NIST Special Publication 500-208

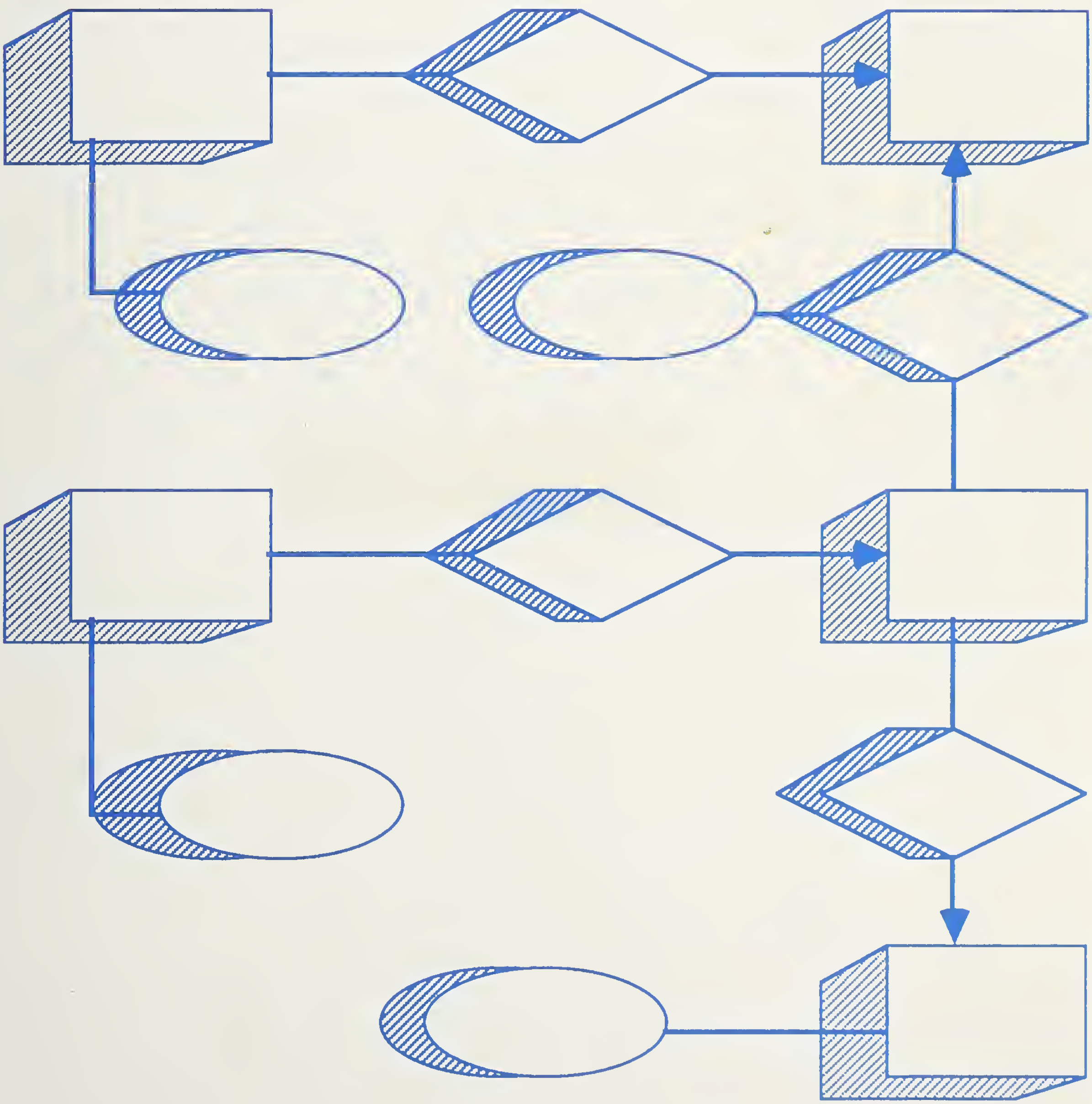
Computer Systems Technology

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards and Technology

NIST

Manual for Data Administration

Judith J. Newton and
Daniel C. Wahl, Editors



The National Institute of Standards and Technology was established in 1988 by Congress to "assist industry in the development of technology . . . needed to improve product quality, to modernize manufacturing processes, to ensure product reliability . . . and to facilitate rapid commercialization . . . of products based on new scientific discoveries."

NIST, originally founded as the National Bureau of Standards in 1901, works to strengthen U.S. industry's competitiveness; advance science and engineering; and improve public health, safety, and the environment. One of the agency's basic functions is to develop, maintain, and retain custody of the national standards of measurement, and provide the means and methods for comparing standards used in science, engineering, manufacturing, commerce, industry, and education with the standards adopted or recognized by the Federal Government.

As an agency of the U.S. Commerce Department's Technology Administration, NIST conducts basic and applied research in the physical sciences and engineering and performs related services. The Institute does generic and precompetitive work on new and advanced technologies. NIST's research facilities are located at Gaithersburg, MD 20899, and at Boulder, CO 80303. Major technical operating units and their principal activities are listed below. For more information contact the Public Inquiries Desk, 301-975-3058.

Technology Services
- Manufacturing Technology Centers Program
- Standards Services
- Technology Commercialization
- Measurement Services
- Technology Evaluation and Assessment
- Information Services

Electronics and Electrical Engineering Laboratory
- Microelectronics
- Law Enforcement Standards
- Electricity
- Semiconductor Electronics
- Electromagnetic Fields[1]
- Electromagnetic Technology[1]

Chemical Science and Technology Laboratory
- Biotechnology
- Chemical Engineering[1]
- Chemical Kinetics and Thermodynamics
- Inorganic Analytical Research
- Organic Analytical Research
- Process Measurements
- Surface and Microanalysis Science
- Thermophysics[2]

Physics Laboratory
- Electron and Optical Physics
- Atomic Physics
- Molecular Physics
- Radiometric Physics
- Quantum Metrology
- Ionizing Radiation
- Time and Frequency[1]
- Quantum Physics[1]

Manufacturing Engineering Laboratory
- Precision Engineering
- Automated Production Technology
- Robot Systems
- Factory Automation
- Fabrication Technology

Materials Science and Engineering Laboratory
- Intelligent Processing of Materials
- Ceramics
- Materials Reliability[1]
- Polymers
- Metallurgy
- Reactor Radiation

Building and Fire Research Laboratory
- Structures
- Building Materials
- Building Environment
- Fire Science and Engineering
- Fire Measurement and Research

Computer Systems Laboratory
- Information Systems Engineering
- Systems and Software Technology
- Computer Security
- Systems and Network Architecture
- Advanced Systems

Computing and Applied Mathematics Laboratory
- Applied and Computational Mathematics[2]
- Statistical Engineering[2]
- Scientific Computing Environments[2]
- Computer Services[2]
- Computer Systems and Communications[2]
- Information Systems

---

[1] At Boulder, CO 80303.
[2] Some elements at Boulder, CO 80303.

NIST Special Publication 500-208

Manual for Data Administration

Judith J. Newton, Editor

Computer Systems Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899

and

Daniel C. Wahl, Editor

Vector Research, Inc.
Arlington, VA

Sponsored by National Capital Region Chapter,
Data Administration Management Association

March 1993

U.S. DEPARTMENT OF COMMERCE
Ronald H. Brown, Secretary

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
Raymond G. Kammer, Acting Director

Reports on Computer Systems Technology

The National Institute of Standards and Technology (NIST) has a unique responsibility for computer systems technology within the Federal government. NIST's Computer Systems Laboratory (CSL) develops standards and guidelines, provides technical assistance, and conducts research for computers and related telecommunications systems to achieve more effective utilization of Federal information technology resources. CSL's responsibilities include development of technical, management, physical, and administrative standards and guidelines for the cost-effective security and privacy of sensitive unclassified information processed in Federal computers. CSL assists agencies in developing security plans and in improving computer security awareness training. This Special Publication 500 series reports CSL research and guidelines to Federal agencies as well as to organizations in industry, government, and academia.

National Institute of Standards and Technology Special Publication 500-208
Natl. Inst. Stand. Technol. Spec. Publ. 500-208, 155 pages (Mar. 1993)
CODEN: NSPUE2

U.S. GOVERNMENT PRINTING OFFICE
WASHINGTON: 1993

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, DC 20402

Foreword

This Manual is the result of two years of research, discussion and development by the Data Administration Standards and Procedures Working Group, sponsored by the National Capital Region chapter of the Data Administration Management Association (DAMA-NCR). DAMA is the professional association for data administrators, and has no ties to particular organizations or vendors.

NIST is publishing this document because it makes a valuable contribution to forwarding the understanding and practice of data administration. As many authors contributed to its creation, the views expressed herein should **not** be assumed to be the "official" views of NIST.

Any use of trade, product or organization names in this publication is for descriptive purposes only and does not imply endorsement by DAMA, NIST or the U.S. Government.

This manual is the product of outstanding efforts by many volunteers. The group is co-chaired by Judith Newton (National Institute of Standards and Technology) and Dan Wahl (Vector Research, Inc.). Primary writers and reviewers for the manual include David Bloom (American Management Systems), Bill Brimberry (National Park Service), Alan Crosby (U.S. Department of the Interior), Connie Davis (Defense Information Systems Agency), Bill McDonald (Bethlehem Steel), Sandra Perez (Concept Technology, Inc.), Joan Sziede (United States Geological Survey), David Stowell (Stowell Associates), Judith Newton, and Dan Wahl. Thao Nguyen (Patent and Trademark Office) compiled the glossary with help from Dan Moreno (U.S. Department of the Interior). Karen Keefer, Fritz Bowers, and Norma Mowry (all of the U.S. Army Depot Systems Command) provided graphics support. Special thanks to Len Brush (Carnegie Mellon University) who edited and produced both draft versions of the manual; Kimberly Wade (Vector Research, Inc.) who produced the final version; and Duane Hufford (American Management Systems), Gary Beuschel (Army Corps of Engineers), and Burton Parker (MITRE) who carefully reviewed the manual.

This document is meant to complement the material found in NIST Special Publication 500-173, **Guide to Data Administration.**

Comments are welcome and can be directed to Dan Wahl (703) 521-5300 or Judith Newton (301) 975-3256. Our mailing address is c/o DAMA-NCR, 3729 Curran Street, McLean, Virginia 22101.

This material is also available through DataMart, a professional online service for data administrators. Contact Joe Girdwood (714) 730-3670 or Mary Jane Frances Smith (301) 948-9262 for access information.

Message from the Chapter President

As president of the National Capital Region chapter of the Data Administration Management Association (DAMA-NCR), I am proud to recommend the results of the past two years' endeavors of our Standards and Procedures Working Group. The Group's co-chairs, Judith Newton of the National Institute of Standards and Technology (Gaithersburg, MD) and Dan Wahl of Vector Research, Inc. (Arlington, VA), have done an outstanding job of bringing together a group of volunteer, working data administrators to present their views on a number of topics at the heart of our discipline.

Data administration's emergence as an independent discipline over the past decade or so has been marked by considerable progress and growth, but also much uncertainty and confusion. Many IRM professionals still do not have good answers to basic questions concerning its definition, scope, usefulness, and relation with other IRM disciplines.

DAMA-NCR is a professional development association devoted to addressing these questions by bringing together active data administrators, professionals working in closely related IRM disciplines, academicians, and vendors. For the past two years, the DAMA-NCR Standards and Procedures Working Group has provided a collegial format for discussions on topics of current interest to its participants. Given their results presented here, we look forward to the Group's initiatives in the coming year.

Carla von Bernewitz
President, DAMA-NCR

Table of Contents

PART 1: DATA ADMINISTRATION OVERVIEW

PART 2: DATA ADMINISTRATION ACTIVITIES

Table of Contents

Table of Contents

Table of Contents

List of Figures

List of Tables

Section 1. Introduction

This document provides guidance to data administrators in developing and carrying out a data administration (DA) function. Its purpose is to assist data administrators with establishing an organization's data administration function, or with adding, reviewing, developing, or implementing related activities and responsibilities. It is written by data administrators for data administrators.

Part 1 orients the data administration function within the general IRM (Information Resource Management) context, and provides guidance in defining an organization's data administration function. We present statements of data administration goals and objectives, as well as discussions of data administration-related responsibilities.

Part 2 describes a number of data administration related activities, each in its own section. Each section is designed to be useful to active data administrators in designing and developing their own organizational standards and procedures.

Appendices contain alternative definitions of data administration, a glossary of DA related terms, and a bibliography.

Section 2. Data Administration Defined

Data administration is emerging as an independent discipline, is evolving as an organizational function, and is not consistently defined. Data administration varies in definition and scope among organizations and even among units within organizations. One of our objectives is to offer usable definition(s) of data administration that are applicable across a range of organizational contexts.

Two frameworks for Data Administration are illustrated by Figure 2-1, "Data Administration Under the IRM Umbrella." The upper framework is one frequently seen in practice; the lower one is based on the **Guide to Data Administration**. In both, IRM is depicted as the overall area of responsibility. The first places Data Administration and Database Administration separately since they are often performed by separate offices, and database administrators do not report directly to data administrators. The **Guide** model places Database Administration under Data Administration because its concepts and principles are a subset of those used in Data Administration - it does not necessarily imply organizational hierarchy.

In both cases, data administrators and database administrators cooperate closely in managing an organization's data. Alternative definitions of data administration are provided in Appendix A. They have been gathered from experts in the discipline as from official government documents.

2.1. Data Administration in Information Resource Management

Because the fundamental objectives of data administration are to maximize the value, quality, and usability of data resources, data administrators must coordinate data requirements and design with the entire organization (management, system designers, database administrators and end users). To promote acceptance and understanding of this role, data administrators and their management must clearly define DA roles and interfaces within the IRM structure and other divisions. Table 2-1 presents a comparison of Data Administration and Information Resource Management.

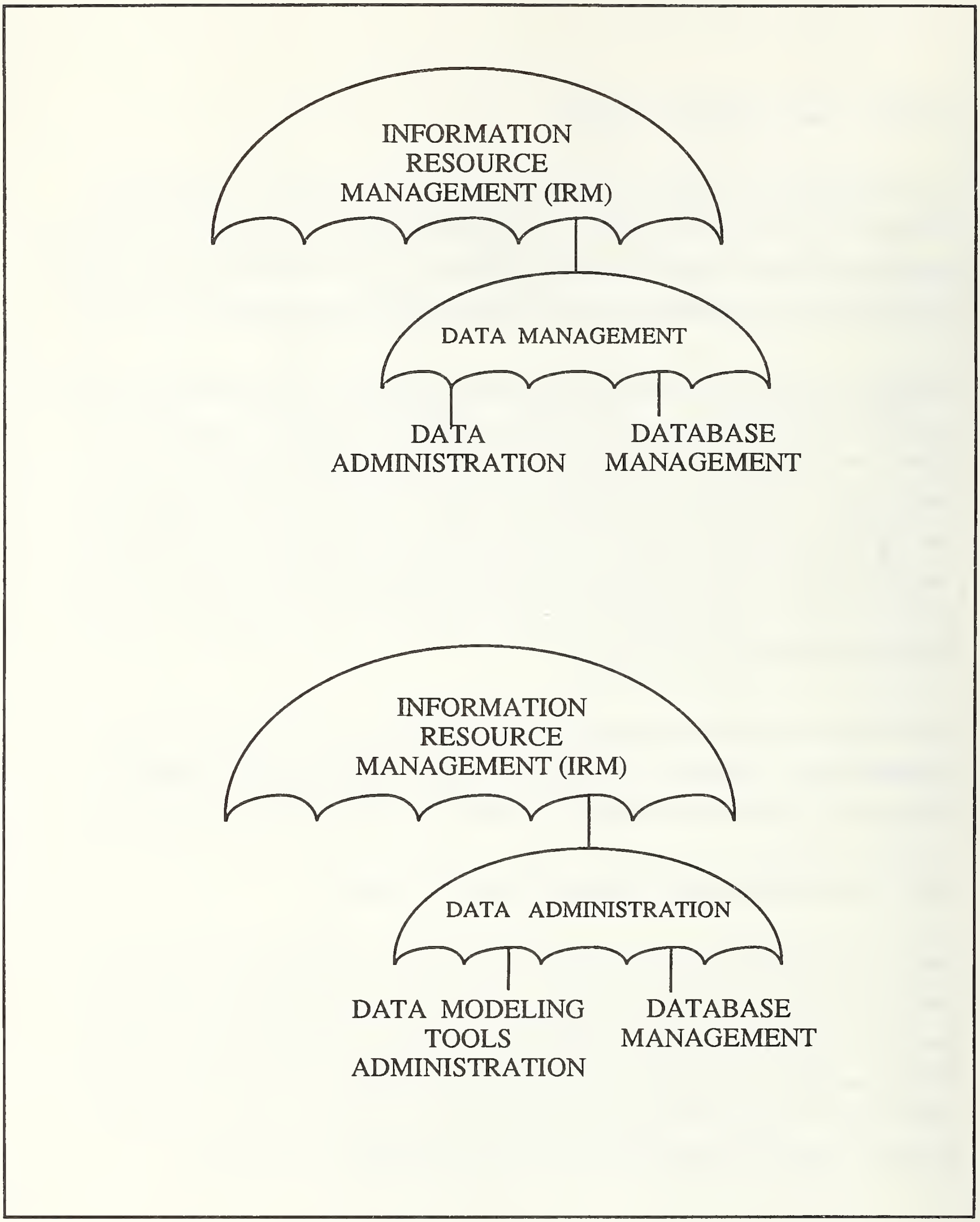


Figure 2-1: Data Administration Under the Information Resource Management (IRM) Umbrella: Two Paradigms.

Table 2-1: Comparison Of Data Administration And Information Resource Management

Area of Comparison	Data Administration	Information Resource Management
Primary Responsibility	Administrative	Managerial
Scope	All databases	All data oriented resources
Data Design	Logical	Responsibility for both logical and physical
Primary Liaison	Management	Management
Range of Concern	Long-term resource planning, cross-system planning and coordination	Data planning, operations
Primary Orientation	Metadata Data resource planning and coordination	Data and information systems

Figure 2-2 illustrates the working relationships and roles typical of a data administration function within an IRM organization.

2.2. Data Administration and Database Administration

The obvious distinction between **data administration** and **database administration** is that data administration is more concerned with data from a user view and functional standpoint while database administration is concerned with the technical issues related to (particular) database management systems.

Data administration functions focus on the conceptual planning and organization of the data resources. This includes data modeling, developing policy and standards for use and management of data, training users, and coordinating other data activities with system designers and users. Database administration, in contrast, deals with the technical operations and physical maintenance of the data resources as part of an automated database management system.

As a planner, a data administrator must coordinate with management and users, while the database administrator, as a technical expert, interacts with users, programmers, analysts, and other system developers. These basic differences are outlined in Table 2-2: Distinctions Between Data Administration and Database Administration.

One area of considerable overlap between the two (DA and DBA) responsibilities is the normalization process. Data administration is responsible for uncovering functional dependencies and providing the DBA function with a fully normalized logical data model. In practice the DBA function, in the implementation process, will either discover that the data reflect additional functional dependencies not uncovered during the logical data modeling process, or that the physical implementation requires override of entity and referential integrity rules. Normalization must be coordinated between the data administration function and database administration function.

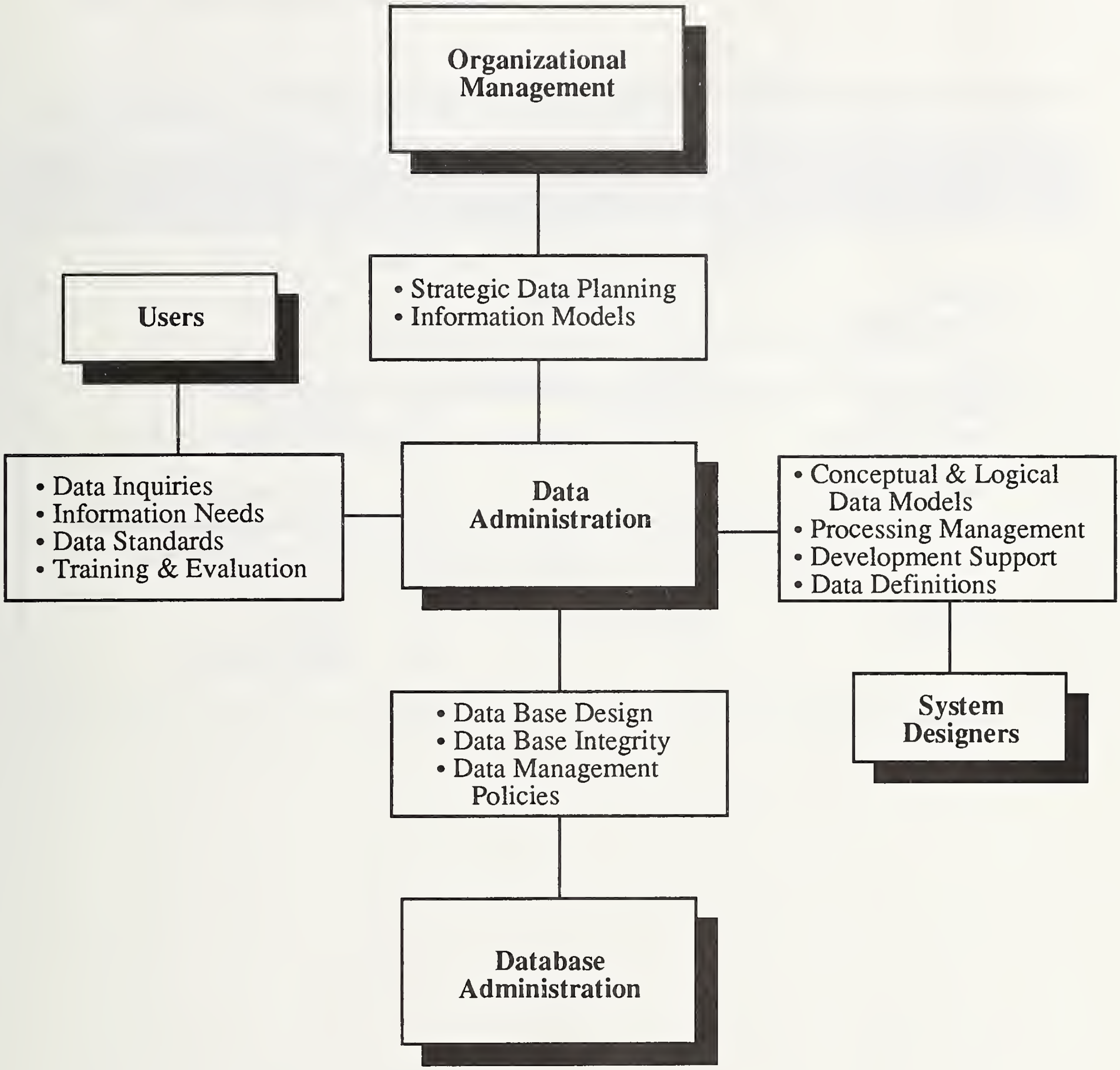


Figure 2-2: Organizational Interfaces of Data Administration.

Table 2-2: Distinctions Between Data Administration And Data Base Administration

Distinguishing Data Characteristics	Data Administration	Database Administration
Primary Responsibility	Administrative	Technical
Scope	Database Independent	All Databases
Data Design	Logical	Physical
Primary Liaison	Management	Programmer, analysts
Range of Concern	Long-term data planning	More concerned with short-term development and use of databases
Primary Orientation	Metadata Data Design	Data Database DBMS specific

Section 3. Data Administration Mission

This section presents mission statements for a data administration function.

- To combine activities, standard methods, human resources and technology for the central planning, documentation and management of data from the perspective of their meaning and value to the organization as a whole.

- To increase system effectiveness by controlling data through uniformity and standardization of data elements, database construction, accessibility procedures, system communication, and maintenance.

- To provide guidance for planning, managing and sharing of data and information effectively and efficiently in automated information systems.

Section 4. Data Administration Goals and Objectives

These statements of goals and objectives are intended as legitimate examples of choices each organization may embrace in the establishment and evolution of the organization's data administration function.

