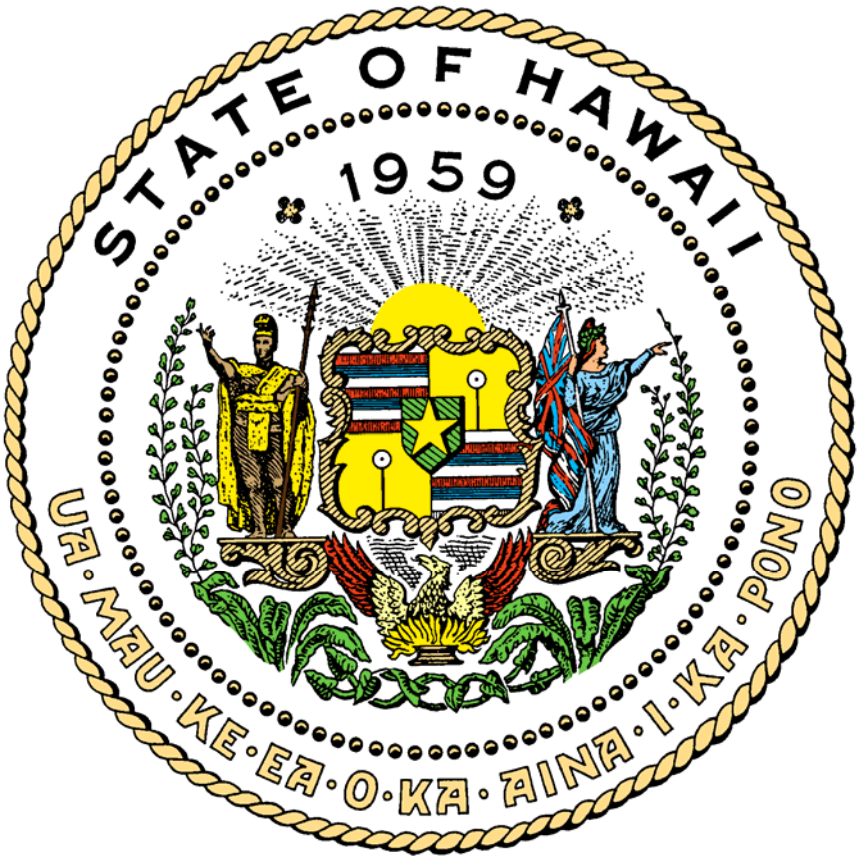
Number: 06.00



Database/Data Dictionary Overview

December 2004

TABLE OF CONTENTS

Department of Accounting and General Services
Information and Communication Services Division

Information Technology Standards

1 INTRODUCTION

This document presents the policies, procedures, standards, conventions, and guidelines for the use of databases (DB) and database management systems (DBMS), at the central-site information processing center of the State of Hawaii, Executive Branch, hereafter referred to as “the State”.

All data administration (DA) and database administration (DBA) functions and facilities for State owned or State controlled data and information will be centrally controlled, regulated and supported by the Department of Accounting and General Services (DAGS), Information and Communication Services Division (ICSD) at a central information processing center.

The contents of the established database conventions in this document are based on the collective experiences of many State users, contractors, and other organizations.

This document is organized into the following seven sections.

1. Introduction – the initial section that provides an overview of the standard.

2. Database Environment - defines the basic concepts and objectives for managing data in State owned or controlled database systems, as well as the physical and logical environments for databases at DAGS-ICSD.

3. Database Administration - describes the State’s statewide policies for the coordination and allocation of duties and responsibilities for database administration, as regards to effective and efficient management of databases.

4. Meta-Data - describes the State’s policies and objectives for the retention and use of meta-data within a state centralized and controlled data dictionary.

5 System Design and Development - provides standards and conventions to be followed for the design and development of application systems within a database environment.

6. Security - provides information on database access controls and standard protection security features for the State’s database environment.

7. Programming - provides standards, conventions and guidelines for the design and construction of program applications in a database environment.

1.1 Purpose

The purpose for this document is to present standards, conventions, and guidelines for the use of any database software installed and supported on the computer systems that is under the jurisdiction of DAGS-ICSD and at other state government agencies in the State of Hawaii Executive Branch of government.

This information technology (IT) standards document will enable user agencies in the State's Executive Branch to have structured, uniform, and consistent processes and procedures for the effective sharing of the State's data resources in order to better serve the people of Hawaii.

This document provides guidance for computer programmers, data processing systems analysts, contractors, and consultants who must design, develop, construct, test, and install programs, systems, and/or job streams that use DBMS in State owned or controlled computers.

1.2 Scope

This document provides an overview of preferred database software that are installed and supported at the State's owned or controlled computer facilities.

It does not address nor discuss any specific syntax, command, or feature associated with any particular DBMS software. Those topics are contained in other Information Technology (IT) Standards in the 06.0x series of the State's official IT Standards reference set.

1.3 Applicability

State computer programmers and data processing systems analysts and any contractor and vendor must follow the standards, procedures, conventions, and guidelines presented in this document. This restriction also includes any consultant hired by the State who will use the State's mainframe computing resources at the State's central computer site.

Failure to follow these procedures, standards, conventions and/or guidelines may result in computer systems error messages or unpredictable results and may result in delays and/or cancellation of a program or job stream.

1.4 Comments And Suggestions

Any State of Hawaii Information Technology (IT) Standards document, reference manual or users guide mentioned in this document are available through the

departmental user agency data processing coordinator (DP Coordinator). Many IT Standards documents may also be accessible on-line via the Internet by clicking on Information Technology Standards on the ICSD home page at:

http://www.hawaii.gov/icsd/

Statewide Forms are also accessible on-line by clicking on Forms Central on the Government in Hawaii home page at:

http://www.ehawaiigov.org/government/html/

Any comment, recommendation, proposal, or suggestion regarding the contents or presentation of this IT Standards document may be sent either via internet email to icsd.admin.ppmo@hawaii.gov or in writing to:

Information and Communication Services Division
Project Planning and Management Office
1151 Punchbowl Street, B10
Honolulu, Hawaii 96813-3024

2 DATABASE ENVIRONMENT

The database environment has been established at the State of Hawaii Executive Branch ("the State") central-site data processing center so that relevant and related data and information may be designed, structured, organized, stored and retrieved in such a manner so that they may be independent of any program or project that actually uses the data in the database.

The primary emphasis of these database IT Standards is focused on setting standard methods, processes, and procedures for the efficient use of shared and limited data information resources.

2.1 Definition: Database

A database (DB) is a collection of interrelated data elements that are grouped and stored together with a minimum of redundancy to effectively serve multiple application systems.

A common and controlled approach is used to add new data elements, modify existing data or data-entity, and access and retrieve data stored in the database.

The data elements are organized and structured to provide a flexible and

dynamic but stable foundation for future application system development.

