Database Applications In Science Data Systems For Low-Cost Satellite Missions

Priscilla L. McKerracher, Hyung S. Han, Douglas B. Holland, Jacqueline A. H. Stock
The Johns Hopkins University Applied Physics Laboratory
(443) 778-4474 (Baltimore)
(240) 228-4474 (Washington, D.C.)
Priscilla.McKerracher@jhuapl.edu, Hyung.Han@jhuapl.edu,
Douglas.Holland@jhuapl.edu, Jacqueline.Stock@jhuapl.edu,

Abstract. As in financial and business applications, satellite systems accumulate data sets over some time period. Certain portions of this data lend themselves to storage within a commercial database system. The advantages and disadvantages of using commercial database systems for storage of data related to satellite systems are discussed. The types of data stored in satellite science data systems are outlined. Case studies from MSX, NEAR and TIMED, three satellite missions developed at The Johns Hopkins University Applied Physics Laboratory, are reviewed. An example of database logical architecture to support science data queries is presented. Implementation lessons in INGRES and ORACLE are noted. Issues related to the user community in each case are discussed. In conclusion, a summary of the lessons learned from all case studies is presented.

Introduction

Within a satellite ground system, commercial Database Management Systems (DBMSs) are found primarily in mission operational systems and in science data systems. These DBMS systems are often referred to as "catalog" systems; they provide index pointers to different types of data. The types of data stored in mission operational systems are spacecraft performance assessment data, telemetry point definitions, spacecraft command definitions, autonomy rule definitions and processor memory maps. The types of data stored in science data systems are primary science data, data summary indices, metadata, and large volume data. This paper presents examples of databases for science data systems from three missions. It reviews the difference in the types of requirements and the level of support for the low-cost, National Aeronautics and Space Administration (NASA)-supported NEAR and TIMED missions versus the Ballistic Missile Defense Organization (BMDO)-supported MSX mission. Lessons learned on each of these missions include reviews on database availability, accessibility, accuracy, sizing and overall cost.

A commercial DBMS, especially a Relational DBMS (RDBMS), offers significant advantages over a flat file system. This is especially true for users who must store a large volume of data and who require centralized control over access to that data. The most common disadvantages of the conventional flat file systems are:

- The means of relating one data item to another is not obvious,
- Custom applications are required to ensure data integrity,
- Record-by-record processing is required for data access.
- Modifying the data record structure, for example adding a new item, requires reprocessing of the data.

A commercial RDBMS offers a set of standard tools to the user. This saves design and development costs for the system developer. The costs of acquiring, maintaining and managing the system must be weighed against these savings. The savings come in the form of a standard database interface known as the Structured Query Language (SQL). SQL is a non-procedural language which allows the developer to work at a higher level of abstraction for data location and manipulation. A commercial RDBMS offers a wide range of tools for defining, storing, querying, manipulating and displaying data. It manages objects within a database and enforces data integrity and data security.

However, commercial database management systems have associated costs and weaknesses. Licensing and maintenance fees can be expensive. Commercial RDBMSs are often developed with the needs of the business customer in mind. Therefore they may not support some features that are standard for satellite data customers, i.e. some types of time conversions and time representations. Although they can provide the potential for a wide flexibility of standard and ad-hoc queries to the user via SQL commands, the developer must often provide a more intuitive interface for novice users. The developer must decide whether to invest the time on a generic user interface that allows complex queries or to provide a simpler user interface that allows only a

subset of queries. Often the satellite data user population requires only a small subset of queries on a regular basis. Queries for data of a certain time-range or for a particular instrument configuration are the most popular. The database developer must always do the work of matching user requirements to database design and customization.

Three Mission Overview

Catalog systems were developed to support the selection of and access to science data for 3 different spacecraft missions developed by the Johns Hopkins University Applied Physics Laboratory (JHU/APL) Space Department. The missions are BMDO's Mid-Course Space Experiment (MSX), NASA's Near Earth Asteroid Rendezvous (NEAR) and NASA's Thermosphere Ionosphere, Mesosphere Energetics and Dynamics Experiment (TIMED). The focus of the overall science mission, the types of science instruments on board, the amount of science data collected, the primary organization of the data, and the distribution of the user community are key elements which influence the design of the respective catalog systems. Therefore these mission elements are briefly reviewed here. Tables 1-3 compare and contrast key elements of these missions.

