
Workbook 2: Using the Huntsville Operations Support Center (HOSC) Databases

HOSC Training Division

May 1999



National Aeronautics and
Space Administration

George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama 35812

Workbook 2:
Using the Huntsville Operations Support Center (HOSC)
Databases

HOSC-WKBK-2050

May 1999

Prepared in support of:
NAS8-44000

Prepared by:
HOSC Training Team

Prepared for:
Mission Systems Division
Mission Operations Laboratory
Science and Engineering Directorate
Marshall Space Flight Center



Welcome

Welcome to Huntsville Operations Support Center (HOSC) training provided by the HOSC Training Team (HTT).

If you are interested in scheduling additional training, submit a training request form via the Internet. The homepage can be accessed at:

http://mole.msfc.nasa.gov/hosc_training/htt.html

Another option is to contact the HOSC Training Coordinator, Dawn Schell, at (256) 461-4927.

Training Contacts

If you have questions regarding training, contact your EO62 training representative or one of the following individuals:

Name	Organization	Phone Number	E-mail Address
Lisa Watson	EO39	544-3523	lisa.watson@msfc.nasa.gov
Kim Lankford, HTT Lead	NTI	461-4536	kim.lankford@ums.msfc.nasa.gov
D.C. (Sparky) Goodman	NTI	461-4545	sparky.goodman@ums.msfc.nasa.gov
Michelle Adrian	NTI	461-4514	michelle.adrian@ums.msfc.nasa.gov
Dawn Schell	NTI	461-4927	dawn.schell@ums.msfc.nasa.gov
Gene Sobczynski	NTI	461-4516	eugene.sobczynski@ums.msfc.nasa.gov
James (Jay) Washburn	NTI	461-4543	james.washburn@ums.msfc.nasa.gov
Anthony (Mac) McCoy	bd Systems	461-4532	anthony.mccoy@ums.msfc.nasa.gov

Workbook Overview

The majority of the workbooks are designed to be self-paced requiring very little assistance from an instructor. The following table lists the workbooks and their associated course number:

Workbook Numbering System

Workbook Number	Title	Associated Course
Workbook 1	HOSC End-User Software Training	HOSC-1000 HOSC-1010 HOSC-1020 HOSC-1030
Workbook 2	Using the Databases	HOSC-2050
Workbook 3	Using the Exception Monitor Application	HOSC-2060
Workbook 4	Using the Display Generation and Operation Applications	HOSC-2070
Workbook 5	Using the Computation Generation and Operation Applications	HOSC-2080
Workbook 6	Using the Scripting Applications	HOSC-2090

Workbook Numbering System (Continued)

Workbook Number	Title	Associated Course
Workbook 7	Using the NRT Data Request Applications	HOSC-2100
Workbook 8	Using Applix	HOSC-2110
Workbook 9	Using FrameMaker	HOSC-2130
Workbook 10	Using Electronic Mail	HOSC-2140
Workbook 11	Using the Payload Information Management System (PIMS)	HOSC-2150 HOSC-2160
Workbook 12	Using the Pixmap Editor	HOSC-2170
Workbook 13	Using the Ground Support Equipment Packets Application	HOSC-2180
Workbook 14	Using the Strip Chart Recorder Applications	HOSC-2190
Workbook 15	Using the End-User Command Applications	HOSC-2120
Workbook 22	Workstation Overview and General Purpose Utilities Training	HOSC-1040
Privileged Applications		
Workbook 16	Using the Privileges within the Database Applications	HOSC-3000
Workbook 17	Using the Command System Management Application	HOSC-3010
Workbook 18	Using the User Configuration Management Application	HOSC-3020
Workbook 19	Using the System Monitor and Control Applications	HOSC-3030 HOSC-3040
Workbook 20	Using the Data Packet Generator Application	HOSC-3050
Workbook 21	Using the Database Monitor and Control Application	HOSC-3060

This workbook covers course:

HOSC-2050 - Using the Huntsville Operations Support Center (HOSC) Databases

This course will provide the trainee with the information necessary to operate the Telemetry Database application, Command Database application, User-Generated Data Element (UDE) Database application, and end-user aspects of the Database Change Request application.

Workbook Layout

The workbooks include a welcome section that details how the document is divided into modules as well as what is contained within each module. The modules include a discussion of the main topic of the module, a step-by-step “Try It...” and review questions. Modules have been included that “put it all together” and provide exercises to reinforce what you have learned.

These particular applications are provided to assist you, the user, in accessing and manipulating the Telemetry Database (TDB), the Command Database (CDB), and the User-Generated Data Element (UDE) Database. Database applications also provide the capabilities to access Local Table and User Copy versions of the database in order to update and edit (if privileged) the database tables and submit change requests using the Database Change Request form. This course will provide a thorough introduction to the purpose and structure of the software and will set you on a path toward understanding the applications and mastering the manipulation of them.

This workbook and the accompanying course is divided into five modules. The first module discusses basic database definitions, what a relational database is, and the forms used to manipulate it. The next three modules discuss, in depth, each of the three databases. The fifth module provides information about making permanent changes to a database and the DBCR process.

Each module features an exercise which will apply the skills you acquire during the course. The five modules include:

- Module 1: *Definitions and Concepts*
- Module 2: *User-Generated Data Element (UDE) Database*
- Module 3: *Telemetry Database (TDB)*
- Module 4: *Command Database (CDB)*
- Module 5: *Documenting Permanent Changes to a Database*

Given the extensive technical requirements, extreme effort has been taken to make the software as intuitive and user-friendly as possible. Hopefully, your experience with the software will be rewarding, and you’ll find it satisfies your individual needs.

Contents

Section	Page
Welcome	i
Training Contacts	ii
Workbook Overview.....	ii
Workbook Layout.....	iv
Module 1	
Definitions and Concepts	1-1
Objectives	1-1
Definitions	1-1
Concepts	1-6
Oracle Forms.....	1-9
Using Form Menus	1-19
Summary.....	1-30
Module 2	
UDE Database	2-1
Objectives	2-1
UDE Concept	2-1
File Menu UDE Options.....	2-2
UDE Database Application.....	2-12
Summary.....	2-18
Module 3	
Telemetry Database	3-1
Objectives	3-1
Telemetry Basics.....	3-1
Telemetry Database and Local Table Operations.....	3-2
Summary.....	3-7
Module 4	
Command Database	4-1
Objectives	4-1
Definition of Terms	4-1
Command Structure	4-3
Command Types.....	4-4

Section	Page
Command Response	4-6
Command Database Forms	4-7
Command Database Concepts	4-8
Summary	4-17
 Module 5	
Documenting Permanent Changes to a Database.....	5-1
Objectives	5-1
Database Change Request Application	5-1
Concepts.....	5-1
Database Change Request Form	5-6
DBCR Processing	5-13
Summary	5-15
 Appendix A	
Exercise Solutions	A-1
 Appendix B	
Supported Uplink Data Types	B-1
 Appendix C	
Abbreviations and Acronym List.....	C-1
 Appendix D	
Glossary	D-1

List of Figures

Figure	Page
Figure 1-1, Database Configuration	1-7
Figure 1-2, Database menu	1-9
Figure 1-3, Telemetry Database Query/Update menu	1-10
Figure 1-4, Forms - Example 1	1-11
Figure 1-5, Forms - Example 2	1-13
Figure 1-6, Typical form	1-14
Figure 1-7, Query/Where dialog box	1-16
Figure 1-8, Message area	1-17
Figure 1-9, Typical form menu bar	1-19
Figure 1-10, Action menu	1-20
Figure 1-11, Message dialog box	1-20
Figure 1-12, Close (Save Changes Confirmation) dialog box	1-21
Figure 1-13, Edit menu	1-22
Figure 1-14, Field menu	1-23
Figure 1-15, Editor dialog box	1-24
Figure 1-16, Record menu	1-25
Figure 1-17, Delete Record (Confirmation Required) dialog box	1-26
Figure 1-18, Query menu	1-27
Figure 1-19, Options menu	1-28
Figure 1-20, Message dialog box	1-28
Figure 1-21, Generate ASCII Report dialog box	1-28
Figure 1-22, Help menu	1-29
Figure 2-1, UDE DB Concept	2-2
Figure 2-2, Launchpad File menu	2-2
Figure 2-3, Retrieve UDE main window	2-3
Figure 2-4, Retrieve UDE (Confirmation Required) dialog box	2-4
Figure 2-5, Retrieve UDE (File Already Exists)	2-5
Figure 2-6, Retrieve UDE Message	2-5
Figure 2-7, Store UDE main window	2-8
Figure 2-8, Store UDE (Confirmation Required) dialog box	2-9
Figure 2-9, Store UDE Message	2-10
Figure 2-10, Database cascade menu	2-12
Figure 2-11, UDE Database main window	2-12
Figure 2-12, Query/Update menu	2-12
Figure 2-13, Edit Personal UDE Attributes form	2-13
Figure 2-14, Delete Record (Confirmation Required) dialog box	2-14
Figure 2-15, View All Users UDE Attributes - Query Only form	2-15
Figure 2-16, Copy UDEs dialog box	2-16
Figure 3-1, Local Table Concept	3-2
Figure 4-1, Command Format Structure	4-3

Figure	Page
Figure 4-2, Command Database forms	4-7
Figure 4-3, Command Database Concept	4-8
Figure 4-4, Command Field Detail Form & Header Field Detail Form	4-9
Figure 4-5, Forms Example	4-11
Figure 4-6, Command Update Form Example	4-13
Figure 5-1, Consolidated Concepts	5-4
Figure 5-2, DCR Database, Query/Update menu	5-6
Figure 5-3, DBCR Form	5-7
Figure 5-4, Incorporate DBCR dialog box	5-13

List of Tables

Table	Page
Workbook Numbering System	ii
Database States	1-4
Input vs. Uplink Data Types	4-10
Enter Query Fields	5-8
Supported Uplink Data Types	B-1

Module 1

Definitions and Concepts

Objectives

In order to access and work within the various databases of the EHS, a fundamental understanding of terms, definitions, and concepts as used in the EHS is necessary. Information presented in this initial module is intrinsic to your understanding of the database applications as it forms the building blocks of which the integrated database is built. In this module you will learn:

- what the databases in the EHS are and how their separate iterations relate to each other
- how the databases relate within the physical constructs of workstations and servers
- how databases are manipulated via Oracle forms
- how forms are related to each other
- the composition of forms and their options

Definitions

Relational Database

All of the databases used in the EHS are based on the “relational” database model and consists of a set of tables peculiar to that particular database. In a relational database, the objects, or “things”, about which data are kept are called entities. Each table in the database contains data about a single type of entity. Some of the tables contain information about data characteristics. Other tables are used to “look up” valid values for a particular type of data. Still other tables contain data about the relationship between the entities that are represented by the other tables. Regardless of whether the table contains data about characteristics, look-up values, or relationships, all of the data for each occurrence of the entity represented by the table is located in a single row of the table.

Telemetry Database

The Telemetry Database (TDB) contains the telemetry definitions needed to drive Huntsville Operations Support Center (HOSC) telemetry processing. The Telemetry Database application provides the capability to access and manipulate data in an EHS project telemetry database (PTDB) which contains the telemetry definitions needed to drive HOSC telemetry processing. This PTDB is the source of the real-time telemetry processing tables found in the telemetry local table.

Local Table

The telemetry local table is a copy of the telemetry database stored on a workstation and server. It contains information necessary to perform decommutation, conversion, calibration, and limit sensing of MSIDs. A local table is maintained for a user account for each Telemetry Database that is on-line for a given project, mission, and operational support mode. Thus, the user account ID, database version, and Mission/Operational Support Mode/Project (MOP) uniquely identifies a local table.

Local Table Change File

The Telemetry Database Local Table menus allow a user to modify and create local table change files. The user is displayed a copy of a specified local table. The user can then edit processing information for an MSID such as the calibration data (polynomial coefficients, point pairs, or state codes), and limit/expected state data (limit caution range, limit warning range, or expected state). A Local Table Change file is exactly that, a file which contains the edited information for MSIDs contained in the Local Table. This information may include limit, calibration, or decommutation data.

A change file may be shared among users. The change files must be stored to UDE and marked shareable by the user. Then users from the same MOP (Mission, Operational Support Mode, and Project) can retrieve and apply the change file to their local table.

User Copy

The user copy data for a specific user consists of copies of all data records which the user is authorized to modify in the associated database, linked to the user by his/her EHS account. The user may modify, delete, and add records in his user copy which can then be attached to a Database Change Request (DBCR). User copy data is available to data owners, project database developers, and MOL database developers only.

Command Database

The Command Database (CDB) provides the capability to access and manipulate data in an EHS project command database (PCDB) which contains the command definitions needed to drive HOSC command processing. The Command Database provides a predefined script which defines the format of each of the tables in the Operational Command Database (OCDB). A process provided by the Command Database retrieves data from the baselined PCDB, including the non-configuration controlled command data set and modifiable command chain definitions, and populates the specified OCDB tables.

The Command Database application provides you with the means to create a user copy of commands you are authorized to access so that you can modify their contents, perform certain validation checks, and submit proposed database changes to the Database Coordination Group (DBCG).

Operational Command Database

The Operational Command Database (OCDB) resides on the command server and is part of the Command Subsystem available only during an operational support mode. Before a mission support activity the PCDB is used to create a unique OCDB. The OCDB contains the command data set and command chain definitions, and the uplink patterns for all pre-defined and complete modifiable commands.

User-Generated Data Element Database

The User-Generated Data Element (UDE) Database provides for centralized storage of data elements developed by other EHS applications. These data elements can include displays, computations, scripts, command update forms, local table change files, Near Real Time data requests, exception monitoring configuration files, message handler configuration files, or strip chart recorder configuration files. A separate UDE database is maintained for each project and mission.

UDEs are console tools on the workstation that enable a user to analyze telemetry and generate commands. The EHS applications enable users to generate and operate a UDE, and users have the capability to run multiple applications and windows simultaneously. Most UDEs which process telemetry or commands are required to be validated prior to operation which ensures that the user has correctly generated the UDE. A user may save his/her UDE on the workstation or store the UDE on the database central server. Storing a UDE in the UDE database provides the means for users to share the files. A UDE marked as shareable in the central database can be downloaded to another user's workstation.

Database Application Privileges

The CDB and TDB applications control the functions that their users are allowed to perform through their respective user interfaces. This is accomplished using Command Database and Telemetry Database application privileges. A Command Database or Telemetry Database application privilege identifies a fixed set of functions or capabilities that a privileged user is able to perform through the Command Database or Telemetry Database user interface, respectively. These privileges are a combination of a user's "user profile" setup through UCM (User Configuration Manager software) and privileges setup by the DBA (Database Administrator).

Following is a list of CDB and TDB application privileges and a summary of the capabilities provided to each.

- Default (no application privilege defined - unprivileged user)
 - User can select the PCDB or PTDB to be accessed, but cannot select user copy data for access
 - User can access a directory of PCDBs or PTDBs
 - User can query any data from the PCDB but cannot update data. User can query non-proprietary data from the PTDB but cannot update data
 - User can generate, view, and print standard and project-specific reports
 - User can access on-line help

- Data Owner
 - User has all the default capabilities listed above
 - User can query proprietary data of which he is the owner
 - User can select his user copy data for access
 - User can make updates to his user copy data
 - User can make updates to non-configuration controlled data of which he is the authorized owner
 - User can incorporate his own DBCRs into his own user copy data
- Project Database Developer (DBD)
 - User has all the capabilities provided to a data owner
 - User can generate command or telemetry database products for external delivery
- Mission Operations Laboratory (MOL) DBD
 - User has all the capabilities provided to a project DBD
 - User has access to working PCDB or PTDB tables
 - User can convert and load external data into a working PCDB or PTDB
 - User can execute a process to validate a PCDB or PTDB
 - User can incorporate approved DBCRs into a PCDB or PTDB revision

Database States

Database states are labels that are attached to specific database revisions to identify the stage of existence of those revisions. For PCDBs, such states include Delivered, Pre-released, and Baselined. For PTDBs, such states include Delivered, Pre-released, Baselined, and Archived. The database state is used for database access control. It also provides a means for a user to request access to a database revision based on its state rather than its revision number. The following table provides a list of available database states and a description of each.

Table 1-1, Database States

DB State	Description
Working Database	Database that is being used to prepare for the next delivery. May originate from new source data or may be last revision with updates. Must pass Data Verification.
Delivered Database	Database that has passed Data Verification and Regression Testing. DBCG controls promotion from Working to Delivered. Delivered status can be approved for non-operational (non-critical) activities or to support HMCG schedules. Database never passes this status unless an operational activity is scheduled.

Table 1-1, Database States

Pre-released Database	Database that has passed Data Verification, Regression Testing, and System Validation and is delivered to Operations. Database that is used to validate UDEs and build new local tables and OCDBs prior to the actual release of the database. Cannot be used for real-time data processing. Database that is used for Operations Validation (Mission Manager Verification).
Baselined Database	Database that has passed Data Verification, Regression Testing, System Validation, and Operations Validation. Database that reflects the current local table and OCDB data that is being used in the HOSC for a particular mission activity.
Archived Database	Telemetry database that no longer reflects the current telemetry local table information. Only one available on-line at a time. Primary use is for Near-Real Time capabilities to support processing data that was recorded under an old database.
Real-time Database	Workstation and Project Server local tables and Operational Command Database (OCDB) used by HOSC software to process telemetry and commands, respectively. These databases are used for real-time processing and are generally subsets of a Pre-released, Baselined, or Archived database.

Concepts

To begin, please direct your attention to Figure 1-1, Database Configuration, for an overview of the database configuration. The EHS is a distributed processing system in that the majority of the processing required by user initiated applications is performed on that user's workstation. The EHS workstation provides a multi-application windowed environment for access to the EHS applications. Each user has certain privileges dictated by the user profile which controls access to the applications.

While just about everything you need is sitting in front of you in the form of an EHS workstation, it may be helpful to have an idea of the bigger picture. The SMAC server maintains the system configuration files for all projects which are available to a project LAN during system initialization and mission operations. The database central servers maintain the configuration controlled databases for all projects and the command server provides a central control point where real-time command images are built and uplinked.

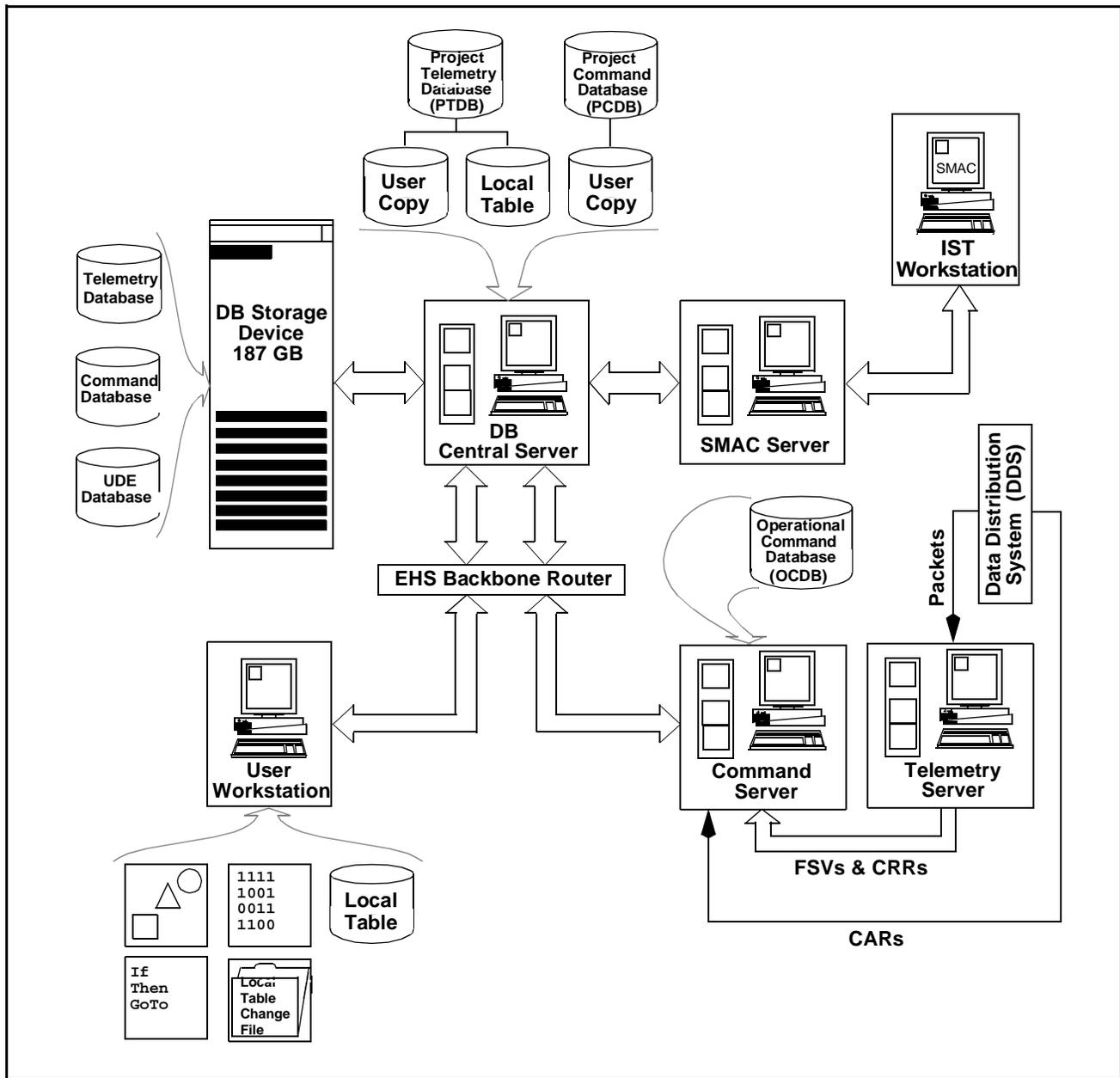


Figure 1-1, Database Configuration

The Database (DB) Central Server hosts the EHS database management system. This system hosts the configuration-controlled telemetry and command databases definition information and provides centralized storage for user-generated data elements (UDEs) (i.e., computations, displays, scripts, etc.). The system provides a configuration management capability for the submission and control of ground processing support data (i.e., telemetry and command databases).

During periods of operational support the required PTDB and PCDB are resident on the DB Central Server. The UDE DB, however, remains in the DB Storage Device with the DB Central Server providing responses to Store/Retrieve requests from individual workstations to the DB Storage Device.

Unlike project servers, command servers and workstations, the DB Central Servers are not initialized for support of a particular EHS mission or activity. Each of the database servers is permanently allocated to an EHS supported project. The Database Servers must be available for development during and between all missions. Also, the Database Servers must be available for development of UDEs between missions. The Database servers are brought down only for maintenance.

The Command Server functions as the EHS gateway for all project commanding activities. This system has the capability to receive commands from local and remote systems, User Ground Support Equipment (UGSE), and remote user facilities. Upon receiving a command, the server executes the appropriate validation processes and checks for hazardous commands before performing the actual command uplink. The system makes command acceptance responses and command activity reports available to the initiating user. The Command Server provides the capability to monitor telemetry values in response to commands and provides messages to the command initiator.

The Telemetry Server receives Space Shuttle telemetry and NASCOM 48-bit block data via the Data Distribution System (DDS). The Telemetry Server also delivers command responses to the commanding subsystem. Command Acceptance Responses (CARs) are received by the Telemetry Server from an intermediate facility in the form of NASCOM 48-bit blocks. The NASCOM block is processed and formatted into a packet and then distributed to the Command Server. Similarly, Flight System Verifiers (FSVs) and Command Reaction Responses (CRRs) are received in the telemetry downlinked from the vehicle, processed, packetized, and distributed to the Command Server.

The Operational Command Database (OCDB) resides on the Command Server. The database tables necessary for the OCDB are created by the Database Monitor and Control utilities (initiated through the SMAC Server) since Database Monitor and Control is responsible for monitoring database disk and table spaces. Command Database provides a process for populating the OCDB tables using data from the baselined PCDB (resident on the DB Central Server) for the specified project and mission/increment. This process retrieves data from the baselined PCDB, including the non-configuration controlled command data set and modifiable command chain definitions, and populates the specified OCDB tables. The uplink patterns for all predefined and complete modifiable commands are generated at this time and stored in the OCDB.

After logging on to an EHS Workstation you may invoke any of the Database applications from the Launchpad by clicking on the **D**atabase menu option and then selecting the desired database application (see Figure 1-2, Database menu).

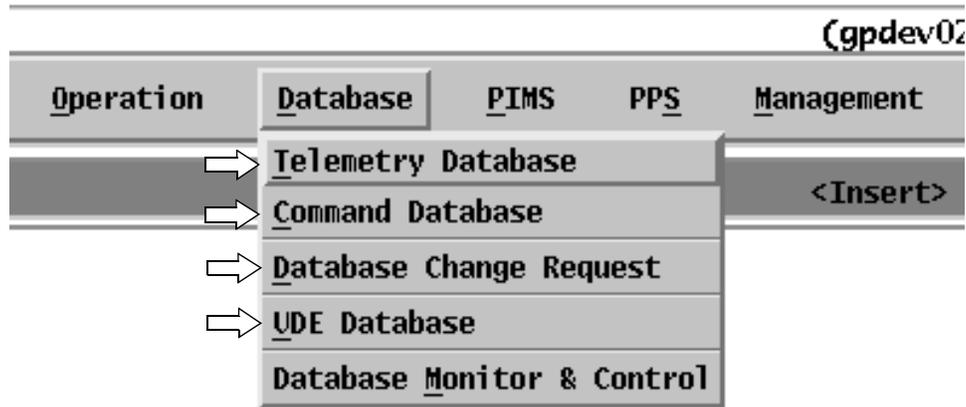


Figure 1-2, Database menu

After initiating one of the database applications, you must first select a database version with which to work. The database that is placed on your workstation is a copy of the project database that is residing on the DB Central Server.

Oracle Forms

Forms are selected through the cascade menus available off the **Query/Update** menu of Database application main windows (see Figure 1-3, Telemetry Database Query/Update menu). Forms are used to both query and update specific items available in the Database. Though each form allows you to search and update different information, all forms have common characteristics, menus and menu items. This section describes the behavior that is common to all these windows.

Note: The distinction between forms, as used within Huntsville Operations Support Center (HOSC) database applications, and mini-applications that are invoked from other HOSC applications, is that forms follow Oracle user interface standards, while HOSC applications follow specific HOSC user interface standards.

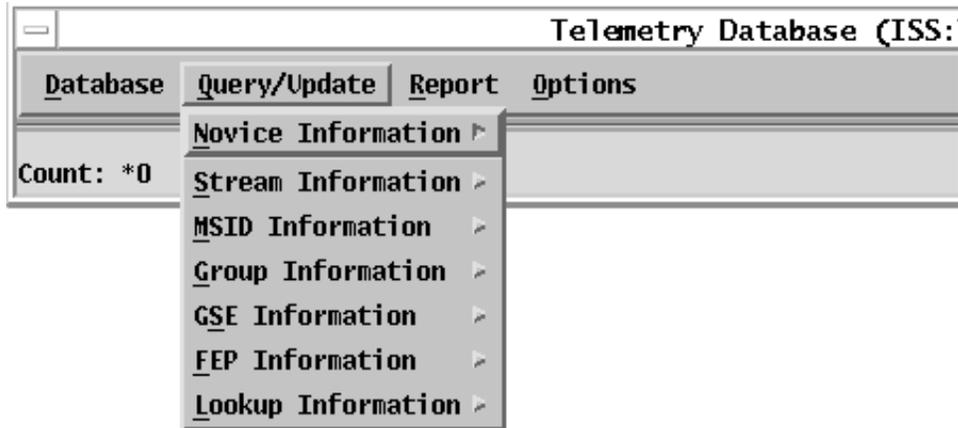
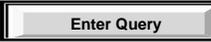


Figure 1-3, Telemetry Database Query/Update menu

Forms operate in one of two modes: **Enter Query** mode and **Normal** mode. Pushing  puts the window in Enter Query mode, and pushing  in Normal mode. In the Enter Query mode the form is initially blank and all fields in the first row of a form are enabled so that selection criteria may be entered. In the Normal mode, all fields with default values are populated with those values and, if you are a privileged user, all fields are enabled.

Just as the entities and tables in the databases are related so are many of the forms, since they are the means of populating the database tables. An example of this can be seen in the forms that are used to define an MSID (see Figure 1-4, Forms - Example 1). The **MSID List - Query Only** form is used to enter MSIDs and their Technical Names into the Measurement Table of the database and in the query mode provides a complete list of all the MSIDs contained within a selected mission database.