RECOGNIZE and **PROMOTE the importance of** **data** and information as valuable resources requiring management of their creation, use, storage, documentation and disposition;

PROMOTE DATA CONSISTENCY AND STANDARDIZATION throughout the organization by developing standards for data element names, definitions, values, formats, metadata and documentation in a central data repository and in databases;

CREATE AN ARCHITECTURE that consolidates the conceptual, logical and physical models of the data to the informational needs and business functions of the enterprise;

MINIMIZE DUPLICATION in collecting, processing, storing and distributing data;

ENCOURAGE AND FACILITATE DATA AND INFORMATION SHARING within the organization and among external user groups;

IMPROVE THE QUALITY, ACCURACY AND INTEGRITY of information and shared data resources; and

IMPROVE DATA MANAGEMENT AND ACCESS through the use of appropriate existing and new methods, tools and technologies.

Section 5. Data Administration Roles and Responsibilities

Data administration roles and responsibilities include:

(1) developing and implementing policies, procedures and standards for creating and handling data in a consistent manner, and providing education and training to promote data management skills.

(2) using techniques and methods to build a broad data architecture and to identify opportunities for sharing commonly-used data through integrated applications and databases.

(3) strategic planning for development of the data architecture and overall management of data resources consistent with organizational goals and business strategies;

(4) developing policies, standards, and procedures for creating, managing and improving the integrity of the data resources;

(5) technical and administrative control of the data repository to improve documentation and coordination;

(6) data modeling in preparation for database design and data migration;

(7) education and training in data management concepts.

The success of a data administration program will be measured by its ability to take on these roles and meet its responsibilities.

Tasks associated with DA roles and responsibilities are presented below.

Strategic Tasks -

(1) participate in strategic systems planning

(2) develop enterprise models for data activities

(3) reconcile enterprise-level models with system-level models

(4) name and define key entities

(5) design sharable data structures to implement strategic plan

Enforcement Tasks -

(1) specify standards, procedures and tools for data design

(2) establish organization-wide standards for naming data objects

(3) establish data management procedures to be used by others, such as system development life cycle procedures

(4) enforce these policies through Quality Assurance - system design

(5) conduct operational audits

(6) maintain integrity of data steward role

Consulting Tasks -

(1) encourage compliance with DA (Data Administration) objectives through QA (Quality Assurance)

(2) conduct educational seminars

(3) participate in project teams

(4) advise policy makers

Library Tasks -

(1) ensure consistent use of editing rules across systems

(2) maintain enterprise models

(3) design models and schemes

(4) maintain central repository of metadata from all systems

(5) assist system designers and others in use of repository

Technical Tasks -

(1) select, install and manage DA support software

(2) provide advice on use of Data Administration support software for system development and Data Administration functions

(3) provide advice on impact of changes to existing system

Section 6. Data Administration Staffing

This section presents data administration responsibilities in two forms. The first is a simple listing of responsibilities. The second is a more formal position description.

6.1. List of Responsibilities

Responsibilities of data administrators may include the following:

(1) Help to evaluate new application proposals to determine feasibility and identify the potential for sharing existing data. Also help identify and prioritize new hardware and software requirements;

(2) Promote education of all managers and users in the general concepts and responsibilities of successful data administration;

(3) Coordinate management and user participation to develop the information models, diagrams and reports that contribute to the data architecture;

(4) Maintain the documentation of all components of the data architecture (conceptual, logical and physical models) through a central data repository;

(5) Develop policies on data-related activities such as data integrity, data security, data inventory, data standards, data sharing and data repository; and

(6) Assist Database Administration with developing technical procedures such as change control, impact analysis, backup and recovery, integrity checks, etc., for preserving the integrity and security of the data resources.

6.2. Position Description

The following is a sample position description for a high level Data Administrator who heads a Data Administration Organization. In this model position, the Data Administrator delegates responsibilities for establishing and implementing one or more of his/her major duties to his/her staff. The staff therefore are given titles that are more descriptive of their specific duties and responsibilities. For instance, a Data Analyst would be responsible for establishing and implementing standards for strategic data planning, logical data modeling and data analysis; while a

Repository Manager would be responsible for establishing and implementing standards for development, population, and maintenance of the Information Resource Dictionary System (IRDS).

Position Title: Data Administrator

Responsibilities:

The Data Administrator is the head of the Data Administration (DA) organization under the Information Resource Management (IRM) umbrella. The DA organization is responsible for the definition, organization, supervision, and protection of data in order to provide good quality, sharable, and accessible data throughout the enterprise. The DA organization establishes and implements policies and procedures to support the missions of the Data Administration program as established by IRM. The Data Administrator interacts politically, diplomatically, and tactfully to sell, market, arbitrate and negotiate with upper management the Data Administration position to implement changes necessary to achieve a viable Data Administration program. Additionally, the DA organization coordinates with Database Administrators, data custodians, managers, end users, project managers, and application developers by providing education and technical support, reviewing feedback and developing good working relationships. The Data Administrator manages a staff that is responsible for establishing and implementing the Data Administration Program.

Major duties:

- Establish and maintain the DA program charter, goals, and objectives. Plan, schedule and monitor activities. Establish and maintain budgets. Define human resource requirements. Obtain resources and support services.

- Manage the development, approval, enforcement, and revision of DA Policies, Standards, Procedures, and associated documentation for strategic data planning, data modeling, data analysis, data element standardization, repository management, data quality improvement, data security, data conversion/transition, DA quality assurance.

- Establish and implement procedures for defining and setting DA quality assurance standards in order to measure, assess and continuously improve the DA Program. Approve recommendations for improvement and publish success stories and lessons learned.

- Manage the establishment and application of procedures and techniques for quantifying, measuring and analyzing the costs for DA activities and data improvements in order to cost justify DA.

- Manage strategic data planning activities and the configuration management of the Strategic Data Plan.

- Manage the function of maintaining the Enterprise Data Model, providing data modeling, data conflict resolution and model integration support to project teams.

- Manage the function of providing data analysis support to ensure the proper translation of logical data models to physical database design and that data elements within existing automated systems are analyzed and linked properly to standard data elements.

- Manage the identification of functional specifications, development and maintenance of the Information Resource Dictionary Systems (IRDS). Manage the functions of populating and administering the IRDS.

- Manage the function of defining, measuring, assessing and continuously improving data and metadata quality.

- Manage the function of protecting the data and metadata by developing data security standards and procedures and adhering to data security classification and Automated Data Processing (ADP) security rules and regulations.

- Manage the development of standard techniques and support for the conversion and transition of data from legacy and manual systems to new automated information systems; as well as the movement of automated data between incompatible automated information systems.

- Keep abreast of new technologies, philosophies, and techniques for DA and Information Technology by researching, training, and consulting. Communicate with and provide DA education for staff, Database Administrators, data custodians, manager, end users, project managers, and application developers.

- Supervise the DA Staff, establishing position descriptions and performance standards.

PART 2: DATA ADMINISTRATION ACTIVITIES

Section 7. Data Standardization

7.1. Introduction

This chapter is devoted to data standards and data standardization. Data standards consist of specifications, goals, objectives, and guidance related to an organization's data resources. They are addressed through a data standards program such as that shown in Figure 7-1.

Data standardization is the process of moving from a current data management environment to one characterized by an established data standards program and strong data discipline. Disciplined data are data that are subject to well-defined and regularly practiced methods, procedures and enforcement mechanisms.

This first subsection provides an overview of a data standards program, including definitions, goals and objectives, a brief description of the standardization process, and comments on the relationship between data standards and the software life cycle.

The second subsection presents the framework for data representation, the third subsection addresses legacy data, and the fourth subsection discusses the concept of linkages.

7.1.1. Data Standardization Goals and Objectives

A number of benefits accrue from establishing a strong data standards program. They include greater sharing of data, higher quality data, improved data consistency, increased data integration, better data synchronization, more secure data, better understanding of data, improved documentation of an organization's information resources, better control of data redundancy, and better control over data updates.

Most of these benefits are interdependent. For example, improved data consistency and synchronization promote increased data integration and greater sharing of data.

7.1.2. Data Standards Program Components

The goal of a data standards program is to cost effectively manage an organization's information resources at the data element level of detail. The data standards program shown in Figure 7-1: Data Standardization Program Components, is designed to achieve this goal over the long term. The following paragraphs briefly discuss each component.

Data Models. Data models are used to identify and structure an organization's information requirements and related business rules. Typically there are three levels of data models. An entity-relationship diagram overview shows the major entities and relationships for a business area. A key-based data model describes the major data structures for a business area in third normal form. And a fully attributed model includes all of a business area's entities, attributes, relationships, and integrity rules. Items in the fully attributed model form the basis for the content of standard data elements.

Framework for Data Representation Standards. The framework for data representation standards is a series of interrelated standards that provides guidance on data formulation and maintenance. It provides the structure for standard data elements.

Standard Data Elements. These are the heart of a data standards program. They accurately, compactly, and flexibly encapsulate an organization's information resources.

Legacy Data Elements. These data elements do not conform to the guidance in the framework for data representation standards. Most approaches to data standards assume that they will disappear over time. In fact, for a variety of reasons, they tend to persist. The approach described here accommodates their persistence through linkages with standard data elements.

Standards-based Application Data Elements. These data elements are used in applications that are developed based on standard data elements. Most approaches accept some

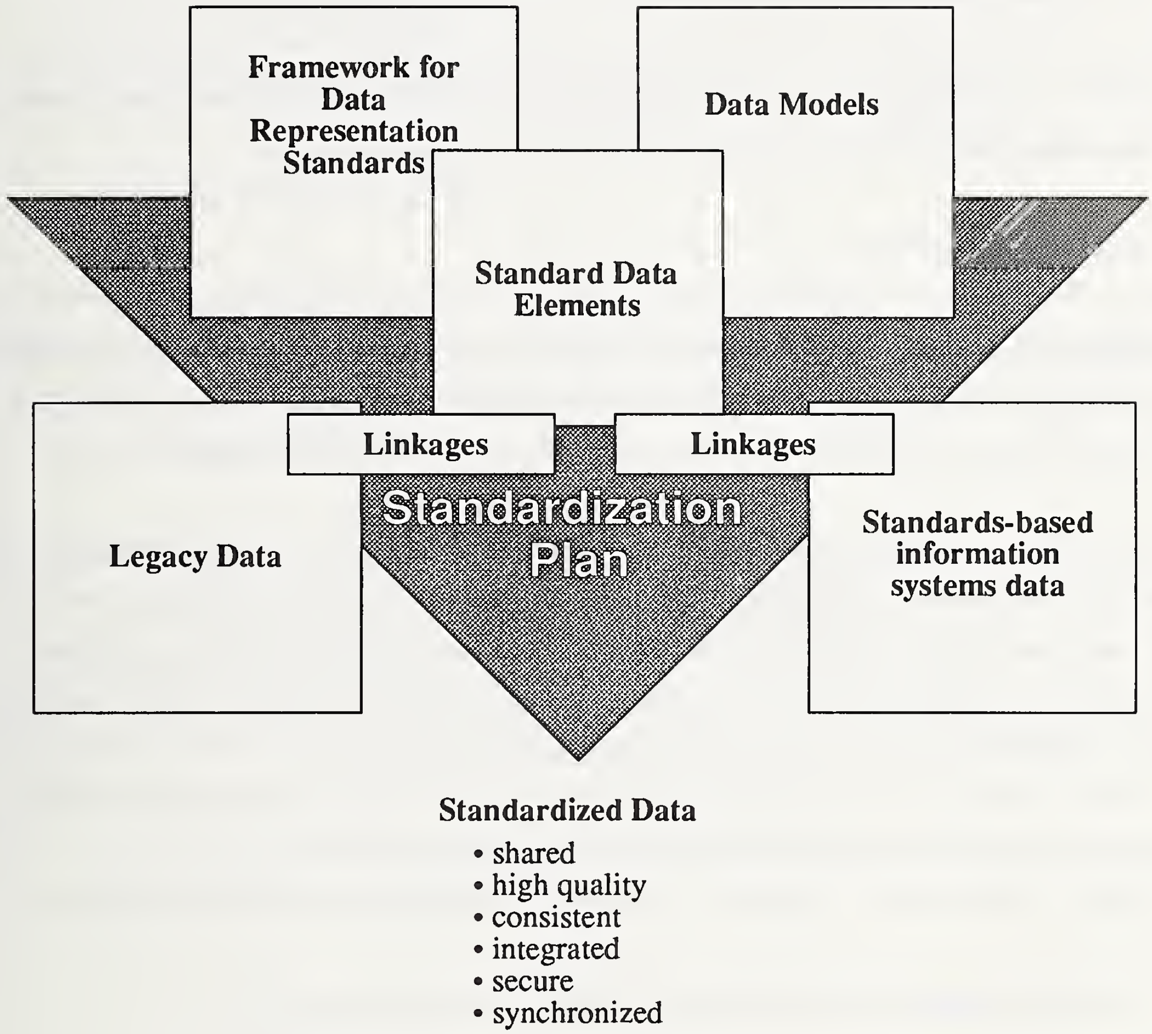


Figure 7-1: Data Standardization Program Components.

deviations from standard data elements for such reasons as performance tuning and space savings. If these deviations are not well specified, however, the link between application data elements and the data requirements and business rules specified in data models is broken. Linkages are a key to maintaining this connection.

Linkages. Linkages connect standard data elements with both legacy data elements and standards-based application data elements. They are discussed in detail in Section 7.3.

7.1.3. Disciplined Data

Disciplined data are data that are subject to well-defined and regularly practiced methods, procedures and enforcement mechanisms. Both the framework for data representation standards discussed in Section 7.2 and the linkages discussed in Section 7.3 are important to data discipline.

Data representation standards specify in detail methods and procedures for developing standard data elements based on items in a data model. Linkages are critical to program scope and to enforcement. Through linkages, legacy data is no longer outside the scope of a data standards program. And linkages prevent standards-based application data elements from drifting away from their standards base - they remain subject to standards enforcement.

Good data discipline promotes greater sharing of data, higher quality data, improved data consistency, increased data integration, better data synchronization, and more secure data.

7.1.4. Data Standards and the Software Life Cycle

Data standards are closely related to the software life cycle in a number of respects. They are closely tied to identification and specification of data requirements in the requirements specification stage. They are used to link standardized data with data in both legacy databases and standards-based databases. In addition, data standards can have an indirect effect on software life cycles through their support for subject area databases.

Yet there are other aspects of data standards that establish it as an infrastructural activity that is strongly separate from individual software life cycles. In order to promote data standardization, integration, and to minimize redundancy, a data standards program must provide guidance and direction across information systems and their life cycles.

An organization's commitment to the goals of a data standards program is reflected in its commitment to the infrastructure that supports, but is separate from, the software life cycle. A good description of one overall data management infrastructure is presented in [WAHL92].

7.2. Data Representation Standards

Consistent and precise methods and structures are necessary to standardize the representation of data for storage, processing and transmission. A series of interrelated standards to provide guidance on data formulation and maintenance will furnish these methods and structures that allow data commonality for data interchange.

A framework upon which to organize data representation standards is essential for transfer of information both within organizations and between national and international partners in information interchange. The concept of the **open systems environment** (OSE) has provided a set of standards addressing hardware, software, and communications; the international standards community has recognized the need to develop a set of data representation standards to assure compatible information interchange spanning localized and distributed national and international networks.

Among the goals of this framework are to provide:

- a vehicle that expedites the open interchange of data within the open systems environment;
- guidance to IRM professionals and data managers on the design and structure of data;
- a supporting structure that will facilitate the migration of current systems.

7.2.1. Data Hierarchy Reference Model

The central question of representation harmonization is: how can a concept be represented in a standardized form intelligible to all parties in the exchange?

The **data hierarchy reference model** describes a three level schema identifying the phases of data element generation. These levels are:

- Conceptual Data Level
- Generic Data Level
- Application Data Level

The conceptual data level includes **data element concepts** derived from a set of high level general concepts (ideas) by application of rules governing structure and context. Data element concepts are classified, organized, and registered in a classification system known as the *Data Element Concept Taxonomy* where they have constraining influences on the attributes of related data element concepts.

Data element concepts are the basis for development of **generic data elements** at the generic data level. Each generic data element is assigned a set of approved attributes, some of which are derived from its related data element concept. Other attributes are added to make the data element concept more specific. Generic data elements are intended to serve as a reference source for application data elements [NEWT91].

The third level of the reference model represents **application data elements** used in actual systems. They are derived from generic data elements by assignment of context-specific attributes and attribute values. Although not all application elements must be based on generic elements, the set of attributes of an application data element must not conflict with those of its related generic data element; in addition, it will require attributes related to the operational environment.

As data elements are developed, they are designed (i.e., defined, named, and attributed) in relation to their level in the framework of standards. Accordingly, data elements developed independently can achieve identical representation, allowing common data interchange and common understanding.

7.2.2. Classification of Data Elements

The process of developing data elements from general concepts begins by the selection of those general concepts which are of interest to the organization and amenable to representation by information processing systems. A concept can be selected by answering the questions: "what is it?" and "how is it represented?" A method to organize this process must result in a set of **class words** which are:

> complete - taken together, the set represents all information concepts of interest to the organization; and
>
> discrete - the definition of each does not overlap the definition of any other.

This requirement is satisfied by describing a taxonomic profile of data elements from concepts through generics. The dependencies of the different logical levels are detailed and specific constraints and relationships such as inheritance of mandatory attributes across levels are imposed.

7.2.3. Basic Attributes of Data Elements

Attribute standardization involves the specification of a standard set of attributes, and their allowable value ranges, independent of the application areas of data elements, tools, and implementation in a dictionary. This is essential to describing a data element completely for use in a variety of functions.

Basic attributes are valid for all data elements, including all levels of abstraction described earlier in this section. They are independent of any function of a data element, such as qualifier, indicator, composite; the classification system; any methodology for design of information processing systems or data interchange messages; and any application environment.

There are four categories of attributes. The first is **definitional**. These attributes describe the semantics and associated relational aspects of a data element. Examples of these attributes include NAME, ABBREVIATED NAME, DEFINITION, DESCRIPTION, LANGUAGE, QUALIFIER OF, QUALIFIED BY, CLASSIFICATION, and PERMISSIBLE INSTANCES.

The second category is **representational**. These attributes correspond to the external level in the 3-schema architecture. It includes REPRESENTATION CATEGORY, FORM OF REPRESENTATION, TYPE OF CHARACTERS, MAXIMUM NUMBER OF CHARACTERS, MINIMUM NUMBER OF CHARACTERS, LAYOUT OF REPRESENTATION, VALIDITY CATEGORY FOR USAGE, and DATA ITEM REPRESENTATIONS.

Attributes categorized as **administrative** record management and control aspects of data elements. These include UNIQUE IDENTIFIER, NAME OF RESPONSIBLE ORGANIZATION, DATE OF APPROVAL, DATE OF LATEST MODIFICATION, STATUS, NAME OF SUBMITTING ORGANIZATION, DATE OF RECEIPT, CHANGE REQUEST NUMBER, PROTECTION CATEGORY, and COMMENTS/REMARKS.

The fourth category is **relational attributes**. These include CROSS REFERENCES, KEYWORDS, ALIASES, and DATA CONCATENATIONS.

7.2.4. Data Definition

Data definition must be a part of any set of standards if consistency in data representation is to be achieved. Since the definition of a data entity can only be expressed as an unformatted block of text in natural language, a standard for data definition must consist not only as of a set of rules (testable against a set of syntactic and lexical conditions) but also as guidelines (more subjective semantic constraints) to be applied against that text.

7.2.4.1. Rules

A data definition shall:

- be unique (within any dictionary in which it appears)
- be stated in the singular
- state what the concept is, not what it is not
- be stated as a descriptive phrase or sentence
- expand acronyms on the first occurrance
- contain only commonly understood abbreviations
- be expressed without embedding definitions of other terms

7.2.4.2. Guidelines

A data definition **should**:

- state the essential meaning of the concept
- be precise and unambiguous
- be concise
- be able to stand alone
- be expressed without embedding rationale, functional usage, domain information, or procedural information
- avoid circular reasoning

7.2.5. Naming Principles for Data Elements

Names are usually the primary means by which human users of data identify and interact with data elements. Names must be clear, brief, rule-conformant, and free of physical context. They are formulated according to rules which are independent of any specific natural language syntax. They imply only concepts which are represented in the definition; ideally, they contain all concepts which are represented in the definition [NEWT88].

A good naming convention reveals ill-formed data by limiting the profusion of modifiers often used to describe a non-atomic element. It reveals coupled elements in which data has been confused with metadata. It is a tool by which information engineering can be advanced.

As naming conventions are established independently from any natural language syntax, names can be customized to suit a particular industry or special interest group, yet be mapped to a meaning-free code or number to facilitate international data exchange.

Naming principles are described in general terms with examples furnished. Rules are derived from the principles by which standard names are developed. These rules form a naming convention. Syntax, semantics and lexical rules vary by organizations such as corporations or standards-setting bodies for business areas.

Data element names are formed of **components**, each assigned meaning (semantics) and relative or absolute position (syntax) within a name. They are subject to lexical rules. They may or may not be delimited by a separator symbol. The set or range of values of each component is rigorously controlled by an authority, e.g., a data administrator within a corporation or an approving committee for an international business area naming standard.

Semantic rules convey meaning through logical reference.

Semantic content of names:

- Components: **class terms** identify the data class, or category, to which a data element belongs; **prime terms** place application data elements within the logical context of the information model; **modifiers** restrict the meaning of other components.

- Separators: delimit components and may or may not be assigned semantic meaning.

Syntax rules relate components in a consistent, specified order.

A naming convention specifies the arrangement of components within the name. This may be <u>relative</u> or <u>absolute</u>.

Relative arrangement specifies components in terms of other components, e.g., a modifier must occur before the component being modified.

Absolute arrangement specifies a fixed occurrence of the component, e.g., that the class term is always the last component in the name.

Lexical rules reduce redundancy and increase precision.

Rules concerning allowed and disallowed words in components, synonyms, abbreviations, component length, etc. are included.

Communication of the rules by which standardized names are formed facilitates data transfer between and within organizations.

7.3. Transitioning to Standard Data

This section describes a process for transitioning from current ("legacy") data representations to a standardized data environment. It begins by describing a process for gathering information on legacy data and then presents a method for linking legacy data with standard data elements.

7.3.1. Legacy Data Survey

In many organizations, there is little centrally available, detailed information on data resources such as systems files and databases. Collecting, maintaining, and analyzing this information can be a time consuming and difficult task. Yet it is vital to know what is in use in order to transition relatively smoothly to a more standardized data environment. This section addresses various activities and concerns associated with conducting a survey of current data.

A legacy data survey project must begin with a clear statement of objectives, and a detailed work plan. This is critical for at least two reasons. First, the project will involve interaction with many people in the organization. If this interaction is not clear and direct, the project may lose support. Second, the project will generate a great deal of information about data elements (metadata). This information must be well organized to promote accurate and timely recording and analysis. Other benefits of a good work plan are faster data collection, and a shallower learning curve for support staff.

Gathering information on operational, legacy data also serves to validate data models. It may capture critical elements of information that were not captured during data modeling for any of a number of reasons. For example, employee turnover can result in a loss of institutional memory. Another possibility is that certain key employees may not have been available to participate in data modeling sessions.

A legacy data work plan should address each of the areas discussed below.

a. *Scope.* Specify information systems to be surveyed. Group information systems by functional area (e.g., financial management, personnel). Put the smaller, relatively simple, and best organized and documented systems at the top of each functional area list. For the first couple of systems, this allows the project team to work through the process fully with relatively easy targets, and to adjust the process before moving on to more challenging sets of systems and databases. It will also help in getting the project team up to speed with technical vocabulary as they move from one functional area to another.

b. *Data Requirements.* Specify data, or metadata, requirements in detail. One objective is to tax each information system only once. A standardization effort will lose support if the project team repeatedly requests input from an information systems maintenance team or users. Critical data element attributes include name(s), information system name, information system owner, definition, representation(s), ownership (individual, position, or responsible office), update cycle, security and access requirements, and valid values or validity rules.

c. *Data Gathering Process.* Specify how the metadata will be obtained. If possible, the project team will gather as much of the attribute information as possible from existing data dictionaries, databases, files, record layout descriptions and other system documentation. Typically, the quality of system documentation will vary widely from system to system. An early review of system documentation will help in planning and scheduling.