MSX - Mission 1

The MSX satellite was launched in April 1996 with a suite of state-of-the art sensors and has collected several terabytes of high-quality data on terrestrial, earth-limb, and celestial backgrounds. The science data catalog system reviewed here supports only the Ultra-Violet Imaging and Spectral Imaging (UVISI) suite of nine sensors. UVISI consists of 4 Imagers and 5 Spectral Imagers (SPIMs) built by the JHU/APL. See Table 2. The uploading of pre-planned command sequences allows UVISI to be configured to point in practically any direction in the celestial sky or towards the earth. Data is organized into files representing these pre-planned experiments known as Data Collection Events (DCE's). The UVISI Catalog system was built primarily to support searches into the data without regard to DCE identifier, ie. by geographic location or by signal strength in a particular wavelength region. The catalog primarily supported local access by one science team: the ShortWave Terrestrial Background Data Analysis Center (STBDAC). As well as standard instrument metadata and pointing data, the catalog contains customized or specialized metadata in the form of compressed science data indices, which are described in the MSX case study.

NEAR - Mission 2

The Near Earth Asteroid Rendezvous (NEAR) spacecraft was launched on February 17, 1996 as one of the first of NASA's Discovery Program: a series of innovative, small-scale, low-cost spacecraft projects. NEAR hosts six scientific instruments composed of an imager, a spectrograph, a rangefinder, spectrometers, and a magnetometer array. See Table 2. NEAR's primary mission, the study of the nature and evolution of S-type asteroids, will commence in February 2000, when it will be placed into orbit around the asteroid Eros. Data is organized primarily by Spacecraft time, which is converted to UTC time via SPICE routines provided by the Jet Propulsion Laboratory (JPL). Data is also organized according to "Times of Interest." These are times identified by the science planning teams which correspond to certain events, such as the time planned for the viewing of a particular region on the asteroid. The catalog must support queries by multiple science teams at multiple remote locations. The catalog contains metadata, pointing index data, and special "plate model" indices that support asteroid geolocation requirements by mapping the instrument data onto triangularly-shaped regions of the asteroid at varying resolutions.

TIMED - Mission 3

The TIMED spacecraft will launch in May of the year 2000 and is the first mission in the NASA Solar Terrestrial Probes Program. Its primary objective is to investigate and to understand the energetics of the Mesosphere and Lower Thermosphere/Ionosphere (MLTI) region of the earth. The TIMED satellite hosts a complement of 4 science instruments consisting of a spectrograph, photometers, spectrometers, radiometers and interferometer. See Table 2. GPS time or spacecraft orbit number is the cross-reference parameter between instrument data files. As in the NEAR mission, the TIMED catalog system must support queries from science teams at multiple remote locations. This catalog also contains metadata and pointing data, as well as specialized timeline metadata which describes events, modes, and anomalies for the spacecraft and instruments.

Table 1. Mission Requirement Summary

	AVAILABLE BUDGET	DATA ARCHIVED	CATALOG SIZE
MSX	Relatively Large	Tera-Bytes	14 GBytes
NEAR	Relatively Very Small	10's of Giga-Bytes	2 GBytes
TIMED	Relatively Small	100's of Giga-Bytes	5 GBytes

Three Mission Comparison

Requirements - These 3 missions demonstrate the ability of the APL Space Department to adjust their spacecraft mission design and supporting development to successfully support the newer low-cost NASA missions. Table 1. summarizes the relative available budgets and the data archive and catalog sizing requirements. The MSX mission and UVISI instrumentation capabilities were by far the most expensive and complex of the three. NEAR and TIMED represent much lower-cost, focused NASA missions. Overall their catalogs performs much simpler indexing functions and support much smaller data archives.