2.2 Definition: Database Management System

A database management system (DBMS) is a set or family of computer systems software that are used “to manage data”.

The term, "to manage data" includes the processes of data definition, modification, documentation, status reporting, access and protection.

2.3 Structure And Function Of A Database And DBMS

The structure for a good database environment is based on establishing flexible, logical database designs and on creation of dynamic logical structures to provide the data with a stable foundation for growth, enhancement and modifications.

The ultimate logical structure of the data should be designed to serve many diverse potential/probable applications in the present and the future, and to allow for change to the database without changing the application programs.

Because future use cannot be fully foreseen, the database should be designed to represent the properties of the data as currently used by an organization.

The grouping of data elements and the associations between the data groups should be defined to reflect and represent the inherent structure, function, and properties of the data currently used in an organization.

The data elements and data groups should not reflect the processes, uses and procedures of application programs.

This organizational philosophy will thus provide stable data structures. These stabilized data elements can better retain their validity and relationships while their use and the physical structure of the files may change.

2.4 Database Objectives

The State’s main objectives for database driven application systems are to:

a. Provide flexibility of access of information stored in a database to implemented application systems.

b. Maximize the efficiency and performance of production systems.

c. Facilitate the sharing of data among different application systems by providing processes that are independent data structures.

d. Reduce data repetition and information redundancy.

e. Maximize consistency and reliability of data between application systems.

f. Maximize the accuracy, integrity, and security of the data maintained within the database.

g. Provide end users with effective access to their data and information.

h. Standardize technical support for end-user computing.

2.5 ICSD'S Database Environments

The State's central data processing center currently supports database management systems (DBMS) accessed by three database products, ADABAS, DB2, and Oracle.

ADABAS, adaptable database, is the State's first DBMS designed to manage and control high demand access requests for very large data files that reside on large centralized mainframe computer systems. Software AG provides ADABAS DBMS.

DB2 is the State's first fully relational DBMS designed to manage and control access requests for large centralized mainframe computer systems and smaller database servers using SQL commands. IBM provides DB2 DBMS

Oracle is a fully relational DBMS designed to manage and control access requests for large centralized mainframe computer systems and smaller database servers using SQL commands. Oracle provides the Oracle DBMS.

The databases reside on various platforms running different Operating System. The DAGS-ICSD supports both test databases and production databases.

SQL stands for "Structured Query Language". It is the industry standard language for relational database management systems. SQL statements are used to perform tasks, such as, update data on a database, and retrieve data from a database.

2.5.1 Test/Development Databases

The purpose for having controlled test/development databases is to

facilitate a flexible and dynamic environment that simulates and supports the development, production; target; and controlled environment for designing, debugging, testing stable, verifiable and reliable database applications or systems by authorized application systems data processing systems analysts and computer programmers.

As part of the State's information security control policy, production data shall not be used to test programs, systems, or projects.

Sensitive, private, or confidential information contained in live, real-time or production data files shall never be selected when extracting typical data elements from production data files to generate data elements for test data files (see IT Standard 08.02 Information Security).

An application system project shall use only one computer platform with test or development database/data files to carry out its design, construction, debugging, testing and training phases.

If separate databases are more effective for testing or development purposes, the personal microcomputer (PC) based version of the database DBMS should be considered as a cost/beneficial approach for this purpose.

The test/development databases are available and supported by the DAGS-ICSD-DBMS during the weekdays (Monday through Friday) from the hours of 7:00 a.m. through 6:00 p.m.

While a particular test/development database may be active and on-line for operational systems reasons outside of these hours, the database will not be accessible for the test/development environment outside of the standard Monday through Friday workday periods, nor on weekends or State holidays.

2.5.2 Production Databases

The purpose for having controlled-access production databases is to ensure a protected and reliable environment for running statewide production applications that need access to secured data that may be shared by a variety of users or agencies.

The state's database policy requires that any production application that uses shared databases must be designed so that data and information is accessed in a secured, reliable, efficient and controlled manner.

Several different production application systems, programs, and/or projects may share production databases on shared computer platforms that use shared resources and may be shared by a number of users who may be in various geographically remote departmental agencies.

As part of the State's information security control policy, production data files shall only be used for production applications and/or systems (see IT Standard 08.02 Information Security).

The selection of the computer host to place an agency's production database files is determined by the DAGS-ICSD-DBA staff based primarily on the concern for workload balancing.

The production databases are available, and supported by the DAGS-ICSD-DBA staff 24 hours per day, seven days a week with the exception of downtime for hardware or software systems maintenance.

The failure to follow conventions and procedures in this document will not only degrade the performance of application systems, but will also impact the efficiency and performance of other production application systems in the shared database environment.

2.5.3 Use Of Database Utilities

The type of utilities run by the users and the DAGS-ICSD-DBA will vary depending on the computer platform and the DBMS software used.

More types of utilities are available to the users in the Test/Development environment than in the Production environment.

The method of securing access to DB utilities also varies. For example, the tool, Securitre, is used on the mainframe to secure ADABAS utilities.

For more details on what utilities are available or how they are secured, refer to the State established IT Standard associated with a particular database software package.

The DAGS-ICSD-DBA staff provides and maintains system utility procedures, scripts, and sample JCL for their use (see the vendor reference manuals and users guides for each specific software).

2.5.4 Backup And Restore Of Databases

The DAGS-ICSD-DBA staff will schedule and run the "Daily Back-up" of all production databases at the central computer site unless an alternate schedule as been agreed to by the owner of the data.

The DAGS-ICSD-DBA staff may also schedule a "Weekly Back-up" of each production database at the central computer site.

The time that the production database backups are run will vary depending on the computer platform and the requirements of the agency that "owns" the data.

The specific number of retained generations of daily backup and weekly backups will be specified in the State established IT Standard associated with a particular database software package.

Every day a copy of the most current backup-generation will be taken to an offsite storage-site for disaster recovery purposes.

A request to restore a production database file from a DAGS-ICSD backup media should only be made if the file is irreparably damaged.

The backup generation data files will not be available for any other purpose, such as, to rerun test or production batch programs.

Project team members are responsible for designing and developing appropriate restart and rerun procedures for any batch job.

2.5.5 Training For Database Software

The DAGS-ICSD-DBA will conduct or will arrange for classes for using the State supported database software.

To schedule participants for any of these announced classes, send an email request for the training to DAGS-ICSD Systems Support Services Branch (SSB).

2.6 DBMS Request System

The "DBMS Request System" (Request System) is an on-line web-based application that is running on the DAGS-ICSD intranet site.

The purpose for the DBMS "Request System" is to give the authorized Departmental Database Liaison (DDL) a tool to submit and track their requests for services that require assistance from DAGS-ICSD-DBA personnel.