Once an MSID has been selected (by executing a query) it can be further defined by either selecting the **MSID Detail...** option from the **Options** menu or clicking  located on the Control Panel which will invoke the **MSID Detail** form. The **MSID Detail** form further defines the information contained in the Measurement Table by defining individual measurements and giving information on data and calibration type.

From the **MSID Detail** form a user can obtain further definition for an MSID by selecting (through the **Options** menu or pushbuttons) the **Decommuration Detail** form, **Calibration Detail** form, or **Limit Sensing Detail** form. If the **Limit Sensing Detail** form is selected, further information can be obtained by clicking on  the **MSID Detail** form for the Switch MSID that is used for limit switching.

The previous example displayed the relationship between list forms and detail forms. The following example (see Figure 1-5, Forms - Example 2) demonstrates how the value contained in a text entry field determines the selection of a detail form.

In this example we are using the **Stream List** form which is used to enter data into the Stream ID Table to define a HOSC-unique stream number for each format and stream. The form contains basic stream identification information and provides the capability to navigate to a stream detail form.

Basically, the form is composed of a table with each row identifying a HOSC-assigned (provided by the project/mission manager) stream number. In this discussion we are particularly interested in the second column of the table. This column is used to define the stream type and is defined as follows:

- “B” - Block
- “P” - Packet
- “S” - Subset
- “T” - TDM

The value of the selected “Type” field determines which detail form will be invoked when either the **Stream Detail...** option or pushbutton is selected. In Figure 1-5 the first row, for Stream Number 1, has been selected, the Type column contains a “T” indicating that a TDM stream is being identified. When **Stream Detail...** is selected, the **TDM Detail** form for stream number 1 will be invoked.

Stream List (ISS-TST1-0004)

Action Edit Field Based Query Options

Stream Number	Type	Stream ID	Format ID	Protocol	Packet ID List	Priority	Source ID	Description
1	P	1001	A	L	0	0	P01-P01001	PACKET 1001 L FORMAT A
2	P	1001	B	L	0	0	P01-P011001	PACKET 1001 L FORMAT B
3	P	1001	C	L	0	0	P01-P012001	PACKET 1001 L FORMAT C
4	P	303	A	L	0	0	P01-P01303	PACKET 303 L FORMAT A
5	P	303	B	L	0	0	P01-P01303	PACKET 303 L FORMAT B
6	P	1000	100	C	1	1	ISS_PACKET	ISS Essential Packet
7	P	1000	101	C	2	2	ISS_PACKET	ISS Rescheduling 1 Packet
8	P	1000	102	C	3	3	ISS_PACKET	ISS Rescheduling 2 Packet
9	P	1011	100	C	04	04	ISS_PACKET	ISS EIS Packet
10	B	13	00		0	0	ELOC_13	ELOC 13 FORMAT 00
11	F	13	141	B	0	0	ELOC_13	ENCAPSULATED ELOC 13 FORMAT 00
12	B	13	01		0	0	ELOC_13	ELOC 13 FORMAT 01

Buttons: Enter Query, Next Record, Info Record, Stream Detail..., Clear Filter, Locate Query, Previous Record, Smart Record, Close

Count: 12 4 (Clear)

The figure shows three overlapping 'Stream Detail' forms, each corresponding to a different 'Type' value from the 'Stream List' table:

- Top Form (Type P):** Shows fields for Packet ID, Packet ID Address, Packet ID Location, Packet ID Size, and Packet ID Value. It also includes a 'Packet Information Definition' section with a tree view of packet types.
- Middle Form (Type B):** Shows fields for Block ID, Block ID Address, Block ID Location, Block ID Size, and Block ID Value. It includes a 'Block Information Definition' section with a table for block definitions.
- Bottom Form (Type F):** Shows fields for Frame ID, Frame ID Address, Frame ID Location, Frame ID Size, and Frame ID Value. It includes a 'Frame Information Definition' section with a table for frame definitions.

Arrows from the 'Stream List' table point to the 'Stream Detail...' button, and then to the respective 'Stream Detail' forms, illustrating that the 'Type' field in the 'Stream List' determines which detail form is invoked.

Value of selected "Type" field determines which detail form will be invoked when Stream Detail... is selected.

Figure 1-5, Forms - Example 2

Form Components

The forms found in the Database applications consist of five major components (see Figure 1-6, Typical form):

- Title bar
- Menu bar
- Work area
- Control panel
- Message area

Each component of the form will be addressed on the following pages.

Note: The **MSID Detail** form is used throughout this module, as an example for displaying the common menus and form components of “query only” forms.

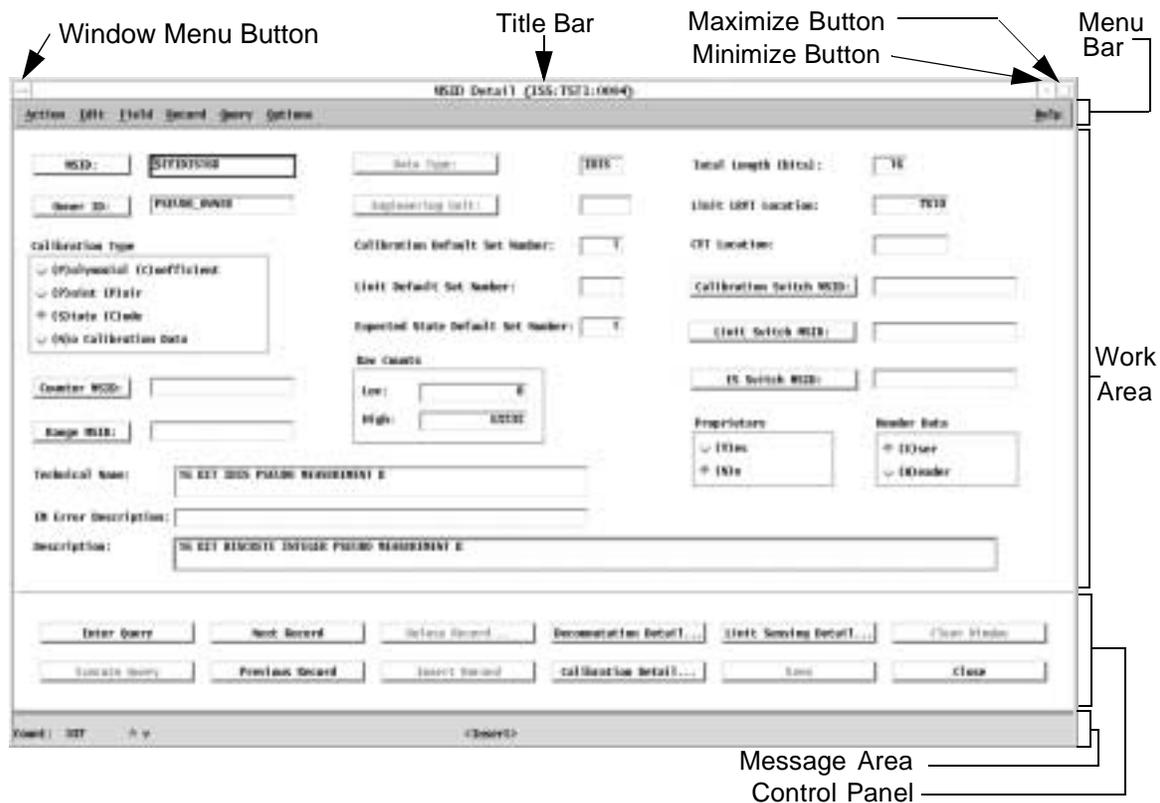


Figure 1-6, Typical form

Title Bar

The form title bar contains the name of the form, project, mission, and revision. This area also reflects when a user copy or Local Table version of the database has been selected.

If you selected a controlled database for access rather than your own user copy data, the title is displayed as follows:

Window Name (<PROJECT> : <MISSION> : <REVISION>)

If you select your own user copy data for access, the window title is displayed as follows:

Window Name (<PROJECT> : <MISSION> : <REVISION> : User Copy)

If you select local table data for access, the window title is displayed as follows:

Window Name (<PROJECT> : <MISSION> : <REVISION> : Local Table)

Menu Bar

The menu bar is organized by function under seven menus: **Action, Edit, Field, Record, Query, Options, and Help. The **Options** menu on each form is unique, but all form **Options** menus include a **Generate Report** menu item. As a result, only the **Generate Report** menu item will be addressed in this module.**

Help: For information on common form menu items, see the “Using Form Menus” section later in this module.

Work Area

The work area of each form provides you with both the capability to define and submit a query, as well as to see the results of that query. If you are working with your user copy or local table, you can also update data within the work area.

When forms are initially invoked, fields within the work area are empty. You enter search criteria on the first row of fields (or use the default) and execute the query. Once the query has been executed, the work area will reflect the results of your query.

Tip: In using any of the database applications a wildcard character of particular interest is the colon (:). Using the colon as a wildcard invokes the **Query/Where** Oracle dialog box (see Figure 1-7, Query/Where dialog box).

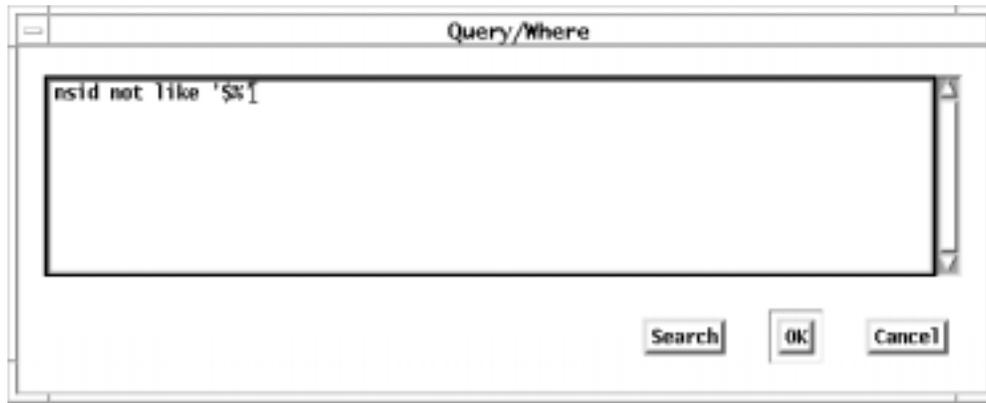


Figure 1-7, Query/Where dialog box

The **Query/Where** dialog box allows you to be either inclusive or exclusive with respect to your search criteria. The text entry field of the dialog box allows you to enter a statement to define your search criteria. Inclusion of the terms “like” or “not like” in the search criteria make the statement either inclusive or exclusive. For example, using the statement “msid not like '%\$%’” will provide a list of all MSIDs that are not pseudo MSIDs. Conversely, the statement “msid like '%\$%’” will provide a list of all MSIDs that are pseudo MSIDs. In order to see the results of your query, click on , and then click on second time (not a double click).

Control Panel

The form control panel includes pushbuttons which allow you to perform many of the more common functions. Pushbuttons available are unique to each form. Any of the following pushbuttons may appear on the control panel of most forms:

<input type="button" value="Enter Query"/>	-clears the work area and becomes grayed out until another query is executed.
<input type="button" value="Execute Query"/>	-executes the query specified by any selection criteria.
<input type="button" value="Next Record"/>	-moves the cursor one record down from where it is currently located within the records of a work area.
<input type="button" value="First Record"/>	-moves the cursor to the top of the list of records.
<input type="button" value="Delete Record"/>	-invokes a dialog box that allows you to mark a record for deletion.
<input type="button" value="Previous Record"/>	-moves the cursor one record up from where it is currently located.
<input type="button" value="Last Record"/>	-moves the cursor to the bottom of the list of records.



-creates a new record and allows you to insert data.



-updates your user copy with changes you've entered on a form.



-clears the work area when selection criteria has been specified.



-closes the form and returns you to the previous window.

Message Area

The form message area is comprised of two lines (see Figure 1-8, Message area):

- Message line
- Status line

The message line displays both Oracle Forms messages and application-specific messages such as hint text and error text. Hint text informs you of some course of action you are expected to take. Error text provides a brief description regarding a particular error that occurred (see Figure 1-8, Message area, Message area).

Tip: After search criteria has been entered in the form, you may use the keyboard accelerators **Ctrl**+**t** (to execute) or **Ctrl**+**b** (to cancel) to either execute/cancel a query instead of using the

Execute Query in the control panel or the **Execute Query** and **Cancel Query** options from the **Query** menu. (See Message line in Figure 1-8, Message Area.)



Figure 1-8, Message area

The status line of forms is the same as that of the selected Database's main window for both the **<List>**, **<Insert>**, and **<Replace>** indicators. The status line of forms is different from that of the selected Database's application main window because it is through forms that you execute database queries.

Prior to executing a query, the **Count:** will show ***0** and the status line will include the phrase **ENTER QUERY**; this is intended to prompt you to enter relevant search criteria on the first line of the form. After you execute a query that results in matching records being found, the **ENTER QUERY** indicator goes away. The **Count:** will show you the number of records retrieved. An asterisk (*) next to the **Count:** indicates that the last record has been fetched. Up and down care **ts()** on the status line indicate

whether there are additional records matching your search criteria above and/or below the selected record. You can use the keyboard arrow keys to move from one selected row to the next.

Note: If no records are returned by a query, the up and down carets (^ v) will not be displayed.

Note: If you are using a list form, the up and down keyboard arrows will select the next or previous record on the list. If you are using a detail form, the up and down keyboard arrows will place the next or previous record's contents in the detail form.

Using Form Menus

The form's menu bar is organized by function under seven menus: **Action**, **Edit**, **Field**, **Record**, **Query**, **Options** and **Help** (see Figure 1-9, Typical form menu bar).

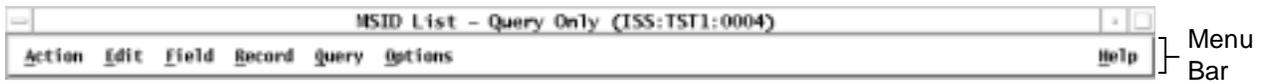


Figure 1-9, Typical form menu bar

The **Action** menu includes menu items that may allow you to save changes, clear the main window, clear selection criteria and unsaved changes, and close the form.

The **Edit** menu includes menu items that allow you to copy any field and cut, copy, paste, and clear text from any field.

The **Field** menu includes menu items that allow you to move between fields. Within this menu, you can also clear and edit fields.

The **Record** menu includes menu items that allow you to move between records and may allow you to mark records for deletion.

The **Query** menu includes menu items to either execute a query or enter a new query.

The **Options** menu includes a menu item that allows you to generate reports based on selection criteria. The resulting report will be viewed in the Oracle Report Previewer.

The **Help** menu provides an index of help topics and provides help on the applicable main window.

Action Menu

Items that may be available under the **Action** menu allow you to perform the following functions: **Save**, **Clear All...**, **Clear Window**, and **Close** (see Figure 1-10, Action menu).

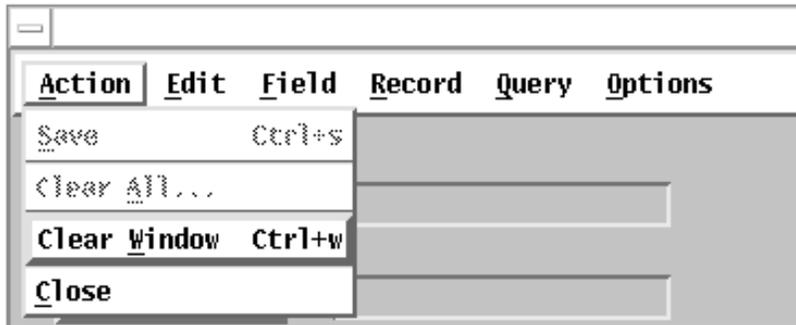


Figure 1-10, Action menu

Save

Save updates the Database with changes you have made and deletes those items you have marked for deletion.

Note: This option is only available if a change is detected.

Clear All...

The **Clear All...** menu item is only available after a query has been executed and a change has been made. This menu item clears the work area and any unsaved changes. You will be asked to confirm this action (see Figure 1-11, Message dialog box).

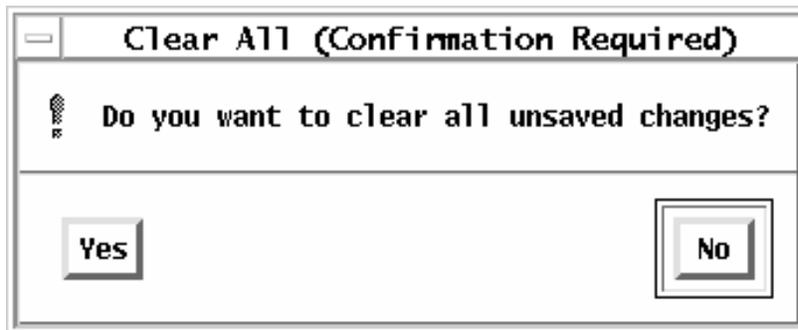


Figure 1-11, Message dialog box

Clear Window

The **Clear Window** menu item is available after selection criteria has been specified. This menu item clears the work area and selection criteria.

Close

The **Close** menu item closes the form and returns you to the selected database's main window. This option is available in all modes and all states.

If you try to close the form and have not saved changes, an alert dialog box will be invoked (see Figure 1-12, Close (Save Changes Confirmation) dialog box).

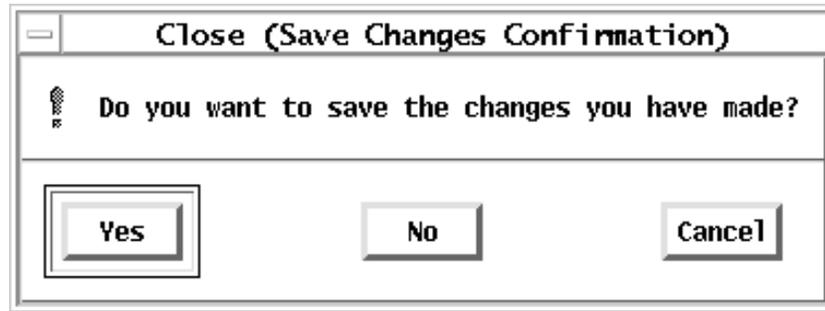


Figure 1-12, Close (Save Changes Confirmation) dialog box

- saves the changes prior to closing the **form** and returns you to the selected database's main window. This is the default pushbutton.
- closes the form without saving changes and returns you to the selected database's main window.
- returns you to the originating form.

Edit Menu

The menu items available under the **Edit** menu allow you to cut, copy, paste, and clear fields within the work area (see Figure 1-13, Edit menu).

Note: Because of the Oracle user interface, copy and paste functions that can be performed with mouse buttons in other HOSC applications cannot be used within database applications. You must use menu items to perform these functions.

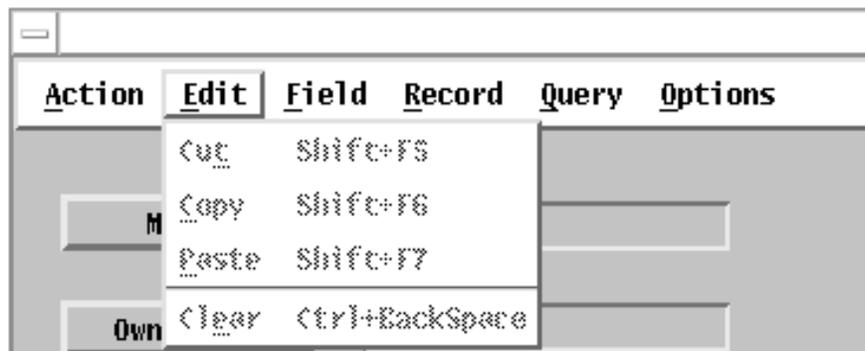


Figure 1-13, Edit menu

Cut

Cut can be used prior to executing a query when you are defining your query criteria. The selection criteria is displayed on the first line below the column headings. You can highlight the field and select **C**ut, then enter a new selection criteria.

Note: The **C**ut menu item is also available after a query has been executed to use with any modifiable fields.

Copy

The **C**opy menu item allows you to copy any highlighted field to the clipboard. Fields copied will not be removed from the work area.

To copy a field, select the field that you wish to copy. Select **C**opy from the **E**dit menu.

Paste

The **P**aste menu item allows you to place a copy of the clipboard into a field.

To paste, click in the field in which you wish to place the contents of the clipboard and then select **P**aste.

Clear

When defining a query in the work area, you can highlight a field and select **C**lear. The **C**lear menu item allows you to clear the selected field. You can then enter a new selection criteria.

Note: After a query has been executed, you can use the **C**lear menu item to clear any modifiable field.

Field Menu

The **Field** menu includes menu items that allow you to move between fields of a record as well as clear and edit a field (see Figure 1-14, Field menu).

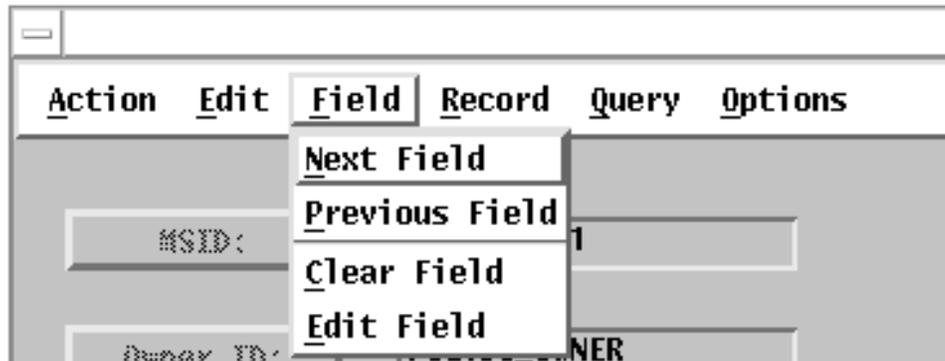


Figure 1-14, Field menu

Next Field

The **Next Field** menu item moves the cursor one field to the right from where it is currently located within a record.

Previous Field

The **Previous Field** moves the cursor one field to the left from where it is currently located within a record.

Clear Field

The **Clear Field** menu item clears the contents of modifiable fields.

Duplicate Field

The **Duplicate Field** makes a copy of the highlighted field and inserts it in the next field available within the database.

Note: This menu item is only available when working with updateable forms.

Edit Field

The **Edit Field** menu item invokes the **Editor** dialog box (see Figure 1-15, Editor dialog box). This dialog box allows you to change the value of modifiable fields.

Note: When defining a query, all entry fields are modifiable.

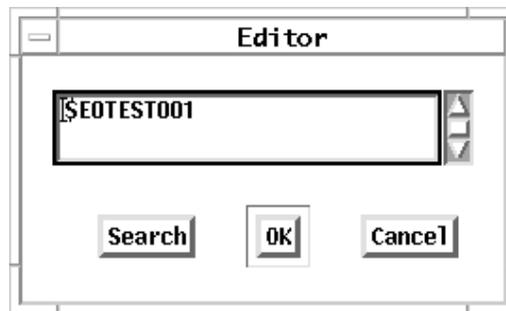


Figure 1-15, Editor dialog box

The **Editor** dialog box contains an input text field that displays the text located within the selected field.



- invokes a **Search/Replace** dialog box.



- implements the changes made on the **Editor** dialog box and returns you to the **Query/Update** window.



- closes the **Editor** dialog box without making any changes.

Record Menu

The **Record** menu includes menu items that allow you to move between records, as well as create, duplicate and delete records (see Figure 1-16, Record menu).

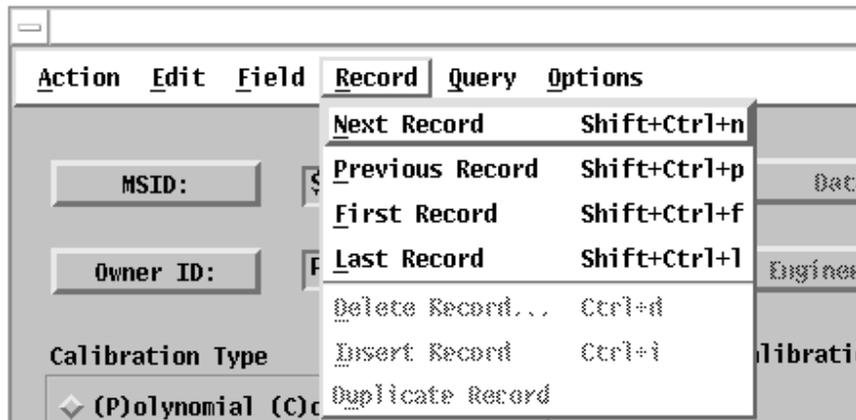


Figure 1-16, Record menu

Next Record

The **Next Record** menu item moves the cursor one record down from where it is currently located; it is insensitive if you are on the last record.

Previous Record

The **Previous Record** menu item moves the cursor one record up from where it is currently located; it is insensitive if you are on the first record.

First Record

The **First Record** menu item moves the cursor to the first of those returned records.

Last Record

The **Last Record** menu item moves the cursor to the last of those returned records.

Note: This menu item is useful when you are trying to determine how many records matched your selection criteria.

Delete Record

The **Delete Record** menu item allows you to mark a record for deletion. The record is not removed from central storage until you click  **Save** from the **Action** menu.

Note: This menu item is only available when working with updateable forms or a user copy of the database.

You will be asked to confirm the deletion (see Figure 1-17, Delete Record (Confirmation Required) dialog box).

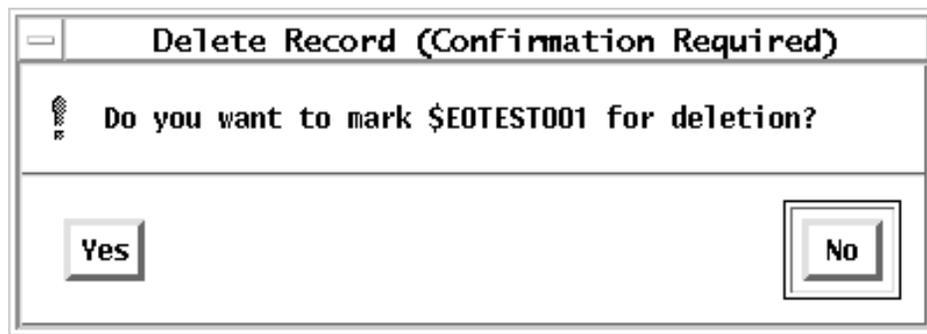


Figure 1-17, Delete Record (Confirmation Required) dialog box

Insert Record

The **Insert Record** inserts a blank record below the selected record. This function provides you with the means to create new PTDB entries.

Note: This menu item is only available when working with updateable forms or a user copy of the database.

Duplicate Record

The **Duplicate Record** menu item makes a copy of the highlighted record and inserts it in the next available record on the form. You can then use this duplicated record as a template for creating a new record.

Note: You must already have a blank record before this function can be implemented.

Query Menu

The **Query** menu includes the **Enter Query** and **Execute Query** menu items (see Figure 1-18, Query menu). An additional menu item is available when working with updateable forms - **Cancel Query**.

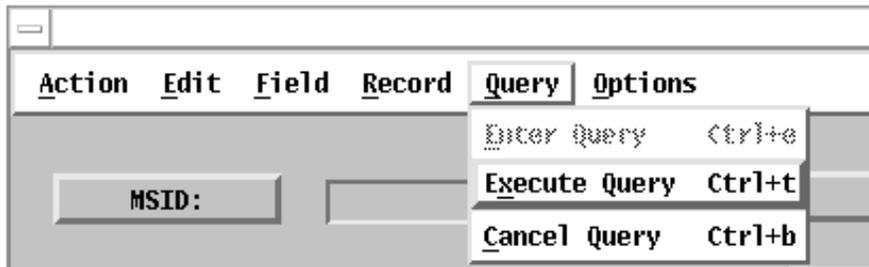


Figure 1-18, Query menu

Note: These menu items provide redundant action for and control panel of all forms.

Enter Query

The **Enter Query** menu item allows you to specify any new selection criteria. If a query has been executed, selecting **Enter Query** will clear the form and becomes insensitive (grayed out) until the new query is executed.

Execute Query

The **Execute Query** menu item initiates a database query based on any specified selection criteria. If you choose to enter no selection criteria and execute a query, all applicable data will be returned.

Note: If no selection criteria is specified, the executed query will return information on all records.

Cancel Query

The **Cancel Query** menu item stops the currently executing query and allows you to enter new selection criteria or execute the default query.

Note: This menu item is only available when working with updateable form.

Options Menu

Forms **Options** menus may include several menu items, but **Generate Report** and **Generate ASCII Report** are common to all forms and therefore, are the only two menu items which will be addressed (see Figure 1-19, Options menu).