RDBMS - The science data catalogs developed for all 3 missions used commercially available Relational Database Management Systems (RDBMSs). However, there was an evolution from an INGRES implementation on MSX to an ORACLE implementation on NEAR and TIMED. ORACLE is the overall market leader; its most powerful features are its portability and scalability. However INGRES supports more user control of the data storage methods available. The developer may select heap, hash, Indexed Sequential Access Method (ISAM) or Balanced Tree (Btree) implementation. The storage method can be changed at any time as the table grows or changes. ORACLE defaults to a system selected implementation.

Types of Data - All catalogs contain configuration information as well as time. Specialized indices exist in the MSX catalog as custom Data Measurement Indices and in the NEAR catalog as Asteroid Plate Model Indices. MSX catalog provides the highest level of detailed metadata and data indices. The TIMED catalog provides the greatest amount of top level information. TIMED users are most concerned about learning what specific products are available on specific days from what instrument systems. MSX catalog users were most interested in discovering whether UVISI data could be found that supported analysis of unique science questions. See Table 3.

User Interface - The interface requirements evolved from local command line access via SQL on MSX to a networked Web access on the NASA missions. The simpler Web accessible catalog systems were more successful. The challenge in all 3 cases was to understand what types of queries the users really needed and how to store indices to the data to support those queries.

Table 2. Comparison of Science Instruments by Mission

MSX - UVISI SUBSET	NEAR	TIMED
4 Imagers • IUW Wide Ultra-Violet • IUN Narrow Ultra-Violet • IVW Wide Visible • IVN Narrow Visible 5 SPIMS • Scanning Spectral Imagers	• MSI - Multi-Spectral Imager • NIS - Infrared Spectrograph • NLR - Laser Rangefinder • XGRS - X-Ray/Gamma-Ray Spectrometer • MAG - Magnetometer	• GUVI - Scanning UV Spectrograph • SEE - Photometers & Spectrometers • SABER - Radiometer • TIDI - Fabry-Perot Interferometer

Table 3. Comparison of Data Tables in Catalog by Mission

	MSX	NEAR	TIMED
Database Primary Key	UT as tenths of seconds since midnight 01/01/2000	Mission Elapsed Time	Universal Time
Core Metadata	Configuration, Alarm	Instrument State and Condition	Data Product Type
Spatial Data	Instrument Pointing	Instrument Pointing and Range, S/C Position	S/C Position and Time, Orbit Number
Associated Metadata	DCE Name, File Registration	Plate Index, Mission Phase, Times of Interest	Data Product Version, Data Product File
Special Metadata	DMIs	Plate Models	Timelines

MSX Case Study

STBDAC and DPC Overview

The UVISI catalog system resides in the Short-wave Terrestrial Backgrounds Data Center (STBDAC), the primary center for analysis of the UVISI data. The STBDAC maintains a local archive of un-calibrated UVISI data files and provides data processing support for conversion of the data to various calibrated formats. Figure 1 illustrates the context of the UVISI STBDAC catalog within the MSX spacecraft to ground system.

A typical 15 to 50 minute experiment, or "Data Collection Event," (DCE) is down-linked from the MSX tape-recorder during regular collection passes and processed in the UVISI Data Processing Center (DPC). The UVISI Data Processing Center "Pipeline" software produces two of the primary data files for loading the catalog system: the Data Measurement Indices (DMI) File and the Data Quality Indices (DQI) File. Information identifying the corresponding Definitive Attitude Files (DAF) also populates the catalog.