Generally, there will need to be follow-up with system or functional experts in order to gather additional information that is unavailable or unclear from system documents. It is a good idea to specify in the work plan a point of contact for each database or system to be surveyed. All significant meetings between the functional or system representatives and the project team should be documented. A good rule of thumb for meetings is to prepare well and present well organized material in order to minimize the time and effort required of representatives of target systems or databases.

d. *Review of Results.* Specify a review process to get functional approval of results. Responsibility for the information collected is critical to acceptance and maintenance of the data element attribute information. The key to getting "sign-off" by an information system owner is a review process. Although it would be helpful if a system representative could review all of the metadata collected for a system or database, this is not often likely to happen. A good alternative is to provide a description of the data sources used for various pieces of metadata. If the description is sufficiently convincing, the system representative will bless the survey results with a stamp of approval.

e. *Storage and Processing of Results.* A data dictionary is the best tool to use to store and process survey results. An early step in the project is to prepare the dictionary to store the data to be developed through the survey. Powerful import facilities can promote electronic update of the dictionary. Flexible processing capabilities can be very useful in identifying outstanding attribute information deficiencies and in populating the dictionary using rule-based routines whenever possible (e.g., for populating the Information System Name for all data elements in a given system).

f. *Consistency Checks.* A legacy data survey provides the very detailed material needed to assess the consistency of legacy data elements across systems. Variations in any aspect of a data element's metadata can adversely affect data sharing and make it more difficult to integrate data. Typically these differences are passed on only haphazardly from generation to generation of data processing professionals or users. Clear statements of distinctions at the attribute level are required for data integration.

Consistency checking can be performed first within a functional area, and then across functional area.

During this step, a list of those data elements that are most commonly used across functional areas can be compiled. These are the data elements that are prime candidates for central quality control and are the keys to data sharing.

7.3.2. Linkages

This section presents a method [HUFF91] for specifying associations between standard and non-standard data elements. Non-standard data elements can be legacy data elements, as well as standards-based data elements that have been modified for physical implementation.

Maintaining linkages between standard data and non-standard data is crucial for data discipline, as well as for tying together physical and logical data models as shown in Figure 7-2. Linkages with legacy data are required in order to bring them into the scope of a data standards program. The typical assumption that legacy data will disappear over time is generally not borne out. Legacy data elements can be well-known, succinct and convenient. In addition, they are often at the heart of key systems that cannot afford to undergo the risk and expense of overhaul required to incorporate data structures based on standard data elements.

Associations between non-standard data elements and standard data elements are often not straight-forward, yet most tools only provide for one-to-one associations between the two. In fact, associations between data elements are characteristically many-to-many rather than one-to-one. Three basic types of associations between data elements are described below:

- **Chained**. An ordered set of data elements linked together (e.g., positions 1-3 describe concept "A", positions 4-5 describe concept "B", etc.). Many dates can be viewed as chained representations of month, day and year.

- **Coupled**. A data element carrying multiple concepts through its assigned name, or its allowable set of coded values. For example, the data element "Four Person Passenger Car Count" not only describes the number of cars, but also describes the type of car (Passenger) and the seating capacity (Four).

- **Multi-purpose**. A data element with multiple uses or definitions. The context of the value contained in the data element changes based on what is described by the record. For example, the value "5" in the data element Vehicle Capacity might mean "five passengers" for a passenger car, but mean "five tons" for a pickup truck.

Types of associations can be captured through a relatively simple set of relational tables that capture the standard data element name, non-standard data element name, the type of association, and attributes related to each type of association.

A **data elements** table can specify all data elements, standards and non-standard. It can be limited to only those data elements that participate in linkages. The key is the data element name or other unique data element identifier.

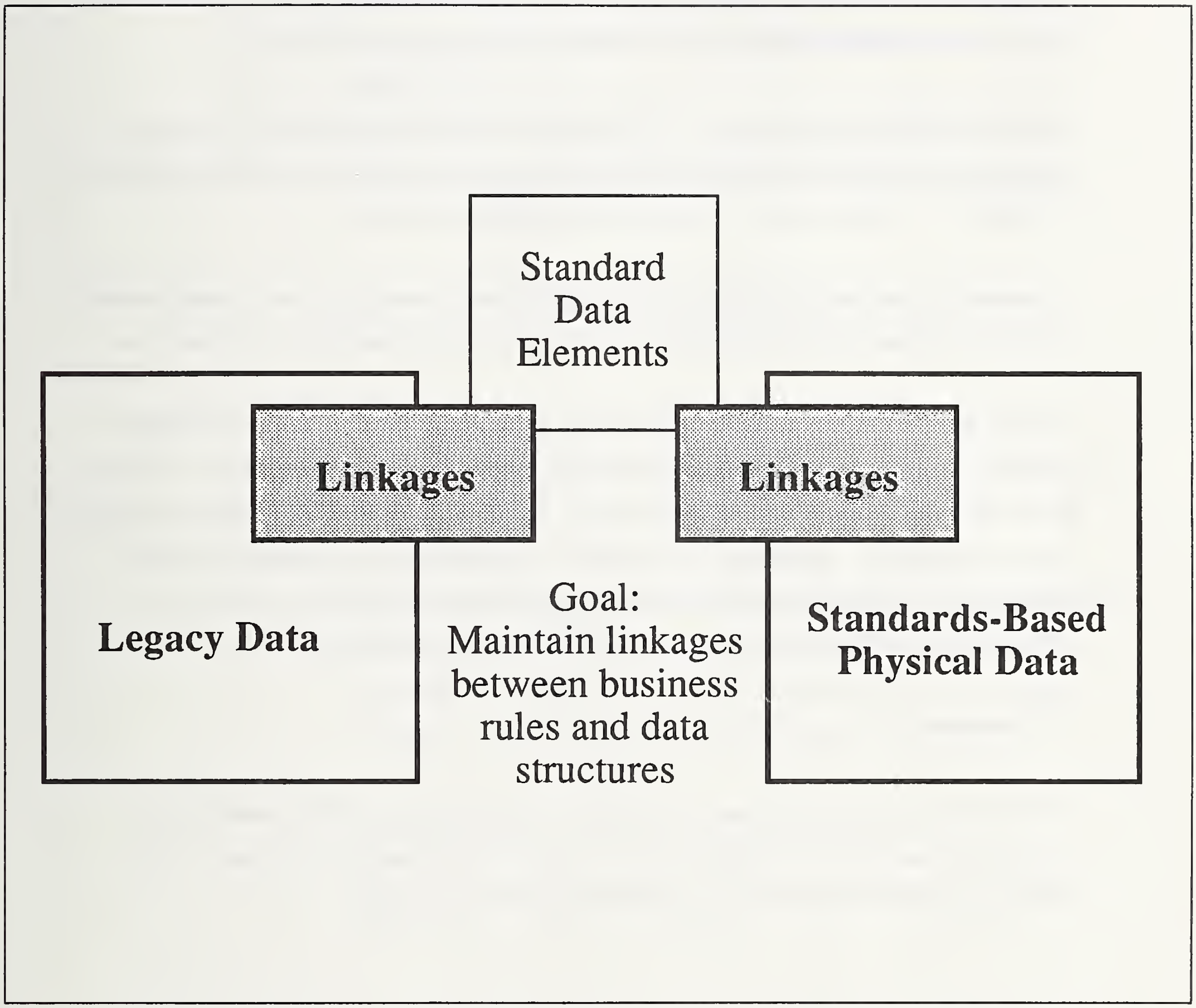


Figure 7-2: Linkages.

An **element association** table contains a unique identifier for each association and a code that identifies the type of association.

An **association members** table identifies the data elements that participate in each of the associations in the element association table and indicates the role each plays in the association.

Association attribute tables capture attributes that further describe an association. There can be separate association tables for each type of associate as each type of association can have its own set of attributes.

An **association type master** table lists all of the types of association. In the basic model specified above, there will be three rows in this table - one each for chained, coupled, and multipurpose.

These are the basic building blocks of data element associations. They can be used to specify complex associations between data elements through layering or nesting. For example, a chained data element may consist of four component parts. One of these may itself be coupled, and another may be multi-purpose. In addition, it is extensible. For example, it is a simple matter to add another type of association to the basic set of three specified above by adding one row to the association type master table, and creating an additional association attribute table.

7.4. Summary

Two data standards program components critical to disciplined data are a framework for data representation standards, and linkages between standard and non-standard data elements. This section has discussed methods and procedures related to each component.

7.5. References

[HUFF91] Hufford, Duane, et. al., *Logical Data Model of Data Element Associations for Mapping Element Problems*, American Management Systems, Inc., April, 1991.

[NEWT88] Newton, Judith J., *Guide on Data Entity Naming Conventions*, Special Publication 500-149, Nat. Inst. Stand. Technol., 1988.

[NEWT91] Newton, Judith J., "Developing and Applying Data Entity Naming Conventions," *Data Resource Management*, Vol. 2, No. 4, Fall 1991.

[WAHL92] Wahl, Dan and Hufford, Duane, *"A Case Study: Implementing and Operating an Atomic Database,"* Data Resource Management, Vol. 3, No. 2, Spring 1992.

NOTE: The data representation standards described in this chapter are under development by the International Standards Organization Standing Committee 14 (ISO/IEC JTC 1/SC 14) in the form of a set of working documents. These are not yet available for general distribution.

Section 8. Automated Tools - Selection and Evaluation

8.1. The CASE Concept

Computer Aided System Engineering (CASE) is the concept that software systems can be engineered in much the same way as hardware systems. The acronym is also defined as Computer Assisted Software Engineering. The concept started with structured programming in which rigorous discipline is applied to the design of a single program and then evolved to a structured approach to the whole life cycle of software and systems development. This approach requires planning and consideration of the interactions and dependencies of a set of software applications as a whole before the development of any discrete part. The complications of studying all of the interactions and dependencies led to the development of automated tools that permit the developers to manage and use the information.

The data processing world is in a continual state of rapid evolution of hardware and software capabilities. Most of the tools available on the market today are designed to work with relational or network databases and third or fourth generation languages. However, there is also a strong movement toward implementing object-oriented environments along with appropriate tools, which means they will soon be available.

8.2. Types of Tools

CASE tools have been developed that can support every stage of the life cycle of an automated data processing system, from initial requirements determination to maintenance of operational systems to reengineering of older systems. Most tools can be used for multiple stages of the life cycle, although few are available that span the entire life cycle from initial analysis and design to code generation and reengineering. The tools provided typically have use in more than one life-cycle phase. A structured diagram can be used to determine initial data requirements or to model requirements for additions and revisions of an existing system. A code generator can be used for initial system development or to produce additional modules. Software packages which contain modules for all phases of system engineering are called Integrated CASE (I-CASE) packages.

8.2.1. Planning Tools

Models

Models are graphic depictions of information and relationships. Environmental models, sometimes called context diagrams, show the relationship of an organization to the rest of the world. Business models show the relationships of parts of the organization. These types of models are used to capture the business concerns in a concise and comprehensive way to aid in the planning of information system activities.

8.2.2. Analysis Tools

Automated analytical methods have been developed for use in the design of software systems. These methods can be used to design new systems and in the analysis of existing systems to determine requirements for enhancements. Each method generally has its own convention for graphic representation of the information and a somewhat different approach to the analysis of data. Several commercial methods are available; there are also unique proprietary methods developed by some companies . Common analysis tools are models and structured diagrams. The differences in both the approach to the analysis of data and notation must be noted when comparing information or charts prepared with different packages. A critical point is that CASE tools have different levels of cross-checking and integrity enforcement. Some tools automatically check that data definitions are consistent across the different models and diagrams and from higher to lower levels of diagrams. Others have little or no checking.

8.2.2.1. Models

The environmental (or context) and business models used in planning information systems activities are also useful in the analysis phase. In addition there are models used specifically in systems analysis. Some show how data items relate to one another in a logical model, or how they are implemented in a database in a physical model. Others attempt to represent the real world in a system model. Some employ object-oriented methods.

8.2.2.2. Structured Diagrams

Structured diagrams are graphic depictions that focus on data from several aspects. Three commonly used structured diagrams are:

- Data Flow Diagrams (DFD) are a method of modeling the movement of data from process to process. A process is an activity, automated or not, that causes some change to occur in a data element. Storage of data when it is not in the process or moving between processes is noted as well. The processes are broad in the first model as are the data elements. Iterative decomposition of the processes and data flows eventually results in atomic data elements that can be defined and used to populate a data dictionary (see sec. 8.4).

- Entity Relationship Diagrams (ERD) are a method of depicting data as static, definable units that possess both distinctive qualities called attributes and relationships to other data items. The linking of explicitly defined data elements through their relationships, such as "part of," or "composed of," can lead to precise definitions of data elements. These definitions with the attributes can lead to population of the data dictionary with the values of elements as well as the relationships that enable definitions of logical groups and key attributes. One ERD method is that of Peter Chen, the principal developer of ERD analysis. Oracle Case*Designer, Yourdon, and many other case tools also offer ERD analysis.

- State Transition Diagrams (STD) are a method of depicting the sequence of changes to data and the events that trigger the changes. These models show stimulus (or input) causing action and resulting in a response (output). This approach is important in modeling of real-time systems where sequence of events and the state of the system at any given point is critical.

8.2.3. Development Tools

Development tools automate to varying degrees the creation of applications on the basis of data and process definitions identified in the analysis phase. The tools may use a text oriented, command line approach or a graphic user interface (GUI). These tools are:

- screen generation of input and output data screens
- code generation of applications from natural language descriptions of desired processes
- report generation based on defined data items
- database schema generation from entities and relationships
- data dictionary generation from ERD's and DFD's
- documentation generation, and
- data Repository generation from multiple data dictionaries.

8.2.4. Change Management Tools

Managers of automated systems need tools to facilitate the management and control of change to the system hardware and software. The dispersed nature of systems developed for personal computers, for use over local area networks, and in distributed environments can lead to chaos without rigorous change management. If the computer operations are a contract operation, then the question of managerial approval and scheduling of the installation of changes are critical. Some CASE packages include the tools to manage change, particularly when the package also includes application code generation tools. CASE packages that are aimed at analysis may have limited change management capability.

Acquisition plans may include configuration management by contractors during the development phase. The procured systems should include the tools to facilitate the management and control of change to the system hardware and software after turnover to the customer. For large minicomputers, as well as mainframes, the volume and variety of operating system changes, hardware fixes and upgrades, and application software updates can quickly become unmanageable.

Change control tools can be acquired with various levels of capability to suit different environments. Users who develop new major systems involving the safety of humans, for example, would require the most rigorous configuration management. Users who modify an older system might be more interested in the capabilities for code comparison and association of proposed changes to other modules. Some of the functions that should be required to manage data prudently are the capability to:

- create and maintain a schedule of changes including such milestones as the date that the testing of the change will begin, the testing window, the date that installation of change will be complete, and the date that the change can be backed out if it fails in production mode;

- associate changes with the required management approval for each configured item;

- associate changes with all affected hardware and software modules;

- monitor the receipt of changes from originator, approval, testing, implementation, back-out, and results;

- generate reports of changes scheduled and implemented by such parameters as originator, type, affected configuration item, or date range;

- maintain permanent audit trails of changes;

- manage a library and cross reference of source code to executable load modules; and

- perform comparisons of source code modules to identify changes.

8.2.5. Reengineering Tools

Reengineering is the concept of taking existing applications and analyzing them for logic, process, and data manipulation, in to create new applications that do the same work more efficiently. Reengineering tools can be used to analyze existing applications code for unexecuted paths, bottlenecks, and erroneous logic so that the code can be modified and reused.

Reverse engineering is the term used to describe the use of tools that break down existing database schemas or applications code into logical elements for analysis and conversion to another database schema or another language and produce new documentation. When reverse engineering is cited as a capability of a CASE tool, it should specify the particular environments in which it is effective and the degree of automation that can be reasonably expected.

Reengineering and reverse engineering are fairly new concepts with little history available on tools providing these capabilities. The principal target for a reengineering tool is the large, critical application that has inadequate documentation and is, therefore, not easily approached by the analytical method.

8.2.6. Data Management Tools

There are potentially three levels at which data can be managed that affects the choice of an appropriate tool. The first level is at the working level of the database administrator (DBA) who is responsible for a discrete collection of data, their definition, accessibility, integrity and physical management. The primary tool at this level is the data dictionary, which is part of or interfaced to the database management system (DBMS) package for which the DBA is responsible.

The second level is the organizational or corporate data level. This is the level of the data administrator (DA) who is responsible for the data of the whole organization. The DA is charged with a wider view of data definitions, accessibility, and integrity and may use an organization-wide data dictionary that interfaces to or includes the common information from each individual data dictionary in the organization.

The third and broadest level is called the repository level. A corporate repository is at the level of a chief information officer (CIO) or information resources management (IRM) official who would be responsible for not only data administration but also the organizational business and information models, inventories of information resources, and other areas declared to be part of IRM by the given organization. The repository concept of IBM, for example, would encompass a broad range of information which is used to integrate activities throughout the application package known as AD/Cycle. The repository can be used to manage not only the data but also the life cycle development activities for the data and applications for which the repository holds descriptions, as described in chapter 10.

8.3. Model and Structured Diagram Management

Data management begins with the first steps of analysis and design of the software system. The modeling and structured analysis methods and tools that can be used to analyze and design a system from the ground up are the same used for management of data for the life of a system. The DA and DBA continue to maintain the models and structured diagrams as systems and requirements change over the life of the system. DA's and DBA's must establish the continuing flow of information that will enable them to know when changes are necessary. Organizational, product line, mission, and technology changes all can require modification or update to business models, logical models, physical models, data flow diagrams, entity-relationship diagrams, or state-transition diagrams.

8.4. Data Dictionary Management

The principal tool that is used in the day-to-day management of data administration is the data dictionary. The data dictionary is a collection of definitions of the data elements that are the core of the information assets of any organization. The data dictionary is developed from logical and physical models of the data, which are produced by one or more structured analysis methods. Maintenance of the data dictionary through the life cycle of the system is critical to effective data management. The means of maintaining the data dictionary depends heavily on the particular software implementation. There are two types of data dictionaries, passive and active.

The passive data dictionary is maintained by the direct intervention of the DA or DBA. The passive data dictionary is integral to the operation of the DBMS, but does not react to changes to the data format or database structure automatically. The passive data dictionary is free-standing. The user must query to get the information and the DA must enter changes in definitions of data elements or relationships. If the data dictionary is not kept in synchronization with the database operations then integrity rules may not be correctly applied, which can compromise the quality of the data. The potential for inconsistency and redundancy of data storage is great. However, the passive data dictionary does not affect the performance of the production systems or halt operations if it crashes.

The active data dictionary is the means by which data quality management tools are made available to the data administrator. The tools include checking data declarations in applications code to the definition in the data dictionary for correctness of type, format, and value range. Some tools can be used at compile time and others at run time. An active data dictionary is integrated into the DBMS operation in such a way that changes to the definitions of a data format or structure are automatically checked through the data dictionary for consistency and waterfall effects. The cross-checking and integrity rule updating are done without direct manual intervention. However, an error in the dictionary processing can prevent or delay the execution of applications.

Planning for the establishment of a data dictionary should include an analysis of the active versus passive in performance and overhead. There is generally a machine performance overhead cost associated with active data dictionary maintenance that can become a performance bottleneck in some environments. There is the personnel resource overhead associated with maintenance of the passive data dictionary. The type of processing environment and the availability of trained personnel are factors to consider.

8.5. CASE Standards

Numerous philosophies of modeling and structured analysis have been implemented in CASE tool packages. In addition to having a different approach to the analysis of data, the developer of each method generally has a different convention for graphic representation of the information. The differences in both the analytical approach and notation are sufficient to confuse the issue if one attempted to compare information or charts prepared with different packages - a strong argument for choosing one package to be the standard for an organization. There are ongoing efforts at the National Institute for Standards and Technology (NIST) to develop a reference model that can be used by the CASE companies to standardize at least part of their packages. This reference model is aimed at the problem of data interchange among different CASE packages because many organizations find that no one package satisfies all of their needs. In addition, the Electronic Industries Association (EIA) CASE Data Interchange Format (CDIF) Technical Committee is developing explicit standards to implement such interchanges.

The NIST standards for the Information Resource Dictionary System (IRDS) and Structured Query Language (SQL) have had an effect on the CASE packages offered. Some manufacturers are adapting their products to conform to these standards where applicable, for example, in generating forms that produce SQL queries or data dictionaries that conform to IRDS. Currently each CASE tool has an internal dictionary or repository for data pertaining to the models built by the tool. A services interface and some means to import and export data from each to the IRDS system will be necessary. Eventually a more integrated approach will be needed to simplify a multitool environment that accesses a single CASE repository and IRDS, which are tied closely together. Figure 8-1 illustrates both the current relationship between CASE tools and an IRDS and the prospective integrated approach.

8.6. Advanced Tools

The attempt to develop artificial intelligence (AI) in computer systems leads to the development of more advanced tools for a variety of functions. The progression from decision support systems that provide a rule-based approach for problem solving, to knowledge-based and expert systems

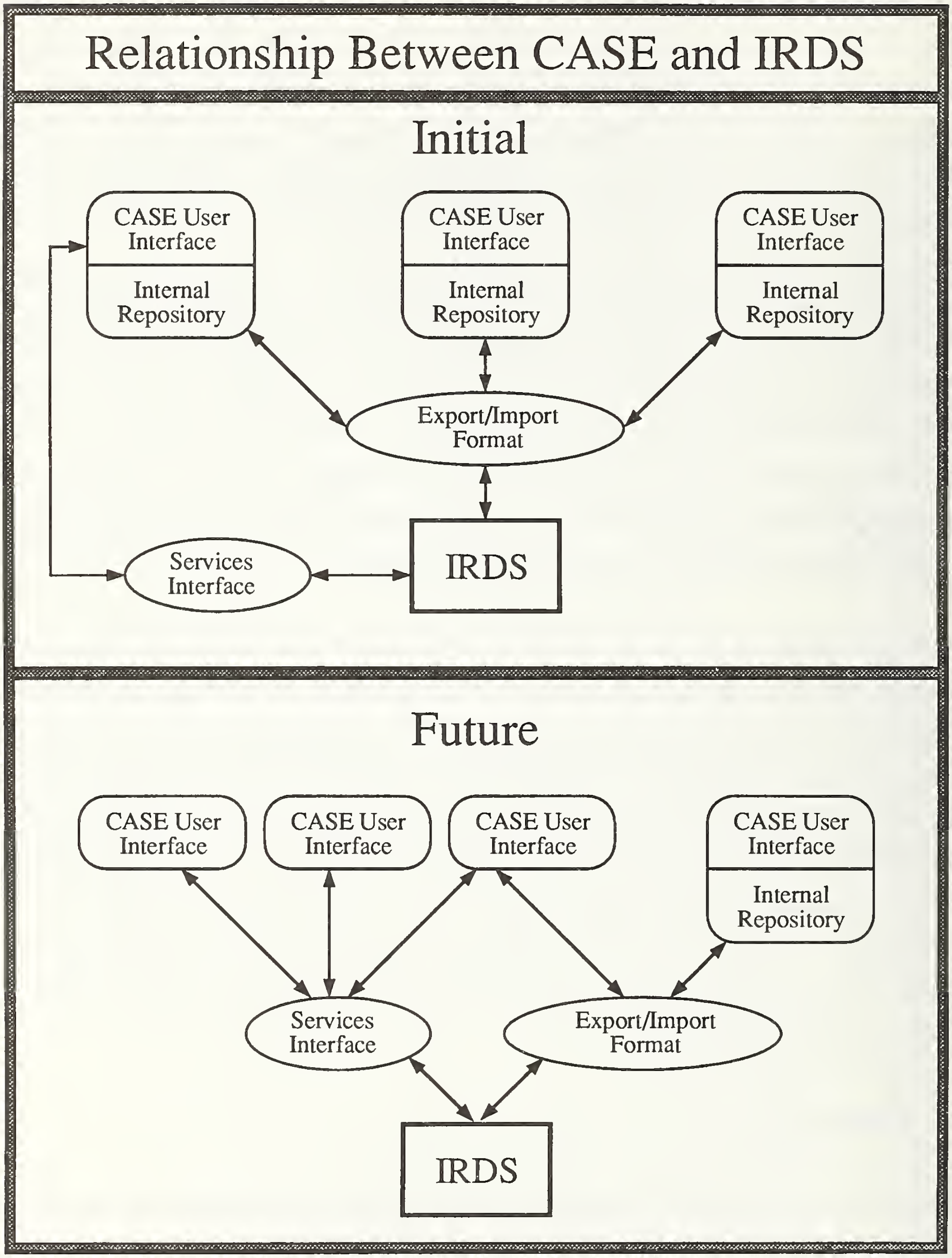


Figure 8-1: Relationship Between CASE and IRDS.

that add a layer of knowledge processing to the decision making capability, to wherever the advances in technology may lead is still in the formative stage.