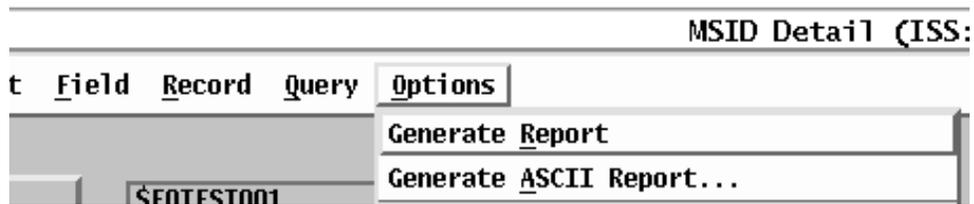


Figure 1-19, Options menu

Generate Report

The **Generate Report** menu item will create a report of the information displayed within the currently shown work area. When you select this menu item the application counts the number of records that will be in the report and provides you with the opportunity to cancel the report (see Figure 1-20, Message dialog box).

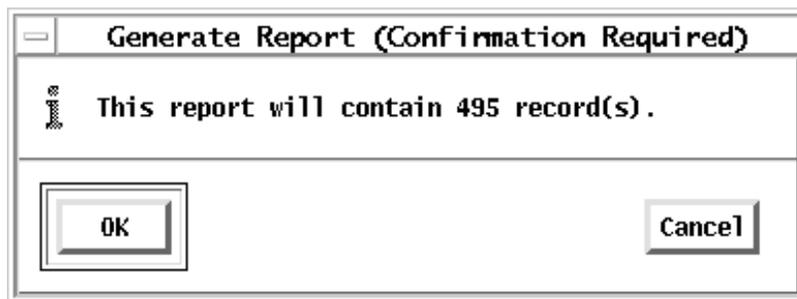


Figure 1-20, Message dialog box

Generate ASCII Report

The **Generate ASCII Report** menu item will create an ASCII report of the information displayed within the currently shown work area. When selected you are prompted to supply an ASCII filename as shown in Figure 1-21.

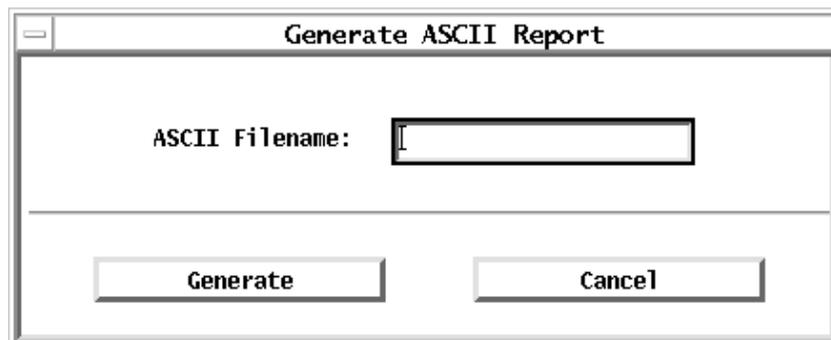


Figure 1-21, Generate ASCII Report dialog box

Help Menu

The **H**elp menu is the last menu item on the **Query/Update** menu bar (see Figure 1-22, Help menu). It provides the capability for users to access information that is specific to the current window of operation. It includes the **O**n **W**indow, **O**n **K**ey, **I**ndex, and **O**n **V**ersion menu items.

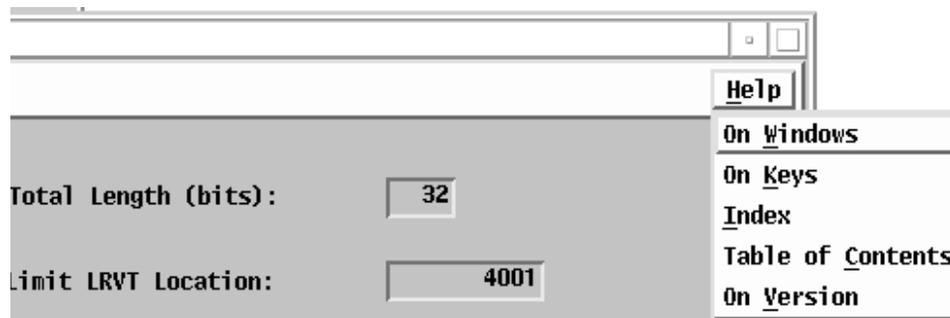


Figure 1-22, Help menu

Note: Forms follow Oracle user interface standards, while HOSC applications follow specific HOSC user interface standards. An inconsistency exists between the two standards in that when an Oracle dialog box is invoked through the **H**elp menu of a form while in the query mode, the dialog box will be displayed to the user behind the calling form. To obtain focus on the Oracle dialog box, simply click on its Title bar.

Summary

This module has laid the groundwork for an understanding of the databases, as used in the EHS, beginning with a definition of the individual databases, their capabilities and states. We defined what your general capabilities as a user are by providing a description of database user privileges. We continued by discussing the operational concepts between EHS workstations, various servers and where the individual databases are resident. We concluded with a discussion of Oracle forms, how they relate to each other, and how they are used.

With the conclusion of this module you should be able to:

- define the terms and definitions pertinent to the EHS database applications
- illustrate the operational relationship between workstations and servers
- demonstrate how to navigate between Oracle forms
- demonstrate the capabilities available via form menus

Questions

Instructions

Indicate the answers for each question below. The correct answers are given immediately following in the **Answers** section.

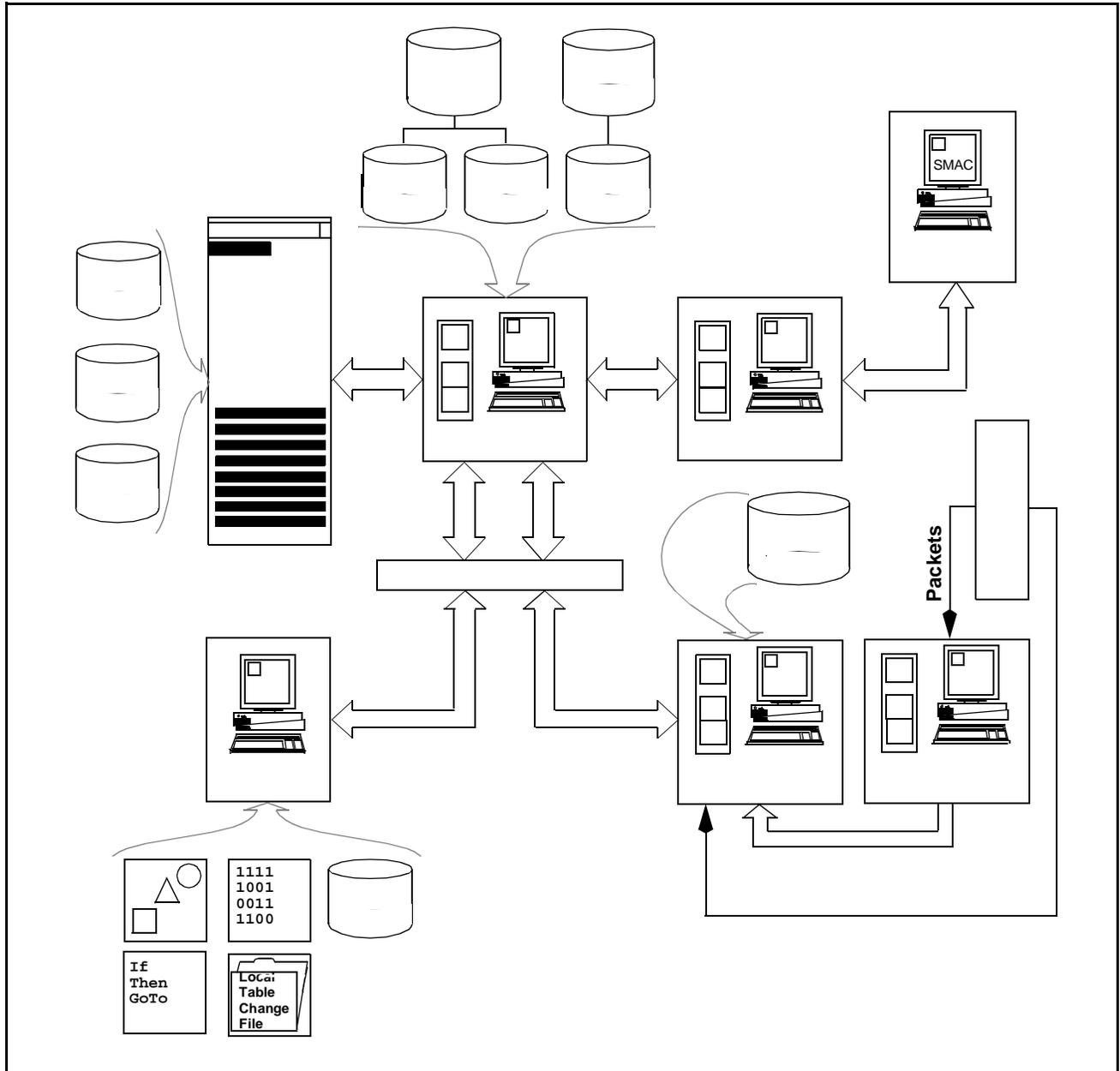
1. Which of the following is not a privilege attributed to an unprivileged user?
 - a. User can access a directory of PCDBs or PTDBs.
 - b. User can access online help.
 - c. User can select his/her user copy for access.
 - d. User can query any data from PCDB.

2. You have been working in your Local Table copy of the PTDB, while reviewing the MSID Detail form you decide that some of the records have to be deleted. What actions are required to accomplish this?

3. Define a Baseline Database.

4. Label the blocks in the Database Configuration diagram using the numbers listed for the following phrases:
 1. Telemetry Database (x2)
 2. Command Database (x2)
 3. UDE Database (x2)
 4. Telemetry Server
 5. SMAC Server
 6. EHS Backbone router
 7. Command Server
 8. DB Storage Device
 9. Project Telemetry Database
 10. Local Table (x2)
 11. IST Workstation
 12. User Copy (x2)
 13. Operational Command Database

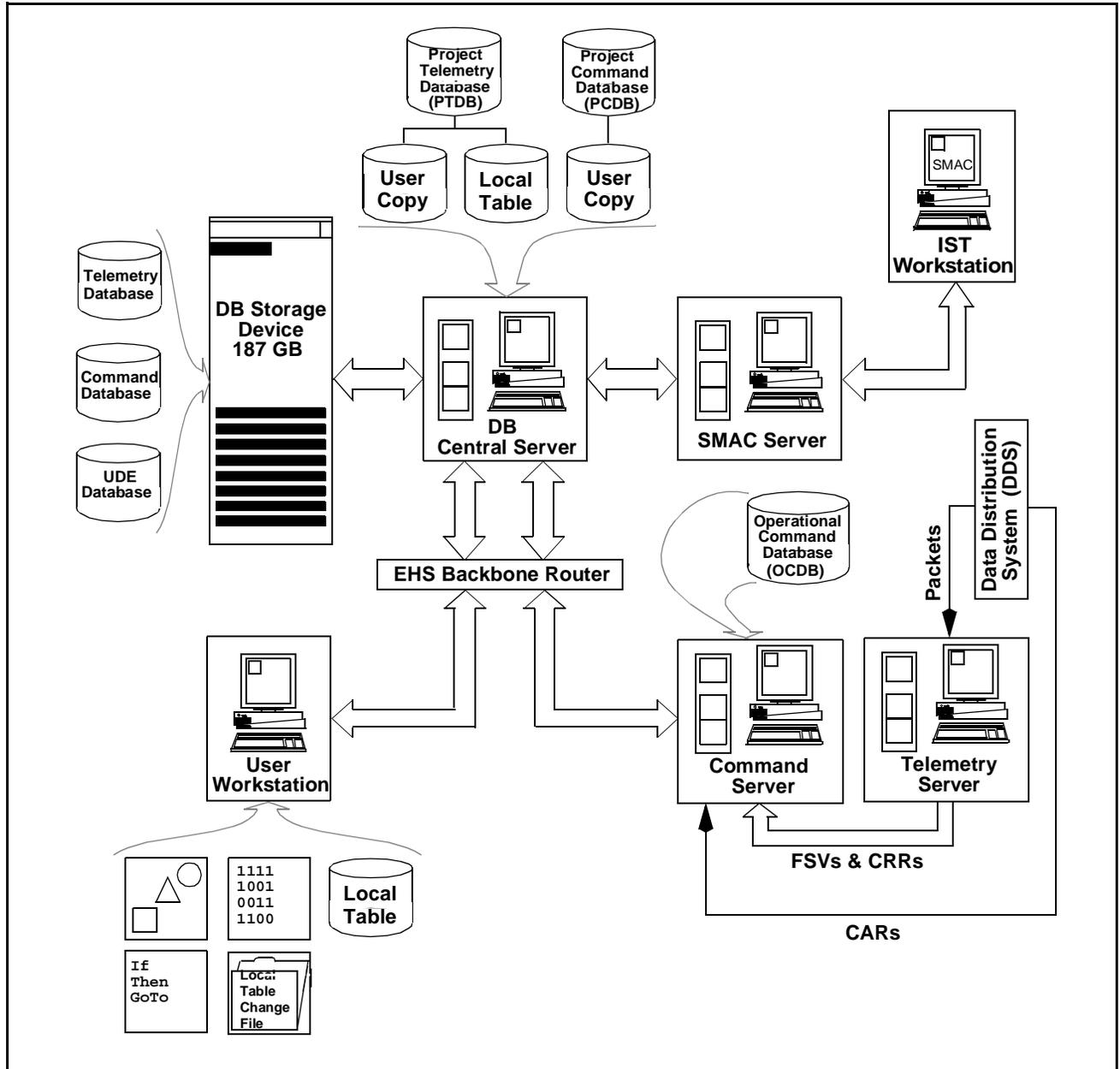
14. Database Central Server
15. User Workstation
16. Project Command Database
17. Data Distribution System (DDS)
18. FSVs & CRRs
19. CARs



Answers

1. c. You must have the privileges of a data owner or above to have a user copy.
2. Select the **D**elete Record option from the **R**ecord menu or click . The record is not removed from central storage until you click  **S**ave from the **A**ction menu.
3. Database that has passed Data Verification, Regression Testing, System Validation, and Operations Validation. Database that reflects the current local table and OCDB data that is being used in the HOSC for a particular mission activity.

4.



Module 2

UDE Database

Objectives

As previously noted, the UDE Database application provides for centralized storage of data elements developed by other EHS applications. In this module you will learn:

- how to store and retrieve UDEs
- how to make UDEs shareable
- how to view all users UDE attributes
- how to copy UDEs between mission databases

UDE Concept

A User-Generated Data Element can be a display, computation, script, change file, etc., that is developed by an EHS application for use during operations. Because of the diversity of applications involved, a practical consideration was to store all these diverse elements in one UDE Database instead of a number of individual databases.

Consider what a UDE is. Basically, it is just a “thing” that was generated for use during operations. As with all things, UDEs have attributes that make them unique. The attributes that we are concerned with are Data Element Type, Name, whether it is shareable or not, the date it was stored, and its size (in bytes).

Aside from a UDE’s purpose during operations, there are only a couple of ways to manipulate them; we can store or retrieve them to the database, or we can edit their attributes (which really just amounts to making them shareable or non-shareable to other users). Because of this, the UDE Database can be accessed via two different methods:

- Launchpad, File menu, Store UDE and Retrieve UDE
- Launchpad, Database menu, UDE Database application

The first method of accessing a UDE, via the **File** menu on the **Launchpad**, simply retrieves the UDE from the database for use in generation or operation applications. If it is being used in a generation application the UDE itself is being developed or edited. If it is being used in an operation application the UDE is being used for the purpose that it was developed for. Store UDE simply puts the UDE away in the UDE Database.

The second method of accessing a UDE, via the **Database** menu on the **Launchpad**, provides you with capabilities to edit the attributes (shareable/non-shareable) of a UDE or copy UDEs from one mission database to another.

Regardless of how you navigated to the UDE application, the underlying concept is the same: UDEs are stored to or retrieved from the UDE Database (via the DB Central Server) from/to your EHS Workstation (see Figure 2-1, UDE DB Concept).

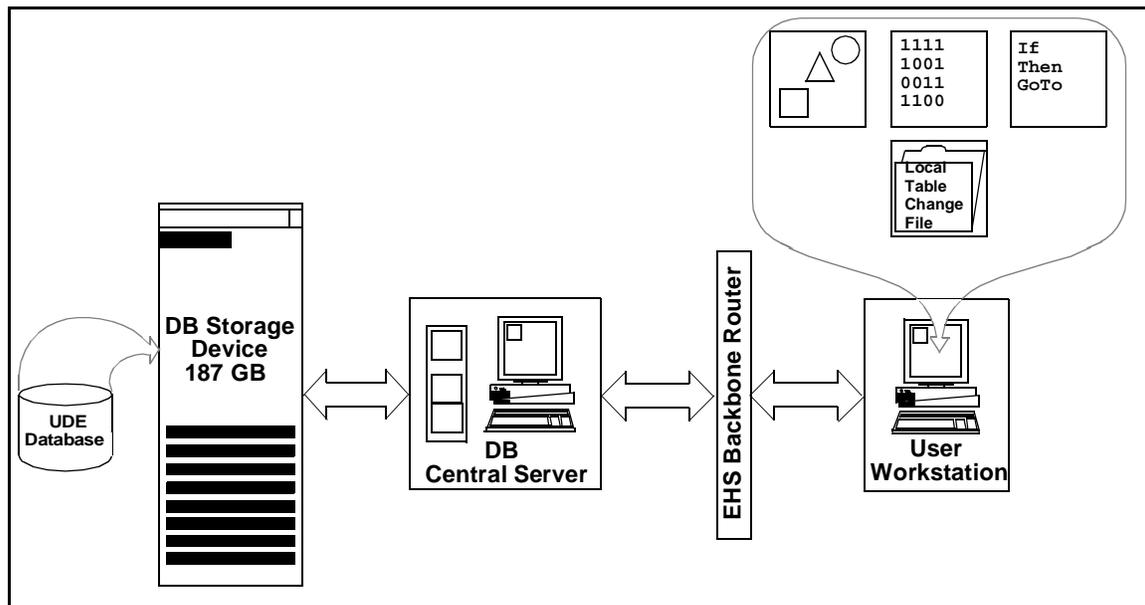


Figure 2-1, UDE DB Concept

File Menu UDE Options

The **F**ile menu, available on the Launchpad includes those applications associated with workstation file manipulations such as opening, storing, printing, etc. (see Figure 2-2, Launchpad File menu).

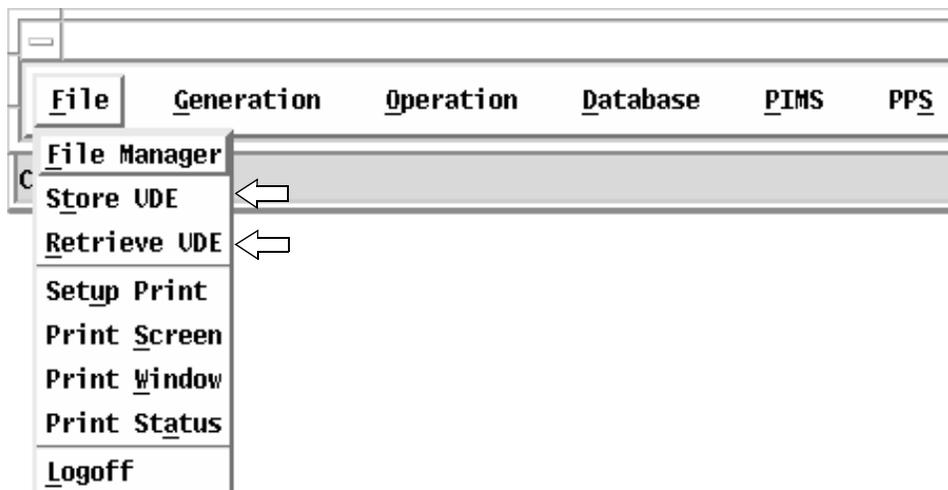


Figure 2-2, Launchpad File menu

As previously mentioned the two options in the **F**ile menu that are available for accessing the UDE

application are **Store UDE** and **Retrieve UDE**. Each of these menu options will be discussed in the following sections. (For purposes of this discussion, the **Retrieve UDE** option will be discussed before the **Store UDE** option.)

Retrieve UDE

The Retrieve UDE application allows you to request that UDEs be retrieved from the UDE database to your local workstation. You can retrieve UDEs that you created or any UDEs marked as shareable within the same project and mission. When the UDE is retrieved, a copy of the UDE is transferred to your local workstation.

When **Retrieve UDE** is selected from the **File** menu the **Retrieve UDE** main window is invoked (see Figure 2-3, Retrieve UDE main window).

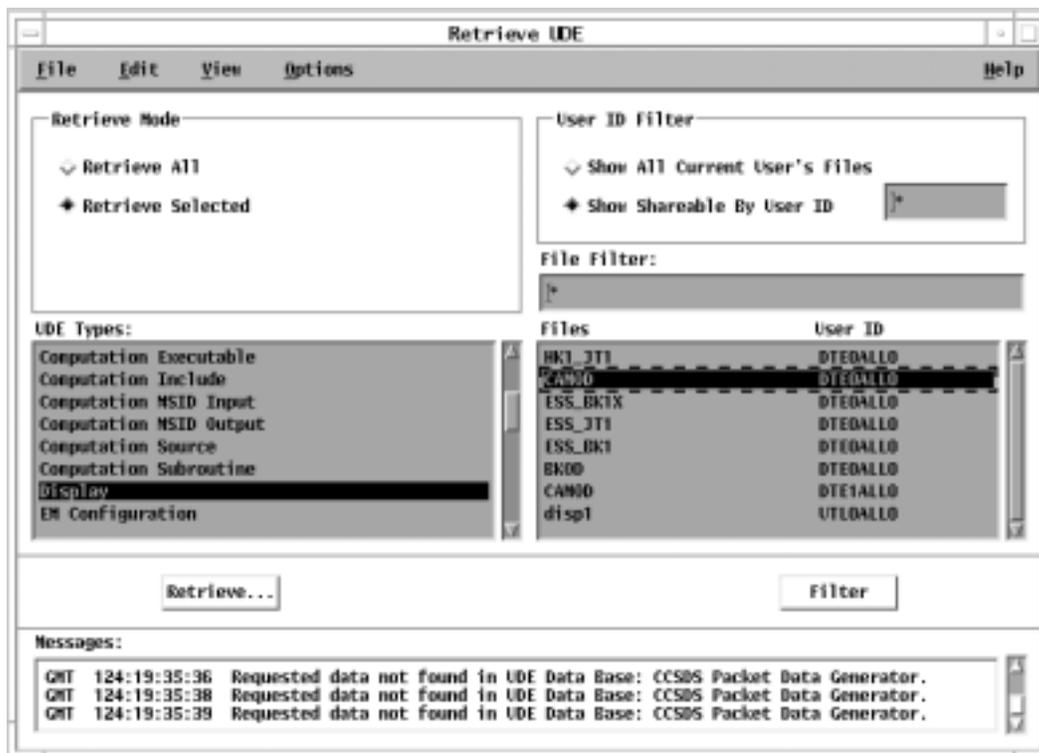


Figure 2-3, Retrieve UDE main window

You may retrieve all UDEs of all UDE types located in central storage to your workstation by selecting **Retrieve All**. The **Retrieve Selected** radio button allows retrieval of selected UDEs of a selected UDE type from central storage and causes the **UDE Types:**, **User ID Filter**, **File Filter:**, and **Files:** frames to become active.

The **UDE Types:** selection list allows you to select the type of UDE whose files you want to see. A selection from this list causes available files to be listed in the **Files:** selection list. Clicking on a file in the **Files:** list selects it for retrieval (a second click on the file will deselect it).

The **User ID Filter** frame allows you to differentiate between all the files belonging to the current user (yourself) for a particular UDE Type and those files that are sharable by User ID for a particular UDE Type. As you remember from Module 1, shareable UDEs can be copied for other users to use besides the owner of the file.

The **File Filter:** input text field provides a means of specifying search criteria for **Files:**. Clicking



lists the files matching your filter criteria in the **Files:** selection list.

When you are satisfied with your selection criteria, clicking  initiates the retrieve process and invokes the Retrieve UDE (Confirmation Required) dialog box (see Figure 2-4, Retrieve UDE (Confirmation Required) dialog box).

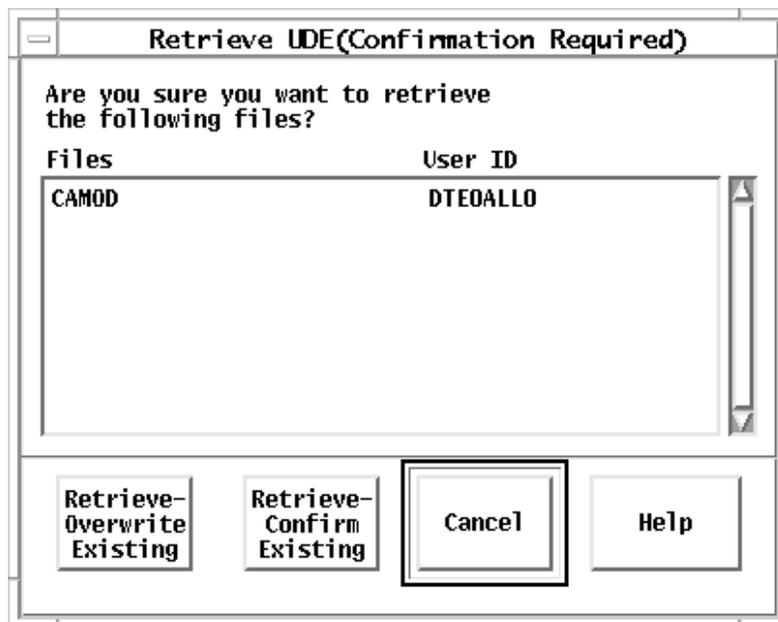
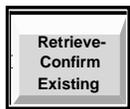


Figure 2-4, Retrieve UDE (Confirmation Required) dialog box

This dialog box presents you with a final decision to continue the retrieve function or abandon it by

selecting , which will return you to the Retrieve UDE main window. Clicking 

retrieves the selected UDEs from central storage overwriting any that may already exist on the

workstation. Clicking  the selected UDEs from central storage invoking a

confirmation dialog box (see Figure 2-5, Retrieve UDE (File Already Exists)) if the file is already resident on the workstation.

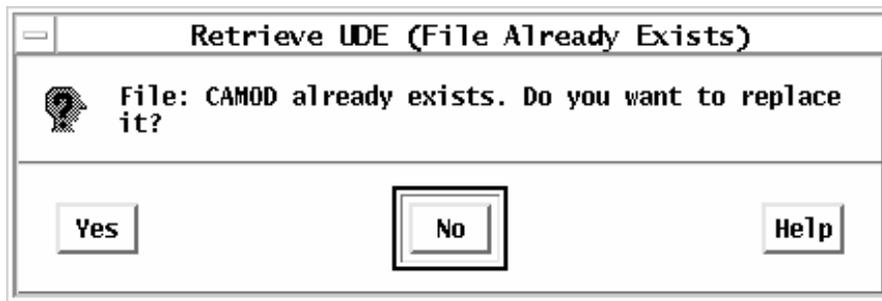


Figure 2-5, Retrieve UDE (File Already Exists)

Whether you have chosen to overwrite or replace any existing UDEs on your workstation the results of the retrieve operation will be delineated in the **Messages:** area of the Retrieve UDE main window (see Figure 2-6, Retrieve UDE Message).