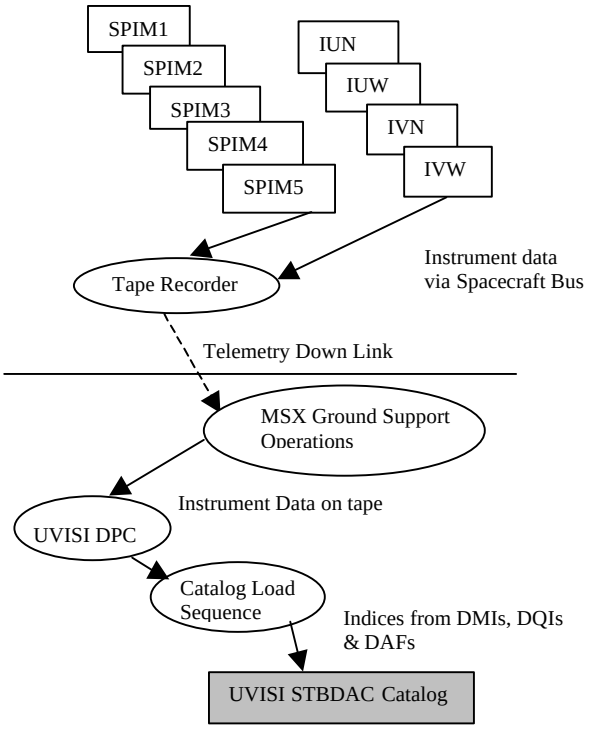

Figure 1. UVISI STBDAC Catalog Context Diagram

Purpose of UVISI STBDAC catalog

The amount of data collected by the UVISI sensors well exceeds the amount easily stored and accessed online. The current archive is on the order of 5 Terabytes. Therefore the UVISI STBDAC catalog was designed to perform 2 purposes:

- ***File Registration***
- ***Content Indexing of Science Data***

File Registration

Because the Data Collection Events are large (1-4 Gigabytes each) and because the amount of UVISI data exceeds our online storage capacity, data is primarily archived to tape. After 2 years of spacecraft operations the STBDAC has over 1400 tapes in storage, each containing data from 1 to 3 Data Collection Events. The file registration process loads all information required to facilitate retrieval of the data files into the STBDAC catalog system. This information includes tape ID number, the names of all files archived on that tape, and backup ID numbers. Storage of this data into a catalog system allows a user to query the catalog on the number of files available, types of files available, and exabyte tape location. An automated software system ensures the accuracy of this catalog. There are currently 154,790 files registered in the catalog system. This represents a total number of 1020 actual events. In addition, 1836 planned but not flown event names are registered.

Content Indexing of Science Data

Although the Registration process allows a user to make queries about a particular file name or type, it does not allow a science user to locate data that might be of interest for a particular type of scientific investigation. However, this functionality is provided by the processing of additional UVISI data indices. For example, a scientist might want to see if there is any UVISI data in a particular spectral region of a particular South American rain forest. In order to locate data of this type, a catalog system would have to support a query to find valid data, given a known latitude and longitude region and a selected UVISI instrument configuration. The design of the UVISI STBDAC catalog supports these types of queries by cataloging the following indices pertaining to UVISI:

- configuration, from DQI file
- alarm status, from DQI file
- science data summaries or averages, from DMI file
- pointing, from DAF file

Configuration Indices **-** The configuration defines the operational state of a particular sensor. Configuration status is reported in DQI files at the same rate at which images are obtained, typically 2 Hz or 4 Hz. The configuration indices include these items

- Telemetry Mode
- Compression Flag
- Filter
- Frame Rate
- Automatic Gain Mode
- Gain State
- Gate State
- Mirror Position
- Scan Size
- Number of X or Y Pixels per Image

Alarm Indices **-** The alarm indices are also extracted from the Data Quality Indices Files. Alarms are defined as states stored as 1 bit. That is, an alarm can have only 2 states: On/Off, Yes/No, etc. There are approximately 50 alarms defined for each of the 9 sensor types. Examples include:

- Data overflow
- Voltage out of range
- Current alarms
- Voltage rate-of-change out of range
- Uncalibrated slit position
- Dark current out of range
- Intensifier state on or off
- Instrument power on or off
- Cover open or closed

Because the developers did not know which alarms would be useful, all available alarms were placed in the catalog. In retrospect, only a handful were actually useful. The cost of extracting, checking, and maintaining correct alarm definitions for 50 states was high. The wide variety of choices was also confusing to the user.