8.7. Availability of Tools

The packages of software tools are available from a variety of software dealers as well as the original developers. The original developers range from computer manufacturing companies, to software development companies, to consulting firms who market the tools that they have developed for their own use. The most common method is to go directly to the developer of the package using the organization's standard acquisition method. A second method available to the U.S. Government is the General Services Administration software schedule procurement. It may be possible to find a package at an advantageous price, although the choice of packages may not include the user's first choice. A third method is to attach the tool package to a procurement of equipment as part of the initial delivery. The justification of an equipment buy can often include the tools that will be useful for making the best use of the equipment. One additional approach can be used if a software development contract is planned. A contract for development of software can require that the CASE tool that is used to develop that software be delivered to the organization along with a fully populated data dictionary as part of the system.

Regardless of procurement method, it is recommended that the purchaser request a demonstration of the capabilities of the prospective packages. Considerations of accessibility across local area networks, compatibility with more than one operating system environment, and platform flexibility may require the choice be made on more than the technical quality of the CASE functions themselves.

8.8. Unavailable Tools

The wide availability of software that performs literally thousands of chores in the data administration area sometimes obscures the fact that there are gaps not yet filled by tools currently available. The principal lack is in the area of quality control and quality management. A fully integrated active data dictionary can solve some of the quality problems, but sometimes at a price in performance. However, there is no good way to assess the effectiveness of the data administration function without tools that will help measure the improvements in data quality or the intangible benefits of better data in a low-redundancy environment. Tool packages that measure performance improvements when data management improves would also help the DA to demonstrate successes or to analyze failures. Statistical analysis packages and spreadsheet programs can, of course, perform some of the required services when the DA is a good statistician. However, application tools designed specifically for data administration quality assessment and quality improvement functions would be welcome.

8.9. Selection and Evaluation Criteria

The proliferation of personal computers with DBMS packages and the potential availability of innumerable CASE tools means that more people are going to establish CASE applications. The potential power of an application developed for a single user is great, but the potential to expend resources and have an unsatisfactory result is also great. The user of the tool can be an acknowledged expert in a technical area yet be relatively unsophisticated in the use of automated tools. Computer programmers and analysts also may have had little opportunity to develop expertise in this area. For this reason it is important to have some criteria with which to judge the potential success of a plan to acquire and use a CASE tool. The criteria can be used for formal contracted efforts or for in-house efforts, which are less formally documented. There are two sets of criteria presented here. The first group of criteria provide a way to look at the capabilities of the CASE tool. The second group of criteria provide a means to look at how the implementation is being approached.

8.9.1. CASE Tool System Criteria

These criteria provide a way to look at the capabilities of the tool under consideration, at a fairly high level.

8.9.1.1. Information

The information presentation of the proposed tool is critical to its success in operation. The key items are the accessibility of the information, its relevance to the user's needs, its timeliness, and its presentation form. In this case some specific questions to consider are:

- Is the information available to users in an easy-to-use presentation?
- Is there an online tutorial for new users?
- Is there an online help module?
- Can users perform their work efficiently?
- Is the command language fairly intuitive or will users require extensive training to make use of it?
- Is the command language consistent among modules, such as use of the same exit command every module, and so on.
- Does the proposed package include parameters to permit tuning the system at the user's discretion? For example, there should be an expert user mode so that explanatory menus can be turned off and commands to adjust default values as desired.

8.9.1.2. Control

A reasonable amount of control is necessary to monitor and regulate the processes of the system. The tool package must include sufficient levels of control to manage and protect the information holdings. Some specific questions are:

- Is there logging of all additions, modifications, or deletions of the data? Are usage histories kept?
- Are the backup and recovery procedures sufficient to recover from failures within the necessary time? How is work in progress recovered after soft or hard crashes?
- Is the package designed so that sensitive data can be identified for restricted access rights?
- Are error messages understandable or are they cryptic?
- Is there password protection at the user level in addition to any host system logon password?
- Is there any cross model integrity checking or enforcement, such as balancing data elements from entity relationship diagrams against those from the process flow models?
- Is there a data dictionary capability to help manage the data?

- Is there a capability to perform version control over the developed models? Is there a system library?
- Is there a means to control the use of the system by users? Is there a warning when a requested operation is so large that it would tie up the system? Are there time-out checks that bump inactive users off the system?

8.9.1.3. Efficiency

Efficiency means doing the necessary work with a minimum of waste. It can be expressed as a ratio of the output (effective work) compared with the input (resources required to perform it). Some factors to consider are:

- Are the processes sufficiently automated that the current staff can manage the current workload? (A bigger workload? A bigger workload with a smaller staff?)
- Are interfaces to other applications clean and easily accomplished? Are modifications to other applications needed for successful operations? If so, are the custodians of those applications willing to accommodate the modifications?
- Can additional models be added to the system easily? Is there flexibility in the applications to accommodate additional data types or new key fields?
- How difficult is it to edit and modify models, diagrams, and dictionary entries?
- If the tool is not an integral part of the DBMS, does it generate compatible data dictionary definitions or schemas?
- Are there batch file manipulation capabilities, such as loading and copying, that require minimal setup?
- Do menus allow experienced users to take shortcuts?

8.9.1.4. Performance

Initial performance evaluations can be made based on information supplied by manufacturers. It may be useful to look at existing similar implementations to estimate performance capabilities. (If the procedures of the organization permit testing of software prior to purchase, some manufactures will supply a package for evaluation purposes, free, in the hope that their package will be selected and subsequently purchased.) Performance includes throughput and response time. Some questions to consider are:

- What is the expected volume of data models, files, diagrams, and so on compared with the potential capability of the chosen tool?
- Are there any inherent limitations that would prevent processing the expected volume or type of tasks?
- Is there a potential conflict for devices or processing capacity in a shared environment?
- Have potential response times for queries been estimated by manufacturer for various sizes of models and databases?

8.9.1.5. Economy

The successful system is one that can be operated within the constraints of the resources. The cost for the level of service and the capacity provided need to be compared with the resources available to the process. The planned environment will dictate the nature of the resources to be evaluated. For example, a CASE tool that uses all of the hard disk capacity of a personal computer system is an unrealistic choice if there are other necessary applications already on that system. Some specific questions are:

- How much memory and disk space is required to support the tool?
- Does the design use dynamic allocation of space or require a set number of bytes, cylinders or packs? Providing excess capacity to allow immediate service of unpredicted workloads may cost more than allocating capacity as demand occurs. If extending space is too complex and time consuming it may pay to preallocate space generously.

- What are the operational costs? Can they be estimated or predicted? Are any software licenses involved? How are updates to licensed software obtained?
- If the tool package involves the use of a computer system controlled by another office, will there be continued support of any required software or hardware devices for the foreseeable future?
- If the planned tool is to be used as an interim solution could the users live with it longer than anticipated?

8.9.2. Implementation Criteria

Most organizations have formal rules pertaining to life cycle management documentation and approval procedures. The manager of a project should check to see if the intended user community and dollar value of the development effort require conformance to these procedures. However, there are other considerations that affect the success of the development effort whether or not a formal life cycle approach is used. Thorough planning can indicate most potential problems. Even in completely internal development efforts planning, and documentation of that planning, is a critical success factor.

- Has an assessment of the planned environment been made, including available resources, sophistication of users, constraints on use of CASE products?
- Is there documentation of the type of tasks to be performed by the tool? Have all of the candidate activities been considered?
- Has a feature analysis been planned to assess the suitability of the tool package for the intended application?
- Does the organization have a preferred methodology that the tool must support or a standard to which it must conform?
- Does the plan include descriptions of database administrative support that will be needed?
- Have demonstration copies been examined, or are there systems available elsewhere with similar uses that could be demonstrated?
- If this is a replacement of an existing tool, is there a plan for data transition?
- Is there a plan for accomplishing interfaces to any existing data dictionaries or databases that may be required? Are all data interfaces identified?

- Is the tool compliant with the GOSIP and POSIX standards for open and portable systems?
- Is there a schedule for the installation, testing, and data transition that takes into account available resources? Is the schedule within management constraints? Does the schedule include appropriate review and approval cycles?
- Have all development costs (staff hours, software costs, computer time, software licenses, maintenance costs, and hardware costs) been estimated?
- Have the communications requirements been analyzed? Will connections to a local area network or wide area network be required? Will the tool of choice operate over a network? Are there additional hardware or software items required to accomplish the connection?
- Has a cost benefit analysis been done? In some cases the tool must be acquired to support critical functions; in other cases the costs may be such that they outweigh the benefits. Has any analysis of other existing applications been made to determine if the functional capability can be acquired through some other means than a new acquisition?
- Has appropriate information on standards and procedures been collected, such as NIST guidelines?

8.9.3. Maintenance Effort Criteria

A significant factor in the use of any CASE tool, indeed any software product, is the level of effort required to keep the product operating properly. The maintenance of a commercial product should be more straightforward than it would be for a customized system, but can still pose problems. Some items to consider in selecting a tool package are:

- How and by whom is the software maintained? If in-house support is planned will there be sufficient support staff available to fix problems quickly? Are there users at night and on weekends? If maintenance of the software will be a contract effort, is the level of support appropriate to the operations of the organization?
- Is the tool supported by a vendor or the original developer? Is there a telephone help service available? What is the strength of the vendor?

- Is the product commercial-off-the-shelf (COTS), customized, developed in-house, or share-ware? Is the product standardized and stable or developmental and volatile? If stable, is it still extensible?
- Are there any operating system dependencies, restrictions, or limitations that would affect the correct operation of the software?
- Does the developer or vendor supply updates, corrections, and new versions to all customers?
- Are there multiple tools in use in the organization that require maintenance?

8.10. Summary

Although the range of CASE tools is broad and choosing the correct one a complex task, the most important thing is to start. It may be necessary to choose an inexpensive package as a learning tool for staff members to gain expertise and insight in the CASE process prior to selecting a standard tool for an organization. It may also be necessary to use more than one tool for different purposes. However, the degree of documentation and rigor imposed by the use of CASE tools can yield benefits to an organization that is in the early stage of CASE sophistication.

Section 9. Data Administration for Security

9.1. Introduction

As with Data Administration, the word *security* has multiple meanings and connotations, depending upon the context within which it is used. Again the first step involves identifying the major meaning clusters and their associated context(s), and then determining those clusters most relevant to the project at hand. Generally, the meanings of *security* can be divided into three major classifications, shown with examples in Table 9-1.

The word *security* is frequently synonymous with *protection*, either in the senses of active defense or of preventive constraints. By far, the most common cluster of meanings of *security* involve the concept of <u>physical protection</u>; this involves things like military or police forces, physical access control, terrorist protections, Acts of God protection, etc.

Table 9-1: SECURITY MEANINGS AND EXAMPLES

Physical Protection
Military/Police Forces
Physical Assess Control and Protection
Terrorist Protection
Acts of God Protection
Database Access Permissions and Protections
Data Back-up, Duplication, Archiving and Secondary Storage
Database Manipulation Permissions
Information Dissemination Control
National Security
Privacy
Confidentiality

Within the database and programming communities, *security* frequently means database access permissions and protections. These include activities like data back-up, data duplication, data archiving and secondary storage; they also include a variety of database manipulation permissions, such as VIEW, UPDATE, etc.

A third set of meanings of *security* involve information dissemination control. These involve data content issues of privacy, confidentiality, National Security, etc. This cluster of meanings must be addressed for most data sets, regardless of the data-source or the data-user communities.

9.2. Data Administration for Security Activities

This sub-section discusses some major data administration activities for security. It includes suggestions for some of the initial activities and starting points for others. It includes some traditional data administration activities and some that are traditional in IRM security analysis.

The following activities are discussed.

- Determining "value" of the confidentiality of the data's content
- Identifying co-labels types (meta-data) to identify data with valid recipients
- Providing rules for co-identifying data and recipient
- Determining granularity of security required for various data
- Determining classification change rules for processed data

9.2.1. Cost / Benefit and Risk

Security, in whatever form, is recognized to have some costs associated with it. Both cost/benefit studies and risk analyses are required in determining security options.

As data administrators, we want quick and not-too-expensive procedures for gauging the cost/benefits for various categories of the data.

The following series of qualitative sorts will determine which data is "most" valuable from a confidentiality viewpoint; the purpose is to quickly focus DA security activities toward those data and issues which are most important. Generally, for data administration purposes, qualitative sorts are both essential and usually sufficient; quantitative cost/benefit studies may be necessary for narrow, specific issues but the broader issues are usually not amenable to meaningful quantitative analysis.

The following is a way of rank-ordering for security value. The first step is to divide the data into major categories reflecting the expected benefit from security measures.

The "benefit" of data security is defined as the *avoided* costs of a security breach. In other words, the benefit of security for various data is ranked by adding the estimated *avoided-costs* from that data falling into the wrong hands. However, the *cost* (in cost/benefit) is the sum of prices required to implement the data security. Since the word *costs* is used in two, opposite senses in the following discussion, be careful to attach the correct meaning.

There may be many criteria for estimating "value" of data security. Rather than get bogged down in discussions of which is most important, have a set of knowledgeable people rapidly, qualitatively rank the security value of the data, i.e., the business cost of improper disclosure, vis-a-vis various criteria.

A qualitative ranking scale for each criteria should be developed and should contain typically, between 5 and 10 ranks. Table 9-2 shows such a ranking relative to dollar-costs of a for-profit organization. The same organization might have a separate ranking for "legal" costs, as shown in Table 9-3. A ranking table of "combined" criteria is shown in Table 9-4.

Table 9-2: DOLLAR COST RANKING CATEGORIES

Major corporate costs (e.g., end of corporation)
Major "Division" costs (e.g., end of division)
Major "Department" costs
Individual costs
No overt costs (social or soft costs only)
No costs

Table 9-3: LEGAL COSTS RANKING CATEGORIES

LEGAL COSTS
Major legal costs (e.g., jail + court)
Moderate legal costs (e.g., court + fines)
Minor legal costs (e.g., negotiated settlement)
No legal cost

Table 9-4: COMBINED CORPORATE COSTS RANKING CATEGORIES

COMBINED DOLLAR/LEGAL COST RANKING CATEGORIES
Major legal costs (e.g., jail + court)
Minor legal costs (e.g., court + fines)
Major corporate costs (e.g., end of corporation)
Major "Division" costs (e.g., end of division)
Major "Department" costs
Individual costs
No overt costs (social or soft costs only)
No costs

Note that there are many ways of forming combined ranking tables, either by leaving the criteria "independent" and interweaving them as shown in Table 9-4, or by formulating a detailed equivalence between "legal-costs" and "dollar-costs." In most cases, the security requirements will be driven by the highest ranked data in each category, resolving the issue of which are the important data to keep secure. This will have been accomplished without falling into the metaphysics of resolving the relative "value" of disjoint criteria.

It may occur that later in the development life cycle decisions will have to be made on which data to be kept secure and the degrees and methods of obtaining security. These decisions will almost NEVER be made by the data administrator; but the DA has provided a mechanism for identifying relative values of such decisions and documenting these decisions.

Governmental and non-profit organizations may utilize other criteria than "dollar-cost." Table 9-5 provides a starting list of such criteria.

Table 9-5: GOVERNMENT AND NON-PROFIT CRITERIA

POSSIBLE OTHER CRITERIA FOR GOVERNMENT AND NON-PROFIT ORGANIZATIONS
Mission Failure
Public Trust Loss
Organizational Credibility Loss
Management Turbulence
Work force Turbulence
Political Damage
No Significant Loss
No Costs

This rapid exercise will usually quickly organize the data by "security-value" and will uncover possible areas of organization miscommunication or misperception.

Risk analyses may be utilized to further refine the costs of security lapses. Risk is the probability of having a security lapse and system security measures may be implemented on a basis weighted with the risk probability. However, the risk probability usually is considered in terms of system protection, rather than in terms of data content. Consequently, while risk assessment is very important for pricing the likely value of a security system, such assessments usually have no direct impact from the data *content*. Consequently, risk analyses will usually be outside the direct interest of the top level data administrator. [A local DA working within a risk analysis organization may be interested in how to organize the data, including results, of such an organization - but the generic DA or top-level DA would not be.]

9.2.2. Security Costs - Co-labels and Granularity

From a data content view, the costs of providing security control are driven by two major factors; the number of required co-labels and the complexity of their associated business rules; and the required co-labeling granularity.

9.2.2.1. Co-labels

Information restricted in any way must have associated information which communicates the restriction and each recipient must have associated information stating their membership among those having access to the information. We define this associated information as a co-label. A co-label is information identifying data with valid recipients. Co-label types are categories of data (or of meta-data) necessary for implementing confidentiality. (Note that "BLANK" or "No co-label" may be the label for unrestricted data.)

Table 9-6 lists names for co-label types. Most of these names are generic, reflect the *content* of what constitutes valid recipients, and can be readily adapted to most organizations.

Note that there will usually be multiple co-label types logically associated with the same data. Further note that there must be business rules associating sets of valid co-labels and these rules can be expressed in terms of logical *and*s, *or*s and *not*s operating on values (or value ranges) of the co-label types. For example, accounting information may be available to: (a) any member of the accounting department (by DEPARTMENT); *or* (b) any senior vice-president or above (by HIERARCHICAL POSITION); *or* (c) to the local project manager of his project (by FUNCTION *and* PROJECT *and* HIERARCHICAL POSITION); *or* (d) Jane Smith (by NAMED INDIVIDUAL).

Clearly, generating and documenting the business rules which allocate data and recipients is a major undertaking and is vital to data administration. Obviously, the "cost" of security will probably increase with the implemented richness of the co-label requirements. Note that the physical implementation of these rules may be simplified in various ways, but these simplifications should be attempted only with the knowledge and concurrence of the DA and after the logical rules are explicitly stated.

Table 9-6: CO-LABEL TYPES

Creator
Department
Function
Project
Location
DoD Classification Level
Hierarchical Position (e.g., Vice President)
Named Individuals
Other
Any logical AND-OR-NOT combination of the above

In Table 9-6, note the co-label *OTHER*. This label serves two purposes on the list. First, it forms a place to hold additional co-label types while determining their validity. Second, it provides a place to deflect suggestions for invalid co-labels without offense nor damage until the problems with the suggestion are understood. For example, people from government security use the concept of *Need to Know* and, when they see the category *OTHER*, they immediately place *Need to Know* there. However, *Need to Know* is not a data content type as are the other co-label types; it is really the result of the use of a set of business rules; the *Need to Know*, for data administration, is those logical combinations of the proposed co-label values which allows access to that data (e.g., recipient *needs to know* because PROJECT involvement *and* HIERARCHICAL POSITION...). While these business rules do need to be fleshed out for each organization, so far, *Need to Know* has not resulted in a fundamentally new co-label type.

Because the *OTHER* category provides a parking place for concepts needing further examination, it also helps to bypass generally fruitless or premature discussions while allowing productive discussions to continue. In addition to its usefulness as a possible data concept, its effectiveness in easing on-going interactions can't be overestimated but need not be disclosed.

Note also that the "valid recipient" of data need not be a person; it may be a computer program, a network communication line, or other combinations of hardware and software. However, the co-label list still seems to work, although particular user communities may choose other names, more familiar in their arena, for the same content.

9.2.2.2. Granularity

Another major data administration cost driver is the granularity of security controls.

Before beginning a discussion of granularity issues, let's provide a few examples of granularity and its impacts.

If all the data is kept on one computer which has no outside access, then only one (implicit) co-label is needed for the data and the granularity of security is very coarse. All the data is available to everyone with access to that computer and to no one not having access.

At the other extreme, data on a computerized network database might be available to anyone, except when certain values of the data are present. For example, location and schedule of a ship might be readily available to anyone, unless the ship were sailing in the Persian Gulf area, in which case its location could be highly restricted. In this example, security co-labels must be applied to all individual data values; all data instances would have to be labeled. This is a hypothetical example of an extremely fine granularity of security control.

A non-computer example of granularity is in the Department of Defense security system. Documents are composed of chapters, pages, paragraphs, sentences and words with accompanying tables, graphs, etc. Paragraphs are each classified according to the highest classification rating of the information in that paragraph, pages are classified according the most classified paragraph;

documents are classified at the rating of the most classified paragraph. Here the granularity of security control is at the document level with an additional labeling at the page and paragraph level.

As a final example, Table 9-7 shows the various levels of granularity within a conceptual data model.

Each type of data has the potential of being organized, for confidentiality purposes, at any of these levels of detail.

Table 9-7: CONCEPTUAL DATA MODEL HIERARCHY

GRANULARITY HIERARCHY IN A CONCEPTUAL DATA MODEL
Set of Databases
Database
Set of Tables
Columns
Rows
Cells (values).

Now we can state rules of thumb on granularity issues.

The finer the granularity of security controls required, the higher the overt costs of security will be. This is clear since the number of co-label values which must be tracked, accessed and compared increases with level of granularity required. The amount of computer overhead, personnel time, data storage and many other cost factors increases with the number of co-label values to track.

However, coarse granularity of security control may introduce a variety of hidden costs. Security control introduced at too coarse a level may unduly restrict data. Then either the goal of data availability is compromised, or workarounds are instituted. In the DoD document example, it is common for highly classified sentences to be eliminated from a copy of the document to provide an unclassified document with most of the information - at the cost of duplicating and having different "versions" of the same information available. The arguments against analogous procedures for database design are well known.

Set the granularity of security at the highest conceptual level possible, consistent with all the data below that level having the same degree of restriction. In an IRM data environment, there are hierarchies of conceptual model types and hierarchies within each model type. The major hierarchies are: business models; data models; and process models. In terms of conceptual granularity, typically business models will provide coarser granularity than data models which are typically coarser than process models. Each model type has an internal hierarchy: business models range from the global corporate information model down to narrow departmental, project, or product models; a typical hierarchical data model, ranging from *Set of Databases* down to *Cells(values)* was shown earlier in Table 9-7.

Data classifications may change for any of several reasons, including data processing, elapsed time, and geographic location. A classification change of data is defined operationally as *any change in the list of authorized users of that data* and that corresponds to a change in co-label values. The rules for changes in classification must also be specified for each set of data.

As processing examples, the salaries of each individual may be kept confidential but the average salary may be public knowledge. Or, the existence and availability of inventory items may be public knowledge, but the total number of certain items may be highly restricted information.

In an IRM situation, the following procedure may help the DA with identifying data classification change situations in processing situations. Start with data flow diagrams (DFDs) and process flow models. Determine whether the security classification of the data output for each process step is the same as the data input. If the classifications differ, then some transformation rule must have been applied. Identify the transformation rule for each case. Work at the coarsest granularity level of DFD which unambiguously transforms or does not transform the data classification; if some portion of the DFD output has a classification change and some does not, then the granularity is too coarse. [Application of rule of thumb #3.]

9.3. Other Factors

A couple of other major factors can be considered in Data Administration for Security. Both the types of database to be considered and the corporate culture, as reflected in structure models, can be major factors for the data administrator.

9.3.1. Database Types

Databases are frequently considered as one of two major types; the operational database and the decision support database.

The operational database is characterized by a fairly stable set of data characteristics; routine operation; and typically having a significant number of its data representing real world items, such as people, inventory, physical dynamics, etc. It is likely to operate in real time situations wherein timeliness of data entry and updates is important. The operational database is a likely target of data administration for security techniques, either during its design phase or as part of an enhancement or integration project.

The decision support database is characterized by snapshot data, non-real time operations, and a large amount of asking unusual questions or attempting to consider the data in new ways.

The thrust of this discussion has been toward operational databases. So far, it appears that little substantial thought has gone into the data administration aspects of security for a decision support database. The DA involved in such a project would be advised to keep his customer informed of the research nature of the undertaking and the unlikelihood of useful results.

9.3.2. Corporate Structures and Security

Control is a significant factor in corporations as it is in security; the way *control* is handled in the corporation is indicative of what may be involved in data administration for security. How is information controlled; who or what decides which data is accessible to whom? The DA might consider the corporations' control structure in preparation for a security project. This may provide indications of what will be possible, or it may provide indications of existing problems, which the DA was called in to correct.

Fortunately, corporate control characteristics are frequently inferable from a knowledge of traditional corporate structure models. Business structures used to be discussed as having one of two shapes; the pyramid structure and the flat structure (See fig. 9-1).