For more information related to the DDL, see 3.5 Departmental Database Liaison Duties and Responsibilities in this document.

2.6.1 Usage By The DDL

Any request by the Departmental Database Liaison (DDL) for any database related service from any member of the DAGS-ICSD-SSB DMS must be made through the on-line DBMS "Request System".

a. The DDL must submit a request for access to the "Request System" to the DAGS-ICSD-SSB Database Management Section (DMS) section chief via the State's e-mail system.

   The e-mail request must include the DDL's first name, middle initial(s), last name, e-mail address, and contact office telephone number with extension (when available).

b. The DAGS-ICSD-SSB DMS staff will be responsible for assigning a unique User-Id for the DDL.

   The DAGS-ICSD-SSB DMS will inform the DDL via e-mail when a User-ID has been assigned for the DDL's use, and provide an explanation as to how the DDL may access the DBMS "Request System" site.

c. The DDL can create a list of pre-authorized departmental users and their e-mail address for DAGS-ICSD-SSB DBA to include in the DAGS-ICSD-SSB DBA e-mail reply distribution list.

d. Requests for proposed enhancements, capabilities, or features for the DBMS "Request System" should be made via the State's e-mail and sent to the DAGS-ICSD-SSB DMS section chief.

2.6.2 Usage In Urgent Situations

Urgent, critical, or time-sensitive questions may be initiated by a phone call to the DAGS-ICSD-SSB DBA.

As soon as possible after the rush or emergency request is made, the DDL must formally go on-line and enter a follow-up entry into the DBMS "Request System".

2.6.3 Usage By The ICSD DMS Staff

The DAGS-ICSD-SSB DMS is responsible for monitoring, supporting and maintaining the DBMS "Request System".

a. The DAGS-ICSD-SSB DMS will use the "Request System" to receive any database and DBMS service request from DDL.

b. The DAGS-ICSD-SSB DBA will use the "Request System" to update the progression of any action that it has taken in response to each request for database related services.

c. The DBMS "Request System" may be used to justify the provision of education or training program for database related products at the State.

2.6.4 Functions Provided By The DBMS Request System

The DBMS "Request System" is an automated method of communication via e-mails between the DAGS-ICSD-SSB DMS staff, and the State's DDLs who need to request services from DAGS-ICSD-SSB DMS personnel.

a. The database service requests and actions provided will be stored in a database. The DAGS-ICSD-SSB DBA and the State's DDL may query and access this database via a key word search.

   The DBMS "Request System" has internal security and controls to allow access restrictions.

b. Any request for services that the DDL enters into the DBMS "Request System" will automatically be forwarded to the DAGS-ICSD-SSB DBA, and to the DAGS-ICSD-SSB DMS Section Chief.

3 DATABASE ADMINISTRATION

All database administration (DBA) functions and facilities for State owned or controlled data will be centrally controlled and supported by DAGS-ICSD.

A State agency or department shall be designated as the Owner of the information originating from the agency or department that is stored in State owned or controlled mainframe computers, mini-computers, networked servers, personal micro-computers, and removable data storage media.

The Owner is responsible for ensuring the integrity and accuracy of information stored in the database hardware resources. For more details of the responsibility of the Owner refer to ICSD IT Standard **08.02 Information Security.**

As part of the State's information security control policy, the Owner shall be responsible for authorizing any and all access to information that falls under the Owner's jurisdiction and for which the Owner is ultimately responsible.

3.1 Database Administration Definition

The Database Administration (DBA) is the centralized control and coordination of the database environment, including database related hardware resources and all tools and utilities used to support DBMS software.

3.2 Database Administration Policy

The State of Hawaii Executive Branch has an adopted Information Technology Master Plan.

The Master Plan establishes the vision, direction, mission, and administrative policies for databases developed and supported in the State's Executive Branch of government.

The Master Plan stipulates, "The State will increase the accessibility, usability, and accuracy of information while striving to reduce the cost of obtaining, processing, and managing information".

This Information Technology Master Plan policy further stipulates that database related policies, processes, procedures, application systems and programs shall be designed, created, developed and established in such a manner as "to decrease data redundancy, to promote sharing of data and systems, and to improve the consistency and accuracy of data".

The Database Management Section (DMS) of the Systems Services Branch (SSB) of ICSD in the Department of Accounting and General Services (DAGS) is responsible for developing, promulgating, implementing, and enforcing the Information Technology Master Plan's policies as they apply to database administration.

3.3 Database Administration Objectives

The following are the State's database administration (DBA) objectives for any database designed, created, developed, and supported for application systems at DAGS-ICSD. The State's database administration (DBA) will:

a. Emphasize good theory and practice in the use of any DBMS.

b. Monitor and control the use of any DBMS to ensure the integrity and security of database data.

c. Monitor and control any DBMS to ensure its efficient and effective use.

d. Monitor to minimize the redundancy of data elements and tables.

e. Implement an effective distributed data management environment.

f. Provide technical support for authorized end-user's access to data.

3.4 Database Management Section Responsibilities

DAGS-ICSD is designated as the Custodian of databases stored in the State's central information processing site. DAGS-ICSD has a database management section (DMS) whose function is to effectively address this custodial responsibility. The DAGS-ICSD-SSB DMS will:

a. Establish standards, conventions, guidelines, and/or procedures to effectively ensure efficient data access controls, security and protection.

b. Provide technical and operational expertise to application systems and projects that are planning, designing and implementing database systems.

c. Maintain, support, and manage all entrusted databases, and database management systems software and their related resources.

The DAGS-ICSD-SSB DMS is administratively authorized to have Custodial authority over all databases at the State's central computer site to ensure that information resources housed in databases are available to, and accessible by approved and authorized departments and user agencies for their operational information processing needs.

Any request by a department or agency for participation of a member from the

DAGS-ICSD-SSB DMS for a departmental or agency application system project must have been presented to the DAGS-ICSD Administrator at the same time that the department or agency has requested approval to proceed with the development of the project.