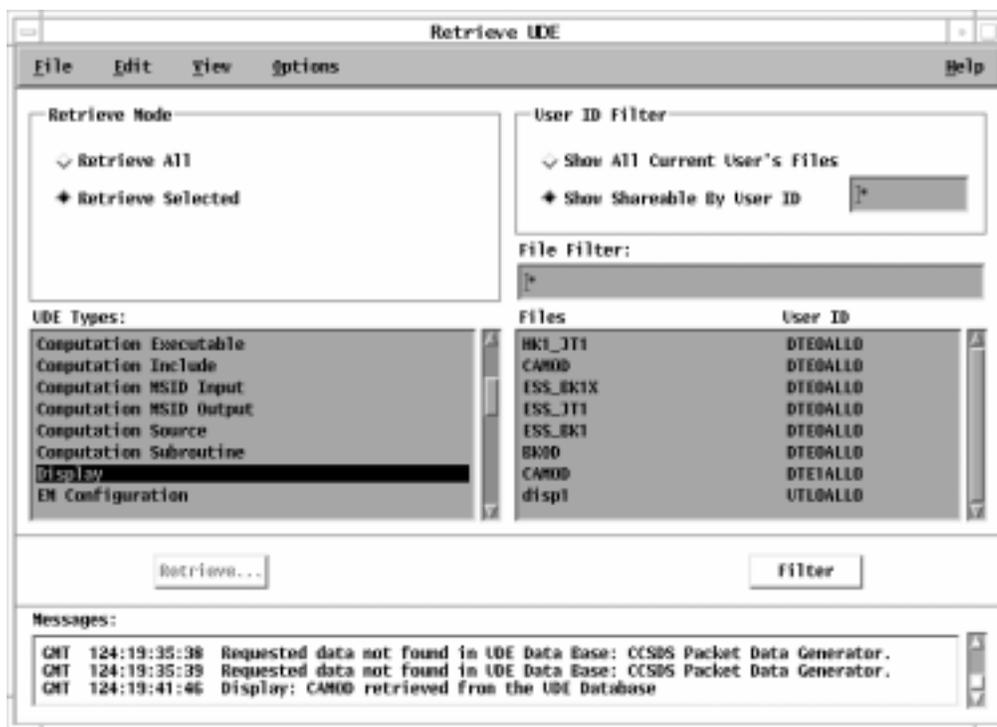


Figure 2-6, Retrieve UDE Message

Once a shareable UDE has been retrieved, you become the owner of it. It is stored under your User ID on the Workstation. If you store it back to the UDE database, it is stored in your storage area. When it is downloaded, the sharable flag defaults to **No**. If the user who retrieved it wishes for the UDE to remain shareable he will have to edit the UDE to make it shareable using the **Edit Personal UDE Attributes** option from the **Query/Update** menu of the **UDE Database** application.

Retrieve All UDEs:

1. Select **Retrieve UDE** from the **File** menu on the **Launchpad**.

2. Select the **Retrieve All** radio button within the **Retrieve Mode** frame. This is the default setting.
3. Click .
4. Confirm the retrieval by clicking  on the **Retrieve UDE (Confirmation Required)** dialog box.

Retrieve Selected UDEs:

1. Select **Retrieve UDE** from the **File** menu on the **Launchpad**.
2. Select the **Retrieve Selected** radio button within the **Retrieve Mode** frame.
3. Select the UDE type from the **UDE Types:** list.
4. Click on file(s) that you want to retrieve.
5. Click .

6. Confirm the function by clicking either   **Retrieve UDE (Confirmation Required)** dialog box.

Exercise

Instructions

Retrieve several UDE files types from the UDE Database to your local workstation.

Why?

Any product you create using the EHS software becomes a UDE and can be stored to and retrieved from the UDE Database. It will then be available to you (and other users if you desire) during future login sessions using any EHS workstation. In this exercise, you will become familiar with retrieving different types of UDEs from the UDE Database to your local workstation.

Try It...

1. Click on the **File** menu on the **Launchpad**.
2. Click on **Retrieve UDE**.
3. Within the **Retrieve Mode** frame, click on the **Retrieve Selected** radio button.
4. From the **UDE Types:** list, select the **Display UDE** file type.

5. Click on **Show Shareable By User ID** radio button and click **Filter**.
6. Select the display **ISSTest** from the **Files** list.
7. Select **Retrieve...** from the **Options** menu,

OR

At the keyboard type  , 

OR

Click  .

8. Perform previous steps 4-7 to retrieve **Script**, **Computation Input**, and **Computation Output** UDE types.

Store UDE

The **Store UDE** application allows you to copy UDEs from your local workstation to the UDE database. Since workstations may be reconfigured between activities, this function allows you to store UDEs so that they will be available to you during a later activity. When the UDE is stored, a copy of the UDE is transferred to the UDE Database.

Note: The Store UDE function provides you with a means to backup files from your workstation.

When Store UDE is selected from the File menu the Store UDE main window is invoked (see Figure 2-7, Store UDE main window).

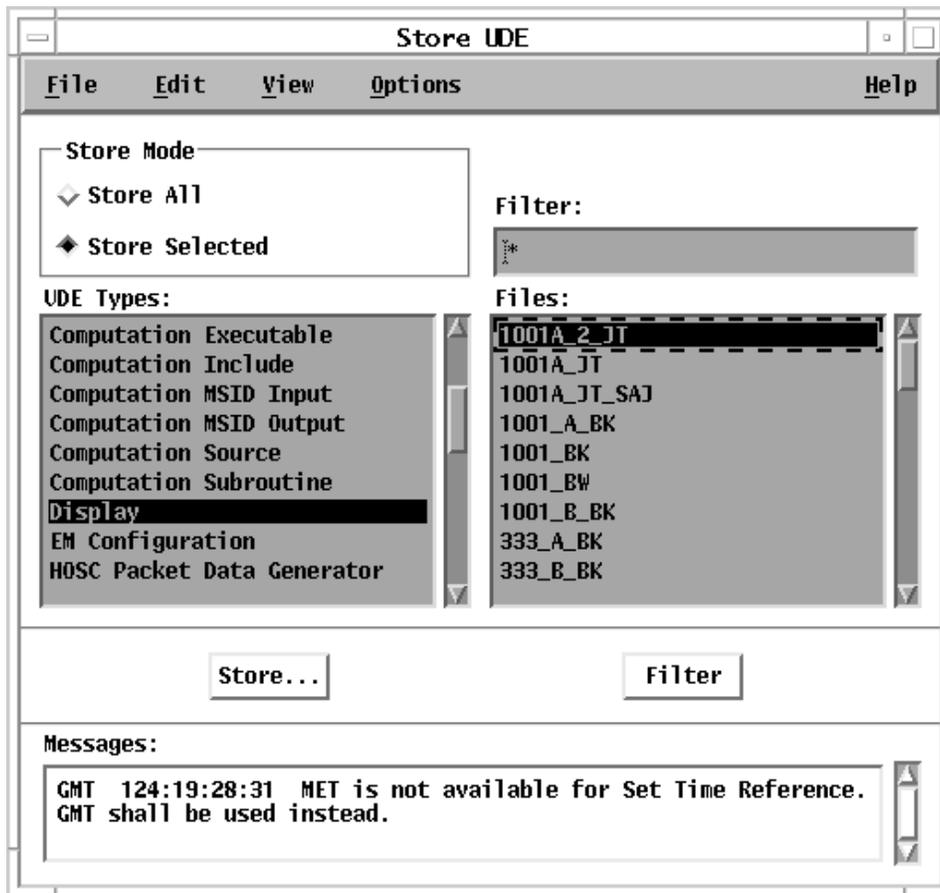


Figure 2-7, Store UDE main window

You may store all UDEs of all UDE types located on your workstation to central storage by selecting **Store All**. The **Store Selected** radio button allows storage of selected UDEs of a selected UDE type to central storage and causes the **UDE Types:**, **Filter:**, and **Files:** frames to become active.

The **UDE Types:** selection list allows you to select the type of UDE whose files you want to see. A selection from this list causes available files to be listed in the **Files:** selection list. Clicking on a file in the **Files:** list selects it for storage (a second click on the file will deselect it).

The **Filter:** input text field provides a means of specifying search criteria for **Files:**. Clicking lists the files matching your filter criteria in the **Files:** selection list.

Clicking the store function and invokes the Store UDE (Confirmation Required) dialog box (see Figure 2-8, Store UDE (Confirmation Required) dialog box).



Figure 2-8, Store UDE (Confirmation Required) dialog box

This dialog box presents you with a final decision to continue the store function or abandon it by selecting **No**, which will return you to the Store UDE main window. Clicking **Yes** continues the store function and returns you to the Store UDE main window.

The results of the store operation will be delineated in the **Messages:** area of the Store UDE main window (see Figure 2-9, Store UDE Message

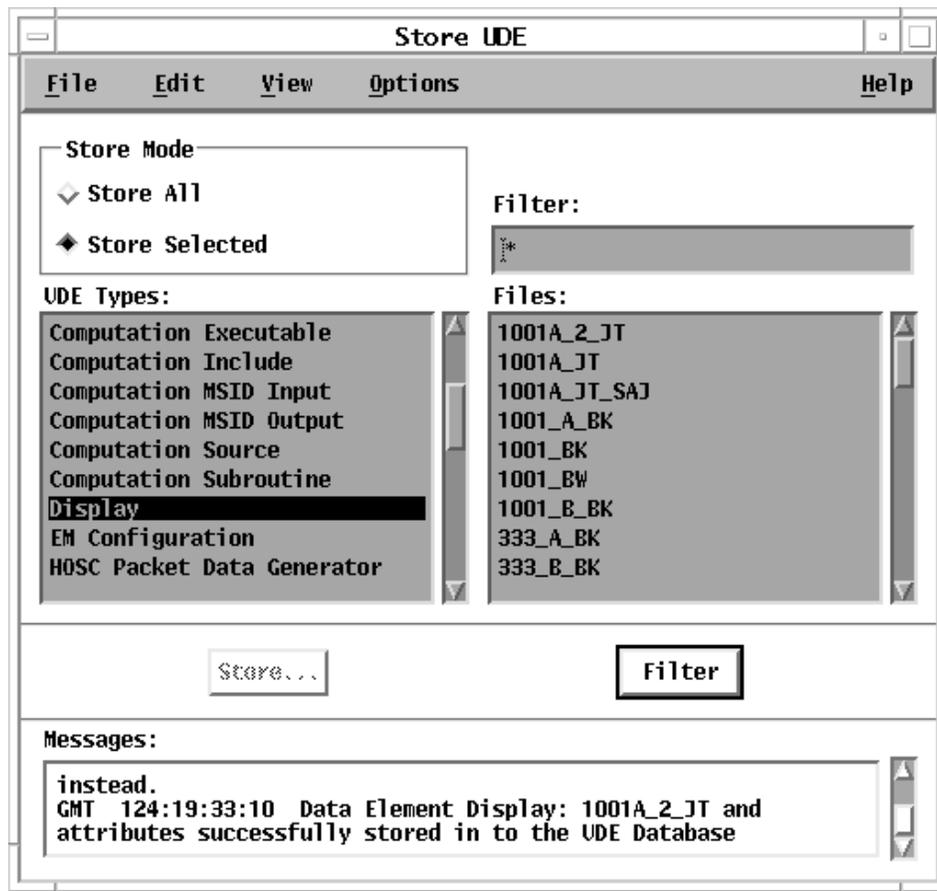


Figure 2-9, Store UDE Message

Store All UDEs:

1. Select **Store UDE** from the **File** menu on the **Launchpad**.
2. Select the **Store All** radio button within the **Store Mode** frame.
3. Click .
4. Confirm the store by clicking  on the **Store UDE (Confirmation Required)** dialog box.

Store Selected UDEs:

1. Select **Store UDE** from the **File** menu on the **Launchpad**.
2. Select the **Store Selected** radio button within the **Store Mode** frame.
3. Select the UDE type from the **UDE Types:** list.
4. Click on file(s) that you want stored.

5. Click .

6. Confirm the store by clicking  on the **Store UDE (Confirmation Required)** dialog box.

CAUTION: If you are storing UDEs that have the same name as a UDE already stored on the central server, this action will overwrite the stored UDE. Use caution when performing this action.

Exercise

Instructions

Store a **Display** UDE type from your local workstation to the UDE database.

Why?

When you create a UDE, it is stored on your local workstation and may not be available to you during future login sessions. The **Store UDE** application provides you with the means to save a permanent copy of your UDE and make it available to other users if you desire. In this exercise, you will practice storing a **Display** UDE type from your local workstation to the UDE Database.

Note: You will learn about making UDEs available to other users in the section titled, **Edit Personal UDE Attributes** later in this module.

Try It...

1. Click on the **File** menu on the **Launchpad**.
2. Click on **Store UDE**.
3. Within the **Store Mode** frame, click on the **Store Selected** radio button.
4. Select **Display** from the **UDE Types**: list.
5. Select the display named **ISSTest** from the **Files**: list.
6. Select **Store...** from the **Options** menu,

OR

At the keyboard type  , 

OR

Click .

Note: You will have to wait a few minutes while the store process completes.

UDE Database Application

To invoke the UDE Database application from the **Launchpad**, click on the **D**atabase menu and then click on **U**DE Database (see Figure 2-10, Database cascade menu).

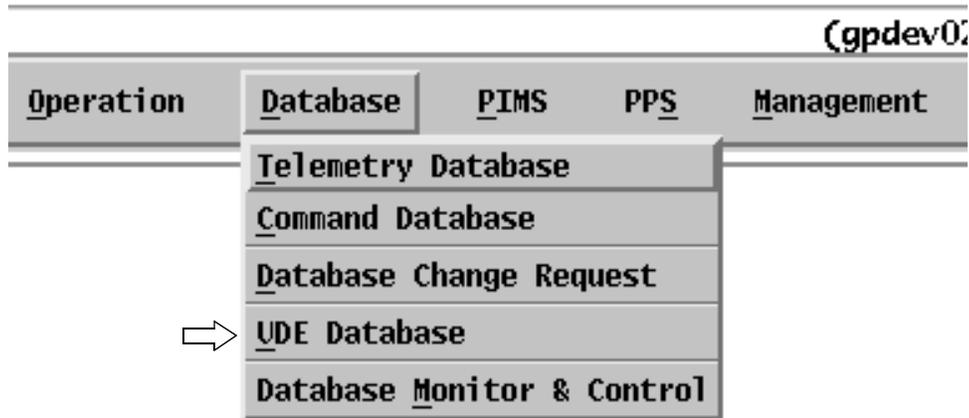


Figure 2-10, Database cascade menu

The main window of the UDE Database application (see Figure 2-11, UDE Database main window) contains menu items (active when a database has been selected) that allow the user to manipulate and maintain UDEs.

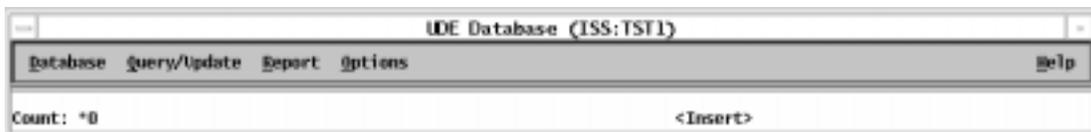


Figure 2-11, UDE Database main window

Query/Update Menu

The menu items under the **Q**uery/Update menu include **E**dit Personal UDE Attributes... and **V**iew All Users UDE Attributes... (see Figure 2-12, Query/Update menu).

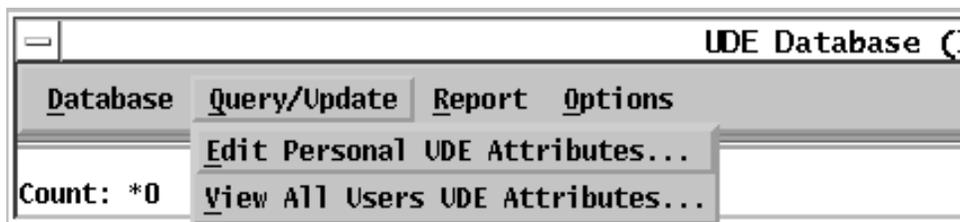


Figure 2-12, Query/Update menu

Edit Personal UDE Attributes...

This menu item invokes a form that allows you to edit the shareable attribute of UDEs that you own (when you retrieve a shareable UDE, you become the owner of that copy). Making a UDE shareable allows other users to access your UDEs. You can also execute queries on those UDEs that you own.

The Edit Personal UDE Attributes form (see Figure 2-13, Edit Personal UDE Attributes form) allows you to:

- Enter and execute queries on your personal User-generated Data Elements (UDEs) within the UDE Database
- Mark a UDE shareable
- Mark a UDE for deletion and confirm the deletion
- Save changes made to your UDEs' attributes

Data Element Type	Data Element Name	Shareable Flag	Stored Date	Data Element Size (Bytes)
Display	dispt.dsp	Y	Mar-19-1999	27
Display	disp2.dsp	N	Mar-29-1999	27
Display	disp3.dsp	N	Mar-29-1999	45
Display	disp4.dsp	N	Mar-29-1999	45

Figure 2-13, Edit Personal UDE Attributes form

Work area fields are empty when the **Edit Personal UDE Attributes** form is initially invoked. You can enter selection criteria or use the default (provides a listing of all of your UDEs for the currently selected database) and execute a query. Once the query has been executed, the work area will contain the results of your query.

If you click **Shareable Flag**, an LOV is invoked that allows you to choose **(Y)es** or **(N)o** as selection criteria. The **Shareable Flag** column identifies **(Y)** or **(N)** whether a UDE is shareable or not. After the query is executed, this is the only field that is modifiable.

(Y)es - Makes the selected UDE shareable so that other users can copy it.

(N)o - Other users cannot access it.

The **D**elete **R**ecord option in the **R**ecord menu or  allows you to mark a record (UDE) for deletion. The UDE is not removed from central storage until you click  **S**ave from the **A**ction menu.

You will be asked to confirm the deletion (see Figure 2-14, Delete Record (Confirmation Required) dialog box).

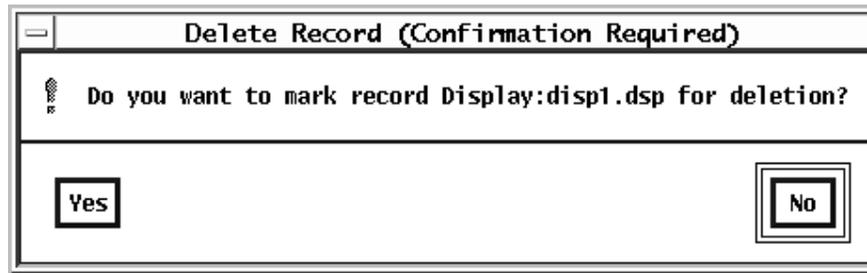
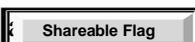


Figure 2-14, Delete Record (Confirmation Required) dialog box

The **S**ave option in the **A**ction menu or  updates the UDE Database with changes you have made.

Note: This option is only available if a change is detected. When save is selected, any changes you have made will be saved.

Make a UDE Shareable:

1. From the **UDE Database** window, select the **E**dit **P**ersonal **U**DE **A**tttributes... menu item from the **Q**uery/**U**ppdate menu to invoke the **E**dit **P**ersonal **U**DE **A**tttributes window.
2. Highlight the UDE that you want to be made shareable or click in the **S**hareable **F**lag column of the desired UDE.
3. Click  **Y**es. The UDE is now marked shareable.
4. Select **S**ave from the **A**ction menu, if you want this change to be updated in the UDE Database.

View All Users UDE Attributes...

This menu item invokes the **View All Users UDE Attributes - Query Only** form (see Figure 2-15, View All Users UDE Attributes - Query Only form). This form allows you to enter and execute queries on other users UDEs within the UDE Database.

User ID	Data Element Type	Data Element Name	Shareable Flag	Stored Date	Data Element Size (Bytes)
DTEDALLO	Computation Description	000_lpr_sd2239.cdd	N	Dec-30-1998	4096
DTEDALLO	Computation Executable	Test_LP.cmp	N	Mar-12-1999	4096
DTEDALLO	Computation MSID Input	1001_A_bk.cni	N	May-04-1999	89
DTEDALLO	Computation MSID Input	CBDATA.cni	N	May-04-1999	89
DTEDALLO	Computation MSID Input	Test_LP.cni	N	May-04-1999	89
DTEDALLO	Computation MSID Input	jtcnp.cni	N	May-04-1999	89
DTEDALLO	Computation MSID Input	jtcnp40_spr.cni	N	Mar-15-1999	4096
DTEDALLO	Computation MSID Output	000_lpr_sd2239.cno	N	May-04-1999	98
DTEDALLO	Computation MSID Output	1001_A_bk.cno	N	May-04-1999	98
DTEDALLO	Computation MSID Output	Test_LP.cno	N	May-04-1999	98

Figure 2-15, View All Users UDE Attributes - Query Only form

To begin with, this is a **Query Only** form which means that it does not allow inserts, updates, or deletes to be made and users are not able to create records. In appearance, the form is very similar to the **Edit Personal UDE Attributes** form with the only difference being the addition of the **User ID** column. The **User ID** column will contain the name of the user who owns each UDE.

Copy UDEs...

Copy UDEs... is the only menu item available under the **Options** menu. It's selection invokes a dialog box that allows you to select UDEs from a different mission and copy them to your current mission database. The **Copy UDEs** dialog box allows you to select a **Source Mission:**, a source **User ID:**, and source **Data Element Type:** from which you can copy UDEs to the mission database you selected when you first entered the application (see Figure 2-16, Copy UDEs dialog box).

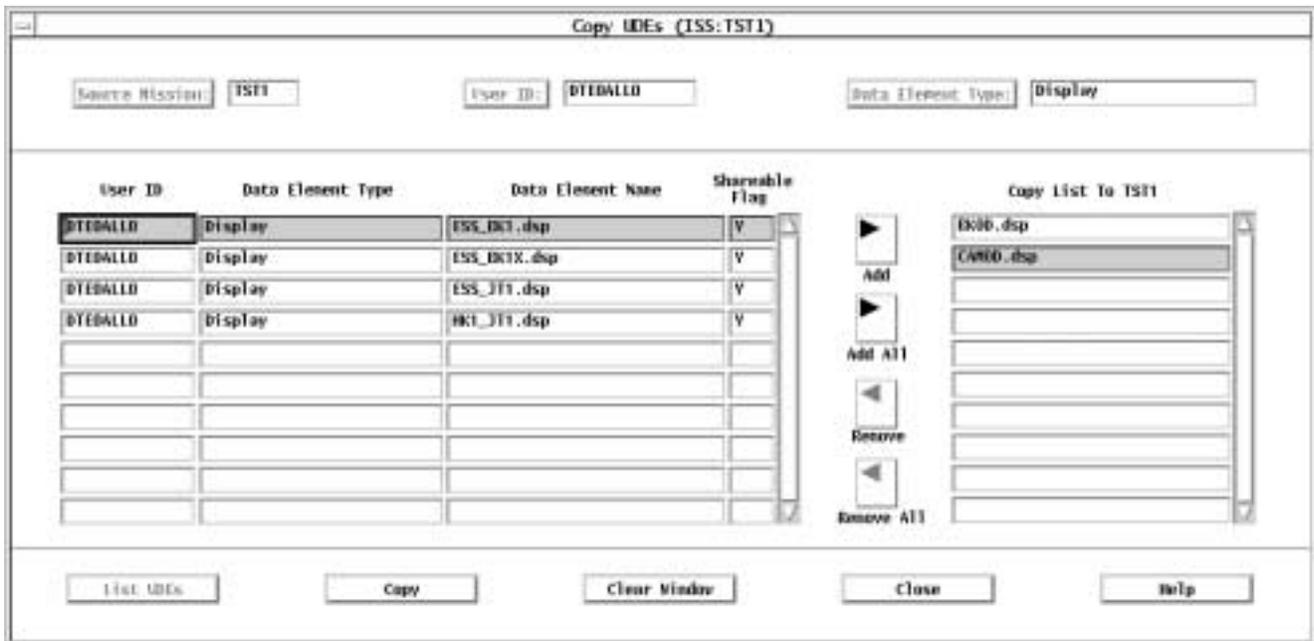


Figure 2-16, Copy UDEs dialog box

Selection criteria is specified in the text entry fields for **Source Mission:**, **User ID:**, and **Data Element Type:** or by making selections from the LOVs. The **Source Mission:**, **User ID:**, and **Data Element Type:** all invoke LOVs from which the appropriate information can be selected. If you know the selection criteria you can enter it in the provided input text fields. Clicking **List UDEs** will provide a list of UDEs, based on your search criteria, on the left side of the dialog box. The **Copy List To mission** column will contain a list of the UDEs you have selected to be copied to your current mission database. UDEs are moved between the lists using the arrow pushbuttons located between the lists. Clicking **Copy**, copies the UDEs specified in the **Copy List To mission** area from the source mission database to the current mission database.

Copy Personal UDEs:

1. Click on the **Options** menu and then select the **Copy UDEs...** menu item.
2. Select the source mission database by typing the mission name in the input text field to the right of **Source Mission:**, or click **Source Mission:** LOV.

Tip: You can execute the query at this point and have all of the shareable UDEs in the source mission database returned to the query results area.

3. Select the user ID by typing the ID in the input text field to the right of , or click to invoke an LOV.

Tip: You can execute the query at this point and have all data element types for the selected user ID returned to the query results area.

4. Select the data element type by entering it in the input text field to the right of , or click to invoke an LOV.
5. Click to execute the query based on the source criteria.
6. Once the query is completed, select the UDE(s) that you want to copy to the current mission by clicking on any field of the desired record.
7. Click (Add) to add the selected UDE to the **Copy List To mission** area.
8. Once all of the desired UDEs are listed in the **Copy List To mission** area, click .
9. Click when the desired UDEs have been copied.

Summary

The UDE Database application provides for centralized storage of data elements developed by other EHS applications. A User-Generated Data Element (UDE) can be a display, computation, script, change file, etc., that is developed for use during operations.

UDEs are stored and retrieved for either generation or operation applications via the **Launchpad - File** menu. Applications available from the **Query/Update** menu of the **UDE Database** main window allow you to edit and view UDE attributes. One attribute of a UDE is its shareable flag, if it is set to yes, the UDE is made shareable so that other users may retrieve it for their own use. UDEs from one mission database may be copied to other mission databases.

With the conclusion of this module you should be able to:

- define the terms and concepts of the UDE Database application
- demonstrate how to store and retrieve UDEs to the UDE Database
- demonstrate how to make a UDE shareable
- demonstrate how to copy a UDE from one mission database to another

Answers

1. The steps necessary to make a UDE shareable are as follows:

From the **UDE Database** window, select the **Edit Personal UDE Attributes...** menu item from the **Query/Update** menu to invoke the **Edit Personal UDE Attributes** window.

Highlight the UDE that you want to be made shareable or click in the **Shareable Flag** column of the desired UDE.

Click  **Yes**. The UDE is now marked shareable.

Select **Save** from the **Action** menu, if you want this change to be updated in the UDE Database.

2. The steps necessary to Store a UDE are as follows:

Select **Store UDE** from the **File** menu on the **Launchpad**.

Select the **Store Selected** radio button within the **Store Mode** frame.

Select the UDE type from the **UDE Types:** list.

Click on file(s) that you want stored.

Click .

Confirm the store by clicking  on the **Store UDE (Confirmation Required)** dialog box.

3. True

4. The steps necessary to Copy a UDE are:

Click on the **Options** menu and then select the **Copy UDEs...** menu item.

Select the source mission database by typing the mission name in the input text field to the right of , or click  LOV.

Select the user ID by typing the ID in the input text field to the right of , or click  LOV.

Select the data element type by entering it in the input text field to the right of , or click  LOV.

Click  to execute the query based on the source criteria.

Once the query is completed, select the UDE(s) that you want to copy to the current mission by clicking on any field of the desired record.

Click  (**Add**) to add the selected UDE to the **Copy List To mission** area.

Once all of the desired UDEs are listed in the **Copy List To mission** area, click



Click  the desired UDEs have been copied.

Module 3

Telemetry Database

Objectives

The Telemetry Database application provides you with the capability to access the telemetry database (TDB) which contains the telemetry definitions needed to drive Huntsville Operations Support Center (HOSC) telemetry processing. In this module you will learn how to:

- modify limits of an MSID
- modify calibration of an MSID
- modify state codes of an MSID
- add and modify an expected state of an MSID
- apply a Local Table Change file to your LocalTable

Telemetry Basics

For a ground system to process telemetry, the structure of the telemetry data must be defined. Each telemetry stream or packet received by the EHS is defined in the telemetry database. The parameters in a stream are assigned a measurement/stimulus identifier (MSID). An MSID is a unique key defined for each parameter in the streams to be processed for a specific project and mission. An MSID is restricted to twenty characters, but can contain any project defined identifier that is unique. MSIDs are not required to follow the MSID naming convention used by the Shuttle project. Any project mnemonic may be an MSID. The telemetry database defines for each MSID; the MSID's location in the stream, data to calibrate the parameter to engineering units, and data to determine if the parameter is within limits or in its expected state. The types of calibration that can be defined for a parameter are polynomial equation, interpolation between point pairs (line segment), or conversion to a state code (for example, ON/OFF). If a parameter has a calibration type of polynomial or point pairs, then caution and warning limit ranges can be defined. If a parameter is calibrated by state codes, then an expected state can be defined. Limit ranges and expected states are used to notify the user when parameters are out of their nominal range or state. Telemetry processing in the EHS is database dependent. A database dependent system allows telemetry processing software to be designed which is independent of telemetry stream definition within the bounds of compliance with MSFC-STD-1274, HOSC Telemetry Format Standard. The information required from the database to perform telemetry processing is stored in tables on each workstation and server called the telemetry local tables.