Image Indices **-** The Data Measurement Indices (DMI) are custom indices defined specifically to provide a means of capturing 'snapshots' of UVISI Imager or SPIM data. In each case particular regions of the 2-Dimensional data "image" are outlined. Then the individual pixel values are summed or averaged over the outlined region at 5 second intervals. Figure 2 illustrates the Imager DMI regions and Figure 3 illustrates the SPIM DMI regions. For the imagers the individual 2-Dimensional scenes are subdivided into 5 regions. One region located at each of the 4 corners and one central region. For each region the following values are calculated

- Mean
- Standard deviation
- 3rd Moment
- Number of Bright Pixels above a selected value.

The Individual 2-Dimensional SPIM scenes are subdivided differently. They are split into as many as seven wavelength bins per SPIM. For each bin the following values are calculated:

- Total Sum
- Standard Deviation

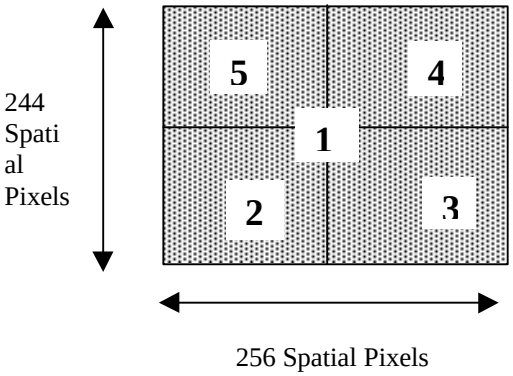

Figure 2. Region Definitions:
Data Measurement Indices for UVISI

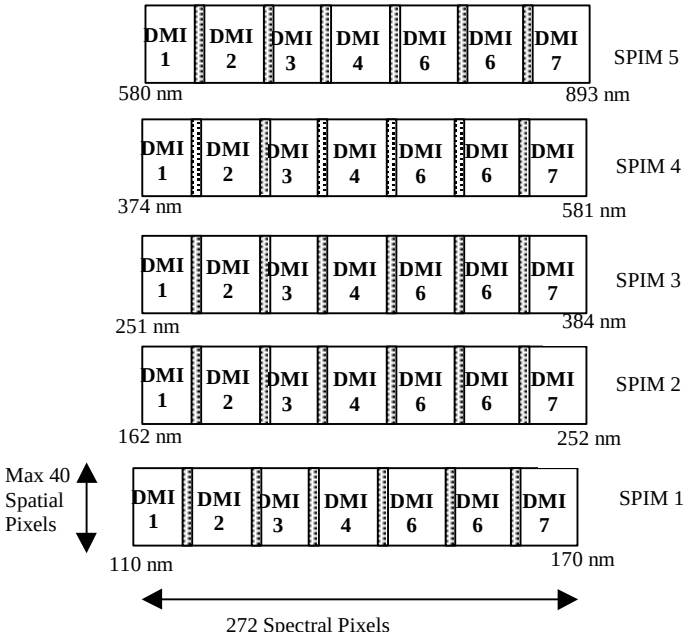

Figure 3. Region Definitions:
Data Measurement Indices for UVISI

Pointing Indices **-** Pointing indices describe either the location on the earth or in the sky which is subtended by the instrument field of view. They also contain parameters that describe the position of the imager viewing angle with respect to other light sources such as the sun or moon. The information required to produce these indices is found in the MSX definitive attitude file (DAF). The DAF contains 25 Hz ephemeris and attitude information. Because these values can be large floating point numbers, they are stored in a compressed format.

Examples of fields populated in the Pointing Indices Tables are:

- ECI J2000 Coordinates
- Line-of-Sight to Boresite Sun Angle
- Geocentric Longitude & Latitude of Boresite Tangent Pierce Point
- Altitude of Boresite Tangent Pierce Point
- Geocentric Longitude & Latitude of MSX Spacecraft

Tables in the Catalog

Catalog tables have been designed to allow a query in any combination of time, DCE, alarms, configuration, pointing, DMIs and file information. All of these data are correlated based on the time and/or DCE. The other table categories are:

- *Alarms* - each entry represents a specific type of alarm, sensor number which had this alarm set, and its start and end time.
- *Configuration* - each entry collectively represents a unique configuration among the sensors with its start and end time. The values for the configuration status are scaled.
- *DMI Sets* - each entry represents one DMI record, and is tagged with the time. A unique set ID is assigned for each DMI record, and this ID is used to build a cross reference to partition and pointing data.
- *DMIs* - each entry is associated with a unique DMI set ID, and represents a specific DMI value for each of the operational sensors.
- *Data Collection Event* - contains planned and actual events. For each event, its DCE ID, operation day, and event start and end time are recorded.
- *Partition* - each entry represents a logical data partition within an event, its start and end time, and a DCE ID that it belongs to.
- *Pointing* - each entry collectively describes the time, direction, and pointing value for each defined pointing item.
- *File Registration* - contains comprehensive information about all the files that have been registered in the Catalog.
- *Cross Reference tables* - in order to speed the query processing, *partition_alarms*, *partition_config* and *partition_pointing* tables are built based on the time intervals.

User Interface

The STBDAC UVISI Catalog employs very little customization of user interfaces. The INGRES standard interfaces via SQL queries allow full query potential, but are only usable by an SQL expert, knowledgeable about the catalog design. Some custom interfaces do allow operations staff to search for particular DCE's by name. This allows operations personnel with no Database Management skills to perform online location and retrieval of files.

There are a few custom reports that can be executed on command line without knowing the SQL. They are:

- *Event Report* - reports all events that have been loaded more than 7 days ago.
- *File Report* - reports all the files registered that are correlated to input DCE.
- *DMI Report* - reports event date and mean values of the 5 regions for the Imagers as a function of the elapsed time since the beginning of the input DCE.
- *Pointing Report* - reports event date, and right ascension angle (longitude) and declination angle (latitude) as a function of the elapsed time since the beginning of the input DCE.
- *DCE List Report* - reports all the DCE IDs that have been loaded in the Catalog.

Expert users of the system can develop custom scripts to extract catalog information and export it to plotting or textual formatting programs. It was envisioned that a user interface would be developed later in the program. And that the interface would be developed to support the types of queries that were found to be most useful. Because such a broad range of queries and responses were possible, including queries that could take a long time (hours) to process, the software development team wished to avoid spending much time and money on a front end that might be later found to not be useful. Unfortunately, a full custom user interface was never developed. The reasons for this are that very few queries were submitted to the catalog system and no budget was available at the appropriate timeframe.

One concern is that very few queries were submitted to the catalog system precisely because there was no user interface! Our hypothesis is that a user interface serves 2 functions:

1. Supports data access by a less expert user,

2. Educates the users as to what types of queries can be made.

The absence of a user interface for novice database users left several of the potential database users, mostly atmospheric chemistry scientists, uncertain as to the types of queries they might help them in their analysis of UVISI data.

Query Examples

- Find the region of geographical coverage (latitude and longitude values) for a particular DCE.

- Find the mean and standard deviation on Region 1 for the Imager1 (Wide-field, Far Ultraviolet) for a particular DCE.

- Find the Oxygen and Nitrogen values for the SPIM1 (Far Ultraviolet) given a particular Solar Zenith angle range for a particular DCE.

Platform

The Catalog uses the INGRES DBMS on an HP 735 System running HP-UNIX operation system. Approximately 36 Gigabytes of online storage have been allocated. Current size of the Catalog is 14.3 Gigabytes.

Catalog Loading

The first step in the catalog loading process is the retrieval of DQI and DMI data from the archived tapes. Registration information loaded previously into the Catalog for each DCE assists the operator in this process via a custom application built using the INGRES 4GL. The subsequent software processes which support loading of the Indices to the catalog system are:

- Indices Processor
- Partition Builder
- Pointing Processor
- Catalog Pointing Reformatter
- Catalog Loader

The method used by the catalog loading system includes first scanning through the instrument configuration indices in order to identify the changes in instrument configurations for a particular DCE. Next the data is partitioned according to these configuration changes into smaller, time-sequential, coherent data segments. Files containing index values for each partition are then created. Finally this data is loaded into the commercial relational database management system. Figure 4 illustrates the data flow of the loading process.