Specifically, to address this database management responsibility, the DAGS-ICSDS-SSB DMS will:

a. Establish the State's conventions and procedures for planning, acquiring, customizing, installing, and maintaining database management system software and support tools.

b. Implement processes and procedures for monitoring, ensuring, and supporting the integrity of physical database structures and application program database libraries.

c. Provide advice and assistance by establishing and implementing policies, procedures, standards, and guidelines for managing database environments, including backup and recovery processes and procedures.

d. Provide technical support to departmental agencies and application development projects that require expertise in database setup, installation, implementation, and problem determination and resolution.

e. Plan, acquire and manage the allocation of data storage resources required for databases and for database management systems.

f. Evaluate, customize, install, maintain, and support database-related application development system software, such as, application builders, code generators, fourth generation languages, and ad hoc query tools.

g. Develop, recommend, and implement plans and procedures to establish effective and efficient communications for database access between host and distributed computer systems.

h. Coordinate the standardization of database management systems within the State to minimize systems maintenance costs, and to promote systems compatibility and interoperability.

i. Develop and implement procedures, and establish standardized criteria to qualitatively measure, analyze, and fine-tune the performance of database management systems.

j. Develop and implement procedures and processes to effectively measure, model, and forecast database usage, and to produce data-storage resource capacity planning estimates.

k. Collect and review job accounting information generated by database management systems for database accounting and billing.

l. Serve as an advisor, consultant and/or member of a technical project team that requires the expertise of a knowledgeable specialist in databases and database management systems.

m. Assist with the development and implementation of safeguards against unauthorized accesses to database and database management systems.

n. Develop and/or evaluate database-related bid specifications published by any agency in the State of Hawaii Executive Branch.

o. Assist in the review of development plans and procurement requests for database-related hardware or software by any agency in the State of Hawaii Executive Branch.

p. Prepare annual budget, projected expenditure, and anticipated project plans for database-related funding needs and services for DAGS-ICSD.

q. Research and provide technical review and evaluation of new industry products and technologies which have the potential to streamline, modify or enhance the State's central-site mainframe database management capabilities.

r. Interface and interact with companies, firms, or organizations of database management system tools, utilities or products.

s. Perform maintenance, service and support of all database-related systems software.

t. Control, customize and perform the scheduled execution of all relevant centralized database tools or utilities.

3.5 Departmental Database Liaison Duties And Responsibilities

The principle duties of the Departmental Database Liaison (DDL) fall into two broad categories:

a. To function as the internal department or agency consultant and advisor for the department or agency on data management and database issues.

b. To function as the main communication liaison between the department or user agency with the DBA in the DAGS-ICSD-SSB-DMS.

3.5.1 DDL Internal Duties And Responsibilities

The DDL is the internal user agency expert for database related issues and concerns. The specific duties and responsibilities of the DDL are:

a. To assist the department or agency with implementing data management tasks, which include specifying, integrating, and coordinating data resources needs and requirements during the creation and annual update of the department or agency's strategic planning, application system development, and/or production applications operations and maintenance.

b. To provide technical assistance for the department or agency's database projects, which include managing, designing, creating, installing, implementing, tracking, and scheduling.

c. To consolidate and coordinate the design, establishment and maintenance of departmental or agency data models of existing applications and those under development (i.e., data descriptions, primary keys, alternate/foreign keys, data-entity relationships).

d. To design, develop, and establish data security specifications for the database environment.

e. To approve, authorize and submit the relevant database security requests to DAGS-ICSD.

f. To develop or arrange for database management training for departmental or agency personnel.

g. To enforce user agency DBMS, Programming, and/or Data Dictionary goals, standards, conventions and procedures that comply with or that exceed those established by ICSD. Any Internal standard, convention, or procedures should focus on uniform and consistent departmental data dictionary usage.

h. To notify or advise departmental or agency application

development staff regarding the availability and usage of facilities available from DAGS-ICSD, which currently includes utility request procedures and the development-staff-run utilities.

3.5.2 DDL External Duties And Responsibilities

The official liaison for the department or agency with other State departments or agencies, the specific duties and responsibilities of the DDL are:

a. To function as the primary contact person to communicate the department or agency's database needs, specifications or requirements to DAGS-ICSD.

b. To present the departmental or agency's views of data models and physical file design for DBMS applications to the DAGS-ICSD DBA.

c. To submit DB2, ORACLE or ADABAS Data File Plans (through the data dictionary tool, PREDICT) to the DAGS-ICSD DBA.

d. To submit database Utility Requests, database Security Requests, and other necessary communications to the DAGS-ICSD DBA.

e. To monitor the agency's database reports either through over-the-counter batch job submission requests, or on-line through SYSAOS. Using reports from either method, the DDL takes appropriate action to initiate changes to data file plans and to recommend submission of data file reorganization requests.

f. To receive priority standing to attend DGS-ICSD classes related to data resource management and database products and tools.

g. To represent the department or agency by attending all DDL user agency meetings scheduled by the DAGS-ICSD DBA.

4 META-DATA

Meta-data is information about data elements and other components of data information systems. Meta-Data is information labeling, cataloging, and descriptive information that is structured in a way to allow data files to be searched and processed by computers that use different platforms.

4.1 META-DATA Policy

Standardized meta-data for all DB2, ORACLE, ADABAS and NATURAL database centric application systems will be maintained within the data dictionary, PREDICT, by the application project team.

PREDICT should be used as the common data store (warehouse) for the departmental user agency's conceptual data model that is designed during the development and upkeep of the department or agency's information technology Master Plan.

PREDICT is the State's standard vehicle for encoding descriptive, administrative, and structural meta-data regarding data elements in database files.

PREDICT is the official data dictionary/repository used at the DAGS-ICSD central computer site, in conjunction with DBMS products, utilities and tools.

PREDICT may be used to produce meta-data to document both ADABAS and non-ADABAS data elements.

It is strongly recommended that PREDICT standard files be designed, created and maintained to ensure uniform and consistent data names and definitions.

4.2 META-DATA Objectives

DAGS-ICSD strongly recommends that all Executive Branch Departments and attached Agencies use the database utility data definition tool, PREDICT, to support the State's meta-data objective for any database environment, which are:

a. To maintain a central repository for the definition of all data elements in the State's centrally controlled DB2, ORACLE, and ADABAS databases.

b. To provide a standard data administration tool that will allow for auditing, controls, and protection of the State's data information resources.

c. To maintain a sufficient level of meta-data definitions to allow the State's data processing personnel to better respond to user requests to develop new application systems

d. To provide a visualization of meta-data definitions that may lead to enhancements of existing systems, by having current information about applications or data elements readily available to the data processing staff.

e. To aid database application systems project teams in identifying inter-agency data sharing potential.

f. To standardize files to ensure data naming and definition standardization and consistency.

4.3 META-DATA Standards And Guidelines

The specifics of meta-data standards are found in the PREDICT Standards, in the State of Hawaii Information Technology (IT) Standards 06.01.

The purpose of the IT 06.01, Predict Standards and Guidelines, is to ensure that the State's meta-data objectives will be met through the maintenance of a central, up-to-date, online repository of relevant information concerning a department or agency's data elements, shared systems, application programs, reports and on-line screens that may use the DB2, ORACLE or ADABAS/ NATURAL database environment.

4.4 META-DATA Responsibilities

The DAGS-ICSD-SSB-DMS will support the State's meta-data environment via the PREDICT system software. The DBA in the DAGS-ICSD-SSB-DMS has the duty and responsible to:

a. Standardize batch-reporting techniques for data dictionary information.

b. Develop, promulgate and enforce standards for database related data-naming conventions, logical file design, data sharing and record keeping.

c. Verify that application project teams document their application system's data dictionary according to Meta-data standards in IT Standards 06.01.

d. Coordinate or provide training for the use of PREDICT.