Telemetry Database and Local Table Operations

The telemetry local table is a copy of the telemetry database stored on a workstation or server (see Figure 3-1, Local Table Concept). It contains information necessary to perform decommutation, conversion, calibration, and limit sensing of MSIDs. There is a unique set of local tables for each user ID and database version. A user may use the Telemetry Database (TDB) and Local Table Operation (LTO) applications to change the information in his/her local table. TDB allows a user to create/modify local table change files and LTO allows the user to apply these change files to the runtime local table. The TDB application displays to the user a copy of a specified local table. The user can then edit processing information for an MSID such as the calibration data and limit/expected state data. Only the information for MSIDs that have been edited are saved to a file, called a local table change file. The Local Table Change file can be marked sharable and then saved to the UDE Database. The capability to edit various characteristics of the local table is configurable per user and is based on the projects requirement to do so. The capability to edit decommutation information is reserved for database developers.

It should be noted that Change Files applied via the Local Table Operations application affect only the copy of the local table that is resident on your work station. For the Change File to be applied on a number of workstations it would have to be marked as sharable, stored to the UDE Database, retrieved to the additional workstations, and then applied to the Local Tables on those workstations using the Local Table Operations application.

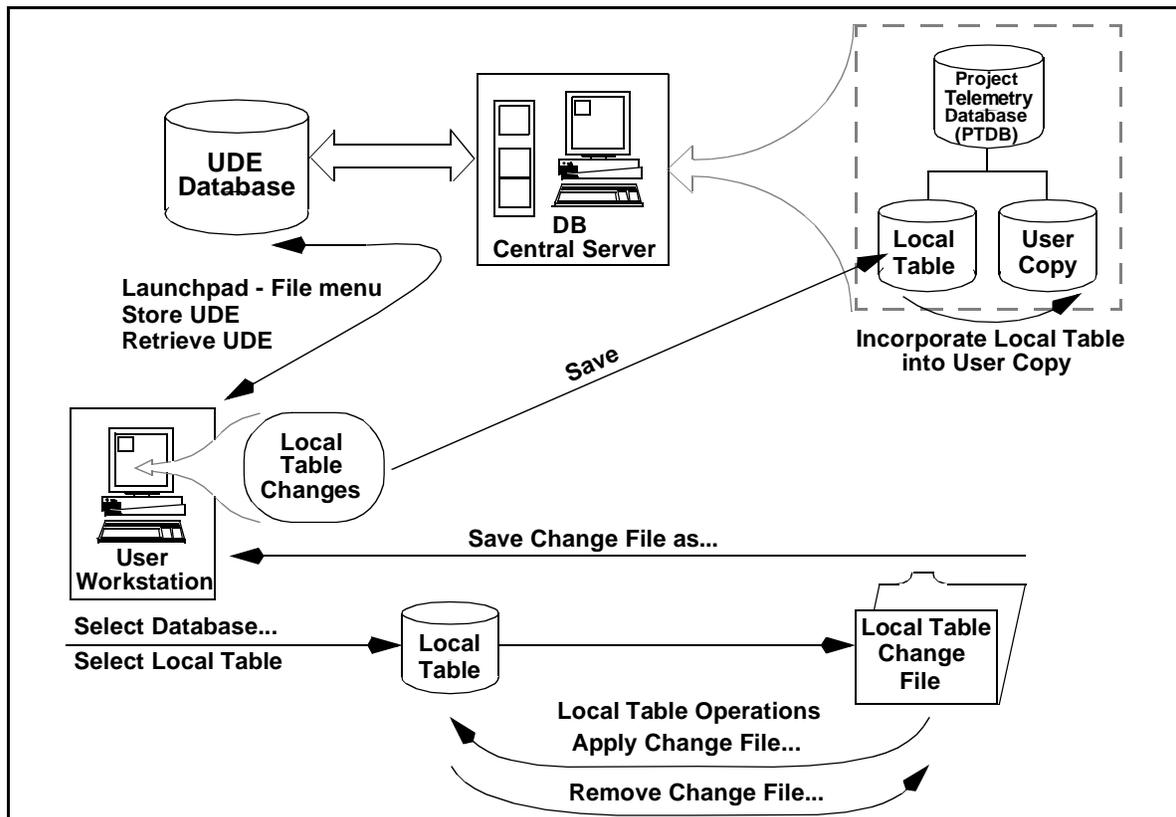


Figure 3-1, Local Table Concept

Exercises

Instructions

The following “Try Its” allow you to make changes to the copy of the Local Table resident on your workstation. Carefully read and complete each step. Refer to the Project-Specific Worksheet for MSIDs.

Scenario:

Certain attributes of MSIDs in the Telemetry Database require changing. You are required to make the changes, save them to a change file, and finally, apply the change file to your local table to verify your changes.

Try It...

Modifying Limits of an MSID that has Limits Defined

1. Invoke the **Telemetry Database** application.
2. From the **Database** menu, select the **Select Database** menu item. Click the **Select Local Table** pushbutton. The first time this option is selected, you will receive an information dialog box that informs you that the local table is going to be populated. To save time, we have built local tables for each user ID.
3. From the **MSID Information** cascade menu, under the **Query/Update** menu, select **Limit Sensing List - Limits**.
4. Click **Execute Query**,

OR

if you want to specify some filter criteria, type your search criteria in the first line prior to clicking the **Execute Query** button.

5. Find and select the MSID from the Project-Specific Worksheet. Change the Caution and Warning limits as specified in the Project-Specific Worksheet. Click **Save**. Click **Close**.

Note: Clicking **Save**, saves your Local Table changes to the PTDB Local Table located on the DB Central Server.

6. Now we are going to save the modified information to a change file that is used to alter the local table on the workstation. Click on the **Save Change File As** option under the **Options** menu. Supply a name and click **Save**.

7. This change file is saved on the workstation as a LocalTable Change File UDE. If you plan on needing it in the future, store it to the UDE database.

Note: The steps needed to apply this change file will be covered later in this exercise.

Try It...

Modifying Limits of an MSID that has No Limits Defined

1. From the **MSID Information** cascade menu, under the **Query/Update** menu, select **MSID Detail**.
2. If you want to specify some filter criteria, type your search criteria in the MSID field prior to clicking the **Execute Query** button. Find and select the MSID from the Project-Specific Worksheet.
3. In order to define limits for an MSID that currently does not have limits, you must first define the **Limit Default Set Number:**. Specify 1 in this field and click **Save**.
4. Click **Enter Query**. Type the specified MSID in the MSID field and click **Execute Query**.
5. Now the **Limit Sensing Detail** button should be active. Click it.
6. Specify the following information with values obtained from the Project-Specific Worksheet:
 - MSID
 - Set Number: (This identifies the current limit set. In EHS, you can have multiple sets of limits)
 - Caution Low:
 - Caution High:
 - Tolerance:
 - Exception Monitor:
 - Warning Low:
 - Warning High:
7. Click **Save**. Click **Close** to close the **Limit Sensing Detail** form. Click **Close** to close the **MSID Detail** form.

Note: Clicking **Save**, saves your Local Table changes to the PTDB Local Table located on the DB Central Server.

8. Now we are going to save the modified information to a change file that is used to alter the local table on the workstation. Click on the **Save**

Change File As option under the **Options** menu. Supply the same name as the first Try It and click **Save**.

9. This change file is saved on the workstation as a Local Table Change File UDE. If you plan on needing it in the future, store it to the UDE database.

Note: The steps needed to apply this change file will be covered later in this exercise.

Additional Exercises

1. Complete steps 1 through 9 above (including limits information) for the MSIDs as listed in the Project Specific Worksheet.

Try It...

Applying the Change File

1. From the **Launchpad, Operation** menu item, select the **Local Table Operation** application.
2. Click on the **Apply Change File** button.
3. Select the change file.
4. Click **Apply**.
5. Open the **ISSTest** display within Display Operation.

Summary

The Telemetry Database contains the telemetry definitions needed to drive Huntsville Operations Support Center telemetry processing. A copy of this database is stored on your workstation as a telemetry local table, which you can modify, saving the changes to a Local Table Change File.

Using the Local Table Operations menu you may apply this change file to the runtime local table on your workstation to see how the database changes you have made react with telemetry data present. A change file may be marked as shareable and stored to the UDE Database, where it can be retrieved by other users and then be applied to their workstations.

With the conclusion of this module you should:

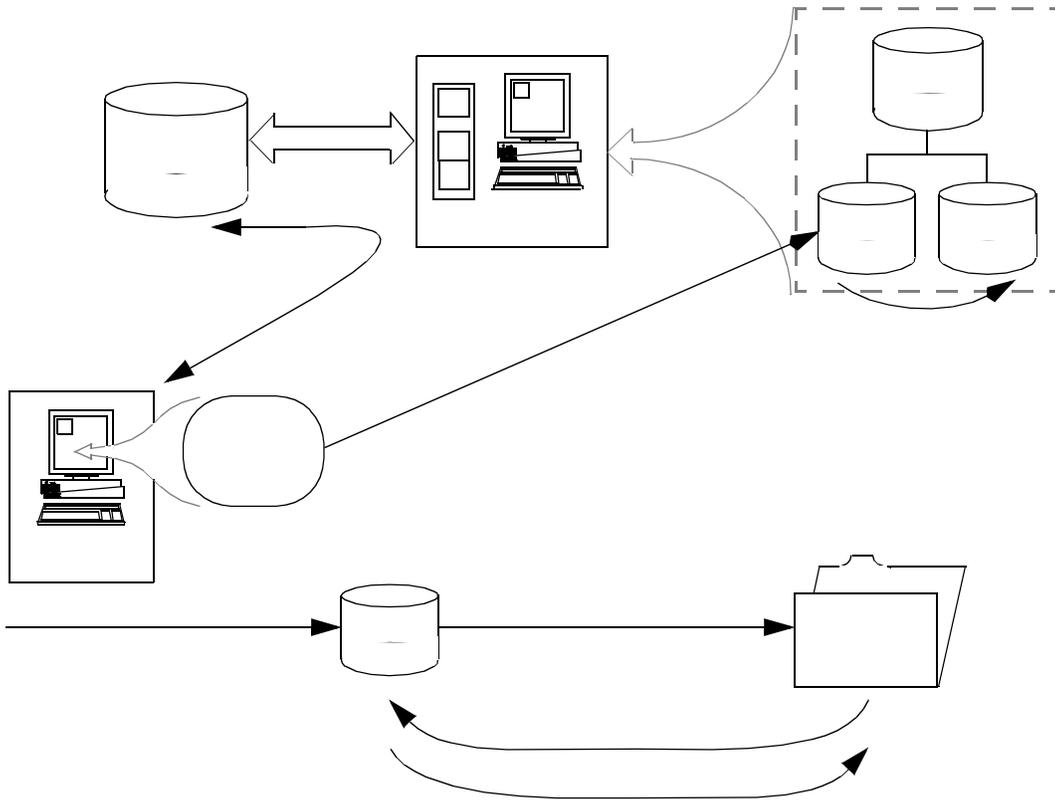
- demonstrate how to modify limits of an MSID
- demonstrate how to modify calibration of an MSID
- demonstrate how to modify state codes of an MSID
- demonstrate how to apply a Local Table Change file to your LocalTable

Questions

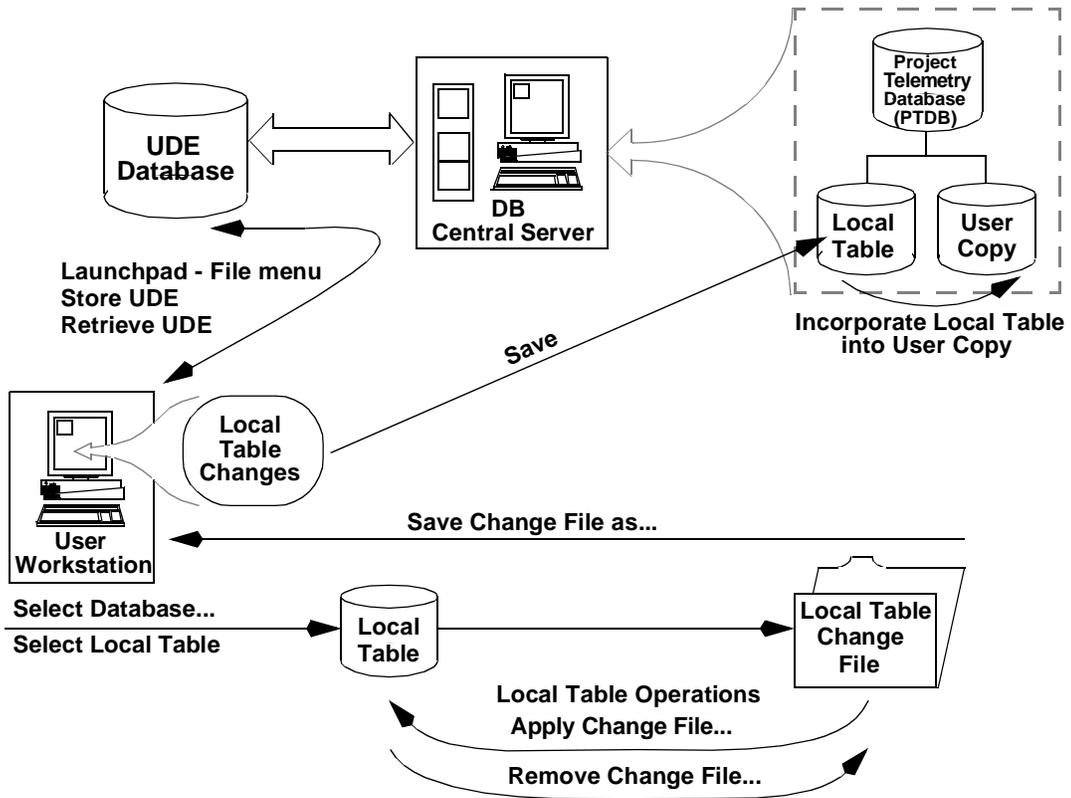
Instructions

Indicate the answers for each question below. The correct answers are given immediately following in the **Answers** section.

1. Label the blocks in the Local Table Concept diagram using the numbers listed for the following phrases:
 1. User Workstation
 2. Project Telemetry Database
 3. UDE Database
 4. DB Central server
 5. Local Table (x2)
 6. Local Table Change File
 7. User Copy
 8. Store UDE/Retrieve UDE
 9. Select Local Table
 10. Save Change File As...
 11. Apply Change File...
 12. Remove Change File...
 13. Local Table Changes
 14. Incorporate Local Table into User Copy
 15. Save



Answers



Module 4

Command Database

Objectives

The Command Database application provides the capability to access and manipulate data in an EHS project command database (PCDB) which contains the command definitions needed to drive HOSC command processing. The Command Database application provides you with the means to create a user copy of commands you are authorized to access so that you can modify their contents, perform certain validation checks, and submit proposed database changes to the Database Coordination Group (DBC). Proposed changes must be submitted to the DBCG before they can be incorporated into a subsequent database version. In this module you will learn:

- the terms and definitions associated with the Command Database
- form usage in the Command Database
- Command Database concepts

Definition of Terms

Though commands are a form of telemetry, they are very specialized in nature, having a unique set of terms, concepts, and nomenclature that should be learned prior to working with the Command Database application forms.

Command

A complete, defined sequence of data (including the command header and command body) which communicates information from a source, either ground or flight-system based, to a payload or spacecraft destination.

Command Body

The portion of the command defined in the Command Database which is used as the transmission vehicle for conveying command data field definitions from the command's source to the command's end-item destination.

Master Command

A command used as a shell or template to be copied, with the exception of overridden fields, to create other commands.

Command Data Field

An element within a command body containing a single data value. Databased attributes associated with a command data field include its field mnemonic, its type (modifiable verses predefined), its initial value, its input data type, its uplink data type, and the fields length.

Command Header

The portion of the command defined in the Command Database (CDB) which contains the information vital to proper routing of the command body to the appropriate destination.

Command Header Data Field

An element within a command header containing a single data value. Databased attributes associated with a command header data field include its field mnemonic, its type (modifiable verses predefined), its initial value, its input data type, its uplink data type, and the fields length.

Command Mnemonic

A reference used to issue a command that uniquely identifies a command within the command database.

Command Response

Information indicating the status of a command as it progresses to its destination.

Input Data Type

The data format in which a user supplies data to command data fields within the Command Database. Prior to uplink, data in each command data field is converted from its input data type to its corresponding uplink data type.

Modifiable Command

A command that is defined in the Command Database containing at least one modifiable command data field that may be updated by a user at any time prior to or during a mission.

Predefined Command

A command that is completely defined in the Command Database prior to a mission. Predefined commands contain no modifiable data fields.

Uplink Data Type

The data format in which the data for each command data field is transmitted to the command's destination. Prior to uplink, data in each command data field is converted from its input data type to its corresponding uplink data type.

Command Class

Commands may be assigned to command classes through the Command Database. Valid command classes for ISS are as follows:

- DLC-USOS Data Load Commands for United States On-Orbit Segment (USOS) destinations
- PC Payload Commands
- SC System Commands

Command Structure

The HOSC command format concept recommends that each command consist of one database-defined command header followed by a separately defined command body. Both the header and the body are comprised of one or more command data fields. All command data fields are individually defined in the delivered Command Database in terms of field length, field input and uplink data types, the field's starting location in a command word, and whether the field data content is predefined or modifiable. The basic command structure is depicted in Figure 4-1, Command Format Structure.

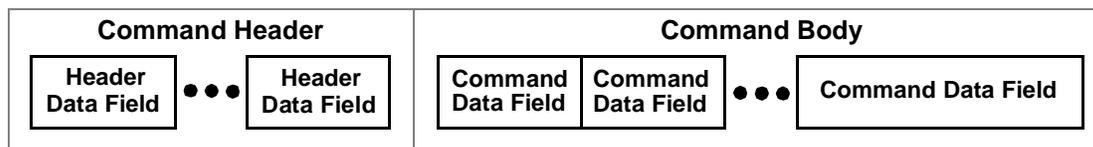


Figure 4-1, Command Format Structure

Command Header

A header consists of one or more header data fields and is referenced in the Command Database by a unique header mnemonic. Each header data field within the header is identified by a field mnemonic, has an individually defined fixed length, and is defined as either predefined or modifiable. Only fixed length command header data fields are allowed.

A header provides onboard routing and destination information for the command once it has been received by the intended spacecraft. A single header definition in the Command Database can have many command body definitions associated with it.

Command Body

A command body is composed of one or more command data fields and is referenced in the Command Database by a unique command mnemonic. Like the header data field, a command data field is identified by a field mnemonic and is defined as either predefined or modifiable. If a command is of fixed length, then each field within the command will have an individually defined fixed length. If the command is of variable length, in addition to any fixed length fields, only one field within the command's definition, the last field, will have an undefined, variable length at the time of command definition. The field's data contents are provided through user input prior to the command's uplink. The field's length is derived from the length of the user input data. Any individual command body definition has only one header definition designated for use during construction of the command's uplinkable bit pattern.

Command Types

The Command Database defines command header structures and command body structures. Each of these structures can be classified as either predefined or modifiable. Modifiable commands may be further sub-classified as fixed length or variable length modifiable commands.

Predefined Fixed Length Command Headers

Predefined command headers are those header formats in which all data used to define the header is delivered with the Command Database and cannot be modified in real-time by the user. The definition of a predefined command header includes, at a minimum, a header mnemonic, one or more predefined fields from which the header is constructed, the predefined data for each field, and the length of each field.

The header length is defined as the summation of the lengths of all header fields and is a fixed, or constant, value. Because the user cannot change the length of any of the fields in a predefined, fixed length header, the total length of the header is also fixed.

Modifiable Fixed Length Command Headers

Modifiable command headers are those header formats in which one or more fields within the header are modifiable in real-time by the user. The definition of a modifiable command header includes, at a minimum, a header mnemonic, one or more modifiable fields, and the length of each field.

The header length is defined as the summation of the lengths of all header fields, both predefined and modifiable. Because the user cannot change the length of any of the fields in a modifiable, fixed length header, the total length of the header is also fixed. Modifiable, variable length headers are not allowed.

In addition to modifiable fields, modifiable command headers may also include one or more predefined fields. Initial values for modifiable fields in the header may be delivered with the Command Database. Users may update modifiable fields with data that is in the input data type defined for that field. It is

important to note that a modifiable field indicates that the user may change the value of the data within the field, but not the length of the field.

Predefined Fixed Length Commands

Predefined commands are those command formats in which all data used to define the command body is delivered with the Command Database and cannot be modified by the user. The definition of a predefined command includes, at a minimum, a command mnemonic, one or more predefined fields from which the command is constructed, the predefined data for each field, and the length of each field. Predefined commands may only be of fixed length type.

The command body length describes the total length of the command body as defined as the summation of the lengths of all fields within the command body. The length of a predefined command is a fixed, or constant, value. Because the length of none of the fields in a predefined command can be changed by the user, the length of a predefined, fixed length command is also fixed.

Modifiable Fixed and Variable Length Commands

Modifiable commands are those command formats in which one or more fields within the command body are modifiable by the user. The definition of a modifiable command includes, at a minimum, a command mnemonic, one or more modifiable fields, and the length of each field. In addition to modifiable fields, modifiable commands may also include one or more predefined fields. Initial values for modifiable fields in the command body may be delivered with the Command Database. Users may update modifiable fields with data that is in the input type defined for that field. Modifiable commands may be either of fixed length type, in which the user cannot change the command's total length, or of variable length type, in which the user can dynamically set the command's total length through the modification of the command's variable length field.

Modifiable Fixed Length Commands

The length of most modifiable commands is a fixed, or constant, value. This length describes the total length of the command body as defined by the concatenation of all fields within the command body, both predefined and modifiable. Because the user cannot change the length of any of the fields in a modifiable, fixed length command, the total length of the command body is also fixed.

Modifiable Variable Length Commands

This special type of modifiable command contains a single, modifiable field which has the unique distinction of having a variable length. The variable length field must be the last field in the command body definition. Modifiable, variable length commands may contain only one variable length field. The length of a variable length field may be between one and four bytes long. The total length of a variable length command is not defined in the database until after the variable length field data is sized and defined by the user. Users may update a variable length command with data, provided that the data is supplied to the database in the hexadecimal input data type. The final length of the command will be

determined using the summation of the length of the uplink data provided by the user in the variable length field and the length of all other fields comprising the command's uplink pattern.

Command Response

For each command uplinked, several types of command responses may be generated by the EHS Command Subsystem to indicate the status of the transmitted command. Command responses convey information about the command, such as its progress through the communication paths, integrity once onboard the flight system, deliverability to the end system or payload, status after execution, or other information.

Command Acceptance Response

A Command Acceptance Response (CAR) is generated by an intermediate transmission facility, for example, the Johnson Space Center (JSC) Control Center Complex (CCC), and indicates the reception of a command, block of commands, or file by that facility. Besides the command definition itself and a CAR time-out value, there is no additional information needed from the project's Command Database for the EHS Command Subsystem to properly process and report CAR data.

Flight System Verifier

A Flight System Verifier (FSV) is generated by a system onboard the spacecraft and will indicate the receipt of a command, block of commands, or file by the onboard system. The receipt of an FSV by the EHS Command Subsystem does not guarantee a command's execution, but only indicates that the command was received onboard. FSVs are defined on a project-wide basis. For each FSV used by a project, a telemetry MSID must be supplied in the delivery of the project's Command Database. The supplied telemetry MSID must be completely defined in the TDB. An FSV time-out value, applicable to all project FSVs, must also be supplied. Each project must also levy requirements that specify how the data received in the identified MSIDs is to be interpreted to determine the "pass" or "fail" status of the FSV associated with the command transmitted.

Command Reaction Response

A Command Reaction Response (CRR) is any telemetry data or group of telemetry data that has been received by the EHS Command Subsystem in response to the execution of a previously uplinked command. CRRs are defined on a per command basis. Each defined command in the Command Database may have one, many, or no CRRs defined. For each individual command requiring a CRR-type confirmation, one or more telemetry MSIDs identifying the telemetry data containing the command's confirmation information must be defined and resident in the project's Command Database. For each of these telemetry MSIDs, an expected value or range of values must be specified. These values are used to establish the "pass" or "fail" status of the associated CRR. Each supplied telemetry MSID must be completely defined in the TDB. A CRR time-out must also be provided on a per-command basis to determine when the EHS Command Subsystem should end the CRR assessment.

Command Database Forms

For the most part, the forms invoked through the menus of the Command Database application, behave and operate just as previously described for the Telemetry Database. There are, however, a couple of distinctions worth mentioning (see Figure 4-2, Command Database forms). Generally, any pushbutton on the Control Panel of a form that allows you to navigate to another form is also presented as a menu option in the **Options** menu of the form.

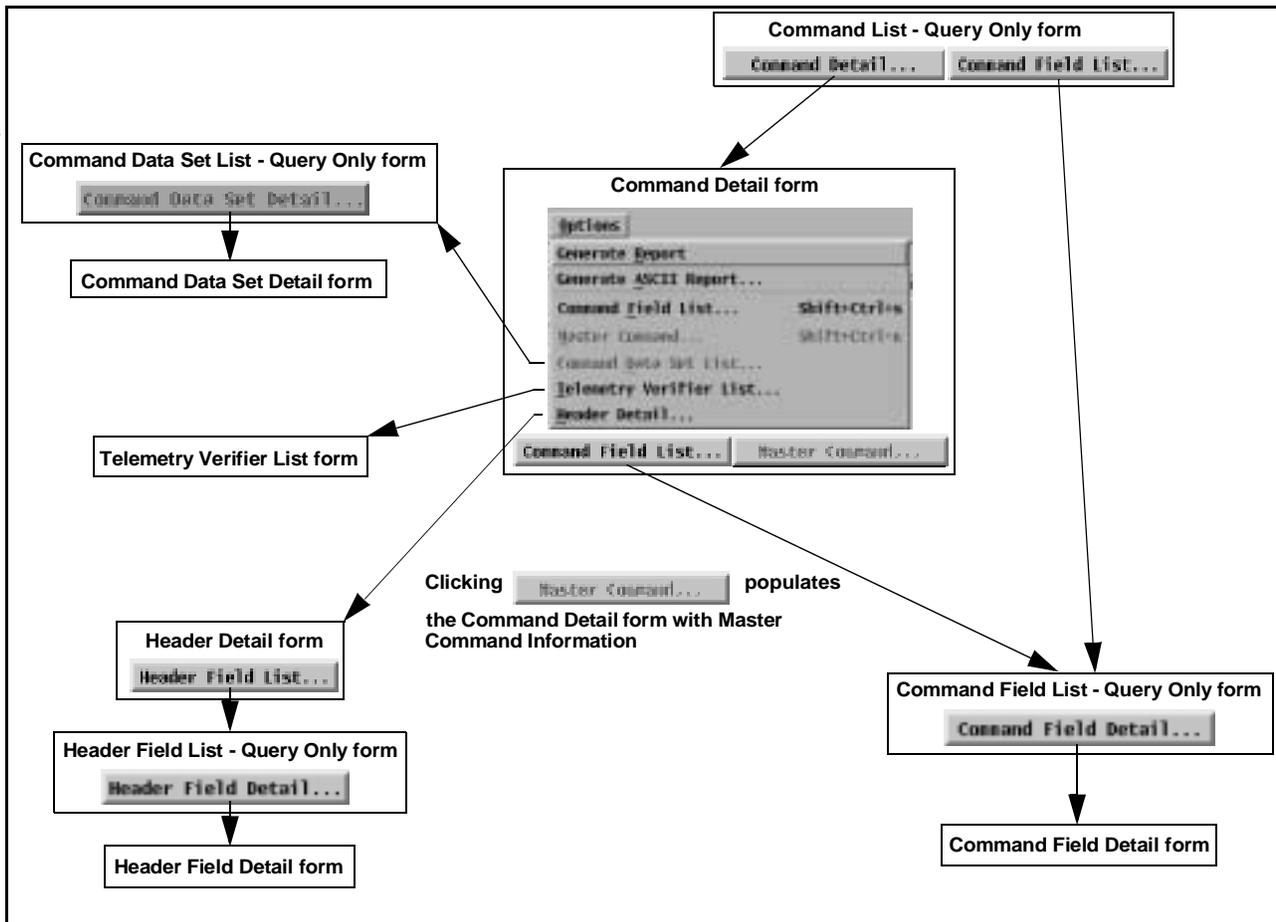


Figure 4-2, Command Database forms

An exception is noted in the **Command Detail** form where **Command Field List...** and **Master Command...** are two options described by pushbuttons while the **Options** menu also provides options for **Command Data Set List...**, **Telemetry Verifier List...**, and **Header Detail...** forms.

