on Structured Logic, Yorktown, November (1982).

P. S. NEWMAN, "Towards an Integrated Development Environment", IBM Systems Journal, Vol. 21, No. 1 (1982).

M. M. PARKER, "Enterprise Information Analysis: Cost-Benefit and the Data Managed System", IBM Systems Journal, Vol. 21, No. 1 (1982).

FOR THE YEAR 1982 (Continued)

M. M. PARKER, "Toward Cost-Benefit Analysis of Data Administration", SHARE Proceedings, Vol. 1, August 1982.

J. G. SAKAMOTO, "Use of DB/DC Data Dictionary to Support Business Systems Planning Studies: An Approach", The Economics of Information Processing, Vol. 1, Publisher: John Wiley & Sons Publishing Co (1982).

J. G. SAKAMOTO, "Use of DB/DC Data Dictionary to Support Business Systems Planning Studies: An Approach", Symposium on The Economics of Information Processing, Sponsored by the IBM Systems Research Institute (1982).

N. C. SHU, "Specifications of Forms Processing and Business Procedures for Office Automation", IEEE Transactions on Software Engineering, Vol. SE-8, No. 5 (1982).

N. C. SHU, D. M. CHOY, V. Y. LUM, "CPAS: An Office Procedure Automation System", IBM Systems Journal, Vol. 21, No. 3 (1982).

R. C. SUMMERS, "Computer Security", chapter of Computer Handbook, Publisher: Van Naustrand Reinhold Publishing (1982).

FOR THE YEAR 1983

C. WOOD, E. B. FERNANDEZ, "Minimization of Demand Paging for the LRU Stack Model of Program Behavior", Information Processing Letters, Vol. 16, No. 2 (1983).

SCIENTIFIC CENTER REPORT INDEXING INFORMATION

1. Author(s): Zachman, John A.	9. Subject Index Terms: Information Systems Architecture; Information Systems; Architecture
2. Title: A Framework for Information Systems Architecture	
3. Originating Department: Los Angeles Scientific Center	
4. Report Number: 320-2785	
5a. Number of pages: 27	5b. Number of References: 14

6a. Date Completed	6b. Date of Initial Printing	6c. Date of Last Printing
March 1986	April 1986	

7. Abstract:

The subject of Information Systems Architecture is currently receiving considerable attention. The increased design scopes and levels of complexity of information systems implementation necessitate the use of some logical construct (or architecture) for defining and controlling the interfaces and the integration of all of the system's components. The amount of capital involved and the increasing dependency of a business' success on its information systems preclude undisciplined approaches to management of those systems.

On the assumption that an understanding of information systems architecture is important to the development of a disciplined approach, the question that naturally arises is "What, in fact, is Information Systems Architecture?" This paper is an attempt to establish an independent definition of architecture and to map that definition on to the area of information systems. Some preliminary conclusions as to the implications of the resultant descriptive framework are drawn.

8. Remarks:

Preliminary

CALS Phase II Architecture

APPENDIX B

STANDARDS

Appendix B
Standards

1. Program Management

Configuration Management

MIL-STD-483	Configuration Management Practices for Systems, Equipment, Munitions, and Computer Programs
MIL-STD-881	Work Breakdown Structures for Defense Materiel Items
MIL-STD-780	Work Unit Codes for Aircraft
MIL-STD-480	Configuration Control - Engineering Changes, Deviations and Waivers

Quality Assurance

MIL-Q-9858	Quality Program Requirements

2. Design

Drawings

DOD-STD-100	Engineering Drawing Practices
MIL-D-1000	Drawings, Engineering and Associated Lists
MIL-D-5480	Data, Engineering and Technical: Reproduction Requirements for

MIL-M-9868	Microfilming of Engineering Documents, 35mm Requirements for
MIL-M-38761	Microfilming and Photographing of Engineering/ Technical Data and Related Documents
MIL-STD-804	Formats and Coding of Aperture, Copy, and Tabulating Cards for Engineering Data Microreproduction System

Specifications

MIL-STD-490	Specification Practices
MIL-S-83490	Specifications, Types and Forms
MIL-STD-961	Outline of Forms and Instructions for the Preparation of Specifications and Associated Documents

3. Systems Engineering

MIL-STD-499	Engineering Management

Reliability

MIL-STD-785	Reliability Program for Systems and Equipment Development and Production
MIL-STD-2155	Failure Reporting, Analysis, and Corrective Action System

Maintainability

MIL-STD-470	Maintainability Program for Systems and Equipment
MIL-STD-471	Maintainability Demonstration

Safety

MIL STD-882	System Safety Program Requirements

Standardization

MIL-STD-680	Contractor Standardization Plans and Management

4. Support

MIL-STD-1388/1	Logistic Support Analysis
MIL-STD-1388/2	DoD Requirements for a Logistic Support Analysis

Maintenance Planning

MIL-STD-1390	Level of Repair
MIL-STD-1629	Procedures for Performing a Failure Mode, Effects, and Criticality Analysis

Support Equipment

MIL-STD-2165	Testability Program for Electronic Systems and Equipments

MIL-C-45662	Calibration System Requirements
MIL-STD-2077	General Requirements for Test Program Sets