4.5 META-DATA References To Technical Manuals

The vendor's PREDICT Reference Manual is in Volume 8 of the State of Hawaii IT Standards set of documentation.

The State's statewide PREDICT standards and conventions manual is IT 06.01, found in the State of Hawaii IT Standards.

Each designated Departmental Database Liaison (DDL) has a complete set of the State of Hawaii Information Technology (IT) Standards set of documents.

5 SYSTEMS DESIGN & DEVELOPMENT

All departments and attached agencies in the State Executive Branch shall use the same approach to methodically assess, design, create, develop and access standardized and consistently structured database centric application systems.

5.1 Systems Development Policy

System design and development within the DB2, ORACLE, ADABAS/NATURAL database management system environments will follow the discrete phased approach presented in the system development methodology tool, SDM/Structured, as well as any other process or procedure required by DAGS-ICSD for effective database operational support and services, as may be presented in any document distributed by DAGS-ICSD.

5.2 System Design Objectives

There are two principle database system design objectives, they are:

a. To design and create flexible and scalable databases and data file structures with primary consideration for potential data sharing, both inter- and intra-departmentally; and for the future growth, enhancement, and updated use of databases and data files in a business and operational environment that have ever changing needs and requirements.

b. To design databases, physical files and data processing application systems that efficiently and effectively use the State's central-site shared computer resources.

5.3 Systems Development Responsibilities

The duties and responsibilities for the designing and creating of databases and data files are shared among the DBA of DAGS-ICSD-SSB-DMS, the departmental database liaison (DDL), the Project Manager, and the application development project team.

5.3.1 The DBA Responsibilities

The DBA of the DAGS-ICSD-SSB will be responsible for reviewing and

evaluating all database related documentation that are required by tasks and activities specified in the State's methodology, SDM/Structured. This includes the review for appropriateness of the design of conceptual data models, physical data structures, and data access techniques.

5.3.2 The DDL And Project Team Responsibilities

The DDL and the application project team will be responsible for understanding and adhering to database and data file design and systems development methods as documented in pertinent the State IT Standards.

5.3.3 The Project Manager

The Project Manager is responsible for coordinating, directing, and tracking an application system's project design, development and implementation.

a. The Project Manager must acquire (or produce) and provide database-related project documentation and information for the DBA of DAGS-ICSD-SSB, as stipulated in tasks or activities of the SDM/Structured methodology. This is required so that the DBA can do both an efficient and effective job of performance evaluation and resource planning/allocation at the appropriate stages in the project development lifecycle.

b. The Project Manager should ensure that sufficient lead-time is provided for each project development methodology phase, task, and activity, so that the planned system development schedule can be reasonably attained.

c. The Project Manager must ensure that required approval and/or authorization has been received by the appropriate person/office for each pertinent phase, task or activity of the application development lifecycle before the project team proceeds to the next project development event.

5.4 System Design Alternatives Analysis

Since the mainframe based database resources are the most expensive components of processing data and information, a critical and careful analysis of the application system needs, requirements and designed solution must be done to ensure that the mainframe environment is the best choice.

Only statewide or widely used systems that need a large, shared database and real time access can justify the expense of the mainframe database environment.

After the mainframe database environment is decided upon, the following design alternatives need to be evaluated:

5.4.1 Storage Alternatives

Alternative storage media should be evaluated that address the concerns of the anticipated access demands, media retention periods and data archiving processes that are based on the kinds and amounts of data that are expected to be stored within a database, such as, DB2, ORACLE or ADABAS.

5.4.2 Access And Timeliness Alternatives

Alternatives should be evaluated for designing transactions and transaction-types that would minimize the number of direct database access calls, and minimize the amount of CPU cycles and processing time that are based on the types and number of online transactions that are expected to be requested by the application system end-users.

Alternative data design features that should be assessed include issues, such as, hardcopy batch reports, online access to batch generated reports, or cooperative systems processing.

The use of cooperative systems processing options should be considered to shift computer-processing time from the expensive central mainframe to the less expensive distributed minicomputers or microcomputers.

5.4.3 Costs And Benefits Of Alternatives

When doing the alternatives design analysis, the overall development (from conceptual through installation), the benefits to the people of Hawaii and to state government, and any projected ongoing maintenance or operational costs should be reasonably quantified to estimate the true-costs and achievable benefits for the database system.

The alternative analysis should take into consideration all projected State development, environmental, and installation costs, which should include DAGS-ICSD costs for database support resources.

In addition to setup implementation factors, cost analysis should consider

the production costs of the system, which include ongoing maintenance, product and/or systems upgrades, and operational costs.

5.5 Systems Development Checklist For Database Tasks

The database related tasks, documentation, reviews and system-walk-through required by SDM/Structured and DAGS-ICSD internal procedures are contained in the table listed below.

SDM/Structured form SDM-1230, Data Dictionary, may be replaced by PREDICT reports because they contains the same information.

NOTE: The Database Task Checklist itemizes documentation and review requirements needed solely for the DBA of DAGS-ICSD-SSB. The database task checklist does not address the needs of the State Data Administrator (DA), Telecommunications Coordinator (TC) or other areas of DAGS-ICSD.

Database Task Checklist

SDM Phase	Task Num	DESCRIPTION	FORM # OR DOCUMENTATION	SUBMIT TO DBA
SRD	4.3	Define the Data Stores of the Current System	1230,1168,1313	
SRD	5.2	Define Data Dictionary of the Current System	1230,1232,PREDICT	
SRD	5.4	Develop Data Store Model Current System	1230,1235,1233	
SRD	6.3	Produce Logical Model for the New System	1220,1230,PREDICT	Yes
SRD	7.2	Define the required Data Characteristics	1230,1179,PREDICT	
SDA	5.3	Define Physical Data Store Requirements	1100,1169,1230,1235,PREDICT	Yes
SES	1.0	Establish the Automated System Environment	1190,1100,1101,1192	
SES	5.0	Design the Automated Data Stores	1230,1313,1510,1235,1850, 1169,1200,1515, 1550,1168	Yes
SES	6.3	Specify File Reconstruction Processes	1100,1101,1850	
SES	7.0	Prepare Acceptance Conversion and Implementation Plan	1100	Yes
SIS	2.0	Design Data File Structures	1168,1230,1313, 1510, 1602, PREDICT	Yes
SIS	--	Create Physical file plan and ANY user view in PREDICT.	PREDICT	
SIS	2.0	Walk-thru physical file design with ICSD-DBA	Meeting	Yes
SIS	2.0	Submit ADABAS File Plans for both Test & Production Databases after ICSD DBA approves the physical file design.	DBMS Request System, PREDICT	Yes
SIS	4.0	Submit RACF Security Requests for both Test and Production Databases	DBMS Request System	Yes