Another distinction worthy of note also exists on the **Command Detail** form involving the **Master Command...** option. Selecting the **Master Command...** menu item (or clicking **Master Command...**) populates the **Command Detail** form with information pertinent to the master command for the currently displayed command. Instead of navigating to another form, this option brings up a copy of the **Command Detail...** form populated with master command information.

Command Database Concepts

As mentioned earlier, the Command Database application is used to modify configuration controlled items contained in the PCDB. Most of the data in the PCDB is configuration controlled. That is, it cannot be changed directly without the concurrence and approval of the Database Coordination Group (DBCG) before being incorporated into a subsequent database revision.

Non-configuration controlled information in the PCDB includes command data set definitions and modifiable command chain definitions. If you are authorized to access these items, you can change them directly from the Command Database application during non-operational support periods without using the DBCR process.

Users may find it helpful to enter their changes into a database and then submit those changes for review and incorporation into the controlled database. This method allows certain validation checks and constraints to be enforced by the user interface and the database before the user even submits the changes. Command Database (and Telemetry Database) provide user copy data for this purpose (see Figure 4-3, Command Database Concept). The user copy data for a specific user consists of copies of all the data records which the user is authorized to modify in the associated database, linked to the user by his EHS account. The user may modify, delete, and add records to his user copy data. User copy data is available to data owners, project database developers, and MOL database developers only.

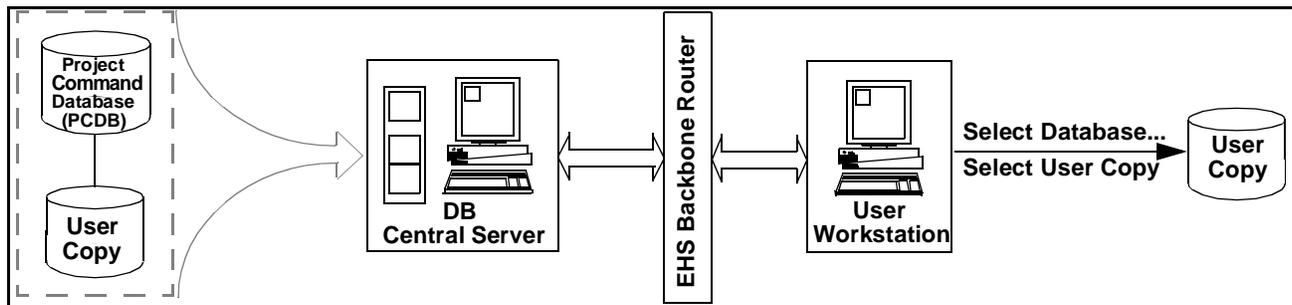


Figure 4-3, Command Database Concept

Input/Uplink Data Types

The **Command Field Detail** form and **Header Field Detail** form (see Figure 4-4, Command Field Detail Form & Header Field Detail Form) provide you with the means to view specific fields within a command or command header. Two of these fields, **Input Data Type:** and **Init Data:**, allow you to determine what needs to be supplied in your command update forms. **Command Update** forms and **Command Header** forms provide you with the mechanism necessary to change data in modifiable fields of commands using the Update Command Mini-application invoked from within the Command Operation application. **Input Data Type:** and **Uplink Data Type:** relate to each other and will be discussed in the remainder of this section.

The image displays two overlapping software windows. The top window, titled 'Command Field Detail (ISS:TST1:0004:User Copy)', contains the following fields: 'Cmd Name' (NODE_REAL_LOAD_CMD), 'Field Name' (ASCE_HEADER), 'Field Type' (F), 'Length' (1), 'Start Word' (3), 'Log Unit' (empty), 'Range Low' (empty), 'Range High' (empty), 'Input Data Type' (E), 'Uplink Data Type' (IBTS), 'Calibration Type' (radio buttons for Polynomial, Phase, State, and No Calibration Data), 'Init Data' (1), 'Var Length' (N), 'Start Bit' (3), 'Calibration Default Set Num' (checkbox), and 'Description' (Modifiable State, ASCE Node Station Node Bit). The bottom window, titled 'Header Field Detail (ISS:TST1:0004)', contains: 'Header ID' (SUBP010_HDR), 'Field Name' (APID), 'Field Type' (radio buttons for Predefined and Modifiable), 'Length' (11), 'Start Word' (1), 'Start Bit' (5), 'Input Data Type' (B), 'Uplink Data Type' (IUNS), 'Cal Type' (checkbox), and 'Description' (Application Process ID (APID)). Both windows have a menu bar with 'Action', 'Edit', 'Field', 'Record', 'Query', and 'Options', and a 'Help' button. The bottom window also has a 'Count: 1' indicator and a status bar with 'Count: 11', 'v', and '<Insert>'.

Figure 4-4, Command Field Detail Form & Header Field Detail Form

The **Input Data Type**: indicates the data representation that will be used when entering data into the **Init Data**: field which contains the current command or header data.

The **Uplink Data Type**: indicates how **Init Data**: will be converted prior to uplink.

This section defines the input and uplink data types that can be assigned to command header and body data fields. The input-to-uplink data conversion mechanism performed by the Command Database simplifies the modifiable data entry process on behalf of the users. When supplying data for a data field, the user provides it in a user-friendly, recognizable, readable format (the input data type). The Command

Database then decalibrates and converts it into a more complex format that is uplinked to the spacecraft (the uplink data type).

When preparing a User Copy (see previous section) and you are supplying data to a data field within the Command Database, the data must always be provided in the data field's input data type. Each field in the Command Database must have an input-to-uplink data type conversion specified. Each field definition may also specify a type of decalibration (either a polynomial coefficient decalibration or a point pair decalibration) to be performed on the data as part of the conversion from input data type to uplink data type. Table 4-1, Input vs. Uplink Data Types, lists the valid input data types and indicates the valid uplink data types to which the input data types can be successfully converted for uplink. The uplink type of ICHK does not have an input data type since this data is not entered by the user, but rather calculated by the HOSC software prior to uplink.

Help: Uplink data types are described in Appendix D, Supported Uplink Data Types.

Table 4-1, Input vs. Uplink Data Types

Input Data Type	Uplink Data Type
A - Alphanumeric String	IDIS - Discrete Integer IUNS - Unsigned Integer SASC - ASCII Characters SEBC - EBCDIC Characters
B - Binary	IDIS - Discrete Integer IPAR - Parity Bit IUNS - Unsigned Integer
D - Decimal (1)	FEEE - IEEE Floating Point FIBM - IBM Floating Point IBCD - Binary Coded Decimal IDSI - Distended Signed Integer IMAG - Signed Integer ITWO - Two's Complement Signed Integer IUNS - Unsigned Integer
G - Scientific Notation (2)	FEEE - IEEE Floating Point FIBM - IBM Floating Point
H - Hexadecimal	IUNS - Unsigned Integer
O - Octal	IUNS - Unsigned Integer

- Notes: (1) Non-exponential input will have a maximum length of 16 characters. Sign and decimal point are optional.
(2) Exponential inputs will have a maximum length of 22 characters. Decimal point is required as is the exponent with its sign. Exponent magnitude may not exceed 75.

If you choose to update a command within the Command Operation application and you do not know the input data type and range, you could invoke the Command Field Detail form to determine the **Input Data Type:** as well as the **Init Data:** (initial value) of the command (see Figure 4-5, Forms Example).

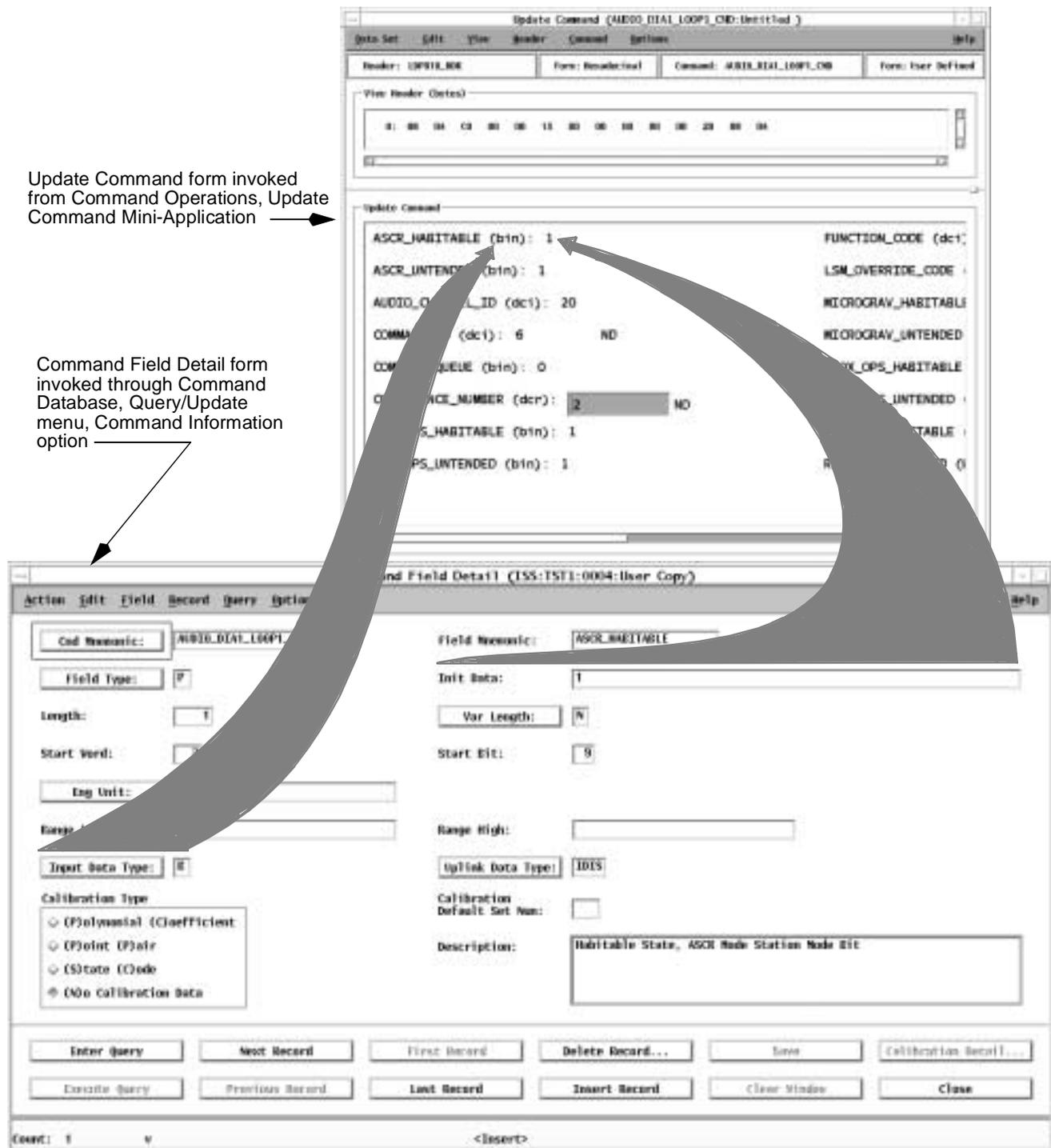


Figure 4-5, Forms Example

Exercises

Instructions

The following “Try It” provides an itemized procedure for researching command mnemonics in the Command Database. Carefully read and complete each step.

Scenario:

You are tasked with updating modifiable commands within the Command Operation application. In order to accomplish this you must first generate command and command header update forms so that they are available on your workstation. Once these forms have been generated, you can edit the modifiable fields for input data type and initial value as defined in the Command database.

Try It...

-
1. From the **Launchpad**, select the **Generation** menu, **Command Update Form Generation** menu option.
 2. From the **Command Update Form Generation** menu bar, select the **Generate Command Form...** menu item from the **File** menu. This invokes the **Generate Command Form** dialog box.
 3. Select a command database to use (choose Baselined).

Note: For purposes of this “Try It” any command from the Mnemonic Selection List will provide the desired result.

4. If you know the command mnemonic, you can enter it in the **Generate:** input text field,

OR,

Supply search criteria in the **Mnemonic Filter:** text field and click



. Your commands that match the filter criteria will be displayed within the mnemonic selection list. If you entered a specific command in the **Mnemonic Filter**, the matching command will be returned to the mnemonic selection list.

5. Select the command that you want to generate an update form for from the mnemonic selection list. The selected command will be displayed in the **Generate:** input text field.

- To further define your form's format, identify whether you want to include all command fields (both predefined and modifiable), only modifiable fields, or if you want to copy a pre-existing master form.

Click on .

Note: If you generate a command update form that only includes modifiable fields, you can later add any or all predefined fields associated with the command using the **Add Predefined Fields** dialog box accessible from the **Edit** menu. See the “Add Predefined Fields...” section later in this module for more information.

- You will be returned to the **Command Update Form Generation** main window where the form will be displayed (see Figure 4-6, Command Update Form Example).

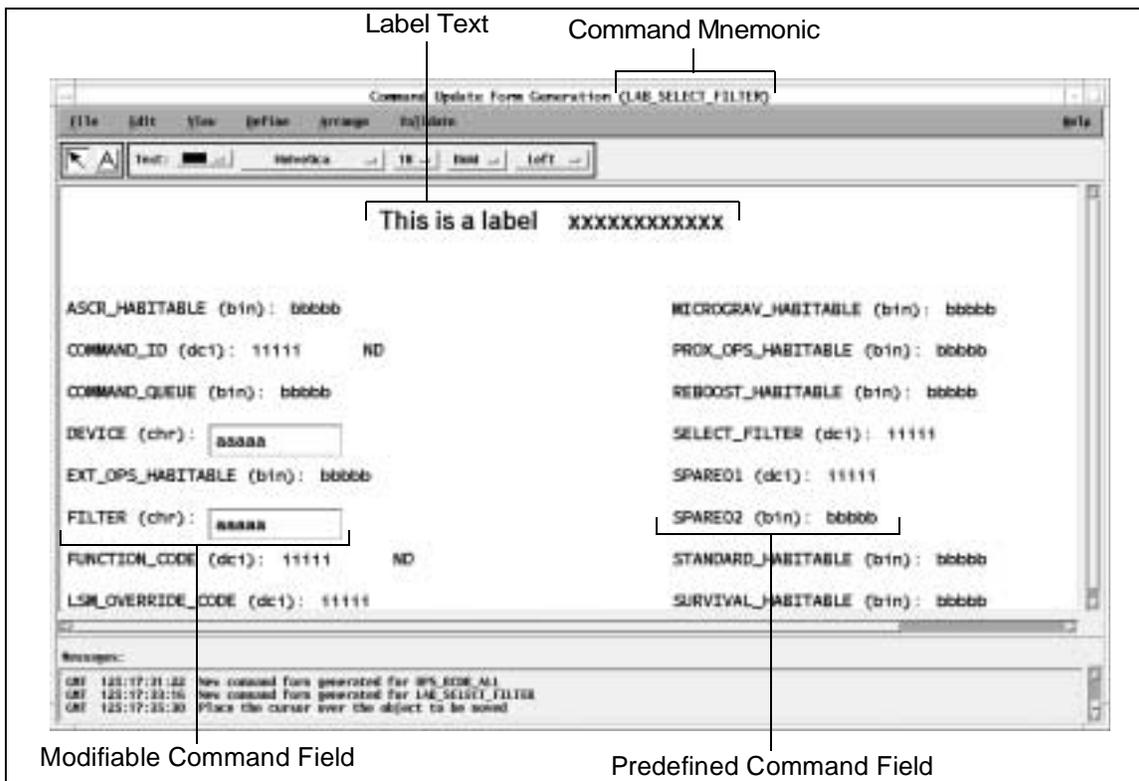


Figure 4-6, Command Update Form Example

- Now that you have generated a Command Form you can verify it's contents against the values contained in the Command Database.
- From the **Launchpad**, Select the **Database** menu, **Command Database** menu option.

10. To access the **Command Field Detail** form, click on **Command Information** under the **Query/Update** menu and then select **Command Field Detail...** The **Command Field Detail** form will be invoked.
11. In the **Cmd Mnemonic:** text entry field enter the command mnemonic for the previously generated form and click . The form will be populated with information peculiar to the specified command mnemonic.
12. Each command will most likely be comprised of several command fields. The **Field Mnemonic:** field can be used to enter a query for a specific command field.
13. Commands may be comprised of both predefined and/or modifiable command fields. Commands that have a modifiable data field must be completed before they can be uplinked. In the **Field Type:** field you can enter either **P** or **M** to display only predefined or modifiable command fields for each command.
14. Locate the **Input Data Type:** entry and compare this with the value displayed in your **Command Update** form.
15. Locate the **Init Data:** entry and compare this with the value displayed in your **Command Update** form.

Note: For steps 14 & 15, any discrepancies between the **Command Field Detail** form and the **Command Update** form are resolved using the Update Command mini-application which is discussed in the End-User Command Applications course (HOSC-2120).

16. When you are satisfied with the contents of your **Command Update** form, select **Save** from the **File** menu.
 17. On the **Command Field Detail** form click .
 18. From the **Command Update Form Generation** menu bar, select the **Generate Header Form...** menu item from the **File** menu. This invokes the **Generate Header Form** dialog box.
 19. Select a command database to use (choose Baselined).
- Note:** For purposes of this “Try It” any header from the Mnemonic Selection List will provide the desired result.
20. If you know the header mnemonic, you can enter it in the **Generate:** input text field,

OR,

Supply search criteria in the **Mnemonic Filter:** text field and click



Note: Steps 21 thru 23 are necessary if you selected the second option in step #20 above. If you opted for the first option in step #20 above, please continue with step #24.

21. Your headers that match the filter criteria will be displayed within the mnemonic selection list.
22. From the mnemonic selection list, select the mnemonic for which you want to generate a header update form. The selected mnemonic will be displayed in the **Generate:** input text field.
23. To further define your form's format, identify whether you want to include all header fields (both predefined and modifiable), or only modifiable fields.
24. Click on . The matching header will be returned to the mnemonic selection list.
25. You will be returned to the **Header Update Form Generation** main window where the form will be displayed.
26. Now that you have generated a Header Form you can verify it's contents against the values contained in the Command Database.
27. From the **Launchpad**, Select the **Database** menu, **Command Database** menu option.
28. To access the **Header Field Detail** form, click on **Header Information** under the **Query/Update** menu and then select **Header Field Detail...** The **Header Field Detail** form will be invoked.
29. In the **Header ID:** text entry field enter the header for the previously generated form and click . The form will be populated with information peculiar to the specified command header.
30. Each header will most likely be comprised of several header fields. The **Field Mnemonic:** field can be used to enter a query for a specific header field.

31. Headers may be comprised of both predefined and/or modifiable header fields. Headers that have a modifiable data field must be completed before they can be uplinked. In the **Field Type:** frame you can select either **P** or **M** to display only predefined or modifiable header fields for each command.
32. Locate the **Input Data Type:** entry and compare this with the value displayed in your **Header Update** form..
33. Locate the **Init Data:** entry and compare this with the value displayed in your **Header Update** form.

Note: For steps 32 & 33, any discrepancies between the **Header Field Detail** form and the **Header Update** form are resolved using the Update Command mini-application which is discussed in the End-User Command Applications course (HOSC-2120).

34. When you are satisfied with the contents of your **Header Update** form, select **S**ave from the **F**ile menu.

35. On the **Header Field Detail** form click  .

Summary

The Command Database contains the command definitions necessary to drive HOSC command processing. A command is comprised of a command header having header data fields and a Command body having command data fields. Headers and commands may be predefined or modifiable, and of fixed or variable length. For each command uplinked, there are several command responses generated which include: Command Acceptance Responses (CAR), Flight System Verifier (FSV), and Command Reaction Responses (CRRs).

Information in the Command Database is manipulated through the forms generated by the Command Database application. As a user authorized to modify data records in the PCDB, you may submit changes via a user copy attached to a DBCR for DBCG approval.

With the conclusion of this module you should be able to:

- define the terms and definitions associated with the Command Database
- demonstrate form usage in the Command Database
- illustrate Command Database concepts

Questions

Instructions

Indicate the answers for each question below. The correct answers are given immediately following in the **Answers** section.

1. What information can be changed in a PCDB without using the DBCR process.
2. What is a CAR? FSV? CRR?
3. Which form would you use to locate the position of a command mnemonic within a command chain?
4. Which form would you use to determine the length, start word, and start bit of a Header Field? Which forms could you have possibly navigated thru, to reach this form?

Answers

1. Non-configuration controlled information including command data set definitions and modifiable command chain definitions.

2. CAR - A Command Acceptance Response is generated by an intermediate transmission facility, for example, the Johnson Space Center (JSC) Control Center Complex (CCC), and indicates the reception of a command, block of commands, or file by that facility.

FSV - A Flight System Verifier is generated by a system onboard the spacecraft and will indicate the receipt of a command, Block of commands, or file by the onboard system.

CRR - A Command Reaction Response is any telemetry data or group of telemetry data that has been received by the EHS Command Subsystem in response to the execution of a previously uplinked command.

3. Command Chain Detail form

4. Header Field Detail form.

Path: Command List form

Command Detail Form

Header Detail form

Header Field List - Query Only form

Header Field Detail form

Module 5

Documenting Permanent Changes to a Database

Objectives

In this module you will:

- consolidate concepts
- learn how to complete a DBCR

Database Change Request Application

The Database Change Request (DCR) application provides the capability for you to input, view, and edit a Database Change Request (DBCR) using an on-line form. A DBCR is a form used to document proposed changes to the configuration controlled data in the PCDB or PTDB. You can also use this form to delete, print, withdraw, and submit for approval the DBCR. If the DBCR is approved, it is incorporated into the Command or Telemetry Database. The DCR application also provides the capability for a user to view and print a directory of submitted DBCRs.

Concepts

Before discussing the DBCR form, we are going to consolidate all of the concepts that we have discussed in the previous modules (see Figure 5-1, Consolidated Concepts).

Telemetry Database - Local Table

Changes in the Telemetry database begin with the creation of a Local Table Change File. From the Telemetry Database main window, **D**atabase menu, select **S**elect **D**atabase..., provide selection criteria for the desired database and then click  , a copy of the baselined local table is displayed. Using the TDB forms perform the required changes to the local table. Once finished, you save the changes to a Local Table Change file by selecting **S**ave **C**hange **F**ile **A**s... from the Telemetry Database main window **O**ptions menu. Remember, only the changes you have made to the local table are saved to the Local Table Change file on your workstation.

Note: Since the Local Table Change file is only saved on your workstation, you may want to store it as a UDE to the UDE database for future retrieval.

Incorporate Change File Into Local Table... (Telemetry Database, **Options** menu) applies the Local Table Change File to the Project Telemetry Database (PTDB) local tables located on the DB Central Server. The database performs extensive rule verification and database comparisons to locate any discrepancies in the data.

Incorporate Local Table into User Copy... (Telemetry Database, **Options** menu) incorporates the opened Local Table into a telemetry database user copy. This allows you to have a User Copy with edits driven by the Local Table Change File submitted with a DBCR without having to re-enter all of the changes into the user copy again. The User Copy is attached to the DBCR by entering “Users Copy” in the **Change Source:** text entry field or by selecting “Users Copy” from the **Change Source:** LOV on the Database Change Request form.

When you are finished operating with the Local Table containing modifications from the change file or have completed work on your DBCR you can return the Local Table to its original content by selecting **Reinitialize Local Table...** from the Telemetry Database (Local Table) **Options** menu. **Reinitialize Local Table...** reinitializes all the MSIDs in the opened Local Table to use the PTDB defaults.

Telemetry Database - User Copy

As previously mentioned, the user copy data is provided so that you may enter changes into a database and then submit those changes for review and incorporation into the controlled database. Using the user copy to record suggested database changes allows certain validation checks and constraints to be enforced by the user interface and the database before you submit the changes in a DBCR. The user copy data for a specific user consists of all the data records which the user (you must be a data owner, project database developer, or MOL database developer) is authorized to modify in the associated database, linked to the user by his EHS account. Using the database forms you may modify, delete, and add records in your user copy data.

From the Telemetry Database main window, **Database** menu, select **Select Database...**, provide selection criteria for the desired database and then click . If a user copy for the specified database does not exist, a user copy will be populated. You will be notified if this needs to occur and that the process may take a few minutes. If a user copy already exists, you will be connected to it and the menu bar should reflect “User Copy” enclosed in parenthesis. After making and saving the required changes to your user copy you can exit the application. The User Copy is attached to the DBCR by entering “Users Copy” in the **Change Source:** text entry field or by selecting “Users Copy” from the **Change Source:** LOV on the Database Change Request form.

If for any reason you wish to discard changes you have made to your user copy and start from scratch, selecting **Reinitialize User Copy...** from the **Options** menu allows you to reinitialize your user copy to the default values defined in the selected “baselined” database.

Command Database - User Copy

The concepts and procedures used for the Command Database - User Copy are the same as those employed for the Telemetry Database - User Copy described in the preceding section. The only

difference being the databases themselves, and by definition, the types of data being manipulated. The user copy of the selected database will be limited to those commands which you are authorized to manipulate

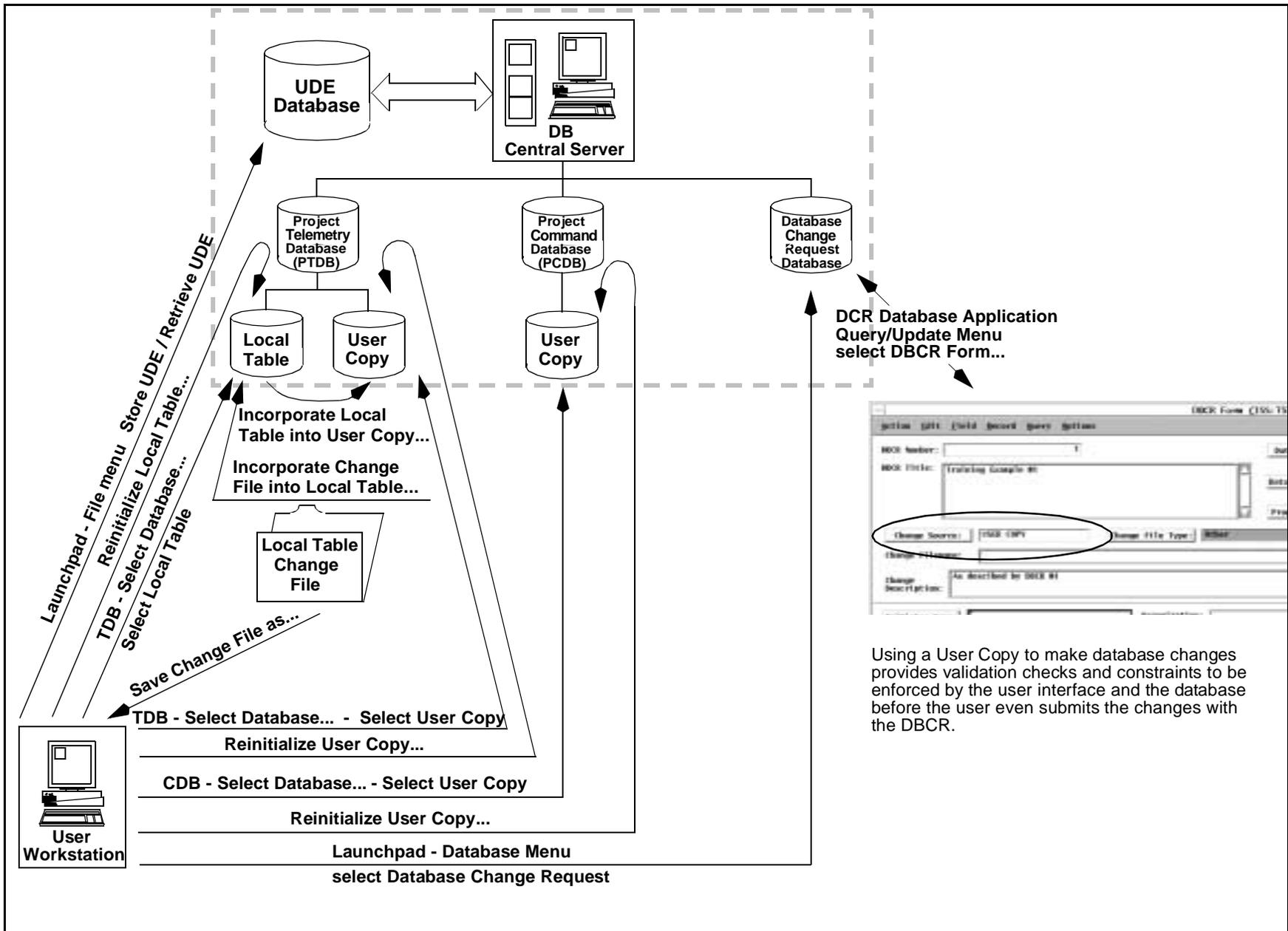


Figure 5-1, Consolidated Concepts

Exercises

Instructions

The following “Try It” creates a User Copy from your Local Table Change File. Carefully read and complete each step.