The pyramid structure is strongly hierarchical, with reasonably clear lines of command and successively smaller numbers of people at the higher levels. It traditionally described large manufacturing and retail businesses.

The flat organization was characterized by very few administrative or management levels and by a large amount of independence at the lower levels. It was described as being typical for R&D organizations.

The corporate culture will likely be quite different between these two types of organizations. Working DA for Security in the pyramid organization, success will probably depend upon convincing the relevant upper level manager(s) to be involved; disputes and resolution of business rules will probably be decided high in the organization; there will normally be someone, or some few, people who could be expected to make decisions. The corporate culture may be partially modified by dictum and such dictates will have some likelihood of being enforced. The DA can provide plans, approaches, standards, and, if convincing, may expect some compliance.

In the "pure" flat organization, the approach will be different. Since there will typically be no single point to resolve issues, the DA must plan on looking for common approaches; must expect to work with committees in which everyone is a peer; and must devise standards and methods whose value is self-evident, or whose implementation is self-fulfilling. Modifications of corporate culture, if any, will be quite small.

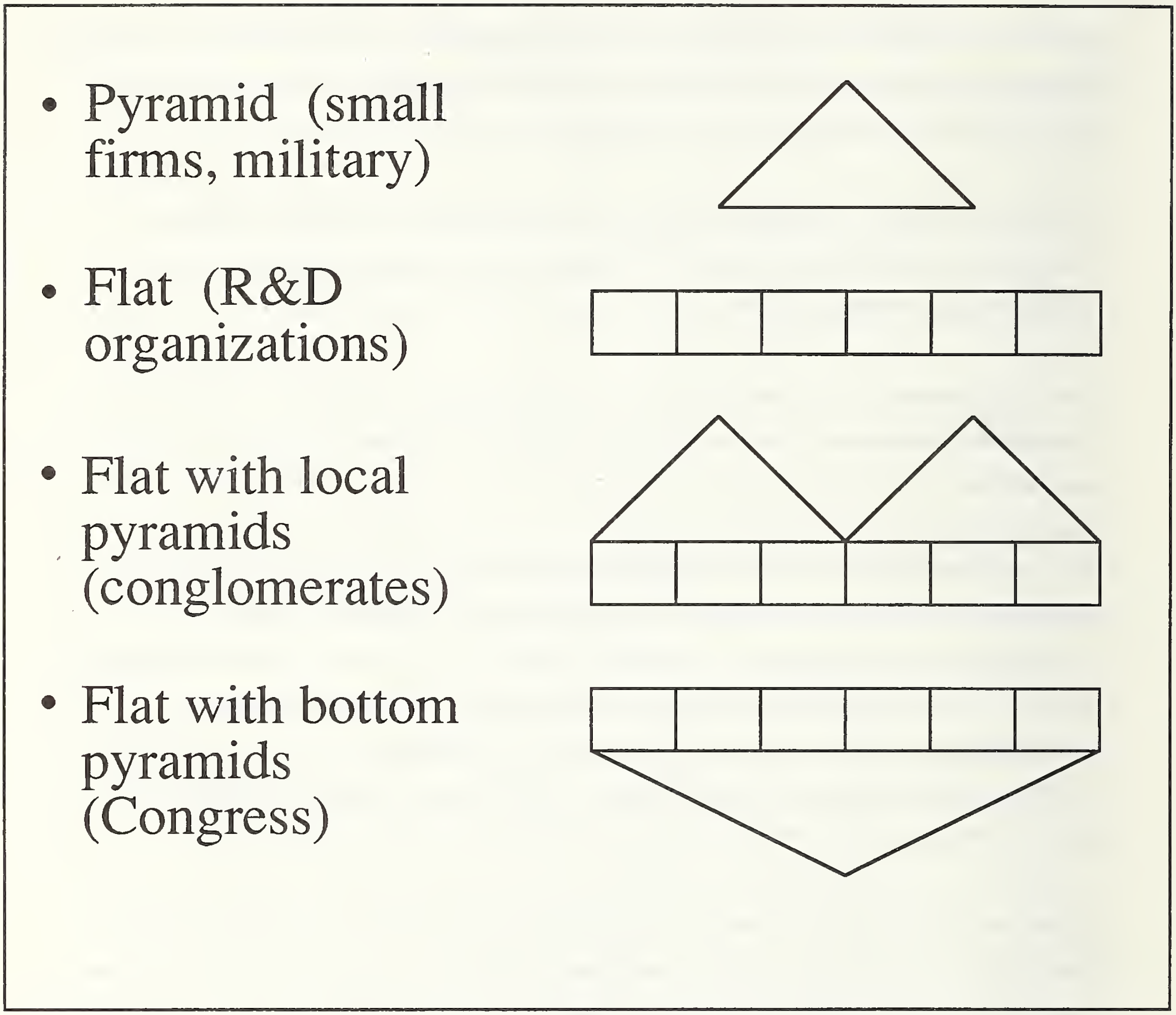


Figure 9-1: Corporate Structures.

More commonly encountered are various admixtures of these two basic control structures; the flat bottom with local pyramids structure is common in conglomerates or in project-oriented organizations. Another variation is the flat-with-a-bottom-pyramid structure. The bottom pyramid is usually a service or support organization to the flat structure, such as computer services to Congress, or in academic institutions.

In the flat-with-local-pyramids, the DA must scope out the extent of each project - is it intended to operate in the "flat" part of the organization, or is it contained within one pyramid (creating databases very like a stovepipe), or is it intended to integrate data across pyramids? In this inverted pyramid case, the hierarchical ability to enforce standards is absent, except for the possible option to refuse service to those not abiding by suitable standards.

While these considerations of corporate control patterns are not rigorous, they do provide reasonable initial hypotheses for what a DA might expect from an unfamiliar organization. These simple control models may imply a significant number of organizational business rules.

9.4. Summary

This section has confined itself to the considerations of a top-level DA and to only one of several aspects implicit in the word "security" or the concept of "protection." The same sort of considerations can expand this approach to the extent and depth required for any particular project.

Section 10. Repository Management

10.1. Introduction

This section describes a model for Repository Management, including the policies, standards, and procedures that should be employed when using a repository. In particular, this section is most applicable where a central enterprise-wide repository is implemented in conjunction with system development project teams who will populate and use it. This section is not intended to provide specifications for the functionality or structure of a repository.

The ANSI X3H4 committee, consisting of representatives from both corporate and government practitioners, leads the U.S. national development effort in standardized specifications for an information repository. The current information repository standard, most commonly referred to as the Information Resource Dictionary System (IRDS), is specified in the standards ANSI X3.138 and FIPS PUB 156.

While this document focuses on Data Administration (DA), many of the entity types maintained in a repository do not necessarily come under the purview of DA. There are variations in the industry regarding how much should come under the control of DA versus, for example, the System Development organization. This chapter does not attempt to identify which parts of the repository come under DA's domain, but the current thinking on this issue covers data stewardship responsibility *at a minimum*. Data stewardship identifies the logical "owners" for describing the meaning and use of the *data* in an organization, and ensures that the data owners carry out that responsibility.

This section is divided into the following subsections:

Repositories and Models
Repository Management
Repository Management Policy
Repository Management Procedures
Repository Management Roles and Responsibilities

10.2. Repositories and Models

Not terribly long ago data administrators and database administrators discussed the benefits of data dictionaries (and their less than wonderful user interfaces!). Data dictionaries are used typically to describe in detail the data collected and processed by the organization's information systems. This includes the *databases, files, records, segments, data elements*, and other related information about data, called meta-data (i.e., "data about data").

Today we speak of information repositories. The term is broader because the information stored in a repository typically will involve more types of information than the classic data dictionary. For example, organizations still care about the entity types identified above, such as *files, records,* and *data elements*. However, the repository can be used to manage all of the organization's information resources by extending these entity types to include *systems, organizations, networks, business functions and processes, data flows*, and many others.

A model is defined here as a subset of the "repository" containing both entity types and entities, i.e., entity type instances. For example, a model can be created for a business area containing the entity types: *function* and *data entity*. Specific instances might be the *Customer Order Processing* entity (of the *function* entity type) and *Customer, Customer Order*, and *Product Inventory* (of the *data entity* entity type). Note that the repository describes not only the entities of interest to an organization, but also the relationships between the entities (actually relationships themselves are generally treated as unique and separate entities in a repository). Thus there may be relationships between the *data entities*, for example, that are part of the model.

The repository, while a useful documentation tool, has other and perhaps more significant purposes, e.g.,

- Planning the organization's information resources, including organizational structure planning and strategic data planning;
- Managing the system life cycle, from the planning to archive phases;
- Developing, implementing, and enforcing data and other standards; and
- Managing the use of an organization's information resources.

The repository accomplishes the above goals by allowing an organization to document any and all aspects of the organization in conjunction with other tools, such as CASE and project management tools. The functionality provided in these tools are generally not inherently part of the repository functionality, although some vendors package I-CASE (integrated CASE) tools, such as the Information Engineering Facility by Texas Instruments, Inc. and Information Engineering Workbench by KnowledgeWare, which have both CASE and some repository functionality.

10.3. Repository Management

As we have learned, implementing a tool, such as an information repository or CASE tool, does not by itself guarantee success. As with most automated tools, a pre-planned and well-thought-out approach is necessary to ensure the tool's success. This approach is called repository management.

For our purposes, when discussing repository management, we refer to the administrative aspects of maintaining information in the repository, as well as tool-specific capabilities such as those listed below.

(1) The definition of an enterprise-wide repository schema, which defines the entity types that the enterprise will describe and the rules for doing so;

(2) Partitioning the repository information into application-specific views;

(3) Controlling access to repository information;

(4) Creating, manipulating, and reporting repository information; and

(5) Managing changes to repository contents by creating entity versions and providing for the migration of entities from one system life cycle phase to another.

Repository management includes tool specific capabilities and the administrative (or management) aspects of maintaining information in the repository. In addition to the five items listed above, repository management should comprehend the policies that an organization sets forth for populating and using the repository, the standards for defining repository information, the procedures for integrating design information across system development project teams, and the management of changes to the repository. Item 1 above, the definition of an enterprise-wide repository schema, is not within the scope of this chapter.

In order to be effective at repository management, the administration of these policies, standards, and procedures should be applied to repository information wherever it exists in the organization. While much of the information may exist in a corporate-wide repository, that information is typically supplied via CASE tools. Thus the local CASE tool "repositories" should be viewed as a physically distributed extension of the corporate repository. This does not mean that the CASE components must mirror all of the corporate-wide repository's components or vice-versa. However, in order to maintain the integrity and consistency of information in the two environments, repository management practices should be defined to encompass the two environments in totality.

Since the information in the repository is developed and used by different players, including Data Administration and system development project teams, it is critical that the information conform to a set of DA standards. In addition, since each player contributes in some way to the overall corporate-wide repository, the integration of new information into the repository should be monitored carefully to maintain the integrity of the overall model.

10.3.1. Management Policy

The following identifies some of the policies that should be addressed for Repository Management:

The Repository Management plan should be developed as part of each system development project's overall plan. This will document the specific approach for *populating* and *maintaining* the repository. Most of this plan should be in accordance with procedures defined for the organization, and may add to them, for instance, by defining specific project team-level roles and responsibilities for repository management. Refer to Intra-team Repository Model and Roles and Responsibilities for more information on this topic.

All DA standards for meta-data naming and definition should be followed when creating or updating repository information. These standards include entity naming, entity description (the set of attribute types which describe an entity), and entity standardization (such as *data element* standardization).

Change control procedures should be developed, and followed when updating the central repository. Change control, which is ideally handled by the repository but may also require some manual activities, includes entity versioning and change audit tracking. In addition, the repository should allow users to migrate entities from one system life cycle phase to another.

Integration reviews should be conducted, as needed, to ensure the repository reflects accurate information of the highest integrity. Depending on the organization, integration review may be the only way to approve changes to a corporate data model. Refer to Inter-team Model Update for more information.

Access privileges for repository information should be defined so it is clear to the organization who can access what, where, when, how, and why. This will require policies for security to protect the integrity of the repository.

10.4. Repository Management Procedures

Repository Management Procedures are described in three subsections:

10.4.1. Repository Standards Dissemination Procedures

Deliverable: Understanding and Acceptance of DA Standards
Primary Role: Information Modeler (DA)

Data Administration defines standards, such as data naming and description standards, and process naming and description standards. These standards should be defined, agreed upon, documented, and disseminated. In addition, these standards should be reflected in the repository schema, i.e., the attributes for the repository entity types, and should be enforced using whatever automated tools or manual procedures are necessary. For example, data naming standards may be enforced in part by automated tools, but may also require inspection by DA to ensure absolute compliance with standards (and, for example to ensure data redundancy is minimized).

While there are few standardized procedures for disseminating standards, it is imperative that all contributors to the repository, particularly system development project teams, have knowledge about the standards that are imposed on them. The following are some suggestions for disseminating standards:

Don't distribute standards and then expect them to be implemented. Implementing standards should be followed up with procedures for ensuring their compliance. This includes repository reviews and "hand-holding" as necessary by DA.

Always test standards prior to distributing them. A new and willing project team is a good tester of new standards.

Try not to impose standards as being a burden on the organization. Although it may seem to increase levels of project team effort, be sure to *illustrate* the benefits of standards for everyone.

Ask for improvements to standards (but don't promise that all change requests can be honored). A procedure/form for requesting changes to the standards can be incorporated into the standards documentation that is disseminated.

Use the repository itself to disseminate standards, as appropriate. This is often done by providing global access to an entity-relationship model (of entities, relationships, and attributes) of the enterprise-wide repository model itself. Thus the naming standard for data elements could be documented in an attribute called *data element name* of the entity *data element.*

10.4.2. Repository Access Control Procedures

Deliverable: Repository Access Privileges, Repository Views/Subsets
Primary Role: Repository Manager (project team and DA role)

Access to the repository should be managed carefully. The corporate repository contains information that is of interest to many different people. But it is also true that different people have interest in the same repository information. This is particularly evident for the business and system analysts who work on system development project teams. Thus the models that are placed in the repository should be selectively parsed out to those teams so as to avoid one team overwriting the models of another. The following are recommendations for providing and controlling access to repository information, but will depend heavily on the capabilities of the organization's toolset:

Define a subset or view of the corporate-wide repository as a distinct "model" with which a project team may access and further develop. From a DA perspective, the portions of the model which relate to data should be distributed to project teams based on data stewardship principles and the actual update of the repository should be accomplished via an integration review.

Note that it will become very important for the DA and/or integration team to keep track of what entities are needed by each project team (see Repository Update Procedures). This can be accomplished using repository view/subset management, functionality that is included in today's automated repositories.

Provide read-only access to entities in the repository that a project team is interested in, but is not responsible for changing.

Provide the project team's repository manager with the capability for further dividing their model into sub-models. This is useful for managing large projects where the amount of information becomes large and the project can best be managed by dividing it into smaller, more cohesive chunks. This capability may be provided by a repository or may be a function of a CASE tool.

Provide the project team's repository manager with the capability for controlling the users' access privileges to the team's model or sub-models. This level of security is a project-specific one and is best accomplished through the project team's Repository Manager. This capability may be provided by a repository or may be a function of a CASE tool.

10.4.3. Repository Update Procedures

The creation and modification of the contents of the repository are usually the outgrowth of system development. When changes are made to the repository the impact on other users of the information should be controlled and managed. Update procedures should indicate how the information in the repository will be changed. For most organizations, the fact that the repository is an automated system does not alleviate the need for manual or other automated procedures. Model update is divided into two categories: Intra-team Model Update and Inter-team Model Update.

Figure 10-1 depicts a repository update environment involving four main activities:

(1) *Define System Project Meta-data* to develop meta-data (e.g., data and process definitions) relevant to a particular system development project, such as a new Personnel System.

(2) *Review Repository for Impacts on System Project* to identify meta-data of the project that does not fit within the standards and objectives within the repository. Deviation from the repository will be due to (1) not referencing objects already defined in the repository, (2) inappropriate naming conventions, and (3) identifying meta-data that either should be in the scope of the project and is not, or should not be in the model and is in the project model.

(3) *Review/Approve Project Meta-data for Impacts on Repository* to identify those meta-data that are new and should be added to the repository or are updates to existing repository meta-data.

(4) *Maintain Information Repository* by completing the updates and additions to the repository.

The four activities above identify *what* should be done. *How* they are accomplished needs to be defined by each organization that is implementing a repository. The solution is partially dependent on the toolset used, such as the repository and/or CASE tools. However, this section will identify some practical considerations for repository update. Repository update is divided into three categories: DA Model Update, Intra-team Model Update, and Inter-team Model Update.

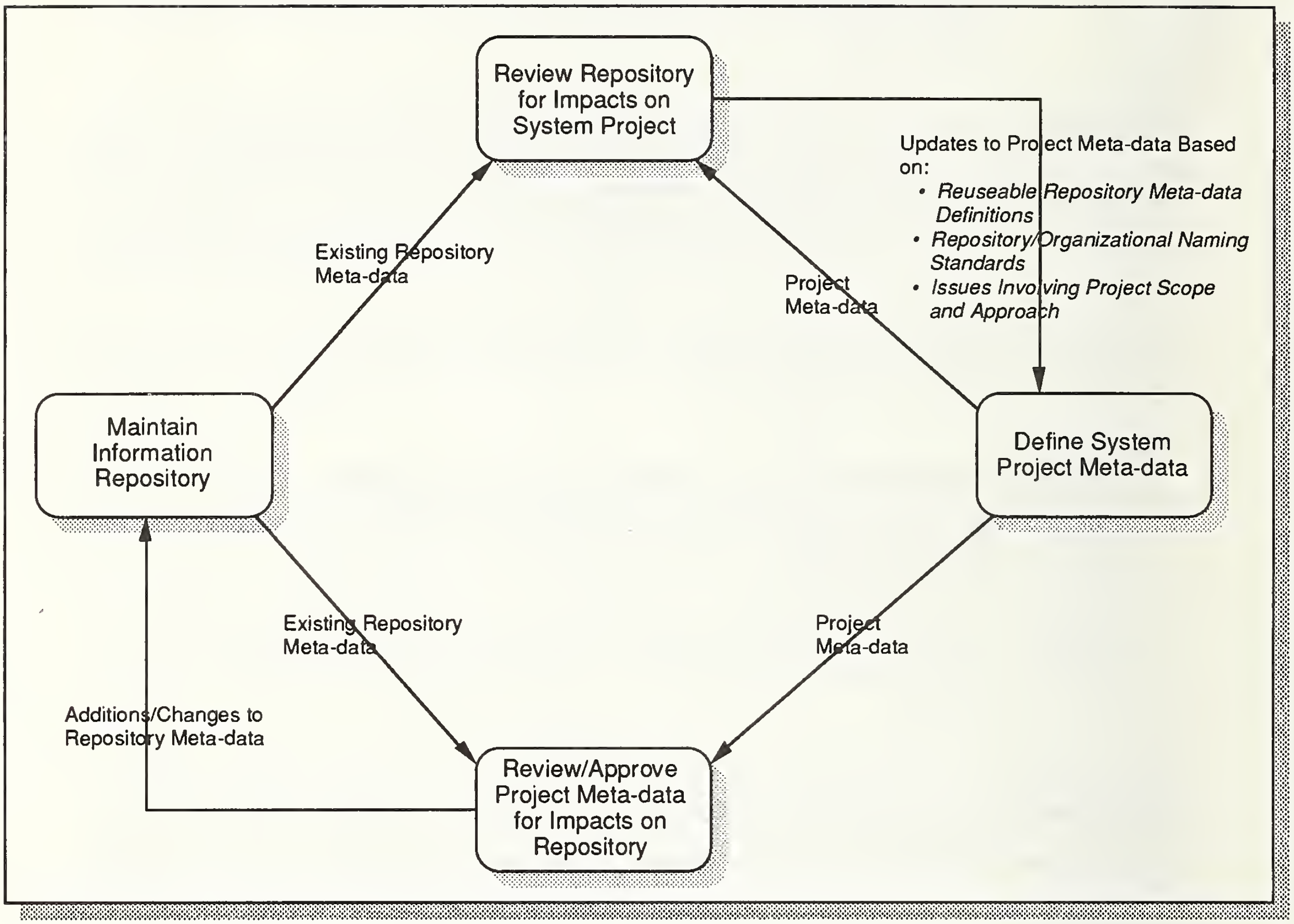


Figure 10-1: Repository Management.

10.5. DA Model Update

The DA organization may decide to add to or change the contents of the repository, for example, to develop an enterprise-wide information architecture or data model. When DA performs these and other similar activities which modify the repository contents, both the Intra-team Model Update and Inter-team Model Update procedures should be followed. In this case, the DA organization is treated as a "project team" and includes in addition to the traditional DA roles (as defined in the Repository Management Roles and Responsibilities) the roles of *Information Analyst* and *Project Team Leader*.

With repository management comes a critical issue regarding how much metadata DA should collect and populate in the CASE and/or repository tools. Many DA organizations attempt to populate the tools by performing analyses and surveys to collect detailed data requirements, such as elements, within the existing systems. This approach towards repository maintenance can be extremely expensive and may provide little or no added value. The problem is that the information that was collected was based on a DA effort and not a specific system project. If, for example, immediately following the data collection effort a project team makes an enhancement to their system to collect additional data neither DA nor the repository will be "aware" of these changes. Thus DA's efforts are now dated and for the repository to be accurate, another data call would be necessary. This mode of maintaining the repository puts DA in constant catch-up with reality and does not provide the kind of proactive decision-making DA needs to ensure that data is being collected and used by the organization according to a master "corporate" plan (such as the results of a strategic data planning effort).

10.5.1. Intra-team Model Update

Deliverables: Project Team Repository Models, Model Changes
Primary Roles: Information Analyst, Project Data Manager, and Team Leader (team roles)

Intra-team model update procedures are defined strictly for the changing of a team's model *prior* to its incorporation into a corporate-wide repository. Intra-team model update procedures allow a project team to work together at a localized level for a period of time (until the next iteration of Inter-team Model Update). The following are suggestions for *Intra-team Model Update*:

The Information Analyst should record and track changes to model information. A repository should create new versions of an entity when the entity is changed, and maintain ties to prior versions for audit and regression purposes if necessary. Since repositories do not always provide a complete audit trail of changes, some manual tracking using change control forms, may be necessary. Note that the forms themselves may be automated and routed to team members electronically, e.g., via E-Mail. In addition, the team should migrate relevant entities from one system life cycle phase to the next, as the project progresses through the system life cycle.

Models should be archived regularly, so that regression to a complete prior state of the model is possible should the model information become corrupted. Archiving involves backing up the model information in a routine fashion, a necessary requirement for the maintenance of any database.

Provide update access to *subsets* of the model to specific analysts on an as-needed basis. For large projects in particular, it will be highly inefficient to provide the entire model to an analyst who may be working on only a small portion of the design. The entire model may be too much information to deal with. Also this would result in less project management control over changing model contents.

Establish a Model Review process *for the team* to approve all changes made by team members to the team's model. The review process should reflect two aspects of approval: (1) Approval with respect to the semantic meaning of the model, i.e., the model accurately represents the team's view; and (2) Approval with respect to DA standards.

Unapproved changes should not be recorded into the team's official model. If unapproved changes have been made to that model, they should be backed out.

Participate in Inter-team model update procedures.

10.5.2. Inter-team Model Update

Deliverables: Updated Central Repository Models, Model Change Requests from DA, Refreshed Project Team Repository Models

Roles: Information Modeler (DA), Information Architect (DA), Local Data Administrators (DA), Repository Manager (DA and team role), Project Data Manager (team role) and Project Team Leader (team role)

Inter-team model update procedures provide for the *integration* of project team models into the enterprise-wide repository. Repository integration at the central repository level is more formal than intra-team update by involving Data Administration and other project teams in the update process. The following are suggestions for *Inter-team Model Update*:

> Team members from one team should maintain close ties with other teams who are responsible for defining entities that the first team requires in its model. This coordination minimizes conflicts when the teams' models are actually reviewed for integration into the central repository. For Data Administration, coordination is most significant with respect to Data Stewardship responsibility, which defines the logical "owners" for describing the meaning and use of data in an organization.
>
> For example, suppose Project Team 1 is responsible for defining Data Entity A. Project Team 2 requires a new data element to be added to Data Entity A. Team 2 should request that Team 1 approve and modify their definition of Data Entity A to accommodate Team 2. The more coordination attempted prior to integration within the central repository, the easier the integration process will be.
>
> DA should encourage coordination between teams by, for example, instituting Local Data Administrators (see Roles and Responsibilities).
>
> An integration review team should be established comprising DA representatives *and* representatives from the various project teams, usually the Team Leader and/or Project Data Manager.