SIS	4.0	Submit ADABAS Protection Log Extract Utility (PLEU) request for audit trail extracts For Test/Development and Production.	DBMS Request System	Yes
SIS	--	Submit Projections of Batch Job Activity and Projections of Online Activity. See APPENDIX C for format of reports	Memo	Yes
SIS	--	Request NATURAL Application Library for Test/Development Database.	DBMS Request System	Yes
SIS	--	Request NATURAL Group ID's in Test/Development environment for staff new to the environment.	DBMS Request System	Yes
SIS	--	Request NATURAL Group ID's of development team be linked to the new Application Library.	DBMS Request System	Yes
SIS	--	Generate DDM's in PREDICT from user views.	PREDICT	
SIS	--	Request files (DDM's) be linked to the new Application Library.	DBMS Request System	Yes
SIS	--	If there is initial data for the new file, create compressed file with ADACMP utility for each Test ADABAS file.		
SIS	--	Request loading of compressed file onto Test/Development Database or create file via PREDICT.	DBMS Request System	Yes
SIS	--	Request files (DDM's) be linked to the Production Library.	DBMS Request System	Yes
TST	6.0	Request APAS Reports from DBMS	DBMS Request System	Yes
TST	6.0	Coordinate Compliance/Performance Review with ICSD-DBMS	Memo	Yes
CNV	2.1	Define File Conversion Requirements	Memo	Yes
CNV	--	Schedule conversion with ICSD-DBMS and ICSD-PSB	Memo	Yes
IMPL	--	Request NATURAL Application Library for Production Database	DBMS Request System	Yes
IMPL	--	Request NATURAL Group ID's in Production environment for users new to the environment.	DBMS Request System	Yes
IMPL	--	Request NATURAL Group ID's of users be linked to the Production Library.	DBMS Request System	Yes
IMPL	--	Request Predict Definition with DDM's and programs be copied to the Production environment from the Test/Development environment.	DBMS Request System	Yes
IMPL	--	If there is initial data for the new file, create compressed file with ADACMP utility for each Production ADABAS file.		
IMPL	--	Request load of compressed file to Production Database or create file via PREDICT	DBMS Request System	Yes

5.6 General Design Methods And Procedures

The logical design of a database is to be independent of the specific DBMS, which will be used to create the physical database and data files.

The SDM/Structured procedures are adequate for producing a viable logical data model.

The actual design of the physical files requires considerations that are unique and specific to each database management tools like, DB2, ORACLE or ADABAS.

5.7 Physical Structuring Design Methods

The following are physical structuring design processes that should be addressed after the logical data model is designed and developed from the conceptual data model.

a. Projections of frequency of requests should be determined for each entity set (daily average and peak access for both online and batch access).

b. Estimations of entity set content and sizes should be determined.

c. The affinity factors between entities should be determined. See Standards, IT 06.01 PREDICT, or the Glossary in this document for a discussion of the affinity factors.

Performance considerations can affect the design of the database. The physical design should be based on an evaluation of the possible options that may be used in loading the data files, and the design of the procedural and processing logic of the application system functions.

The following physical structure design procedures should be implemented:

5.7.1 Describe Each Application Program

The description of each application program in the developing system should include specifications that:

a. Describes the manner in which each record type may be processed.

b. Describes the access path and the sequence of required records and/or record-types linked to each type of information request.

c. Determines the request frequency, volume of the run, and the times that data should be made available.

5.7.2 Prioritize Application Programs

Each program in the application system should be rated and prioritized. The priority level for each program should be based on identifying the criticalness of the functions and performance of the application program.

Critical-performance considerations include:

a. An assessment of data storage volume allocations.

b. The determination of the frequency of data access requests.

c. Any mandatory deadlines.

d. Any effect of data requests on the performance and scheduling of other hardware or software system.

e. The minimum performance impact on other programs and systems.

5.7.3 Optimize Database Performance

The performance of each critical function may be optimized by shortening the access paths. This would include optimizing the processing logic, and eliminating any database feature that may increase system resource demand or overhead.

The expected performance of each critical DBMS feature or function and the performance of other system functions on the database should also be determined.

5.7.4 Denormalize The Database

After the physical structure design of the database super entity is addressed, the normalized logical data file structures can then be denormalized to improve the database performance.

The denormalization process, however, may reduce the overall flexibility of the data-file design.

With the logical model and the physical structure design information, the trade-offs between the following considerations should be evaluated during the denormalization of the data file design to determine the final design of the physical file structures.

Trade-off consideration features:

a. Application system performance
b. Data independence
c. Accessibility of information
d. Audit and security considerations
e. Currency and availability of information
f. Ease of scheduling
g. Impact on current users of the database
h. Disk space usage and allocation

5.7.5 Denormalization Process

The denormalization process should take place for the following reasons:

a. To improve the performance of application programs when they access the database.

b. To reduce the cost of processing the information request.

As a part of the denormalization process, the limitations of the DBMS data files should be a major consideration. Refer to the vendor's database management documentation for any specific limitation.

5.7.6 Denormalization Considerations

The process of denormalization and finalizing the database data file physical design should take into consideration the following design approaches and guidelines:

a. Consider modification of the data file design to combine two data files into one, such as making one data file a periodic group within another. This should be done when there is a high affinity factor between entities and a high access frequency.

b. Consider the use of multiple record type data files. This may be done when different types of entities are frequently accessed together by a common key and/or there is infrequent reason to read through an entire data file for a given entity.

   Table files and other data files that are only accessed randomly, a few records at a time, are appropriate multiple record type files.

c. Use the Predict Standard File to standardize field definition, provide field usage documentation, and aid in field definition changes.

d. Consider defining additional descriptors (keys), super-descriptors and sub-descriptors based on performance considerations (sorts, inquiries, etc).

   Descriptors that are used together in searches or sorts should be combined together as super-descriptors. This will simplify searches and will also save space.

   If a descriptor is used in combination with other descriptors in selecting records, but it provides very little search refinement in the selection, then the field should not be defined as a descriptor at all. In this case, the records selected could be read to determine if they meet the total selection criteria. This will reduce the amount of allocated file space and will also reduce the time for file updates by eliminating an update to the descriptor for the field.

e. Do not store data elements that can easily be computed, determined or derived.

f. Determine the best storage method for data elements based on the primary usage. For example, numeric data that is used mostly for retrieval on inquiries should be stored unpacked (format "U"). Numeric data that is used in computations should be stored in packed ("P" format).

g. Order data elements within data files to ensure productive optimum performance and intuitive ease-of-use. The following is a suggested approach to this ordering:

   1) The preferred sequence ordering for data fields within records when based on attribute/format is:

i. All FI (fixed) fields
ii. All "normal" fields (not FI, MU, or NU)
iii. All NU and non-MU fields
iv. MU (multiple occurrence) fields
v. PE (periodic groups)

2) Frequently used data fields should be placed closer to the front of the record, this may override the sequence convention proposed above

3) Put NU fields most likely to be null adjacent to each other.