Provisioning

MIL-STD-1561	Uniform DoD Provisioning Procedures
MIL-STD-789	Procurement Method Coding of Replenishment Spare Parts

Packaging, Handling, Storage, and Transportation

MIL-STD-648	Design Criteria for Specialized Shipping Containers
MIL-E-17555	Packaging and Packing of Electronic and Electrical Equipment, Accessories and Repair Parts
MIL-STD-1367	Packaging, Handling, Storage, and Transportability Program Requirements
MIL-STD-2073	Packaging Requirements

Technical Publications

MIL-M-15701	Content Requirements for Technical Manuals: Equipment and Systems
MIL-STD-7298	Manuals, Commercial Off-the-Shelf

MIL-M-24100	Manuals Technical: Functionally Oriented Maintenance Manuals (FOMM) for Equipment and Systems
MIL-M-38807	Manuals, Technical: Illustrated Parts Breakdown, Preparation of
MIL-M-38784	Manuals, Technical: General Style and Format Requirements
MIL-STD-1685	Comprehensibility Standards for Technical Manuals
MIL-M-85337	Requirements for Technical Manual Quality Assurance Program

<u>Standardization</u>

MIL-STD-680	Contractor Standardization Plans and Management

Preliminary

CALS Phase II Architecture

APPENDIX C

GLOSSARY

CALS Phase II Glossary

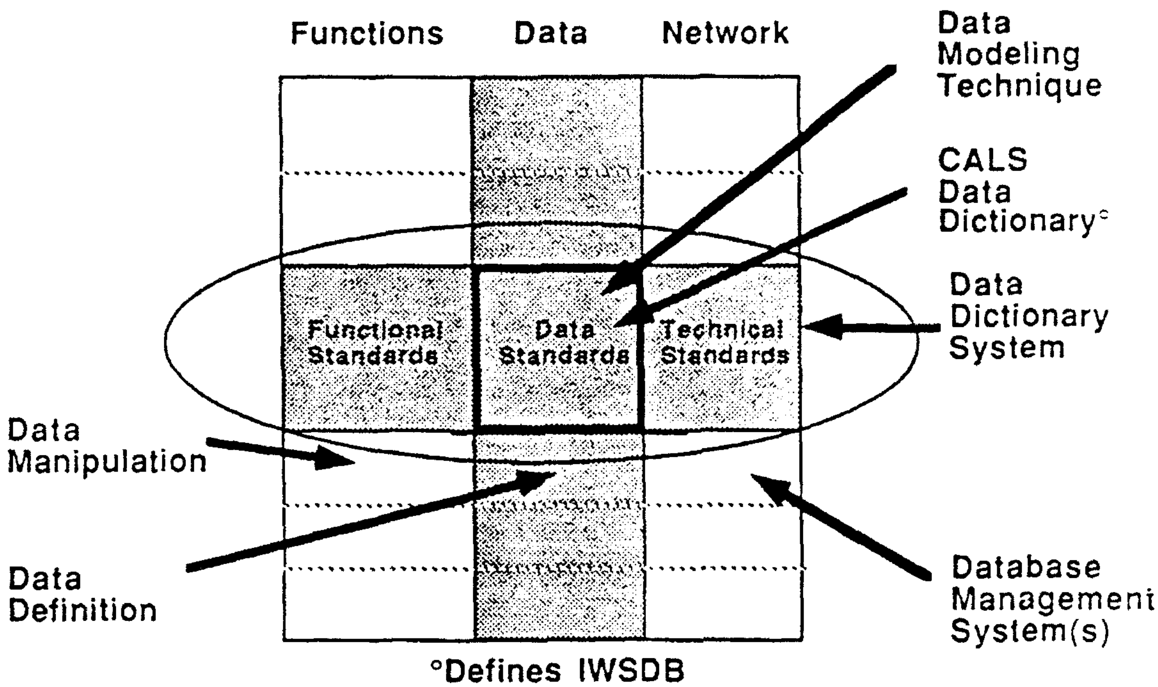
This section contains a set of definitions for terms used in this report.

CALS Data Dictionary - A set of data standards, defined in IDEF1X, that are derived from the data element descriptions contained in existing and future functional standards as well as from other sources, such as PDES, that will be used as guidelines for the construction and verification of Integrated Weapon Systems Databases supporting CALS Phase II services.

CALS Data Dictionary System - A specific set of computer programs that are used to develop, verify, validate, and manage data standards contained in the CALS Data Dictionary.

Contractor Integrated Technical Information Service (CITIS) - A specific implementation of CALS Phase II services on a specific weapon system program. CITIS results in an Integrated Weapon System Database (IWSDB) constructed in compliance with the Data Standards in the CALS Data Dictionary. A CITIS delivery system must comply with appropriate Technical Standards in MIL-STD-1840A.

Integrated Weapon System Database (IWSDB) - A specific implementation of Data Standards from the CALS Data Dictionary that supports a specific CITIS. An IWSDB will inevitably contain data in addition to those specified in the CALS Data Dictionary but required by a specific weapon system program.


Functions
Data
Network
Data Modeling Technique
CALS Data Dictionary°
Data Dictionary System
Functional Standards
Data Standards
Technical Standards
Data Manipulation
Data Definition
Database Management System(s)
°Defines IWSDB