Changes made to team models should be reported to an integration team *prior* to the integration review meeting. This pre-review by the integration team allows issues that are inherent in the requested changes (such as non-compliance with DA standards or blatant inconsistencies across project team models) to be resolved outside of the formal review process. In this way, less time is spent quibbling over less important issues and less frustration is felt by all those involved.

The integration review should address each change (or summarized changes as appropriate) for approval by the integration team. The Information Architect (DA) should review changes with respect to maintaining the integrity of the enterprise-wide *business* model, or at least the corporate *data* model. (This will depend upon the scope of responsibility of the particular organization's DA). The Information Modeler (DA) should examine changes with respect to DA standards.

The integration team should consolidate the results of the review process, highlighting the approved changes to the central repository. In addition, the integration team should provide these changes in automated (and potentially manual form) to the specific project teams that are affected by the changes. This includes teams that maintain the entities that were changed and teams that merely reference the entities within their own models.

Project teams should be given procedures for refreshing their view of the repository, i.e., their models, with the approved changes from the integration team. The specific procedures for accomplishing this will depend entirely on the technical architecture of the repository and CASE tools.

10.6. Repository Management Roles and Responsibilities

Repository Management roles and responsibility have been discussed throughout this chapter. This section consolidates the responsibilities for each role. Keep in mind that these are roles, and not persons. Thus a single individual might fulfill more than one role.

Project Team Leader (project team role)

The Project Team Leader provides the following services:

Plan, coordinate, and monitor progress of project team's efforts.

Lead analysis and design by applying appropriate techniques.

Review/approve changes to the team's model.

Identify and resolve issues within the project team.

Participate in Inter-team Model Update, particularly at integration review meetings, as appropriate.

Project Data Manager (project team role)

The Project Data Manager provides the following services:

Lead data analysis for project team.

Assist project team members with data analysis, data modeling, and implementation of DA standards.

Identify and resolve data modeling and data integration issues.

Participate in Inter-team Model Update, particularly at integration review meetings, as appropriate.

Information Analyst (project team role)

The Information Analyst provides the following services:

Perform process and data analysis.

Access team model and/or subsets of the model to recommend changes to project team's model.

Participate in Intra-team Model Update.

Repository Manager (DA and project team role)

The Project Team Repository Manager provides the following services:

Create subsets/views of the team's model.

Provide team members with access privileges to the team's model subsets/views.

Establish procedures for and execute backup, recovery, and update of the team's model.

Coordinate with the DA Repository Manager to ensure approved model changes are reflected in the corporate repository after integration, and to ensure team's repository is refreshed with updated corporate repository after integration.

Provide reports to project team based on model contents.

The DA Team Repository Manager provides the following services:

Create subsets/views of the corporate model for access by project teams.

Provide enterprise with access privileges to repository subsets/views.

Establish procedures for and execute backup, recovery, and update of the repository.

Coordinate with the Project Team Repository Managers to ensure approved model changes are reflected in the corporate repository after integration, and to ensure team repositories are refreshed with updated corporate repository after integration.

Provide reports to DA of repository contents.

Information Architect (DA role)

The Information Architect provides the following services:

Provide initial corporate information architecture, if applicable (i.e., the results of an enterprise-wide study, such as an Information Systems Plan or Strategic Data Plan).

Establish Inter-team Update procedures.

Coordinate Inter-team Update.

Identify issues impacting data integration.

Recommend resolution of problems not resolved by integration review.

Information Modeler (DA role)

The Information Modeler provides the following services:

Provide data modeling guidance.

Facilitate consistent data modeling techniques across project teams as issues arise.

Review proposed data model changes (in corporate repository) and advise integration review team concerning the impact of particular changes.

Maintain the corporate data model to support integration.

Local Data Administrator (DA role)

The Local Data Administrator is usually assigned either to a particular user organization, a functional area within the organization, or some other basis for grouping project teams into a common category. Thus the LDA is a "fielded" DA role. It is often politically sound for the DA organization to hire an individual from the user organization to fulfill this DA role.
The Local Data Administrator provides the following services:

Identify data modeling and data integration issues affecting project teams.

Facilitate resolution of issues affecting multiple teams (this usually based on the assignment of the LDA to specific project teams, either by organization or some other grouping).

Identify data modeling and data integration issues affecting the corporate data model. This includes review based on corporate DA standards.

Participate in project team model reviews.

Participate in Inter-team Update, particularly integration review meetings.

Assist project teams with information analysis and data modeling.

Information Analyst Consultant (DA role)

The Information Analyst Consultant provides the following services:

Provide consultation on function and/or data modeling to project teams.

Assist project teams with Repository Management procedures and applying DA standards.

Work with the Local Data Administrator to ensure standards are met and integration issues are resolved across project teams.

Section 11. Strategic Data Planning

(Note: This section is provided with the expectation that additions, revisions and enhancements will be made as the concept of Strategic Data Planning evolves and matures.)

11.1. Introduction

Strategic Planning is the process of deciding on objectives of the organization, on changes in these objectives, on the processes and resources used to attain these objectives, and on the policies that are to govern the acquisition, use, and disposition of these resources.
Strategic Data Planning (SDP) is the establishment of a long-term direction for the effective use of data in support of the mission, objectives and processes of the enterprise. SDP assumes a recognition that data is a valuable corporate resource. As a corporate resource data must be effectively managed in order to promote system interoperability, system standardization, universal access, flexible and adaptable systems, and reduction in the number of systems and the amount of data necessary to accomplish the organization's goals. One of the earliest enterprise-wide planning methodologies was introduced by IBM in 1970 as its Business System Planning (BSP) program.

Although numerous planning methodologies have emerged since and the emphasis has shifted from process to data, most of the objectives from the early approaches remain the same:

For executive management:

- an evaluation of the effectiveness of current information
- an assessment of future information needs based on business priorities
- "horizontal" availability of information

For functional and operational management:

- data to be used and shared by all authorized users
- systems that meet business requirements
- data represents what it is supposed to represent

For I/S management:

- a better long-range planning base for data processing resources and funding
- a reduction in data management costs
- a model for all future data design activities
- an understanding of the business

Based on these goals and objectives, a high-level strategic data plan is developed for the enterprise. SDP is the activity that produces a "plan" in the form of a set of architectures that in turn comprise various representations of business requirements including data models, process models and other deliverables identified further in this section. The data model for example, defines a future, fully-integrated view of the organization's logically related data or subject-oriented base of data. The "view" and, in fact, the strategic plan should extend beyond that which focuses **exclusively** on data, to include considerations for the "application" of data in support of enterprise business processes.

Also, the "plan" can and should be produced based on **all** data needs of the **entire** enterprise. But as valuable and as attractive as an "all and entire" approach might seem, there are several factors that must be considered at the outset in order to determine if a scope of "all data" and the "entire enterprise" is possible, practical and cost-effective. In general, some of those factors can be classified as organizational, geographical, political, financial and architectural. More about a "modified" strategic data plan is presented later in this section.

Assuming SDP for the entire enterprise is presumed both practical and cost effective, it is important that the types of data environments to be included be explicitly identified as within the scope of the SDP activity. In keeping with the distinctions made in an earlier section of the document between Data Administration and the broader function of Information Resource Management, the scope of SDP will be limited to "data" as they support operational (day-to-day) and informational (planning/decision-support) activities of the enterprise - most often, but not necessarily exclusively, in the form of computer systems. No particular platform or environment is considered during the SDP activity, although it would be a consideration in the subsequent activity of application planning.

11.2. Strategic Data Planning Principles

The SDP activity is based on the premise that data is an asset that can be effectively managed to improve the quality of products and services and to increase the capability to compete as a business.

Supporting principles include:

- The Strategic Data Plan must directly support the mission, goals and objectives of the enterprise as a whole.
- The Strategic Data Plan must support the strategic, tactical and operational activities of the enterprise.
- The Strategic Data Plan must promote the quality and integrity of data.
- The Strategic Data Plan must be resilient to organizational change.
- The Strategic Data Plan must provide for informed decisions to be made with regard to implementation.

11.3. Strategic Data Planning Approach

While the major emphasis of Strategic Data Planning activity is of course, on data, any approach to the effort must consider the context within which data are applied or used. For the purpose of Strategic Data Planning, that context is the enterprise (the business organization) or more specifically the business processes that support the mission, goals and purpose of the enterprise.

So, the two primary and fundamental entities to be considered or analyzed in the Strategic Data Planning activity are: Process and Data.

Techniques used to implement the concept might vary and indeed there are numerous "methodologies" that purport to deliver an enterprise-wide strategic data plan. The current, comprehensive and probably most widely accepted approaches to data planning <u>and</u> development are based on the concept of <u>information engineering:</u>

- an interlocking set of formal techniques in which business models, data models and process models are built into a comprehensive knowledge base through the use of automated tools and are used to plan, develop and maintain information systems.

- the discipline of structuring data according to the global business needs of the enterprise.

- the application of a science by which the properties of data are made useful to people in systems and processes.

All three descriptions of information engineering as a "technique," "discipline" and "science" are appropriate for the Strategic Data Planning activity.

The following description of the SDP activity is presented within a general framework of information engineering within which also, plans, architectures and models play a major role. The SDP activity is divided into two sub-activities:

- Enterprise Analysis
- Strategic Data Plan Development

11.4. Enterprise Analysis

In general the SDP activity is begun by examining all available inputs from the enterprise including:

- Mission/Objectives/Inhibitors
- Business Strategy
- Business Processes

- Data (required by Business Processes)
- Data Quality

These entities, characteristics and associated assessments are assembled to form an Enterprise Model which is represented in both graphic and textual forms.

11.4.1. Mission / Objectives / Inhibitors

Most organizations have developed a formal mission statement and set of business objectives and/or critical success factors (CSF's). But in the absence of a formal statement of mission and objectives, the SDP activity should include the development of these statements as a first step. The value of identifying objectives, CSF's, etc., is that they help to establish an appropriate focus, from a business perspective, on processes and data. The objectives and CSF's themselves are likely to point to key processes. Follow-on measurements of whether objectives are met or CSF's are achieved probably will point to strategic data of the enterprise.

"Inhibitors," on the other hand, typically are not available. It is important to identify inhibitors to the mission and objectives in order to help provide a focus for the subsequent activity of developing an application architecture (plan) and priorities for application development. Other potential benefits in identifying inhibitors are described in the section on Data Quality.

11.4.2. Business Strategy

Long and short-term strategies for fulfilling the mission and objectives of the enterprise must be examined and understood. Business plans including major projects, product/service sales projections, tactical plans and schedules should be considered.

11.4.3. Business Processes

Business Process identification and definition is important to development of the SDP, at a minimum, because it provides for a common understanding among all involved about what the business does to accomplish its overall mission and objectives, and it provides a basis for defining which processes use strategic data. Business process identification proceeds with development of a model of the business in terms of decisions and activities required to manage the resources of the enterprise.

11.4.4. Data

Once the Business Processes have been defined and using them as the context within which to identify data, high-level categories or subject areas of lasting interest to the enterprise should be identified. These subject areas serve as a starting point for a detailed description of data required to support the business, regardless of the current or future environment or platform on which the data resides or will reside.

This first step results in what an authority on "information architecture" refers to as the "scope description" of data for the enterprise. Data identification and definition continues with development of a more definitive "model of the business" such as the one proposed by John Zachman, and includes sufficient detail to represent the data requirements of the enterprise regardless of organizational constraints.

11.4.5. Data Quality

When the preceding entities have been addressed, a large part of the enterprise model will have been developed in terms of both an "executive view" and an "operational view." However, one of the most significant activities in the SDP process remains to be accomplished: data quality assessment. This critical activity measures the state-of-data used or required by the enterprise and helps to establish a strategy for the future-state-of-data as a major resource supporting the enterprise.

Data quality assessment is a critical step that is missing or lightly treated in many methodologies. Even in some cases where data quality is addressed, it is restricted to analysis of data as it resides in computer systems.

As in the case with development of the process and data portions of the enterprise model, the data quality assessment activity ignores any particular environment, platform, or medium. So that whether the data "resides" in an automated system, a file drawer, or a file folder, it is unimportant at this point in the SDP activity.

Starting with an appropriate level of the data entities included in the Enterprise Data Model, an assessment of the data is made in terms of certain characteristics:

Criticality - data are (or are not) required by the receiving process(es) to the degree that the process would be impeded without them.

Availability - (Accessibility) data is (or are not) available (might exist but not provided) to the process, or are (or are not) accessible (cannot be obtained conveniently or economically) to the process.

Reliability - (Accuracy) data can (or cannot) be relied upon to base decisions or they conflict with other data of the same type but originate from another process.

Content - (Granularity) data includes too much or insufficient detail to be of value without additional cost, processing or interpretation.

Timeliness - data are provided too late to be of value.

Periodicity - data are provided either too frequently or not frequently enough to be of value.

Medium - the means or method by which the data are conveyed or transferred from process to process is (in)appropriate to the extent that it affects their value or requires additional handling or processing.

Each data class can be assessed based on the preceding characteristics, and, where deficiencies exist, a degree of deficiency could be applied; for example, on a scale of 1 (slightly deficient on occasion) to 5 (unusable on a regular basis). In terms of the data's availability/accessibility, the assessment results in either a "yes" or "no" condition. Analysis of deficiencies can produce a data quality index on which to base decisions with regard to improvement.

The data quality assessment should continue with the identification of each data class's originating process (there should be only one) and all of the receiving processes (there can be more than one).

11.5. Strategic Data Plan Development

Analysis of the data deficiencies, the severity of the deficiencies and the respective originating and receiving processes should result in the identification of ranked, candidate targets (processes and data) for improvement. Cost of the data deficiencies and therefore the value of any improvement can also be calculated based on the impact to the receiving process(es).

Further analysis will determine if improvements should be in the form of computer applications, revised/enhanced computer applications, new, revised or reengineered processes, organizational changes etc. The types of improvement and the improvements themselves will be influenced by several factors:

- SDP Objectives (Why was the SDP activity undertaken? Do the proposed improvements support the objectives?)

- The State-of-Information Systems (Are none, some or all major foundation or operational support systems in place? Are decision/planning support applications installed? What is the data quality index for data in systems that are in place? Can any systems be reengineered to combine function, reduce data redundancy etc.?)

- Process Improvement (Can the efficiency or effectiveness of a process be improved by correcting a data deficiency? In correcting the deficiency does correction include automating the process or a portion of the process?)

All of the resulting conclusions and recommendations should be "packaged" as the Strategic Data Plan in the form of:

- Data Architecture: (1) "the overall structure of the data of the enterprise including the definition of subject databases, a distribution schema, i.e., where is the data distributed throughout the enterprise - sometimes referred to as a geographic architecture, a definition of the major data policy decisions as well as logical and physical definitions of the data structures of the enterprise"; (2) "the identification, definition and graphic representation of the sets of data by subject area required to support the objectives and strategies of the enterprise." (3) "a logical grouping of the most important and frequently used data in the enterprise containing both current and future data needs."

- Application Architecture: (1) "the identification and definition of key applications, their interdependence, among each other their deployment strategy, and other policies and strategies required to bring about the strategic opportunities of the enterprise"; (2) "the systems 'blueprint' that represents data requirements in support of business processes"; (3) "the identification of computerized portions of business processes in support of objectives and strategies and data needs of the enterprise and classified in terms of operational support and planning/decision support."

The definitions provided for Data and Application Architectures should be sufficient to establish a baseline understanding of the content (or intent) of the Strategic Data Plan. Some "planning" methodologies would include a Technological Architecture and there is nothing to preclude that as a requirement or a deliverable in the Strategic Data Plan, particularly because of the natural relationship between data and their processing, storage and delivery media. However a description of a Technological Architecture is not provided in this release of the Model Standards.

Both the Data and Application Architectures as suggested in the definitions will include various graphic and narrative descriptions of the objects and relationships included in the respective architectures including:

Association Matrices

Process to Subject Area
Process to Entity
Process to Organization
Entity to Critical Success Factor
Process to Application (Present/Proposed)
Organization to Entity
Entity to Data Source
Information Need to CSF
Data Entity to Information Need

Models

Process Decomposition
Data Flow Diagram
Entity Relationship Diagram
Entity Attribute Diagram (at a minimum including entity identifier or name)
Event Diagram
State Transition Diagram

Object Definition (instances of)

Processes
Subject Areas
Entities
Relationships
Attributes
Applications
Critical Success Factors
Information Need

11.6. Strategic Data Planning Methodology

Methods and techniques for performing processes and producing deliverables are presented more explicitly and in more detail in numerous commercially-available methodologies. The "right" methods depend to a great degree on the particular planning requirements and objectives of the enterprise.

It is important for Data Administration to define or select an appropriate development methodology for the SDP. This includes definition of the pieces of the plan that will be developed, their sequence of development, and their level of detail. It also includes how best to scope the project so that it will be manageable and organizationally appropriate, how integration of piecemeal development will occur and whether data model development will proceed top-down, bottom up, or both.

An organization must determine whether a "modified" approach to strategic data planning should be taken. Information engineering methodologies generally specify the development sequence of each information architecture piece; however, few organizations follow this to the letter. In addition, an organization will frequently have several of the plan pieces already developed in an adaptable format. It is up to DA to provide guidelines regarding the usability of existing architecture pieces and the best order of development for pieces that are missing.

Data Administration may also set SDP guidelines based on the size of the enterprise. A smaller organization with relatively simple information needs may not need the level of detail that a larger organization might. For example, a large organization may need to have detailed function and information models for each of the initially identified functional areas as well as for each information system. It may also, as a result, need to develop procedures for the integration of the plans. A smaller organization, however, may have only one consolidated function and data model for the whole organization and no need for information system models. Thus, it is important for Data Administration to adapt these methodologies to the enterprise via architecture development procedures.

Procedures for scoping the enterprise of the SDP process include guidelines regarding the necessity of breaking down the larger enterprise into manageable, easily modeled pieces. The guidelines might include guidance on bounding an area of breakdown, whether the breakdown must be tied to well-defined business or organizational units, and how the breakdown should relate to the enterprise mission. For example, an enterprise might have in its mission statement an outline of the organizational areas of concern or operation. Thus, for the sake of consistency with the mission, it may be preferable to base any further scoping on this original definition.

These are the considerations that must occur in the process of determining whether a "modified" approach to strategic data planning should be taken.

11.7. Strategic Data Planning Roles and Responsibilities

SDP must involve organizational management, data and database administrators, and potential and current information system users. SDP development will usually be done by a project team of mixed backgrounds. This team may include functional users, systems analysts, information system developers, DA representatives, etc. Upper management must be involved so that current organizational strategic plans may be developed or accessed for integration into the data model. Data and database administrators will likely be necessary to identify modeling and integration issues. Users will be expressing their information requirements to the data model developers.

11.8. Strategic Data Planning Procedures

Certain processes in the SDP methodology will require adherence to specific procedures with regard to how processes are performed in order to produce consistent results (outputs and deliverables).

11.9. Strategic Data Planning Tools

The Strategic Data Plan will usually be developed via a CASE or I-CASE (integrated-CASE) tool. CASE tools use graphical interfaces and central data repositories to simplify SDP activities like data model development and maintenance. CASE tools have both repository and manipulative capabilities. Thus, organizational strategic planning information may reside in the tool and this, and other data-related information, may be manipulated as needed for in-depth analysis.

Section 12. Information Resource Dictionary System (IRDS)

12.1. Introduction

An Information Resource Dictionary System (IRDS) is a software system that controls, describes, protects, documents and manages an enterprise's information resources. Applications of an IRDS include information resource management, database administration, data administration, CASE tool support, model representation and integration, document administration, forms control, facilities management, configuration management (hardware, software, system, etc.) source and object library management, software life cycle and project management, documentation and organizational planning.

12.2. IRDS Standards

There are several related IRDS Standards:

- the ANSI IRDS family
- the ISO IRDS
- IRDS II.

These standards are discussed in the following three subsections.

12.2.1 The ANSI IRDS Family

This family consists of three compatible standards:

- American National Standard X3.138-1988, the Information Resource Dictionary System. This is the original ANSI IRDS standard. It specifies two user interfaces, a Command Language and a Panel Interface, and includes the Basic Functional Schema, and optional facilities for IRDS Security, Extensible Life Cycle Phases, Procedures, and Entity Lists. This standard was adopted by the Federal Government in 1989 as FIPS 156.

- American National Standard X3.185-1992, the IRDS Services Interface. This is a full-function programmatic interface to the contents of the IRDS. Any programming language or system that provides an external call mechanism and that supports character, integer, and real data types can be used to write software that, through the Services Interface, can populate, navigate, and maintain the contents of an IRDS.

- American National Standard X3.195-1991, the IRDS Export/Import File Format. This standard provides a controlled mechanism for moving data from one IRDS to another. The standard defines a linear representation of the concepts and structures used in an IRDS, and specifies a mapping of these concepts and structures into the representation.

Both the Services Interface and the Export/Import File Format are completely compatible with X3.138-1988, and can be considered extensions of its functionality.

The ANSI IRDS family uses a binary ERA (Entity Relationship Attribute) data modeling approach to describe the standard. The major constructs include entity, relationship, attribute, and attribute group. Attributes are associated with both entities and relationships. All objects must be of one type. Relationships are directed and may have constraints (1:1, 1:N, existence, optional, etc.) associated with participating entities.

The architecture of the ANSI IRDS family includes four levels:

(1) IRD Schema Definition--defined by the implementor and contains the types of all objects that can be defined at the next lower level. Examples include the meta-entity-types like entity-type, relationship-type, attribute-type, and attribute-group-type.

(2) IRD Schema--defines the types that will appear in the IRD and also includes control mechanisms. Examples include the entity types, relationship types, attribute types, and attributed group types like element, record-contains-field, length, etc.

(3) IRD--describes the entities and relationships and their properties for the environment being modeled. Examples include entities, relationships, and attributes like employee-no, employee-uses-document, etc.

(4) "Real World" Information Resources--not part of the standard but rather the instances of the information resource environment. Examples include 222-222-2222, summary-record of 222-222-2222, etc.

These levels act as level pairs of type followed by instances of type.

In addition, there are schema maintenance and output commands, IRD maintenance and output commands, as well as IRD to IRD interface commands.

12.2.2. The ISO IRDS

This standard received final approval as IS 10728 in May, 1992. It is a call level programmatic interface, comparable in scope to the ANSI IRDS Services Interface. The standard is specified using a relational SQL data model; the major constructs include tables and columns. Although the facilities and structures can be closely mapped to respective features of the ANSI IRDS Services Interface, there are incompatibilities in such areas as IRDS security and version control.

12.3. IRDS II -- The Next Generation

Work is currently in process to develop the next generation of IRDS. There are several activities that include defining a framework and reference model, services, administration services, information models, and IRDS conceptual schema. As part of this effort, requirements are being developed, as well as strategic position and objectives for IRDS II. The current thinking on standardization objectives and IRDS objectives is extracted from a working document and presented in the following two sections.

12.3.1. Objectives of IRDS Standardization Activities

The major objectives of the IRDS standardization activities include:

a. To enable the adoption of a single, world-wide standard that satisfies the known needs of enterprises for managing their information resources.

b. To provide specifications that facilitate the cost-effective implementation and utilization of IRDS technology.

c. To provide a cohesive and comprehensive model of IRDS technology standardization requirements.

d. To identify services and functionality that will enable the effective and efficient management and utilization of enterprise information resources, minimize the cost of designing, developing, operating, and maintaining business functions and information systems, enable an open market for tools, and enable the quality management of IRDS functionality and content.