4) MU fields should be null suppressed unless relative positions need to be maintained.

5) Avoid use of MU fields within periodic PE groups.

6) All fields in PE groups should be NU (null suppressed) to allow individual occurrences to be bypassed or completely deleted.

5.8 Recovery/Restart Design Methods

To facilitate the recovery or restart ability after a transaction request to search a database is initiated, individual transactions within an application system should be controlled via the use of ET (End Transaction) logic.

Batch application programs should be designed to use ET logic when accessing database data files to allow for restarts of batch applications.

The recovery/restart design methods should be developed to address most common/typical system failures that would not result in the probable destruction of all or a portion of a database data file.

A model for batch job design that allows for a simple rerun/restart process can be obtained from the DBA of the DAGS-ICSD-SSB.

For catastrophic errors that require the restoration of data files and/or the back out of update transactions, a request to the DBA of DAGS-ICSD-SSB must be made to recover a data file from a daily systems database backup.

The restoration of a data file will result in the loss of all data transactions that had been entered against the data file. After the execution of the backup recovery

job, all transactions against the data file must be re-entered.

Make sure when requesting a restore that you have taken full consideration of data referential integrity. That is, that you are restoring all data files with related data. (As a reminder, ADABAS does not maintain referential integrity for the application.)

Recovery of non-ADABAS files is the responsibility of the project that develops the system, not the DBA of DAGS-ICSD-SSB.

RECOMENDATION: DO NOT DESIGN TRANSACTIONS THAT WOULD UPDATE BOTH ADABAS FILES AND NON-ADABAS FILES.

6 DATA SECURITY

Security and privacy of data elements, tables and files are essential for the reliability, integrity and accuracy of the information stored in the database.

6.1 Data Security At DAGS-ICSD

DAGS-ICSD is committed to provide safeguards and reasonable assurances to protect the data and information under its care against unauthorized access.

Database data files that are shared and accessed amongst inter-departmental applications and/or other departments or agency must be regulated and controlled.

DAGS-ICSD has policies and procedures established for the ADABAS database management environment and the non-ADABAS DBMS environment. Refer to IT Standard **08.01 Security Overview and 08.02 Information Security** for additional Data Security Standards.

6.1.1 ADABAS Data File Security

In the ADABAS database environment, the file security tool, SECURITRE, and the file access tool, RACF, will be used to protect all data files.

6.1.2 Non-ADABAS Mainframe Computer Based DBMS Systems

For any non-ADABAS managed data file that is stored in the mainframe computer at DAGS-ICSD, the tool, RACF, will be used to protect all of these data files.

6.1.3 Non-ADABAS Non-Mainframe Computer Based DBMS Systems

For any non-ADABAS managed data file that is not stored in the mainframe computer at DAGS-ICSD, the DBMS software used in the application system will be used to protect the database data files.

6.2 Securitre

Securitre is a mainframe computer based software package marketed by Treehouse Software, Inc. It provides an Interface between ADABAS/NATURAL and a System Security Facility (SSF), such as ACF2, Top Secret, or RACF. SECURITRE secures ADABAS at the database, file and field levels. ADABAS Utilities, NATURAL, and NATURAL Utilities can also be secured with Securitre.

The DBA of DAGS-ICSD-SSB will coordinate with the DAGS-ICSD-SSB System Security Section to setup the RACF definitions.

6.3 RACF

Resource Access Control Facility (RACF) is a mainframe computer based software package marketed by IBM. It provides security for all data files on the mainframe platform. RACF protects system resources, and it controls what users can query or access in the computer operating system.

6.4 NATURAL Security System

NATURAL Security is a mainframe computer based software package marketed by Software AG. NATURAL Security will be used to control access to application programs, access to data files, and NATURAL resource usage.

Information and standards governing the use of the NATURAL Security System at the State's central computing site at DAGS-ICSD are found in IT Standards 06.02 NATURAL.

6.5 ADABAS Protection Log Extract Utility (PLEU)

Protection Log Extract Utility (PLEU) is a mainframe computer based software package marketed by Computer-Associates (CA). The PLEU will be used to provide audit trails of update activity on all critical and/or sensitive data maintained in ADABAS files.

The audit trail job streams, "XWAX#," (where "#" represents the database-identifier, DBID), are submitted daily (after the scheduled production database backup jobs) to process User specified extract statements. These jobs are used to extract information from the ADABAS Protection Log (PLOG) for the desired ADABAS data files.

If the protection log extracted data is needed for an application job, the project team must have filed documented operations control procedures to hold that application system job until after the scheduled production job stream that processes "XWAX#" is completed successfully.

The protection log contains the "before" and "after" images (in compressed format) of all ADABAS records that were updated during the previous day's session.

An unlimited number of uniquely formatted requests to generate output-extracted files may be produced with only one pass through the protection log.

Each protection log extract output record will contain a header. The format is found in CA-PLEU User Guide under “CA-PLEU Output Files”.

The EXTRACT statements are used to specify the record selection criteria for an individual ADABAS file with extracted output datasets. There must be at least one EXTRACT statement for each execution of the audit trail utility. Each extract request can select records from the protection log based on its:

- ADABAS file number
- Date
- Time
- Terminal-id
- User-id; and
- "Before" and/or "after" images.

The User through card image commands specifies the format of the selected records and the criteria used for the selection of the records.

The rules for the output format specifications are the same as those specified in the "ADABAS Command Reference Manual" with a few exceptions. Details and restrictions for the "FORMAT" specifications are in the CA-PLEU User Guide.

6.5.1 PLEU User Responsibilities

The format of each extracted file is defined by the FORMAT specifications

provided by the User as part of each "EXTRACT" statement.

6.5.2 PLEU Project Team Responsibilities

The project team must specify the ADABAS file number from which data will be extracted. This is the standard ADABAS file number between 1 and 5000. The format for this parameter is:

FILE=999

The project team must also provide the FORMAT buffer definition of the format for which all records extracted will be written.

The application project team is responsible for coding the EXTRACT statements. The requests for extracts must be planned for during the design phase of the application system.

The application project team is responsible for identifying all the fields that they anticipate may be needed for audit trail verification.

7 PROGRAMMING WITH ADABAS

NATURAL is the preferred programming language that should be used to access data elements and information in ADABAS data files at the State central computer site. This would ensure that all access requests would be controlled by NATURAL Security controls.