Scenario

You have been working on some changes to the Telemetry Database, a local table change file has been created and applied to your workstations local table. You now wish to create a User Copy containing your proposed database changes.

Try It...

1. Invoke the **Telemetry Database** application.
2. From the **D**atabase menu, select the **S**elect Database menu item. Click the **S**elect Local Table pushbutton. The first time this option is selected, you will receive an information dialog box that informs you that the local table is going to be populated.
3. From the **O**ptions menu select **I**ncorporate Local Table into User Copy... Clicking in the dialog box that is presented incorporates your local table data into your user copy.
4. From the **O**ptions menu select **R**einitialize Local Table... Clicking in the dialog box that is presented reinitializes the local table to the database defaults.

Database Change Request Form

The **Database Change Request** form is the heart of the DCR application. It provides you with the means to create, modify, save, and delete Database Change Requests (DBCRs). It also allows you to query the various project databases for DBCRs that have been written, regardless of their current state. When invoked directly from the **Query/Update** menu (see Figure 5-2, DCR Database, Query/Update menu), the **Database Change Request** form is blank. You can use this blank form to create a new DBCR, or you can use it to define and execute a query of project DBCRs.

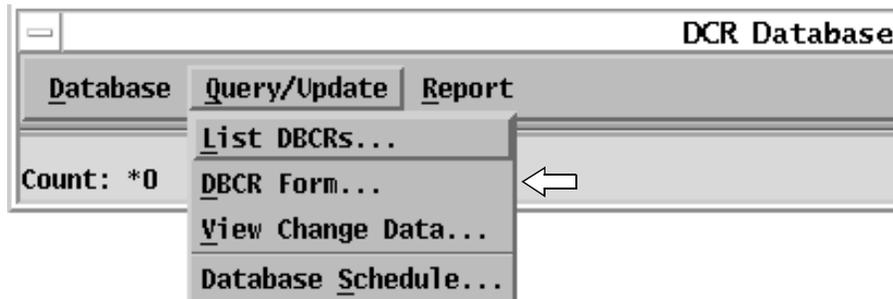


Figure 5-2, DCR Database, Query/Update menu

The work area contains numerous text entry fields (see Figure 5-3, DBCR Form). The work area can be divided into several sections and depending on your responsibilities and privileges, you may have the ability to modify information within these areas. Regardless of your privileges, you'll be able to execute a query and view information relevant to any submitted DBCR.

The screenshot shows a software window titled "DBCR Form (ISS:TSFI)". The window has a menu bar with "Action", "Edit", "Field", "Record", "Query", "Options", and "Help". The form is organized into several sections:

- Initiator Section:** Contains fields for "DBCR Number", "DBCR Title", "Database System", "Database Revision", "Proprietary Flag", "Create Date", "User ID", "Change Source", "Change File Type", "Workstation ID", "Change Filename", and "Change Description".
- Dispositioner Section:** Contains fields for "Initiator Name", "Organization", "Initiator Phone", "Initiator Mail Code", "Requester Name", "Organization", "Requester Phone", "Requester Mail Code", "Submit Flag", "Submit Date", "Withdraw Flag", and "Withdraw Date".
- Dispositioner Table:** A table with columns: "Dispositioner", "Dispositioner User ID", "Disposition State", "Disposition Revision", "Disposition Date", and "Disposition Comment".
- Assignee Section:** Contains fields for "Assignee Name", "Organization", "Assignee Phone", and "Assignee Mail Code".
- Implementor Table:** A table with columns: "Implementor", "Implementor User ID", "Implementation Revision", and "Implementation Date".
- Toolbar:** Includes buttons for "Enter Query", "Next Record", "Define Record...", "Copy DBCR...", "Save", "Execute Query", "Previous Record", "Insert Record", "View Change Info...", "Clear Window", and "Close".
- Status Bar:** Shows "Count: *0", "ENTER QUERY", and "<Desert>".

Figure 5-3, DBCR Form

This form includes sections which are specific to the following types of users:

- Initiator - anyone can initiate and submit a DBCR. To do so, he/she must complete the top portion of the DBCR form and change the **Submit Flag**: to **Y**. Only an initiator of a DBCR form can submit or withdraw a DBCR. Once a DBCR has been submitted, it cannot be modified. It may however, be withdrawn.
- Dispositioner - is a member of the Database Coordination Group (DBCX). This individual's responsibility it is to designate someone to oversee the implementation of an individual DBCR. The Dispositioner uses his portion of the form to make those assignments. Once the assessment of a DBCRs impact is known, the DBCX or HMCX will either approve or disapprove the DBCR. The Dispositioner is responsible for updating this portion of the form.
- Implementor - is a database administrator or database developer whose responsibility it is to implement any approved database changes. An implementor will also have the responsibility for updating implementation information within this portion of the form.

DBCR - Enter Query Mode

When you invoke the **Database Change Request** form directly from the DCR Database's **Query/Update** menu, all of the fields are empty and the form is in the Enter Query mode. You can press

 to execute the default query which shows detail information on all DBCRs for the selected database, or you can enter information in specific text entry fields and then execute a query. The following table identifies those fields on the **Database Change Request** form on which you can query, either through entering information directly into a text entry field or by selecting information from an LOV and having the system enter the information into the text entry field for you.

Table 5-1, Enter Query Fields

Field Titles	Text Entry Only	Text Entry with LOV Selection
DBCR Number: (Wildcards, while they can be entered, cannot be used here to retrieve a subset of DBCRs)	X	
Database System:		X
DBCR Title:	X	
Database Revision: (TDB & CDB only)		X
Proprietary Flag:		X
Change Source:		X
Change File Type:		X
Workstation ID:	X	
Change Description:	X	
Initiator Name:		X
(Initiator) Organization:	X	
Requestor Name:		X
(Requestor) Organization:	X	
Submit Flag:		X
Withdraw Flag:		X
Assignee Name:		X
(Assignee) Organization:		X

Once a query is executed, the work area will display the information relevant for a single DBCR. If your query resulted in more than one DBCR matching your query criteria, the status bar will indicate how many records (DBCRs) have been retrieved. You can use the up and down arrow keys on the keyboard, or click  and  to view other records (DBCRs) that met your search criteria.

Create DBCR

When a user has determined that a DBCR should be created, the user will enter the DCR application and pull up a DBCR form in order to create a DBCR describing changes affecting the TDB, CDB, UDE, or ODE databases. The following rules apply to DBCRs:

- At any point during the entering of information into the DBCR, a user can save the incomplete DBCR to the DCR Database.
 - If a DBCR has been saved, but not yet submitted, a user may delete the DBCR.
 - Only the originator of a DBCR may update appropriate fields of a saved DBCR which has not been submitted.
- After a user has submitted a DBCR, the only change he can make to the DBCR is to withdraw it.
 - If additional changes are required for the withdrawn DBCR, the user can make a copy of the withdrawn DBCR and make the required changes and submit the new DBCR.
 - The withdrawn DBCR cannot be updated and will be retained for tracking purposes.

The procedure for submitting a DBCR for the databases are as follows:

1. Ensure that you have already gone to the Telemetry or Command Database application and created a user copy which contains proposed database changes.
2. From the Launchpad go to the Database menu and select **Database Change Request**. From the **DCR Database** window go to the **Database** menu and click on **Select Database...**
3. On the **Select Database** dialog box, either accept the default mission as shown, type the desired mission in the text entry field, or click on  Mission LOV and select the desired mission. Once the desired mission is displayed within the text entry field, click on .
4. From the **DCR Database** window, go to the **Query/Update** menu and select **DBCR Form...**
5. Exit Query Mode by going to the **Query** menu and selecting **Cancel Query**.
6. On the **Database Change Request** form, enter a title for your DBCR in the **DBCR Title:** input text field.
7. Go to the **Database System:** text entry field and type CDB or TDB; or click on  invoke an LOV and select either **CDB** or **TDB**, depending on which user copy you modified. If you

selected **TDB** the **Database Revision:** and **Proprietary Flag:** text entry fields will be completed by the system.

8. Modify the **Database Revision:** and **Proprietary Flag:** input text fields if necessary. You can either type the appropriate entries for each field or you can click on   to select the entries from an LOV.
9. In the **Change Source:** text entry field either type **Users Copy** or click  invoke an LOV and select **Users Copy** from the list provided.
10. Enter a description of the proposed changes in the **Change Description:** input text field.
11. In the **Initiator:** input text field type your name (group you represent) or if you previously created a DBCR you can click on  and select your name from the list provided. If you select your name from the LOV, the **Organization:**, **Initiator Phone:**, and **Initiator Mail Code:** text entry fields will be completed by the system automatically. If you don't select your name from the **Initiator LOV**, then you must complete the (Initiator) **Organization:**, **Initiator Phone:**, and **Initiator Mail Code:** text entry fields.
12. If you are creating this DBCR for someone other than yourself, identify this individual in the **Requestor Name:** text entry field. If this person was previously identified as a requestor of a DBCR you can click on  and you can select that individual's name from the list provided. If you select the requestor's name from the LOV, the **Organization:**, **Requestor Phone:**, and **Requestor Mail Code:** text entry fields will be completed by the system automatically. If you don't select the requestor from the **Requestor LOV**, then you must complete the (Requestor) **Organization:**, **Requestor Phone:**, and **Requestor Mail Code:** text entry fields.
13. To submit this DBCR to the Database Coordination Group for approval, you must change the **Submit Flag:** from **N** to **Y**. You can directly edit the text entry field or you can invoke the **Submit Flag LOV** and select **Yes** from the list provided.
14. Once you've completed the above steps, clicking :
 - saves your DBCR to the DCR database and associates a number with it. The number is displayed in the **DBCR Number:** text field.
 - compares your user copy to the database against which your changes are being proposed and attaches these changes to the DBCR
 - completes the remaining system-provided text entries on the form to include both the **Creation Date:** and the **Submit Date:**

Exercises

Instructions

The following “Try It” allows you to create and submit a DBCR. Carefully read and complete each step.

Scenario

Having incorporated your proposed database changes to a User Copy you now wish to submit a DBCR so that these changes may be incorporated into the next database release

Try It...

1. From the Launchpad go to the **D**atabase menu and select **D**atabase **C**hange **R**equest. From the **DCR Database** window go to the **D**atabase menu and click on **S**elect **D**atabase....
2. On the **Select Database** dialog box, accept the default mission as shown.
3. From the **DCR Database** window, go to the **Q**uery/**U**pdate menu and select **D**BCR **F**orm....
4. Exit Query Mode by going to the **Q**uery menu and selecting **C**ancel **Q**uery.
5. On the **Database Change Request** form, enter a title for your DBCR in the **DBCR Title:** input text field. (Up to 256 characters, begin with trngxxx).
6. Go to the **Database System:** text entry field and type TDB; or click on  to invoke an LOV and select **TDB**. With **TDB** selected the **Database Revision:** and **Proprietary Flag:** text entry fields will be completed by the system.
7. Modify the **Database Revision:** and **Proprietary Flag:** input text fields if necessary. You can either type the appropriate entries for each field or you can click on   entries from an LOV.
8. In the **Change Source:** text entry field either type **Users Copy** or click on  and select **Users Copy** from the list provided.
9. Enter a description of the proposed changes in the **Change Description:** input text field.

10. In the **Initiator:** input text field type your name (group you represent). Complete the (Initiator) **Organization:**, **Initiator Phone:**, and **Initiator Mail Code:** text entry fields.
11. To submit this DBCR to the Database Coordination Group for approval, you must change the **Submit Flag:** from **N** to **Y**. You can directly edit the text entry field or you can invoke the **Submit Flag LOV** and select **Yes** from the list provided.
12. Once you've completed the above steps, click :

DBCR Processing

Once a DBCR has been submitted it is reviewed by the Database Coordination Group (DCBG) who either approve or disapprove the DBCR. When a determination has been made, the Dispositioner section of the DBCR (see Figure 5-3, DBCR Form) is completed. Information included here consists of the Dispositioner's name, the Disposition State (A or D), the database revision to be affected by the changes, and any comments the Dispositioner may feel necessary. The Disposition Date is automatically supplied by the application when the changes to DBCR are saved. The Dispositioner must also enter information regarding who is assigned overall responsibility for implementation of an approved DBCR.

Next, the DBCR becomes the responsibility of the Implementor who completes his section of the DBCR by entering his name, User ID, and the database revision against which the DBCRs changes have been implemented. The Implementation date is supplied by the application when the changes to the DBCR have been saved.

Actual incorporation of the DBCR is accomplished via the **Incorporate DBCR** dialog box (see Figure 5-4, Incorporate DBCR dialog box) off of the CDB or TDB main window **Options** menu.

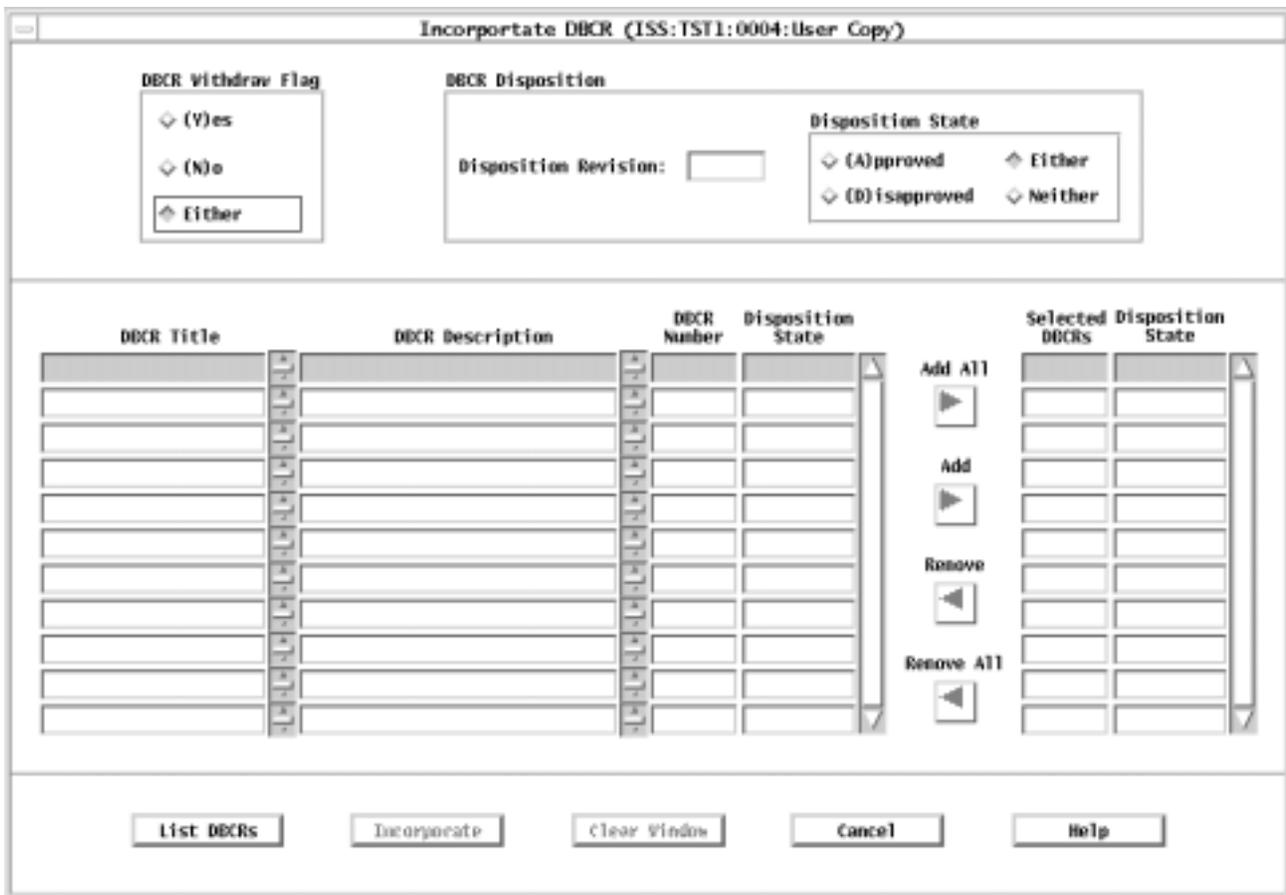


Figure 5-4, Incorporate DBCR dialog box

Prior to the building of a new revision of a PTDB or PCDB, the Database Coordination Group (DBC G) will produce a list of the DBCRs approved for incorporation into the revision and assign them to an MOL database developer to perform the incorporations. A set of database tables are created for the new revision. These tables are initially empty, except for the Telemetry System table which contains information about the revision as a whole. The database developer proceeds to incorporate the approved DBCRs as specified by the DBCG into the empty database tables. In this way, the revision is actually built from the approved DBCRs.

The **DBCR** area of the **Incorporate DBCR** dialog box contains two frames containing radio buttons, **DBCR Withdraw Flag** and **DBCR Disposition**. These radio buttons allow you to specify selection criteria for the types of DBCRs that will be returned to the list DBCR area when . The **DBCR Disposition** frame also contains a **Disposition Revision:** text entry field for specifying selection criteria.

The **List DBCR** area will contain a list of DBCRs that match the criteria specified in the DBCR area.

The **Selected DBCRs** area will contain a list of the DBCRs that you have selected to be incorporated into your current mission database.

 (**Add All**),  (**Add**),  (**Remove**), and  (**Remove All**) are placed between the lists to facilitate moving DBCRs from one list to the other.

Clicking , incorporates the DBCRs listed in the **Selected DBCRs** list in the mission database that you are currently accessing.

Summary

The Database Change Request (DCR) application provides the capability for you to input, view, and edit a Database Change Request (DBCR) using an on-line form. A DBCR is a form used to document proposed changes to the configuration controlled data in the PCDB or PTDB.

Changes to the PTDB begin with the creation of a Local Table Change File which can then be incorporated into the local table on the Database Central server. This incorporation allows you to see how your changes respond to telemetry data and also provides for certain validation checks and constraints to be enforced by the user interface and database. If you wish to proceed, you can incorporate the opened local table into a telemetry database user copy which may be attached to a DBCR for DBCG approval.

A DBCR begins with an initiator completing his portion of the DBCR and attaching a user copy. The form is then reviewed by the DBCG who either approve or disapprove the changes, completing the Dispositioner portion of the DBCR and assigning it to a database administrator or database developer. The DBCR is implemented via the CDB or TDB Incorporate DBCR dialog box and the Implementor portion of the DBCR is completed.

With the conclusion of this module you should be able to:

- state the procedure for making a permanent change to a database
- demonstrate how to complete and submit a DBCR

Answers

1. A telemetry local table is a copy of the project telemetry database while a user copy consists of all the data records which the user is authorized to modify in the associated database.
2. The steps necessary to include a local table change file into a user copy are:
 - Incorporate Change File into Local Table...
 - Incorporate Local Table into User Copy...
3. TDB or CDB main window, Options menu, Incorporate DBCR...
4. False. Only the originator of a DBCR may update appropriate fields of a saved DBCR which has not been submitted.

Appendix A

Exercise Solutions

Module 3 - Telemetry Database

Modify limits for the specified MSIDs

1. Select **MSID Detail** form.
2. Find and select the MSID from the Project-Specific Worksheet and execute query.
3. In order to define limits for an MSID that currently does not have limits, you must first define the **Limit Default Set Number:**. Specify 1 in this field and click **Save**.
4. Click **Enter Query**. Type the specified MSID in the MSID field and click **Execute Query**.
5. Now the **Limit Sensing Detail** button should be active. Click it.
6. Specify the following information:
 - MSID (Find and select from Project Specific Worksheet)
 - Set Number: 1 (This identifies the current limit set. In EHS, you can have multiple sets of limits)
 - Caution Low: from Project-Specific Worksheet
 - Caution High: from Project-Specific Worksheet
 - Tolerance: from Project-Specific Worksheet
 - Exception Monitor: None
 - Warning Low: from Project-Specific Worksheet
 - Warning High: from Project-Specific Worksheet
7. Click **Save**. Click **Close** to close the **Limit Sensing Detail** form. Click **Close** to close the **MSID Detail** form.
8. Now we are going to save the modified information to a change file that is used to alter the local table on the workstation. Click on the **Save Change File As** option under the **Options** menu. Supply the same name as the one in the Try Its and click **Save**.

9. This change file is saved on the workstation as a Local Table Change File UDE. If you plan on needing it in the future, store it to the UDE database.

Appendix B

Supported Uplink Data Types

Table D-1, Supported Uplink Data Types

Uplink Data Type	Min/Max Lengths (bits)	Valid Lengths (bits)	Description												
FEEE IEEE Floating Point		32 - 64	IEEE floating point data will be converted from the input format to either a single-precision (32 bits) or double-precision (64 bits) IEEE floating point value, stored in the defined field and rounded as necessary. No field padding is performed.												
FIBM (IBM Floating Point)		32 - 64	<p>A parameter that is represented as an IBM 370 floating point single precision (32 bits) or double precision (64 bits). The number consists of a sign, an exponent, and a fraction. The quantity expressed by this number is the product of the fraction and the number 16 raised to the power of the exponent. The format for a double-precision floating point is as follows:</p> <p>SIGN: 0 = Positive, 1 = Negative EXPONENT: Binary numbers with a range of 0-127. The exponent is derived by subtracting 64 from the characteristic. Therefore, the range of the exponent is -64 through +63. FRACTION: Binary numbers less than 1 with the decimal point to the left of the high order digit (bit 25). The range of the fraction is 2^{-1} (.5) to 2^{-24} ($5.960464498 \times 10^{-8}$) for single precision and 2^{-1} (.5) to 2^{-56} ($1.387778783 \times 10^{-17}$) for double precision.</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">CONTENT</th> <th style="text-align: left;">SIGN</th> <th style="text-align: left;">EXPONENT</th> <th style="text-align: left;">FRACTION</th> </tr> </thead> <tbody> <tr> <td>Bit no.</td> <td style="text-align: center;">0</td> <td style="text-align: center;">1 - 7</td> <td style="text-align: center;">8 - 63</td> </tr> <tr> <td>Bit wt.</td> <td style="text-align: center;">S</td> <td style="text-align: center;">$2^6 - 2^0$</td> <td style="text-align: center;">$2^{-1} - 2^{-56}$</td> </tr> </tbody> </table> <p>A single precision parameter has the same format but the LSB is 2^{-24}.</p>	CONTENT	SIGN	EXPONENT	FRACTION	Bit no.	0	1 - 7	8 - 63	Bit wt.	S	$2^6 - 2^0$	$2^{-1} - 2^{-56}$
CONTENT	SIGN	EXPONENT	FRACTION												
Bit no.	0	1 - 7	8 - 63												
Bit wt.	S	$2^6 - 2^0$	$2^{-1} - 2^{-56}$												
IBCD Binary Coded Decimal		4/8/12/ 16	Each decimal digit of a binary coded decimal value will be converted to binary integer representation and stored right-justified in the corresponding 4 bits of the defined field. Each 4-bit segment will be padded with leading zeros as necessary to fill the 4 bits.												

Uplink Data Type	Min/Max Lengths (bits)	Valid Lengths (bits)	Description
ICLK Checksum		8	Checksum data will be calculated prior to uplink, converted to binary integer representation, and stored right-justified in the least significant 8 bits of the last command data word of the command, padded with leading zeroes as necessary to fill the 8 bits.
IDIS (Discrete Integer)		1	Discrete data will be converted to binary integer representation and stored in the defined field. No padding is performed.
IDSI Distended Signed Integer		16 & 32	Distended signed integers will be converted from the input format to binary integer representation and stored right-justified within the 12 least significant bits of the defined field. The field will then be padded with leading zeros as necessary to fill the field. If the input value is positive, the high order bit of the field is then set to zero to indicate a positive value. If the value is negative, the high order bit of the field is then set to one to indicate a negative value.
IMAG (Signed Integer)	2 - 32		A parameter that may be 2 to 32 bits in length where the MSB is the sign (0 = Positive, 1 = Negative). The remaining bits represent the magnitude of the data value. Signed integers will be converted from the input format to binary integer representation, ignoring the sign, and stored right-justified within the defined field. If the input value is positive, it will be padded with leading zeros and the high order bit of the parameter will be set to zero. If the input value is negative, it will be padded with leading zeros and the high order bit will be set to one.
IPAR Parity Bit		1	Parity bit data will be calculated prior to uplink, converted to binary integer representation, and stored in the defined field. No field padding is performed.