12.3.2. Objectives of IRDS II

The objectives of IRDS II include:

a. To enable the effective and efficient management and utilization of enterprise information resources. This will be accomplished by providing specifications that enable the:

- effective management and utilization of the representation of information resources;
- effective integration of an enterprise model and submodel;
- sharing of information resources;
- reuse of information resources;
- elimination of redundant information resources;
- describing of all models of information resources to allow for effective utilization of multiple modeling paradigms, representing the dynamics of business functions, and automated systems and processes, modeling all data types, and the evolution of models and modeling paradigms;
- simulation of business functions, and automated systems and processes;
- interoperability of IRDS; and,

- analysis of impact on the enterprise information environment caused by changes to information resources.

b. To minimize the cost of developing, operating, and maintaining information systems. This will be accomplished by providing specifications that facilitate:

  - providing sharable, reusable and nonredundant information about information systems throughout their life-cycle;
  - sharing of descriptions of information systems within and between enterprises; and,
  - managing the evolution of information systems.

c. To enable an open market for tools by providing specification for:

  - interfaces that allow for bulk transfer of IRDS managed data (schema and data);
  - interfaces that enable access to and management of IRDS managed data by local and distributed tools; and,
  - toolset integration with respect to the information being shared by tools belonging to the toolset.

d. Enable the quality management of IRDS functionality and content by providing specifications for:

  - the automatic enforcement of policy, standards, conventions and metrics with regard to IRDS content;
  - tool access to policies, standards, conventions, and metrics;
  - unambiguous identification of information resource descriptions and models (schema and data);
  - controlling access to IRDS content and functionality; and,
  - managing the quality of IRDS content, including controlling the effect of multiple, concurrent, distributed processors on IRDS content.

12.3.3. IRDS II Technology

12.3.3.1. IRDS II Framework

Human and machine users interact through various applications or directly with the IRDS through IRDS technology components that access the IRDS objectbase. Network services can also "plug in" with the "plug and play" feature.

The IRDS II framework includes a services interface system with a client server architecture coupled with a kernal subsystem and technology interface.

There are also various "plug and play" subsystems that include IRDS reference management, IRDS model management, IRDS directory management, and other future IRDS extensions subsystems. The reference management subsystem includes tools like an index, dictionary, glossary, encyclopedia, thesaurus, etc. Model management subsystem manages various models and support integration and translation. The directory management subsystem provides address and content management of data resources.

12.3.3.2. IRDS II Schema

The conceptual schema for IRDS II uses a conceptual graph data model to represent the conceptual schema for the IRDS. This model supports both data and process or behavior aspects of a system. The major constructs include concepts and conceptual relations. Type hierarchies are used for both the concepts and conceptual relations. There are also various operations that can be performed at various levels of description of the schema.

The conceptual schema is founded on the assumption that meaning of information can be separated from the specific languages used to represent it. Thus the capturing of meaning or semantics of information is a major goal to effectively communicate information.

The IRDS conceptual schema is based on logic, mathematics, linguistics, and other disciplines.

The IRDS II conceptual schema taxonomy includes various dimensions: layers of schema definition, types of model expressiveness, three schema architecture, views and view integration, life-cycle phases, and model generality.

The layers of schema definition include the IRDS Definition Schema (defines the primitive concepts, object types and operations, as well as the fundamental lexical and syntactic categories), the IRDS Normative Schema (provides the complete set of formal modeling constructs), Modeling Schemas (define the framework for capturing the conceptual content of a model), and Application Schemas (define the types of objects that exist in some domain of discourse).

The levels of description of the IRDS include IRDS Definition (defines the structure and capability of an IRDS), IRDS (describes the information systems), Information Systems (databases and processing systems that describe the enterprise objects), and Enterprise (actual objects in the real world).

The IRDS can describe itself and can be one of the information systems that belongs to an organization.

In order to meet a wide variety of requirements, different languages can be used to represent the conceptual schema. There are many existing languages that will need to be mapped to the IRDS Normative Languages (conceptual graphs, predicate calculus, KIF (knowledge interchange format, or some other logic based language). Stylized natural languages can also be mapped. The normative languages are derived from the defining language (predicate calculus).

12.4. Summary

The overview of IRDS technology has presented the ANSI IRDS family, the ISO IRDS, and IRDS II. The major differences include the data modeling approach, the expressive capability, and the functions and services provided.

12.5. References

ANSI, American National Standard X3.138-1988. *Information Resource Dictionary System.* American National Standards Institute, New York, 1988.

ANSI, American National Standard X3.185-1992. IRDS Services Interface, American National Standards Institute, NY, 1992.

ANSI, American National Standard X3.195-1991. IRDS Export/Import File Format, American National Standards Institute, NY, 1991.

Berube, J., A Compatibility Analysis of the ANSI and ISO IRDS Services Interfaces, NISTIR 4904, National Institute of Standards and Technology, Gaithersburg, MD, 1992.

Goldfine, A., and Konig, P., A Technical Overview of the Information Resource Dictionary System (Second Edition), NBSIR 88-3700, National Bureau of Standards, Gaithersburg, MD, 1988.

ISO, *Concepts and Terminology for the Conceptual Schema,* ISO TR9007, 1987.

ISO, *Information Technology - Information Resource Dictionary System (IRDS) Framework,* International Standard ISO/IEC 10027, 1990.

ISO, Information Technology——Information Resource Dictionary System (IRDS) Services Interface, IS 10728, 1992.

Sowa, J.F., "Towards the Expressive Power of Natural Language", in *Principles of Semantic Networks* (J.F. Sowa, ed.), Morgan Kaufmann Publishers, San Mateo, CA, 1991, pp. 157-189.

APPENDIX A: Definitions of Data Administration

(See Appendix C for full citations)

ANSI X3.172-1990: The function of controlling the acquisition, analysis, storage, retrieval, and distribution of data. Synonymous with data management.

ARMY REGULATION AR25-9 (data administrator): An individual or organizational unit generally responsible for the design, modeling, analysis, and management of data.

NIST SP 500-173: The management of information describing the data, functions, operations, and structure of automatic data processing systems (ADP) and databases. Data administration collects descriptive information about an organization's data, functions, and operations to support the development of the ADP systems, to provide full system documentation during systems operation and maintenance, and to support redesign efforts required as the needs of an organization evolve.

DEPT OF INTERIOR 375: A function including development and coordination of the policies, procedures, practices, and plans for the capture, correction, storage, and use of data.

Durrell: The overall objective of data administration is to plan, document, manage, and control the information resources of an entire organization. The role of data administration is to integrate and manage corporation-wide information resources by utilizing data dictionaries and well-designed data structures. Data administration determines the contents and boundaries of each database. Data administration should develop a business systems plan for data usage throughout an organization.

Martin: (Data Administrator) is the individual with an overview understanding of an organization's data. The function is responsible for the most cost-effective organization and use of an enterprise's data resources. The data administrator is ultimately responsible for designing the data model and obtaining agreement about the definitions of data which are maintained in the data dictionary.

Date: Data Administration determines the data which should be stored in the database, and establishes policies for maintaining and dealing with that data once they have been stored. Data

Administration is responsible for understanding the data, and the need of the enterprise with respect to the data, at a senior management level.

Bureau of Land Management (BLM): The administrative function of planning the data resource and establishing policies and procedures for managing the BLM's data as a corporate resource. It includes carrying out the processes of strategic data planning, data and process modeling (both conceptual and logical), and developing policies and procedures to meet managers' and users' existing and future information needs.

NBS SP 500-152: The responsibility for definition, organization, supervision, and protection of data within an enterprise or organization.

APPENDIX B: Model Data Administration Standards Glossary

This glossary is based primarily on ANSI X3.172-1990 (same as FIPS Pub 11-3, Dictionary for Information Systems).

Abbreviated Name

The official short name or acronym of the name of the data element.

Active Data Dictionary System

Data dictionary system that controls access to the database system.

Alias

An alternate name for a dictionary member. Aliases can be either specific or general. A specific alias is a named alias, such as a COBOL, JCL, or IMS. A general alias is any alternate name.

Analysis

The methodical investigation of a problem and the separation of the problem into smaller related units for further detailed study.

Analyst

A person who defines problems and develops procedures for problems' solutions.

Application

A particular kind of work that a user performs on a computer.

Archival Database

A historical copy of a database saved at a significant point in time for use in recovery or restoration of the database.

Attribute

A descriptive characteristic or property of the data.

Attribute Type

A specified class of attributes, each of which is associated in the same way with a member of one class of entities.

Automated Information System (AIS)

A combination of information, computer, and telecommunications resources and other information technology that collects, records, processes, stores, communicates, retrieves, and displays data.

Automatic Data Processing

Data processing by means of one or more devices that use common storage for all or part of a computer program, and also for all or part of the data necessary for execution of the program; which execute user-written or user-designated programs; perform user-designated symbol manipulation such as arithmetic operations, logic operations, or character-string manipulations; and execute programs that modify themselves during their execution. Automatic data processing may be performed by a standalone unit or by several connected units.

Batch Processing

The processing of data or the accomplishment of jobs accumulated in advance in such a manner that the user cannot further influence the processing while it is in progress.

Character

A member of a set of elements that is used for the organization, control, or representation of information; the elements may be letters, digits, punctuation marks, or other symbols.

Class Attributes

General information regarding entity classes.

Class Word

Basic nature description of a data item (e.g., name, number).

Classification

A reference to a grouping of data elements or objects.

DA Team Repository Manager

An individual who creates subsets/views of the corporate model for access by project teams; provides enterprises with access privileges to repository subsets/views; establishes procedures for and executes backup, recovery, and update of the repository; coordinates with the Project Team Repository Managers to ensure approved model changes are reflected in the corporate repository after integration, and to ensure team repositories are refreshed with updated corporate repository after integration; and provides reports to DA of repository contents.

Data

Facts or figures from which a conclusion can be drawn. Data representation of facts, concepts, or instructions in a formalized manner suitable for communication, interpretation, or processing by humans or by automatic means. Any representations such as characters or analog quantities to which meaning is, or might be, assigned.

Data Administration

The function of controlling the acquisition, analysis, storage, retrieval, and distribution of data; and the responsibility for definition, organization, supervision, and protection of data within an enterprise or organization.

Data Administrator

A person or group that ensures the utility of data used within an organization by defining data policies and standards, planning for the efficient use of data, coordinating data structures among organizational components, performing logical database designs, and defining data security procedures.

Database Administration

The process of controlling the content, design, and use of one or more databases to avoid uncontrolled redundancies and to enhance development.

Database Administrator

The person or group that defines, organizes, manages, controls, and protects a database.

Database Format

The notation of the internal storage format (picture) of a data element according to the programming language convention.

Data Communication

The transfer of data between functional units by means of data transmission according to a protocol.

Data Definition Language

A programming language used to define the logical and physical structure of a database; that is, the language used for defining the database schema.

Data Definition

The process of creating a schema by identifying and describing data elements and their relationships that make up the database structure.

Data Dictionary

A logically centralized repository of all definitive information about the relevant data in an enterprise, including characteristics, relationships, usage and responsibility. It can also be an automated facility that supports the data administration function by supporting the logical centralization of data about data.

Data Dictionary Controller

Individual responsible for the operation of the data dictionary. The controller's responsibilities include system backup, system security, access control, upload and maintenance of the UDS, member definitions, and new releases.

Data Dictionary System

A computer software system that maintains and manages a data dictionary.

Data Dictionary Systems Administrator

An individual who is responsible for the implementation and operation of data dictionary. Responsibilities include installation, system backup, creation of the dictionary structure, member definitions, and system upgrade to new releases.

Data Element

A basic unit of information having a meaning and subcategories (data items) of distinct units and values.

Data Element Definition

A logical description of a unit of information within the organization's resources.

Data Encryption

A process that converts the characters of the data from a meaningful sequence into an apparently random sequence.

Data Encyclopedia

A database consisting of metadata (data about the organization's data elements).

Data Flow Diagrams (DFD)

Method of modeling the flow of data from process to process.

Data Glossary

A reference document that lists all of the data elements stored in a database and provides for each element a definition of its meaning and specification of its uses in that database; the glossary may be included in a data dictionary, or it may be published separately for easy reference.

Data Integrity

The state that exists when data is handled as intended and is not exposed to accidental or malicious modification, destruction, or disclosure.

Data Item

A subunit of descriptive information or value classified under a data element.

Data Management

The function of managing data used in manual or automated information systems. It includes the activities of strategic data planning, data element standardization, information management control, data synchronization, and database development and maintenance.

Data Model

(1) The user's logical view of the data in contrast to the physically stored data. (2) A description of the organization of data in a manner that reflects the information structure of an enterprise.

Data Modeling

The process of identifying the data, its attributes, and relationships or associations with other data.

Data Object

An element of data structure such as a file, an array, or an operand, that is needed for the execution of programs and that is named or otherwise specified by the allowable character set of the language of the program.

Data Protection

The establishment and enforcement of appropriate administrative, technical, or physical means to guard against the unauthorized interrogation procedures and use of data.

Data Quality

The correctness, timeliness, accuracy, completeness, relevance, and accessibility that make data appropriate for use.

Data Relatability

Ability to establish logical relationships between different types of records, usually in different files.

Data Resource Management

The responsibilities for planning and controlling the data resources and functions of an organization which relate to collecting, cataloging, processing, storing, communicating, and disposing of data consistent with the overall goals and objectives of an enterprise.

Data Security

The protection of data from accidental or intentional modification or destruction and from accidental or intentional disclosure to unauthorized personnel.

Data Validation

The checking of data for correctness or compliance with applicable standards, rules, and conventions.

Data Value

An instance of a data item. One of the allowable values of a data element.

Database

A collection of interrelated data, often with controlled redundancy, organized according to a schema to serve one or more applications; the data are stored so that they can be used by different programs without concern for the data structure or organization.

Database Administrator

An individual or Organizational unit generally responsible for all physical activities relating to maintenance, storage, control, and modifications of databases. Provides information to the DA on organizational use of data within the subject database.

Database Identification

A predetermined outline for uniquely identifying related groups of information.

Database Management System (DBMS)

(1) An integrated set of computer programs that collectively provide all of the capabilities required for centralized management, organization, and control of access to a database that is shared by many users. (2) A computer-based system used to establish, make available, and maintain the integrity of a database, that may be invoked by non programmers or by application programs to define, create, revise, retire, interrogate, and process transactions; and to update, back up, recover, validate, secure, and monitor the database.

Distributed Database

(1) A database that is not stored in a central location, but is dispersed over a network of interconnected computers. (2) A database under the overall control of a central database management system, but whose storage devices are not all attached to the same processor. (3) A database that is physically located in two or more distinct locations.

Distributed Processing

Data processing in which some or all of the processing, storage, and control functions, in addition to input/output function are dispersed among data processing stations.

Distributed System

A system that performs distributed processing.

Document Administrator

The person who defines, organizes, manages, controls, and protects documents.

Domain

A description or enumeration of permissible instances of the corresponding data element concept.

Element

The basic unit of data that can be identified and described.

Entity Relationship Diagrams (ERDs)

Method of depicting the data as definable units which possess both distinctive qualities called attributes and relationships to other data items.

Entity Relationship Model

Graphic method of representing entities, attributes, and relationships.

Entity-Relationship Data Model

A data model based on the concept of entities and relationships among entities, and of the attributes of entities and relationships.

File

A collection of records, logical or physical. A physical file is also known as a dataset.

File Maintenance

The activity of keeping a file up to date by adding, changing, or deleting data.

Flowchart

A graphical representation in which symbols are used to represent such things as operations, data, flow direction, and equipment, for the definition, analysis, or solution of a problem. Synonymous with flow diagram.

Form of Representation

Name or description of the form of representation for the data element, e.g., "quantitative value," "coded," "picture."

Functional Analysis

A systematic investigation of the functions of a real or planned system.

Functional Area

A range of subject matter grouped under a single heading because of its similarity in use or genesis.

Functional Design

The design of the functional units of a system, regardless of their physical representations.

Functional Diagram

A diagram that represents the working relationships among the parts of a system.

Group

A collection of members which can be dealt with as a unit.

Imagery

Collectively, the representations of objects reproduced electronically or by optical means on film, electronic displayed devices, or other media.

Information

The meaning assigned to data by means of the known conventions used in their representation.

Information Analysis

A systematic investigation of the information and its flow in a real or planned system.

Information Analyst

An individual who performs processing data analysis and accesses team model and/or subsets of the model in order to recommend changes to project team's model.

Information Analyst Consultant

An individual who provides consultation on function and/or data modeling to project teams; assists project teams with Repository Management procedures and applying DA standards; and works with the Local Data Administrator to ensure standards are met and integration issues are resolved across project teams.

Information Architect

An individual who provides initial corporate information architecture, if applicable (i.e., the results of an enterprise-wide study, such as an Information Systems Plan or strategic data plan); establishes and coordinates update procedures; identifies issues impacting data integration; and recommends resolution of problems not resolved by integration review.

Information Interchange

The process of sending and receiving data in such a manner that the information content or meaning assigned to the data is not altered during the transmission.

Information Management

The application of general management principles including planning, directing, and controlling the processing, the handling, and the uses of an organization's information.

Information Modeler

An individual who provides the data modeling guidance; facilitates consistent data modeling techniques across project teams as issues arise; reviews proposed data model changes (in corporate repository) and advises integration review team concerning the impact of particular changes; and maintains the corporate data model to support integration.

Information Processing System

A system that performs data processing integrated with processes such as office automation and data communication.

Information Resource Dictionary (IRD)

A collection of the entities, relationships, and attributes that are used by an organization to model its information environment.

Information Resource Dictionary System (IRDS)

A computer software system that maintains and manages an information resource dictionary.

Information Resource Management (IRM)

The policy, action, and procedures concerning information, both automated and non automated, that management establishes to serve the overall current and future needs of an enterprise.

Information Resources

All information created manually or by automated means that an enterprise treats as a resource for decision making and problem solving.

Information System

(1) A combination of information, information technology, and personnel resources that collects, records, processes, stores, communicates, retrieves, and displays either manually or with varying degrees of automation.

(2) A system that consists of people, machines, and methods for organizations to accomplish specified operations on data that represent information. An information system may include data devices; office machines; communications equipment, peripheral equipment; and associated data media and accessories.

Input Process

The process of transmitting data from peripheral equipment or auxiliary storage to internal storage.

Keys

An identifier within a set of data elements.

Keywords

One or more predefined words used for searching, accessing, or retrieving information from an automated system.

Layout of Representation

The layout of characters used in a character string.

Local Data Administrator

An individual who identifies data modeling and data integration issues affecting project teams; facilitates resolution of issues affecting multiple teams (this usually based on the assignment of the DA to specific project teams, either by organization or some other grouping); identifies data modeling and data integration issues affecting the corporate data model; and assists project teams with information analysis and data modeling.

Management Information System (MIS)

The total flow of information within an enterprise that supports the decision-making functions of management at all organizational levels of the enterprise.

Medium/Media

The means or method by which the data is conveyed or transferred from process to process.

Metadata

Information about an organization's information and data activities. This includes the characteristics, resources, usage, activities, systems and holdings of data. Metadata are stored in a data dictionary.

Model

Graphic depictions of information and its relationships.

Models

A subset of the repository containing both entity types and entities.

Modification

An addition, change, or deletion of stored data.

Modifier

In data naming, a detail word about a data item.

Name

A given name or words which identify a data element within one dictionary of data elements.

Nonautomated

Manual, without benefit or hindrance of machines.

Notation

A set of symbols and the rules for their use, for the representation of data.

Organizational Model

An approach to data management based on normalized tables of data and their manipulations which includes relationships between data entities.

Passive Data Dictionary System

Data dictionary system that has no control over database system access.

Password

A character string that enables a user to have fuli or limited access to a system or to a set of data.

Physical Locations

An actual area or address where data resides on storage media.

Prime Word

Functional or logical word about a data item.

Programming Name or Symbolic Name

A shortened name of the data element in conformity with the programming language conventions and used in programs.

Project Data Manager

An individual who leads data analysis for project team; assists project team members with data analysis, data modeling, and implementation of DA standards; and identifies and resolves data modeling and data integration issues.

Project Planning

The activities concerned with the specification of the components, timing, resources, and procedures of a project.

Project Team Leader

An individual who plans, coordinates, and monitors progress of project team's efforts; leads analysis and design by applying appropriate techniques; reviews/approves changes to the team's model; and identifies and resolves issues within the project team.

Protection Category

The indication of the level of information protections, including methods for protection if applicable (e.g., "encrypted").

Qualified by

A unique reference to one or more data elements within the same dictionary by which the meaning of the data element can be qualified.

Qualifier of

A unique reference to one or more data elements within the same dictionary of which the meaning can be qualified by this data element.

Quality Assurance

(1) The planned systematic activities necessary to ensure that a component, module, or system conforms to established technical requirements. (2) All actions that are taken to ensure that a development or organization delivers products that meet performance requirements and adhere to standards and procedures. (3) The policy, procedures, and systematic actions established in an enterprise for the purpose of providing and maintaining some degree of confidence in data integrity and accuracy throughout the life cycle of the data, which includes input, update, manipulation, and output.

Range

A set of permissible values.

Record

A group of related data elements treated as a unit.

Recovery

A process in which a specified data station resolves conflicting or erroneous conditions arising during the transfer of data.

Reengineering

Concept of taking existing applications, analyzing them for logic, process, and data manipulation, in order to create new applications that do the same work more efficiently.

Relation

An association between two members.

Relationship

A special type of entity that is used to indicate a dependency and association, or a link that may be inherent between two entities or among attributes of the same entity, and that is represented or recorded in a database.

Relationship & Data Flows

A representation of logical connections concerning units of information.

Relationship Type

A specified class of relationships, each of which is associated in the same way with a member of one class of entities.

Reorganization

A major change in the way a database is logically or physically structured.

Repository

A collection of all the organization's information resources.

Repository Manager

An individual who is responsible for creating subsets/views of the team's model; provide team members with access privileges to the team's model subset/views; establish procedures for and executes backup, recovery, and update of the team's model; coordinates with the DA Repository Manager to ensure approved model changes are reflected in the corporate repository after integration and to ensure team's repository is refreshed with updated corporate repository after integration; and provides reports to project team based on model contents.

Representation Category

Names or description of the category of representation for the data element.

Semantics

The relationships between symbols and their meanings.

Sequential Batch Processing

A mode of operating a computer in which a run must be completed before another run can be started.

Set

A finite or infinite number of objects, entities, or concepts, that have a given property or properties in common.

Standard

An exact value, a physical entity, or an abstract concept established and defined by authority, custom, or common consent to serve as a reference, model, or file in measuring quantities or qualities; establishing practices or procedures; or evaluating results.

Standardization

The need for common definitions of data items, in terms of both the precise definition of a data item name and its storage format in the database.

State Transition Diagrams (STDs)

Method of depicting the sequence of changes to data and the events which trigger the changes.

Strategic Data Planning

The establishment of a long-term direction for the effective use of data in support of the mission, objectives and processes of the enterprise.

Strategic Data Planning Approach

Concept to support the mission, goals, and purpose of the enterprise.

Structural Relations

The attributes defined for the structural relations between data elements and the other system and/or message elements used for controlling the structures of application systems, edit messages and data models.

Structured Diagrams

Graphic depictions which focus on data from several aspects. Three common used structured diagrams are DFD, ERD, and STD.

Subset

A set, each element of which is an element of another specified set.

Symbology

Any graphic representation of concepts or physical objects.

Synonymous Name

A name or word(s) that differs from the given name but has the same meaning.

System

People, machines, and methods organized to accomplish a set of specific functions.

System Administrator

Individual responsible for the implementation of a data administration environment.

System Analysis

A systematic investigation of a real or planned system to determine the functions of the system and how they relate to each other and to any other system.

System Definitions

Descriptions of related applications within an organization.

System Description

Documentation that describes the system design and that defines the organization, essential characteristics, and the hardware and software requirements of the system.

System Design

A process of defining the hardware and software architecture, components, modules, interfaces, and data for a system to satisfy specified requirements.

System Development

A process that begins with requirement analysis and includes system design, implementation, and documentation.

System Documentation

The collection of documents that describe the requirements, capabilities, limitations, design, operation, and maintenance of an information processing system.

System Integration

The progressive linking and testing of system components into a complete system.

System Integrity

The state that exists when there is complete assurance that, under all conditions, an automatic data processing system is based on the logical correctness of the hardware and software that implement the protection mechanisms and data integrity.

System Support

The continued provision of services and material necessary for the use and improvement of a system after the system has been adopted.

Transaction

(1) A command, message, or input record that explicitly or implicitly calls for a processing action, such as updating a file. (2) An exchange between an end user and an interactive system. (3) In a database management system, a unit of processing activity that accomplishes a specific purpose such as a retrieval, an update, a modification, or a deletion of one or more data elements of a storage structure.

Transaction File

A file that contains relatively transient data, that, for a given application, is processed together with the appropriate master file.

Transaction Processing

A sequence of operations on a database that is viewed by the user as a single, individual operation.

Transfer

(1) In text processing, the movement of selected recorded text from one element of a recording medium to another. (2) To send data from one storage location to another.