For specifics related to the State's Data Security Standards, refer to IT Standard **11.01 Host Mainframe Program Planning and Design** and **08.02 Information Security**.

7.1 ADABAS Program Design And Performance

The cardinal rule for efficient database centric application program design is: "***Minimize access to the ADABAS nucleus***."

On average, approximately 75% of the total CPU usage to execute an application is used by the ADABAS nucleus.

7.1.1 ADABAS Program Design Considerations

a. Applications must be designed with a minimum number of calls to ADABAS. The DIRECT CALL to ADABAS adds significantly to the CPU usage and to the physical I/O (input/output).

b. Online (real-time interactive) programs are allowed to issue no more than 20 ADABAS calls per logical transaction as identified by the CICS assigned transaction number.

c. Standards for the efficient design and coding of NATURAL programs are to be found in the NATURAL Standards manual.

Additional programming design standards and guidelines for both NATURAL and 3GL programming can be found in the document, “Unicenter CA-APAS Insight Monitor for ADABAS User Guide”, under the section titled, “Tuning Guidelines”.

7.1.2 ADABAS Program Performance Conventions

a. Never use the internal NATURAL "SORT" and "SORTED BY" statements.

b. All data file sorts must be done as standalone sorts executed external to the application.

c. The Multi-fetch option must be used for all batch programs that are not developed to perform ADABAS data file updates.

7.2 ADABAS Direct CALL Subroutine

The format for "call" subroutines from native mode batch programs, like COBOL, should be:

"CALL ADABAS"

More details and restrictions for " Direct Call " statements can be found in the ADABAS Command Reference Manual under “ADABAS COMMANDS”.

7.2.1 ADABAS Programming Guidelines

When creating a program the following should be considered.

a. The COBOL/VS "DYNAM" option should be used to compile batch programs. This option will conserve library disk space, and will optimize link-edited modules.

b. For CICS programs, include Software AG module ADASTWA

c. For Batch programs, include Software AG module ADAUSER

c. For greater efficiency, use the "COMMAND ID's" for the read and update commands.

Additional programming guidelines may be found in the vendor's ADABAS Command Reference Manual in the "GENERAL PROGRAMMING CONSIDERATIONS" section.

7.3 Programming References To Technical Manuals

Technical manuals for all Computer-Associates (CA) products can be viewed as listed below over the Internet. For an individual to access the Internet manuals, a Support Connect ID must be attained first at:

http://supportconnect.ca.com/sc/index.jsp

The State of Hawaii account number is 101263.

Reference documents for CA APAS may be found at:

http://support.ca.com/apas_space_supp.html

Reference manuals for CA Endevor may be found at:

http://support.ca.com/endv_supp.html

Reference manual for CA Spool may be found at:

http://support.ca.com/spoolsupp.html

Technical manuals for all Software Ag products such as ADABAS and NATURAL can be viewed as listed below over the Internet. To access the manuals a ServLine24 ID must be attained first by sending an Email to:

wecare@softwareagusa.com
The ICSD-DBA can send the request for the new ID.

Reference manuals for Software AG products may be found at:

http://servline24.softwareag.com/SecuredServices/

Reference manual for Treehouse product such as SECURITRE may be found at:

http://www.treehouse.com/proddownld.asp

APPENDIX

GLOSSARY OF DATABASE TERMS

Affinity Factor	The ratio of occurrences in which one logical entity is accessed together with another in the organization's daily activities. An affinity factor of '1' for entity-1 to entity-2 indicates that whenever entity-1 is used, entity-2 will also be used. Affinity factors are used during physical file design as an indicator of the potential benefit of denormalization.
Data element	A single item of data; therefore, one attribute of an entity.
Data redundancy	Redundancy occurs when the same data element is stored in multiple data files. This practice can lead to inconsistencies in the value or definition of a data element.
Denormalization	The process of modifying the normalized entity sets (files) to improve performance and to increase the usefulness of the data files to current and future applications.
Entity	Any thing, an object, a person, or a relationship, about which information is required to run the business of an organization.
Entity attributes	The data elements which describe an entity.
Entity identifier	The entity attribute (key), or combination of entity attributes (concatenated key) which uniquely defines each occurrence.
Entity record	The set of attributes for a given entity.
Entity set	The set of all occurrences of entity records for a given entity.
Fragmented Map	A portion of the integrated map which is an individual user-view (a single process) of an application system.
Integrated Map	A chart of all data files for an application system, which shows all access paths between the data files.
Logical structuring	The process that involves evaluating the normalized entities. The individual entities have been defined in terms of the primary key of

each entity, and of the data elements that are a part of each entity set. The user views as defined for the applications are then "proven" against the entity sets, insuring that all required data elements could be accessed.

Main Menu — An application design concept where the main functions of an application are displayed to a user, giving the user the option of selecting the particular function he/she desires to execute. This menu may normally be redisplayed to the user at any point in an application so that another function may be selected or the application may be terminated.

Meta Data — This is data used to describe data, or it is the documentation of data elements, data files, etc. For the database, this descriptive information is stored in a special file called the data dictionary.

Normalization — The process which takes an initial grouping of data elements (super entity) and breaks this group into multiple smaller groups of data elements. All of the data elements within each of the small groups are dependent on a single key.

Primary access — This is the primary data access point to access the data elements in an entity set by using the entity identifier as the search argument.

Program/Data Independence — is achieved by separating the descriptions of data elements from the program that accesses the data elements. This allows the data attributes and order of the data elements to be changed without affecting the application program thus reducing program maintenance and intervention requirements.

Relationship — The organizational association of structures that ties (or links) two data elements or two entities together. Relationships are determined by departmental policies and by data usage.

"One-to-one relationship" is when each occurrence of one entity is related to only one occurrence of another entity.

"One-to-many relationship" is when each occurrence of one entity is related to one or more occurrences of another entity.

"Many-to-many relationship" is when each occurrence of the first entity is related to one or more occurrences of

another entity, which in turn is related to one or more occurrences of the first entity.

"Simple relationship" is a one-to-one relationship.

"Complex relationship" may be one-to-many or many-to-many.

Search argument	A value of an entity attribute used to gain access to an entity set.
Secondary access	The access point to enter an entity set, by using an attribute value other than the entity identifier as the primary search argument.
Start-up program	The program that is required to establish global variables and to call the main menu program for an application. It is executed automatically when a user logs on to a production application.
Super entity	The initial grouping of all data elements that comprise a database once the data needs have been determined. The normalization process breaks this single entity group into multiple entity groups.
Third-Normal-Form	The final grouping of data elements once the normalization process has been completed. In this form, every data element in an entity set is directly related to the set by a unique primary key.
User View	A selected number of fields from a database file that are grouped together under a separate name but access the same file. The purpose of separation may be to secure data (by omitting a field name when access is disallowed) or to create a logical subgroup of data elements within a data file.