Uplink Data Type	Min/Max Lengths (bits)	Valid Lengths (bits)	Description																																																																																																																					
ITWO (Two's Compliment Signed Integer)	2 - 32		<p>A parameter that may be 2 to 32 bits in length where the MSB is the sign (0 = Positive, 1 = Negative). The remaining bits contain the absolute value for the positive sign and the two's compliment of the absolute value for the negative sign. An example of a parameter encoded to 8 bits using two's complement for a negative number follows:</p> <table border="1" data-bbox="683 541 1349 1060"> <thead> <tr> <th></th> <th colspan="8">Bit Pattern</th> </tr> <tr> <th>Counts</th> <th>0</th> <th>1</th> <th>2</th> <th>3</th> <th>4</th> <th>5</th> <th>6</th> <th>7</th> </tr> </thead> <tbody> <tr> <td>+127</td> <td>0</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>+126</td> <td>0</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>.</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>.</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>+001</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>000</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>-001</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>.</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>.</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>-127</td> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>-128</td> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> </tbody> </table>		Bit Pattern								Counts	0	1	2	3	4	5	6	7	+127	0	1	1	1	1	1	1	1	+126	0	1	1	1	1	1	1	0	.									.									+001	0	0	0	0	0	0	0	1	000	0	0	0	0	0	0	0	0	-001	1	1	1	1	1	1	1	1	.									.									-127	1	0	0	0	0	0	0	1	-128	1	0	0	0	0	0	0	0
	Bit Pattern																																																																																																																							
Counts	0	1	2	3	4	5	6	7																																																																																																																
+127	0	1	1	1	1	1	1	1																																																																																																																
+126	0	1	1	1	1	1	1	0																																																																																																																
.																																																																																																																								
.																																																																																																																								
+001	0	0	0	0	0	0	0	1																																																																																																																
000	0	0	0	0	0	0	0	0																																																																																																																
-001	1	1	1	1	1	1	1	1																																																																																																																
.																																																																																																																								
.																																																																																																																								
-127	1	0	0	0	0	0	0	1																																																																																																																
-128	1	0	0	0	0	0	0	0																																																																																																																
IUNS (Unsigned Integer)	2 - 32		<p>A data type consisting of 2 to 32 bits and represents a binary integer. Unsigned integers will be converted to binary integer representation and stored right-justified within the defined field, padded with leading zeros as necessary to fill the field.</p>																																																																																																																					
SASC (ASCII Character String)	8 - 1024		<p>A text parameter represented by American Standard Codes for Information Interchange (ASCII) character string: 8 bits per character and a maximum of 128 characters per string. ASCII character data will be translated as required, stored left-justified within the defined field, and padded with trailing blanks as necessary to fill the field.</p>																																																																																																																					
SEBC (EBCDIC Character String)	8 - 1024		<p>A text parameter represented as an Extended Binary-Coded Decimal Interchange Code (EBCDIC) character string: 8 bits per character and a maximum of 128 characters per string. EBCDIC character data will be translated as required, stored left-justified within the defined field, and padded with trailing blanks as necessary to fill the field.</p>																																																																																																																					

Appendix C

Abbreviations and Acronym List

A

A/G	Air-to-Ground
A/M	Automatic/Manual
ADQ	Average Data Quality
AIS	Automated Information Security
ANSI	American National Standards Institute
AOS	Acquisition of Signal
API	Application Programming Interface
APID	Application Process Identifier
	Application Process Interface Definition
APT	Active Process Table
AR	Action Request
ASCII	American Standard Code for Information Interchange
AST	Active Server Table
ATT	Attitude

B

BFS	Backup Flight System
BG	Bit-contiguous Group
BPDU	Bitstream Protocol Data Unit

C	
C	Counter-dependent
C	C Programming Language
CADU	Channel Access Data Unit
CAP	Command Acceptance Pattern
CAR	Command Acceptance Response
CCBD	Configuration Control Board Directive
CCP	Central Command Processor
CCSDS	Consultative Committee for Space Data Systems
CDB	Command Database
CDD	Command Data Definition
CDQ	Current Data Quality
CLI	Command Line Interface
CM	Configuration Management
CMATS	Configuration Management Asset Tracking System
CMD	Command
CNT	Countdown Time
COTS	Commercial-Off-The-Shelf
CPU	Central Processing Unit
CRC	Circular Redundancy Check
CRR	Command Reaction Response
CSCI	Computer Software Configuration Item
CSM	Command System Management
CSS	Command System Services

CSS	Coarse Sun Sensor
CUI	Common User Interface
D	
DADS	Data Acquisition and Distribution Services
DARL	Database Access Routine Library
DB	Database
DBA	Database Administrator
DBC	Database Coordination Group
DBCR	Database Change Request
DBD	Database Developer
DCM	Document Configuration Management
DCR	Database Change Request
DCRG	Distributed Control Room Graphics
DDQ	Data Data Quality
DDS	Data Distribution System
DEMOS	Distributed Earth Model Orbiter Simulation
DG	Display Generation
DMC	Database Monitor and Control
DMC	Data Management Checklist
DO	Display Operation
DOSH	Database Operational Support History
DP	Distribute Packet
DPU	Data Processing Unit

DQ Data Quality
DSID Data Stream Identifier
DSN Deep Space Network

E

EC Experiment Computer
ECR Engineering Change Request
EGSE Experiment Ground Support Equipment
EHS Enhanced HOSC System
ELF Extremely Low Frequency
EM Exception Monitor
EML Extract MSID Library
ES Expected State

F

FDDI Fiber Distributed Data Interface
FEP Front-End Processor
FEPSC Front-End Processor Status and Control
FIFO First-In-First-Out
FPTNM Foot-Pounds to Newton-Meters
FSS Fine Sun Sensor
FSV Flight System Verifier
FTAM File Transfer Access and Management
FTP File Transfer Protocol

G

GB	Gigabyte
GCID	Ground Correlation Identification
GMT	Greenwich Mean Time
GPC	General Purpose Computer
GPS	Global Positioning System
GSE	Ground Support Equipment
GSFC	Goddard Space Flight Center
GUI	Graphical User Interface

H

H/W	Hardware
HAMASE	HOSC Automated Model and Screen Editor
HAPS	HOSC Advance Planning System
HASA	HOSC Administrative Software Account
HASS	HOSC Activity Scheduling System
HCR	HOSC Change Request
HLOG	HOSC Automated Logging System
HOSC	Huntsville Operations Support Center
HPR	HOSC Problem Report
HSR	HOSC Support Request
HTT	HOSC Training Team
HUA	HOSC User Assistance

HViDS	HOSC Video Distribution System
HVoDS	HOSC Voice Distribution System
I	
I/O	Input/Output
ICD	Interface Control Document
ID	Identification
IDD	Interface Description Document
IDQ	Instantaneous Data Quality
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
ISS	International Space Station
IST	Integrated Support Team
J	
JSC	Johnson Space Center
K	
kbps	kilobits per second
KMRTS	Kennedy Marshall Redundant Transmission System
KSC	Kennedy Space Center
L	
LAN	Local Area Network

LDP	Logical Destination Processor
LES	Limit/Expected State Sensing
LOS	Loss of Signal
LOV	List of Values
LPS	Launch Processing System
LTG	Local Table Generation
LTO	Local Table Operation
M	
M	Multi-syllable
MB	Megabyte
MCC	Mission Control Center
MCCU	Mission Control Center Upgrade
MDM	Multiplexer/Demultiplexer
MET	Mission Elapsed Time
MF	Maintenance Fixtures
MH	Message Handler
MOC	Mission Operations Computer
MOL	Mission Operations Laboratory
MOP	Mission, Operational Support Mode, and Project
MSFC	Marshall Space Flight Center
MSID	Measurement/Stimulus Identifier
MSL	Microgravity Science Laboratory
MTBF	Mean Time Between Failure

MUPS	Momentum Unloading Propulsion System
N	
N	Normal
NASA	National Aeronautics and Space Administration
NASCOM	NASA Communications
NASDA	National Space Development Agency (Japan)
NCC	Network Control Center
NDE	Non-operational Development Environment
NDL	NRT Data Log
NDL	Near Real-Time Data Logger
NEMS	NASA Equipment Management System
NG	Bit Non-contiguous Group
NGT	NASA Ground Terminal
NRT	Near Real-Time
NSOC	NASDA Space Operations Center
NSTS	National Space Transportation System
NTP	Network Time Protocol

O

OCC	Operations Control Center
OCDB	Operational Command Database
OCR	Operations Change Request
OD	Operational Downlink

OI	Operational Instrumentation
OS	Operating System
OSF	Open Software Foundation
OTE	Operational Test Equipment
P	
PAP	Payload Activity Plan
PB	Playback
PC	Polynomial Coefficient
PC	Personal Computer
PCAD	Pointing Control and Aspect Determination
PCDB	Project Command Database
PCM	PIMS Configuration Management
PDI	Payload Data Interleaver
PDRF	Playback Data Request Form
PDSS	Payload Data System Services
PFS	Primary Flight System
PI	Principal Investigator
PID	Process Identifier
PIMS	Payload Information Management System
POCC	Payload Operations Control Center
POD	Payload Operations Director
POIC	Payload Operations Integration Center
PP	Point Pair

PPS	Payload Planning System
PRT	Packet Routing Table
psi	pounds per square inch
PTC	Payload Training Complex
PTDB	Project Telemetry Database

R

R	Range-dependent
RAM	Random Access Memory
RCS	Reaction Control System
RDBMS	Relational Database Management System
RDRP	Raw Data Record Playback
RID	Review Item Discrepancy
RPM	Rounds per Minute
RR	Replanning Request
RSS	Resident Size
RT	Real-time
RTAS	Radians to Arcsecs
RTD	Radians to Degrees
RTDS	Real-time Data System
RTS	Requirements Tracking System
RUM	Remote User Machine
RW	Reaction Wheel

S

S	Super
S&E	Science and Engineering
SC	State Code
SC	Subsystem Computer
SCM	Status and Configuration Manager
SCR	Strip Chart Recorder
SDT	Shuttle Data Tape
SGI	Silicon Graphics Indy™
SGI	Silicon Graphics Incorporated
SL	Spacelab
SM	System Monitor
SMAC	System Monitor and Control
SMCM	System Monitor and Control Configuration Manager
SN	Space Network
SNMP	Simple Network Management Protocol
SOA	Science Operations Area
SPL	Scratchpad Line
SQL	Structured Query Language
SRD	Serial Receive Device
SRS	Software Requirements Specification
SS	System Services
SSCC	Space Station Control Center
SSME	Space Shuttle Main Engine

SSUP	System Services User Profile
STS	Space Transportation System
T	
T	Typical
TBD	To Be Determined
TBS	To Be Supplied
TCP/IP	Transmission Control Protocol/Internet Protocol
TDB	Telemetry Database
TDM	Time Division Multiplexer
TDRSS	Tracking and Data Relay Satellite System
TDS	Time Distribution System
TNS	Telemetry and Network Services
TNSDP	Telemetry and Network Services Distribute Packet
TTY	Teletype
U	
UDE	User-generated Data Element
UDSM	User Data Summary Message
UFT	Unrestricted File Transfer
UGSE	User Ground Support Equipment
UI	User Interface
UPAR	User Profile Access Routine
UPD	User Performance Data

V

VC	Virtual Channel
VCDU	Virtual Channel Data Unit
VCID	Virtual Channel Identifier
VMS	Virtual Memory System
VV	Verification and Validation

W

WCP	Workstation Command Processor
WEX	Workstation Executive
WSGT	White Sands Ground Terminal

Appendix D

Glossary

Accelerator	A sequence of keys that provides immediate access to application functions. For example, Ctrl + N Ctrl key + N key to invoke the N ew menu item.
Activation Type	Method used within local table application to activate a group. The defined methods are Time and Control.
Active Window	The workstation window that has input focus and in which keyboard entries impact and may appear. See “Input Focus.”
Analog	A mechanism in which data is represented by continuously variable physical quantities.
Application Main Window	The primary window of a software application.
Application Process Identifier (APID)	The APID is an 11-bit field that is included in Consultative Committee for Space Data Systems (CCSDS) headers. It uniquely identifies the vehicle that created the source packet.
Application Title Bar	The bar at the top of a main window that consists of the window menu button, the title area, and the minimize and maximize buttons.
Apply Pushbutton	A pushbutton that implements any changes made within its dialog box, but leaves the dialog box on the screen so that additional changes can be made. See “OK Pushbutton.”
Approve	In PIMS, this action is taken by a reviewer to signify his approval that a document, change request or data request be placed in the baselined state.
Archived Database	A telemetry database that no longer reflects the current real-time telemetry characteristic information. Only one archive database is available online at a time.
Attributes Defaults Bar	The area below the menu bar on the main window where application Text: , Line: , and Fill: default attributes are set.
Avtec™	A manufacturer of telemetry transmit and receive devices used in the Huntsville Operations Support Center (HOSC) Enhanced HOSC System (EHS) as the primary telemetry processing hardware devices.
Baselined	In PIMS, the final state of the review cycle. When a document, change request or data request has been approved by all reviewers, the approver may place it in the baselined state.

Baselined Database	Database that reflects the current real-time telemetry or command characteristic information for a particular mission activity. Baselined databases have completed validation.
Bitstream Protocol Data Unit (BPDU)	A protocol data unit of the bitstream function having a format of a header followed by a fixed length block of contiguous bitstream data.
Block	NASA Communications (NASCOM) 4800-bit block format utilized for the transfer of data via the GSFC/MSFC Multiplexer/Demultiplexer (MDM) system.
Calibrated	Three types of calibration exist for telemetry samples: polynomial, point pair interpolation, and state code conversion. If calibration is requested, Telemetry and Network Services (TNS) automatically converts the unprocessed sample and then performs calibration on the sample for that Measurement/Stimulus Identifier (MSID) as defined in the local table.
Cancel Pushbutton	A pushbutton that allows a user to exit a dialog box without implementing any changes.
Cascade Menu	A sub-menu or menu-within-a-menu that appears when you highlight a menu function that has an arrow to the right of its name. Cascade menus are used to group similar functions together beneath the pulldown menu.
Caution	A standard icon used throughout the user guide set to represent destructive actions which could result in loss of data.
Caution Limits	A range defined by a high and low value for an analog MSID in the Telemetry Database (TDB) and Local Table. A color code (yellow) represents values within those ranges in the application.
CCSDS Packet	A source packet comprised of a 6-octet, CCSDS defined primary header followed by an optional secondary header and source data which together may not exceed 65535 octets.
Channel Access Data Unit (CADU)	Protocol data unit used for transmission from the ISS to the PDSS. A CADU consists of a CVCDU that has been prefixed and delimited by a synchronization marker.
Click	The action of pressing and releasing a mouse button. Typically, this is a left mouse button action.
Coded Virtual Channel Data Unit (CVCDU)	A VCDU to which a block of error-correcting Reed Solomon (RS) check symbols has been attached.
Command System Manager	The position in charge of controlling the commanding system utilizing the Command System Management software. For AXAF projects, this is known as the PAYCOM position.

Commercial-Off-The-Shelf (COTS) Software	Software applications that have been purchased from a commercial software vendor as opposed to those that were developed internally.
Computation	A FORTRAN or C program used to further manipulate telemetry parameters. These programs are created by the Computation Generation application and are executed in Computation Operation.
Configuration Management (CM) Tools	Institutional applications that allow users to access and perform tasks, such as tracking requirements and equipment, scheduling resources, and logging into automated problem report systems.
Control Indicator	Used to indicate that the group will be activated for limit/expected state (LES) sensing with either the control MSID or a control MSID plus delay time.
Control Panel	The area of a window where application pushbuttons and other graphical components are located.
Converted	The process of translating raw telemetry data into an American National Standards Institute (ANSI) standard data representation so that the sample can be properly interpreted by the machine which processes the data.
Counter-dependent	A parameter whose occurrence in telemetry is dependent on an incrementing or decrementing counter in the data.
Critical Command	A command whose initiation and execution could possibly cause damage to a payload or spacecraft and impair the mission.
Database Administrator	An individual who is primarily responsible for managing the RDBMS engine and administering database accounts. He/she also has the privilege to edit restricted database fields in any database, but is normally not recommended to edit data values that drive the telemetry and command processing for the EHS system.
Database Coordination Group	A working group which includes representatives from the appropriate project operations personnel, project source DB developers, MOL DB developers and the HOSC validation team. Review and approve/dissapprove DBCRs, resolve conflicts and evaluate any DB related issues.
Database Developer	An individual that has the privilege to edit restricted fields (e.g., decom, etc.) for both operational and non-operational databases that drive telemetry and command processing for the EHS system.
Dataset	A saved set of a command's modifiable fields used to update a command prior to being transmitted.

Data Stream Identifier (DSID)	A field within a Secondary EHS Protocol Header for PDSS Payload Data used as a unique identifier for the data stream. This bit denotes if the type of data contained therein is CCSDS packet data (0) or BPDU (1).
Delivered Database	A database must be delivered before it can become pre-released. A delivered database has not been validated for operational testing.
Delta Limit	Maximum acceptable difference between consecutive samples of a parameter.
Desktop	The computer monitor backdrop area on which all windows are opened. May also be referred to as workspace.
Development	In PIMS, the first state of the review cycle in which a document, change request or data request is still being written or is being updated.
Direction Keys	A group of computer keyboard arrow keys which allow users to move up, down, left, and right within an application or menus.
Disapprove	In PIMS, the action taken by a reviewer to signify disapproval and recommendation against moving a document, change request or data request into the baselined state.
Discrete Values	Telemetry values that have states (e.g., on or off).
Double-click	The action of pressing and releasing a mouse button twice in rapid succession.
Drag	To press and hold down a mouse button while moving the mouse on the desktop (and the pointer on the screen). Typically, dragging is used while moving and resizing windows.
Drawing Tools Palette	A group of tool buttons that is used to create graphic objects in order to display telemetry data, initiate commands, and start scripts and computations. The palette is located on its own floating dialog box or the application window.
Dump	During periods when communications with the spacecraft are unavailable, data is recorded onboard and played back during the next period when communications resume. This data, as it is being recorded onboard, is encoded with an onboard embedded time and is referred to as dump data. When a near real-time (NRT) request is written specifying that dump data is desired, the onboard embedded time is used to fulfill the request.
Dynamic Objects	Graphical objects that represent updating telemetry data.
Ellipse	A geometric shape which can be created on a display (i.e., a plane of a cone, an oval shape, etc.).
Expected State	Text state code which indicates the nominal value of a parameter.

Expert Mouse Actions	Clicks or double-clicks of mouse buttons which are non-standard and which activate special functions.
Filter	The filter function is used within a dialog box to refine and define subsets of files you want to work with using a string search and wildcard. Characters can be used to implement the filter function.
Fonts	A style of printed text characters.
Graphical User Interface (GUI)	A way of interacting with computers using graphics-oriented software and hardware.
Grayed out	A menu selection item that has been made insensitive, which is visually shown by making the menu text gray rather than black. Items that are grayed out are not currently available.
Greenwich Mean Time (GMT)	The solar time for the meridian passing through Greenwich, England. It is used as a basis for calculating time throughout most of the world. Displayed within the HOSC, it follows the format ddd:hh:mm:ss.
Grid	A pattern of horizontal and vertical lines forming squares of uniform size on a display, used as a reference for locating points.
Group Parameter Composition	Parameter composition where the bits of a parameter are contiguous and a multiple occurrence of that parameter exists as a group of samples.
Groups	MSIDs which have been grouped together, primarily for use with the Exception Monitor (EM) application.
Hazardous Command	A command whose initiation and execution could pose a threat to human life or the entire mission.
Help	A standard icon used throughout the user guide set to indicate that a cross-reference is provided to assist in solving problems or to answer questions.
Huntsville Operations Support Center (HOSC)	A facility located at the Marshall Space Flight Center (MSFC) that provides scientists and engineers the tools necessary for monitoring, commanding, and controlling various elements of space vehicle, payload, and science experiments. Support consists of real-time operations planning and analysis, inter- and intra-center ground operations coordination, facility and data system resource planning and scheduling, data systems monitor and control operations, and data flow coordination.
I-beam Insertion Bar	A graphical image used to represent the insertion point of text in a text entry area which provides a visual cue that text entry is anticipated by the system.

Icon	A graphical representation of an object on the desktop. Objects can be minimized (iconified) to clear a cluttered workspace, and restored (opened), as needed.
Input Focus	A window or window element that is activated, and available for subsequent actions. Input focus is usually indicated by highlighting or changing the color of the activated element.
Input Slider	An input object that allows users to change values of pseudo parameters and computational constants assigned to objects. Pseudos can be used in other applications (i.e., scripts, computations, etc.).
Insensitive	An object or area of an application window that does not have input focus.
Integrated Support Team (IST)	Institutional groups at the HOSC responsible for configuring, monitoring, and resolving problems with computer systems and application software.
Launchpad	A floating menu bar that is used to initiate all HOSC software applications.
Legend	A table that labels parameters plotted on a chart or grid.
Limit Delta	Maximum acceptable difference between consecutive samples of a parameter.
Limit/Expected State Sensing (LES)	A configurable option in Display Operation that allows the user to select whether he wants to see limit violation status or not. The incoming data is compared against the Local Table limits.
Limits	Defined ranges for a measurement which are used to indicate off-nominal conditions: Caution High, Caution Low, Warning High, and Warning Low.
Line Plot	A plot that uses lines to represent the relationships among telemetry values.
Local Table	A subset of the TDB stored on a workstation or server used for telemetry processing.
Maximize Button	A control button that is located to the right of the application title bar. When pressed, this button enlarges the application window to its largest state.
Menu Bar	The area at the top of a window that contains the titles of pull-down menus.
Merge	The combining of data from different sources for a specific time slice. During merge, the best (cleanest) data from each source will be used to create a contiguous segment of data for the specified time slice.
Message Area	The part of the application window where system messages/responses are shown.

Message Dialog Box	An area that provides information, gives the current status of data, asks questions, issues warnings, or draws attention to errors.
Mini-Application	A secondary main window activated from within a main window application.
Minimize button	A control button located to the right of the application title bar. When pressed, it iconifies the window.
Mission, Operational Support Mode, and Project (MOP)	A MOP is what delineates one EHS activity from another. MOP information is available in the common configuration file on every node.
Mnemonic	An underlined character on a menu item, that allows users to initiate the item by typing letters on a keyboard. A user-friendly name used to reference a command residing in the command database.
Mode Independent	Mode Independent is used to describe any process that is not dependent on a data mode.
Modifiable Commands	Commands containing at least one data field which can be updated during operational activities prior to their uplink transmission.
Mouse	A pointing device that is used along with a keyboard in point-and-click user interfaces. The mouse used with HOSC workstations contains three mouse buttons. The left mouse button is used to activate and select items on windows. The middle mouse button is used for move functions. The right mouse button is used to access popup menus.
MSID Text Field	An output object for viewing telemetry containing a label for the telemetry parameter, as well as the current value of the parameter displayed in a specified format (i.e., decimal, hex, octal, binary, American Standard Code for Information Interchange (ASCII), etc.).
Multiple Drawing Mode	A mode that allows users to draw multiple objects of the same type.
Native Data Type	Defined in the database and indicates how the MSID data will be interpreted in the HOSC.
Nominal	A color code indicating expected conditions within defined limits of parameters.
Non-Shareable	A flag has been set to not allow other users to retrieve your User-generated Data Element (UDE) from the UDE Database and use it on their local workstation.
Normal	A telemetered parameter that occurs once per packet.
Note	A standard icon used throughout the user guide set to direct your attention to specific items of concern.

OK Pushbutton	A pushbutton that implements any changes specified within a dialog box. The dialog box is dismissed after this pushbutton has been selected.
Option Menu Button	A pushbutton which, when clicked, displays a menu of related options. The selected option is shown as the pushbutton label.
Output Slider	An object that displays telemetry parameters.
Packet	A data unit comprised of octets that a source application generates.
Parameter Composition	Describes how the bits of a parameter can be arranged in a packet for a sample(s) of that parameter.
Pixmap Object	A picture that can be either drawn using the pixmap editor, or scanned and assigned using the pixmap editor.
Playback	Playback data can originate either internally or from some other facility. Project servers in the HOSC receive Playback telemetry streams from the HOSC Data Distribution System (DDS) and perform the same processing as would be performed on real-time telemetry streams.
Pointer	Sometimes called the mouse cursor, the pointer shows the location of the mouse on the desktop. The pointer's shape depends on its mode. (e.g., on a window frame, the pointer is an arrowhead, while you are waiting for an action to complete, the pointer becomes clock).
Point Pair Calibration	A measurement which is calibrated using a series of linear segments. The linear segments are defined by a pair of points for each segment. Each point consists of a raw count value and a corresponding engineering unit value.
Pointer Shapes	A graphical shape that a pointer assumes in the drawing mode (e.g., cross-hairs, I-beams, hour-glasses, etc.).
Polynomial Coefficient Calibration	A measurement is calibrated using the following polynomial calibration equation: where: eu - engineering units cnts - counts $eu = COEF0 + (cnts1 \times COEF1) + (cnts2 \times COEF2) + (cnts3 \times COEF3) + (cnts4 \times COEF4) + (cnts5 \times COEF5) + (cnts6 \times COEF6) + (cnts7 \times COEF7) + (cnts8 \times COEF8) + (cnts9 \times COEF9).$
Popup Menu	A menu that is invoked when the right mouse button is clicked. Functions available are the most common and vary from application to application.
Predefined Commands	Commands completely defined prior to an operational activity. Predefined commands contain no modifiable data fields.

Pre-released Database	A database that has been validated for operational testing. It is used to validate UDEs (displays, comps, etc.) prior to the baseline release of the database.
Project Telemetry Database	Contained within the Telemetry Database, includes the telemetry definitions needed to drive HOSC telemetry processing for a specific project/mission. The source of the real-time telemetry processing tables found in the Telemetry Local Table identified by a project/mission/revision prefix. Also included are tables to contain user copy data, an error log, and an Initial Load Table.
Protocol	1: Provides the formulas for passing messages, specifies the details of message formats, and describes how to handle error conditions. More important, it allows us to discuss communication standards independent of any particular vendor's network hardware. A communication protocol allows one to specify or understand data communication without depending on detailed knowledge of a particular vendor's network hardware. 2: A term referring to the type of source data used in the construction of an EHS packet. EHS Packet Protocols include: "C" - CCSDS packet, "P" - pseudotelemetry packet, "T" - encapsulated TDM packet, "B" - encapsulated block packet, and "D" - TDS packets.
Pseudo MSID/Parameter	A parameter identification (ID) that has been assigned to contain the output from a computation.
Pseudo Packet	A telemetry packet consisting of external pseudo MSIDs. External pseudo MSIDs are generated (either by EHS computations or scripts), packetized and multicast on the project LAN.
Pulldown Menu	A list or menu of possible options that is hidden under a general phrase and invoked by clicking the left mouse button.
Pushbutton	A control that causes an immediate action. To press a pushbutton on the screen, point to it and click the left mouse button.
Radiobuttons	A group of buttons that allows users to make only one selection at a time. Radiobuttons are small diamond-shaped buttons.
Range-dependent	A parameter whose occurrence in telemetry is dependent on the value of a range parameter.
Real-time Data	Real-time data is telemetered to the HOSC and distributed for immediate use. Real-time telemetry data, received into the HOSC system and written to the NRT log, is indexed by its time stamps and other identifying information. When an NRT request is submitted that covers a particular time slice, this indexing information is used to meet that request.

Recall Text	Area within an application that allows users to input up to 256 characters.
Release	In PIMS, an action taken by the manager of a document or request that releases it from the baselined state back into the development state for modifications.
Resize Borders	The area that surrounds the framed area of an application, and is used to change the height or width of the window.
Resize Handles	Up to eight handles surrounding an object that allows users to resize objects, displays, or windows.
Review	In PIMS, the second state of the review cycle in which the document, change request or data request has been written and submitted for review and approval.
Sample Composition	Describes how the samples of a parameter are arranged in a major frame.
Scatter Plot	A plot that uses unconnected dots to represent the relationships among telemetry values.
Scratchpad Line (SPL) Directives	Provides users with the capability to start and stop displays, computations, and scripts. A user may also uplink and modify commands and update pseudo MSIDs through the use of SPL directives.
Script	A file containing a sequence of directives that can be invoked in a single step.
Scroll Bar	A control that allows the contents of a window area to be displayed without resizing a window or list.
Select Button	The mouse button used for most operations. By default the select button is the left mouse button.
Set Pushbutton	A pushbutton that allows a user to implement changes based on selections made within a dialog box. Reacts like the OK pushbutton and closes the dialog box.
Setup Message Area	The dialog box that allows users to change the number of lines displayed within the message area of the application main window.
Shareable	A flag has been set to allow other users to retrieve your UDE from the UDE Database and use it on their local workstation.
Shotgun	Parameter composition when the bits of a parameter are scattered in multiple non-contiguous words of a major frame.
Slider Box	A graphical component of the scroll bar, which is dragged to provide a different view of the same file, list, or text area.
State Code Calibration	A measurement is converted to a text state code.
Static Object	A graphical object that is not receiving telemetry data.
Status Bar	A feature that allows the viewing of application critical configurations within the main window.

Submit	In PIMS, an action taken by the manager of a document, change request or data request in which the document or request is placed in a state of review and approval and released from development.
Subset	A collection of measurements from the total measurement set that is bounded as an integer number of octets but does not constitute the packet itself. A mini-packet.
Super	A parameter that occurs more than once per packet.
Superseded Database	If a baselined database already exists for a project and mission, before a new baselined database can be released, the current baselined database is designated as superseded.
Switch MSID	A parameter whose value determines which limit or calibration set will be used for the specified MSID.
TDM	Time Division Multiplexed - a technique for transmitting multiple parameters within a single serial bit stream by interleaving them, one after the other.
Time Indicator	Indicates that a group will be activated based on a defined start time and deactivated based on a defined stop time.
Time Plot	A plot against time containing up to four Y-Axis parameters.
Time Reference	A time format that is represented in either GMT or Mission Elapsed Time (MET).
Time Tag	A time reference marking an event. For example, a parameter goes out-of-limits at 230:16:00:00. The time tag for the out-of-limit event is 230:16:00:00.
Tip	A standard icon used within the HOSC user guide set to indicate that suggestions or hints are provided.
Togglebuttons	Small buttons that can be switched "on" or "off." To switch a togglebutton, point to it and click the select button. Black indicates that the desired attribute is in effect or "on."
Tolerance	Number of times the MSID exceeds the limit value before an EM warning message is issued.
Typical	Parameter composition when the bits of a parameter are contiguous.
Unprocessed	Raw telemetry data.
User-generated Data Element (UDE)	A user-generated file. For example, a display, script, computation, pixmap, etc., is a UDE.
User Data Summary Message (UDSM)	A PDSS Data Quality packet. Information contained in the packet include: the start/stop time of the UDSM report period, the number of unplanned LOS occurrences, the number of BPDUs or packets per DSID received, the number of VCDU sequence counter errors, and the number of packet sequence counter errors per DSID.

Virtual Channel (VC)	A CCSDS construct whereby a physical communications channel is shared among different users, each of whom uses part of the available bandwidth for a virtual channel of CCSDS CADUs incorporating predetermined CCSDS identifiers for each user.
Virtual Channel Identifier (VCID)	A binary identifier located within the VCDU header, which when concatenated with the spacecraft identifier, uniquely identifies a particular spacecraft virtual channel.
Virtual Channel Data Unit (VCDU)	A CCSDS data set of specific structure and fixed length, which includes CCSDS specified headers and into which user data is packaged for transmission over the space-to-ground link.
Warning Limits	A color code (red) representing limit violations of a parameter.
Wildcard	Placeholders for other characters in a string. Three wildcards are permitted in most HOSC applications. The “*” which represents any combination of characters, and the “?” which represents any single character. A blank can be used to replace a single “*” to indicate “all”. Database applications use Oracle as their basis and therefore “%” is used like the “*” and an underscore character “_” is used like the “?”. Blank operates the same in database applications as othe HOSC applications and represents “all.”
Window Menu	The menu that appears when you press the window menu button, which is located to the left of the application title bar on a window frame. Every window has a system menu that enables you to control the position of the window.
Workspace	The area on a terminal where the windows of a user’s environment appear. The workspace is sometimes referred to as a desktop or root window.
XY Plot	A plot that contains one X-Axis and up to four Y- Axes parameters.