Type of Characters

The maximum number of characters that appear in an element of the character representation must be specified. Possible types are: "alphabetic," "alphanumeric," and "numeric."

Unique Identifier

A unique, language independent, identifier (tag) of a data element within a dictionary.

User

Any person referencing or populating data.

User Manual

Documentation that describes how to use a functional unit, and that may include description of the rights and responsibilities of the user, the owner, and the supplier of the unit.

Validation

Tests to determine whether an implemented system fulfills its requirements.

Validation/Verification rules

The rules/instructions applied for validating/verifying data elements occurring in actual communication and/or databases, in addition to the formal screening based on the requirements laid down in the basic attributes.

Variable

A quantity that can assume any of a given set of values.

Verification

Tests of a system under development to prove that it meets all of its specified requirements for a particular stage of the system life cycle.

Verify

To determine whether a transcription of data or other operation has been accomplished accurately.

APPENDIX C: Bibliography

Braithwaite, K. S., Data Administration: Selected Topics of Data Control, New York, John Wiley and Sons, 1985.

Durell, William R., Confessions of a Data Administrator, Data Administration Inc., 301 N. Harrison Street, Bldg B, Suite 203, Princeton, NJ 08540.

Durell, William R., Data Administration: A Practical Guide to Successful Data Management, New York, McGraw-Hill Book Company, Inc., 1985.

Durell, William R., Data Administration Handbook, McGraw-Hill Book Co., 1985.

Durell, William R., The ABC's of Information Resource Management, Data Admin. Inc., 301 N. Harrison Street, Bldg B, Suite 203, Princeton, NJ 0854.

Fleming, Candace C. and von Halle, Barbara, Handbook of Relational Database Design, Addison-Wesley Publishing Company, 1989.

Holloway, Simon, Data administration, Aldershot, Hants, England; Gower Technical Press, Brookfield, VT, USA, 1988.

Inmon, W. H., Data Architecture: The Information Paradigm, QED Information Sciences, Inc., QED Plaza, 170 Linden St., Wellesley, MA, 1989.

Jones, Dennis P., Data and Information for Executive Decision in

Higher Education, National Center for Higher Education Management Systems, Bolder, 1982.

Lefkovits, Henry C., Sibley, Edgar H., and Lefkovits, Sandra L., Information Resource/Data Dictionary Systems, QED Information Sciences, Inc., QED Plaza, 170 Linden St., Wellesley, MA, 1983.

Leong-Hong, Belkis W. and Plagman, Bernard K., Data Dictionary/Directory Systems: Administration, Implementation and Usage, John Wiley and Sons, Inc., 1982.

Martin, James, Strategic Data-Planning Methodologies, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1982.

Martin, James, Managing the Data-Base Environment, two volumes, Technical Report No. 11, Savant Institute, 2 New Street, Carnforth, Lancashire, England, 1980.

Martin, James, and Finkelstein, Clint, Information Engineering, two volume report, Savant Institute, 2 New Street, Carnforth, Lancashire, England, 1982.

McLaughlin, Gerald W., McLaughlin, Josetta S., and Howard, Richard D., Decision Support in the Information Age, Managing Information in Higher Education: New Directions for Institutional Research, No. 55, Fall 1987, 81-91.

Nolan, Richard, Managing the Data Resource Function, West Publishing, 1974.

Ross, Ronald G., Data Dictionaries and Data Administration, AMACOM, 135 West 50th Street, New York, NY, 1981.

Ross, Ronald G., Entity Modeling: Techniques and Application, Database Research Group, Inc., 129 Tremont Street, Suite 500, Boston, MA 1987.

Turk, Tom, Planning and Designing the Database Environment, Van Nostrand Reinhold, 1985.

Wiersma, William, Research Methods in Education: An Introduction, 2nd ed, Itasca, Il, F.E. Peacock Publishers.

VIDEO

Data Architecture: An Information Systems Strategy, W. H. Inmon, QMS Information Sciences, Inc., QED Plaza, 170 Linden St., Wellesley, MA, 1990. 120-minute VHS tape. 0-89435-357-8.

PERIODICALS

Data Architecture: The Newsletter for Corporate Data Planners and Designers, Data Modeling Group, 1380 North Crossing Dr., Atlanta, GA 30329, (404) 633-9153, corresp to: PO Box 29673, zip 30359-0673, $120/year for six issues.

The masthead says: "DATA ARCHITECTURE is designed to provide accurate and authoritative information on data and process modeling."

Database, A Quarterly Publication of the Special Interest Group on Business Data Processing of the Association for Computing Machinery, ACM, 11 West 42nd St., New York, NY 10036.

Database Newsletter, editor Ronald Ross (617) 423-2029 $114/year.

Database Programming & Design 1-800-451-2248 $37/year.

DATAMATION 212-645-0067 Free.

Data Resource Management, Auerbach Publications (800) 950-1201, quarterly, $110/yr.

MIS Week 215-630-1522 Free to qualified individuals on a controlled basis.

ARTICLES

Ackley, Dave, "A True Database Environment," Computerworld, Volume 17, No. 6, Feb. 6, 1983, In Depth, pp. 53-61.

Adams, Mary M., "Data Administration Function at the Core of System Success," Computerworld, XVIII (September 24, 1984), SR/40.

Appleton, Daniel S., "The Technology of Data Integration," Datamation, Vol. 31, No. 21, Nov. 1, 1985, pp. 106-116.

Appleton, Daniel S., "Information Asset Management," Datamation, Vol. 32, No. 3, Feb. 1, 1986, pp. 71-76.

Appleton, Daniel S., "Rule-Based Data Resource Management, Datamation, Vol. 32, No. 9, May 1, 1986, pp. 86-99.

Beeler, Jeffry, "Data Liberation: Dama on the Barricades," Computerworld, Volume 24, No. 35, Sept. 2, 1985.

Betts, Mitch, "Administrators: Integrate Your Databases," Computerworld, Sept. 24, 1984, pp. 16.

Christoff, Kurt A., "You Don't Need a DBMS to 'Do Database'," Infosystems, Jan. 1985, pp. 94-95.

Connall, Duncan G., "Choosing Standard Dictionary Names," Database Newsletter, Vol. 12, No. 2, March/April 1984, pp. 1-12.

Coulson, Christopher J., "The End User Challenge," ICP Data Processing Management, Vol. 9, No. 2, Summer 1984, pp. 16.

Curry, Diana, "The Customer Connection: Data Transfer at Indiana University," CAUSE/EFFECT, volume 12, number 1 (Spring 1989), 42-47.

Curtice, Robert M., "Normalization Facilitates Data Design Process," Computerworld, Vol. XIX, No. 25, Special Report, SR/14, June 24, 1985.

Durell, William R., "Keepers of the Corporate Data," Computerworld, July 30, 1984, pp. In Depth, Page ID/5.

Durell, William R., "The Politics of Data," Computerworld, Sept. 9, 1985, Vol. 19, No. 36, In Depth (ID)/25.

English, Larry P., "Organizing and Managing Data Administration," CDP paper, January 27, 1986.

English, Larry P., "Results of the 1987 Advanced IRM Survey," Database Newsletter, Vol. 16, No. 1, Jan./Feb. 1988, pp. 1.

Faissol, Sergio, "Which Data is Which?" Computerworld, Vol. 21, No. 27, July 6, 1987, pp. 53.

Flavin, Matt, "The Fundamental Concepts of Information Modeling," Englewood Cliffs, NJ, Prentice-Hall, 1981.

Friedman, L. Jeanne, "Higher Purposes for Data Dictionaries," Computerworld, Vol. 21, No. 32, Aug. 10, 1987, pp. S17.

Gallant, John, "User Warns Against Using Multiple Data Dictionaries," Computerworld, Vol. XIX, No. 24, June 13, 1985, pp. 13.

Gilhooley, Ian A., "Managing the Corporate Database," Data Base Management, Auerbach Publishers, Inc.

Gillenson, Mark L., "Trends in Data Administration," MIS Quarterly, IX (December, 1985), 317-325.

Gillin, Paul, "Distributed Data: What You Can Do," Computerworld, Vol. 18, No. 46, Nov. 12, 1984.

Gillin, Paul, "Tool Brightens Steel Firms 'Black Hole,' Unleashed Data Dictionary's Information," Computerworld, July 11, 1983, Vol. 17, No. 28, pp. 55-58.

Gillin, Paul, "Subject Databases Prescribed for Application Portfolio Ills," Computerworld, Vol. 18, No. 46, Nov. 12, 1984.

Goodbody, Terry, "Information Systems Engineering," Computerworld, Vol. 17, No. 9, Feb. 28, 1983, pp. In Depth 9-18.

Gose, Frank, "Data Administration at a Regional University: A Case Study," presented at Rocky Mountain Association for Institutional Research, Flagstaff, AZ, October 7 - 9, 1987.

Gose, Frank, "Data Integrity: Why Aren't the Data Accurate?", presented at the Annual Association for Institutional Research Forum, Baltimore, MD, April 30 - May 3, 1989.

Green, Carolyn W., "The Importance of Measuring Results," Database Newsletter, Vol. 17, No. 1, Jan./Feb. 1989.

Hansen, William L., "Establishing a Data Administration Charter," Database Newsletter, Vol. 12, No. 6, Nov./Dec. 1984, pp. 10.

Haughey, Thomas F., "Information Engineering: Problems Impeding Acceptance," Database Newsletter, Vol. 16, No. 3, May/June 1988.

Hawker, Neil, "Data Administration: The Way to Make Management Aware," Datalink, July 30, 1984, pp. 8-9.

Heagerty, R. and Smith, T., "Data Administration - How to Triumph Over Your Pressing Problems," Computer Weekly, No. 888, Nov. 24, 1983, pp. 23.

Heagerty, R. and Smith, T., "The People Who Matter in the

Administration," Computer Weekly, No. 890, Dec. 8, 1983, pp. 19.

Heagerty, R. and Smith, T., "Why It Is a Good Thing to Treat Data Management with Care," Computer Weekly, No. 887, Nov. 17, 1983, pp. 24.

Howard, Richard D., Gerald W. McLaughlin, and Josetta S. McLaughlin, "Bridging the Gap between the Database and User in a Distributed Environment," CAUSE/EFFECT, volume 12, number 2. (Summer 1989), 19-25. ** see this article for additional references **

Holloway, Simon, "Data Administration in the Organization," Data Processing, XXVIII (May, 1986), 195-198.

Hunt, Kevin F., "Putting Life Into Data Administration," Data Management, Nov. 1983, pp. 36-39.

Inmon, W. H., "Data Management at the Crossroads," Data Management, Vol. 25, No. 10, October, 1987, pp. 17-18.

Inmon, W. H., "The Hardest Sell," Database Programming & Design, June 1988.

Kahn, Beverly K., "Some Realities of Data Administration," Communications of the ACM, Vol. 26, No. 10, Boston University, Boston, MA 02215, Oct. 1983.

Kahn, Beverly K., "An Environmentally Dependent Framework for Data Dictionary Systems," MIS Quarterly, Vol. 9, No. 3, Sept. 1985, pp. 199-220.

Kull, David, "Data Dictionaries Can Point the Way," Computer

& Communication Decisions, July 1987, pp. 69.

Lacob, Miriam, "Do You Need a Data Czar?" Computer Decisions, Vol. 18, No. 6, March 1986, pp. 50-54.

Lipton, R., "Do Executives Need Access to Corporate Database?" Business Computer Systems, Vol. 4, No. 9, Sept. 1985, pp. 29-30.

Minami, Warren N., "Data Administration - Key to Better Management," in Journal of Systems Management, XXVII (May, 1976), 40-44.

Modell, Martin E., "Management's Perspective of Data Bases," Database Management, Auerbach Publishers, Inc.

Modell, Martin E., "Management's Perspective of Data Dictionary Systems," Database Management, Auerbach Publishers, Inc.

Moss, Clifton L., "The Information Systems Plan and the Data Base Plan," Journal of Information Systems Management, Vol. 1, No. 3, Summer 1984, pp. 20-26.

Nerad, Richard A., "Data Administration as the Nerve Center of a Company's Computer Activity," in Data Management, XI (October, 1973), 26-31.

Nesbit, Irene S., "On Thin Ice: Micros and Data Integrity," Datamation, Nov. 1, 1985, Vol. 31, No. 21, pp. 80-84.

Netherton, James, "An Apology for Data," CAUSE/EFFECT, volume 12, number 1 (Spring 1989),3,6.

Newton, Judith, "Observations on Data Element Naming Conventions," Proceedings, Trends and Applications 1984, IEEE Computer Society Press, pp. 153-157.

Newton, Judith, "Designing Data Entity Naming Conventions," Proceedings of the Sixth International Conference on Entity-Relationship Approach, November 1987, pp. 407-419.

Newton, Judith, "Developing and Applying Data Entity Naming Conventions," Data Resource Management v2, n4, Fall 1991, pp. 63-68.

Nolan, Richard L., "Managing the Crises of Data Processing," Harvard Business Review, March - April 1979.

Odell, James J., "Organizational Structure for Data Administration," Database Newsletter, Vol. 12, No. 6, Nov./Dec. 84, pp. 1.

Paul, Lois, "New Type of Data Management Tool Urged," Computerworld, Vol. 17, No. 24, June 13, 1983, pp. 11.

Paul, Lois, "Survey Finds Data Administration Flourishing," Computerworld, Vol. 17, No. 5, Jan. 31, 1983, pp. 8.

Paul, Lois, "Steps to Shared Database Success Outlined," Computerworld, May 25, 1981, pp. 23.

Ravindra, Palangala S., "Data Administration: An Old Function Adopts to its New Role," in Journal of Information Systems Management, III (Fall, 1986), 47-51.

Rollier, Bruce, "Two Teams, One Goal," Datamation, April 1982, pp. 182-184.

Ross, Ronald G., "Collision Course? The Information Center vs. Data Administration," Computerworld, Vol. 16, No. 5, PP: In Depth 1-4, Feb. 1, 1982.

Ross, Ronald G., "The Data Resource Stewardship Pyramid," Database Newsletter, Vol. 12, No. 2, March/April 1984, pp. 3-9.

Ross, Steven J., "Data Administration: Control, Structure, and Organization," Database Management, Auerbach Publishers, Inc.

Rush, Gary, "A Fast Way to Define System Requirements," Computerworld, Oct. 7, 1985, pp. In Depth, pp. 11.

Saltman, Roy G., "Data Element Standards: Communication Standards for End Users," Proceedings of Computer Standards Conference - 1986, IEEE Computer Society, pp. 34-37.

Schultz, Brad, "Data as a Corporate Resource: Divide & Conquer," Computerworld, December 28, 1981/January 4, 1982, Vol. 16, No. 1, pp. 61-63.

Shah, Arvind D. "Data Administration: It's Crucial," Datamation, Vol. 30, No. 1, Jan. 1984, pp. 187-188, 190, 192.

Sloan, Ken, "All Your Assets On-Line," Computerworld, Vol. XXII, No. 28, In Depth, July 11, pp. 55-58.

Sweet, Frank, "The Name Game," Reader's Forum, Datamation, Mar. 1, 1985, Vol. 31, No. 5, pp. 177-178.

Sweet, Frank, "When Should You Add to Your Dictionary?", Database Programming & Design, May, 1988, pp. 28-33.

Symons, C. R., and Tijsma, P., "A Systematic and Practical Approach to the Definition of Data," The Computer Journal (The British Computer Society), Vol. 25, 1982.

Tillmann, George D., "Data Administration Vs. Database Administration - There is a Difference," Infosystems, Vol. 31, No. 2, Booz Allen & Hamilton, Inc., New York, NY, Feb. 1984, pp. 98-102.

Tillman, George, "Why Data Administration Fails," Computerworld, Vol. 21, No. 36, Sept. 7, 1987, In Depth, pp. 73-76.

Vacca, J. R., "DA is Watchdog Against Data Proliferation," Information Systems News, Jan. 9, 1984, pp. 18, 20.

Viles, Art, "Business Systems Planning, A Newsletter Interview," Database Newsletter, Vol. 12, No. 6, Nov./Dec. 84, pp. 1.

von Halle, Barbara, "Twelve Steps for Getting Started On the Long Path Toward Data Sharing," The Relational Journal for DB2 Users.

Wahl, Dan and Hufford, Duane, "A Case Study: Implementing and Operating an Atomic Database," Data Resource Management, Vol. 3, No. 2, Spring 1992.

Weldon, Jay-Louise, "The Elusive Data Administration Function," Journal of Information Systems Management, Vol. 1, No. 3, Summer 1984, pp. 70-75.

Wilson, Andrew H. and Iyer, Shekar, "Data Resource Management," Canadian Datasystem, Nov. 1983, pp. 45, 51, 55.

Winsberg, Paul, "Smoothing the Data Administration Path," Computerworld, Vol. 22, No. 11, March 14, 1988, pp. S14-S15.

Womeldorff, Thomas, "The Pursuit of Data Integration," Computerworld, Vol. 18, No. 12, Nov. 12, 1984, pp. In Depth, ID/29.

Zachman, J. A., "A Framework for Information Systems Architecture," IBM Systems Journal, Vol. 26, No. 3, 1987, pp. 276.

Federal Information Processing Standards (FIPS), Information Resource Dictionary System (IRDS), FIPS PUB 156, Nat. Inst. Stand. Technol., Gaithersburg, MD, April 5, 1989

Fong, Elizabeth N., and Goldfine, Alan H., Editors, Information Management Directions: The Integration Challenge, NIST Special Publication 500-167, Nat. Inst. Stand. Technol., Gaithersburg, MD, September 1989

Fong, Elizabeth N., and Goldfine, Alan H., Editors, Database Directions: Information Resource Management --Making It Work, NBS Special Publication 500-139, Nat. Bur. Stand. (U.S.), Gaithersburg, MD, June 1986

Fong, Elizabeth N., Henderson, Margaret W., Jefferson, David K., and Sullivan, Joan M., Guide on Logical Database Design, NBS Special Publication 500-122, Nat. Bur. Stand. (U.S.), Gaithersburg, MD, February 1985

Goldfine, Alan H., Editor, Proceedings of the Federal Information Processing Standards (FIPS) Workshop on Information Resource Dictionary System (IRDS) Applications, NISTIR 88-3896, Nat. Inst. Stand. Technol., Gaithersburg, MD, December 1988

Goldfine, Alan H., Editor, Database Directions: Information Resource Management -- Strategies and Tools, NBS Special Publication 500-92, Nat. Bur. Stand. (U.S.), Gaithersburg, MD, September 1982

Goldfine, Alan H. and Kirkendall, Thomasin, The ICST-NBS Information Resource Dictionary System Command Language Prototype, NBSIR 88-3830, Nat. Bur. Stand. (U.S.), Gaithersburg, MD, August 1988

Konig, Patricia and Goldfine, Alan H., A Technical Overview of the Information Resource Dictionary System, NBSIR 88-3700 (Supersedes NBSIR 85-3164), Nat. Bur. Stand. (U.S.), Gaithersburg, MD, January 1988

Law, Margaret H., and Rosen, Bruce K., Guide to Data Administration, NIST Special Publication 500-173, Nat. Inst. Stand. Technol., October 1989

Law, Margaret Henderson, Guide to Information Resource Dictionary System Applications: General Concepts and Strategic Systems Planning, NBS Special Publication 500-152, Nat. Bur. Stand. (U.S.), Gaithersburg, MD, 1988

Newton, Judith J., Guide on Data Entity Naming Conventions, NBS Special Publication 500-149, Nat. Bur. Stand. (U.S.), Gaithersburg, MD, October 1987

Newton, Judith J. and Spielman, Frankie E., Data Administration: Management and Practice, Proceedings of the First DAMA Symposium, NIST Special Publication 500-159, Nat. Inst. Stand. Technol., Gaithersburg, MD, October 1988

Newton, Judith J. and Spielman, Frankie E., Data Administration, Standards and Techniques, Proceedings of the Second Annual DAMA

Symposium, NISTIR 90-4292, Nat. Inst. Stand. Technol., Gaithersburg, MD, April 1990

Newton, Judith J., The Naming Forum: Proceedings of the IRDS Workshop on Data Entity Naming Conventions, NISTIR 4374, Nat. Inst. Stand. Technol., Gaithersburg, MD, July 1990.

Newton, Judith J., Melley, Mary Lou, and Higgins, Holly, Managing Data: From Vision to Reality, Proceedings of the Fourth DAMA Symposium, NISTIR 4843, Nat. Inst. Stand. Technol., Gaithersburg, MD, May 1992.

Spielman, Frankie E., Editor, Data Administration Workshop Proceedings, NBSIR 86-3324, Nat. Bur. Stand. (U.S.), Gaithersburg, MD, February 1986

USER'S GROUPS

CAUSE, 737 Twenty-Ninth Street, Boulder, CO 80303, (303) 449-4430, CAUSE@COLORADO.BITNET. CAUSE is The Association for the Management of Information Technology in Higher Education.

Data Administration User's Group, PO Box 1137, Bayonne, NJ 07002, (201) 956-7200. They sponsor a Data Administration Conference (in San Francisco Nov 19-22, this past year). Cost was $625 ($575 for DAMA members).

GUIDE Internation Corp, 111 East Wacher Drive, Suite 600, Chicago, IL 60601, (312) 644-6610. GUIDE is an IBM User's Group which sponsors three conferences three times a year (March, July, and November). It publishes conference proceedings, Guide Project Papers (GPPs), and other publications.

GUIDE PUBLICATIONS

Data Administration Methodology, GPP-30 (No longer available).

Data Administration in a Distributed Data Environment, GPP-79, 1982.

Data Administration in Support of the Information Customer, GPP-156, 1986.

Data Resource Management Planning Methodology, GPP-84, 1982.

Establishing the Data Administration Function, GPP-66, 1977.

Selling Data Administration for the 80's and Beyond, GPP-198, 1988.

☆ U.S. GOVERNMENT PRINTING OFFICE:1993-341-832/84027

ANNOUNCEMENT OF NEW PUBLICATIONS ON INTEGRATED SERVICES DIGITAL NETWORK

Superintendent of Documents
Government Printing Office
Washington, DC 20402

Dear Sir:

Please add my name to the announcement list of new publications to be issued in the series: National Institute of Standards and Technology Special Publication 823-.

Name ______________________________

Company ______________________________

Address ______________________________

City ______________ State ______________ Zip Code ______________

(Notification key N-503)

NIST *Technical Publications*

Periodical

Journal of Research of the National Institute of Standards and Technology—Reports NIST research and development in those disciplines of the physical and engineering sciences in which the Institute is active. These include physics, chemistry, engineering, mathematics, and computer sciences. Papers cover a broad range of subjects, with major emphasis on measurement methodology and the basic technology underlying standardization. Also included from time to time are survey articles on topics closely related to the Institute's technical and scientific programs. Issued six times a year.

Nonperiodicals

Monographs—Major contributions to the technical literature on various subjects related to the Institute's scientific and technical activities.

Handbooks—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications—Include proceedings of conferences sponsored by NIST, NIST annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

Applied Mathematics Series—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

National Standard Reference Data Series—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a worldwide program coordinated by NIST under the authority of the National Standard Data Act (Public Law 90-396). NOTE: The Journal of Physical and Chemical Reference Data (JPCRD) is published bimonthly for NIST by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements are available from ACS, 1155 Sixteenth St., NW., Washington, DC 20056.

Building Science Series—Disseminates technical information developed at the Institute on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

Technical Notes—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NIST under the sponsorship of other government agencies.

Voluntary Product Standards—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The standards establish nationally recognized requirements for products, and provide all concerned interests with a basis for common understanding of the characteristics of the products. NIST administers this program in support of the efforts of private-sector standardizing organizations.

Consumer Information Series—Practical information, based on NIST research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

Order the **above** *NIST publications from: Superintendent of Documents, Government Printing Office, Washington, DC 20402.*

Order the **following** *NIST publications—FIPS and NISTIRs—from the National Technical Information Service, Springfield, VA 22161.*

Federal Information Processing Standards Publications (FIPS PUB)—Publications in this series collectively constitute the Federal Information Processing Standards Register. The Register serves as the official source of information in the Federal Government regarding standards issued by NIST pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

NIST Interagency Reports (NISTIR)—A special series of interim or final reports on work performed by NIST for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Service, Springfield, VA 22161, in paper copy or microfiche form.

U.S. Department of Commerce
National Institute of Standards and Technology
Gaithersburg, MD 20899

Official Business
Penalty for Private Use $300