***Indices Processor* (IP) -** The UVISI IP reads the DMI and DQI files for each unclassified event, creates data files that are loaded into the alarms, configuration, DMI tables in the Catalog, and creates a partition specification file that defines the start and stop partition times for the Partition Builder.

***Partition Builder* (PB) -** A data partition is a logical grouping of all data associated with a time interval. The Partition Builder uses the partition specification generated by the IP to create data partitions and a pointing data requirement file (PRF) for the Pointing Processor, and builds links between the partition and the pointing, and builds links among the DMI sets, partitions and pointing values.

***Pointing Processor* (PP) -** The PRF generated by the PB contains one record for each of the DQI records. The Pointing Processor uses the times in this file to search for the matching pointing data in the Definitive Attitude file. It also uses the mirror position and operational status for each of the SPIMs specified in this file to calculate the tangent height values for each of the operational SPIMs. Pointing data is required for the beginning and ending of each partition and for each DMI record that falls within a partition.

***Catalog Pointing Reformatter* (CPR) -** The CPR converts the pointing values into a compressed form, and creates pointing data files that are loaded into the pointing tables. The Catalog supports 36 pointing items. Due to the number of the pointing data items that are cataloged, each is compressed into either 2- or 4-byte integer or 4-byte floating point using a set of conversion routines that keep the accuracy of the data values within required precision.

***Catalog Loader* (CL) -** The Catalog is only loaded in a batch mode with new event data when all the data files from each software process are successfully generated. The cross reference tables are built at the end of loading. Since the catalog loading to the database requires DBA privilege, a customized client-server system (DBA Server) was developed. The server

constantly runs as a privileged user while waiting for any request for access to the database. After authorizing the client, the request from the client is then executed as this privileged user. Any requests to the database requiring a DBA privilege can be submitted through this server. This allows the operations staff to submit non-trivial predefined database jobs without actually acquiring full access the DBA account.

Catalog Maintenance

Because of the large size of the catalog system the catalog tables are divided across a multi-disk system. At one point size of the check-point file exceeded the size supported by the UNIX system. Therefore it was necessary to back-up entire large disk systems. . Maintenance of the catalog system is expensive in the areas of system administration, database administration and operations. Reduced funding impacted the status and reliability of the catalog.

Lessons Learned

- Catalog was bigger than expected: An unachieved goal was to have two types of catalogs.
  - Staging catalog holding only 7 days of data to support faster loading and query access.
  - Historical catalog holding the entire set of data for the mission.
- Back-ups took longer than expected - The database size became too big. Since even the OS (HP-UX 9.X) had a file size limit (maximum 2GB), vendor provided regular checkpoint utility became useless.
- Loading times were longer than expected - On average per event it took 1/2 to 1 hour to generate data files and 2 to 3 hours to load them into the catalog. This time was cut down to 10 to 25 minutes per event by the following actions:
  - Bypass the INGRES logging system to significantly reduced the I/O time.
  - Drop indices before loading and rebuild them after loading.
  - Build cross reference tables using only current event that is being loaded.
- Occasional inconsistencies in the database occurred when there was a table locking problem with other user transactions. Whenever this happened, it required a simple patch operation to repair.
- Lack of simple user interface severely limited use of catalog.
- Should have built a simple user interface early on that only allowed simple queries, i.e Location or time queries.
- Interpretation of Data Quality from Catalog variables is a complex activity. That is how does the user know which alarms should be checked during a query?
- The catalog development process was hindered by the attempt to develop a catalog system without a detailed knowledge of exactly how to interpret data quality from specific alarm indicators, DMI values, and pointing values as they related to illumination conditions.
- Unexpected artifacts such as solar glare and non-meaningful DMI values during instrument configuration changes further complicated the interpretation of the data.
- A less automated solution provided a more cost-effective, although less detailed method for cataloging the UVISI data.. A composite master list of all UVISI experiments was created in a simple EXCEL spreadsheet. This list has the date, time, event number and a simple comment on the purpose of the experiment. However, location information in both geomagnetic and geographic coordinates is missing from this list.

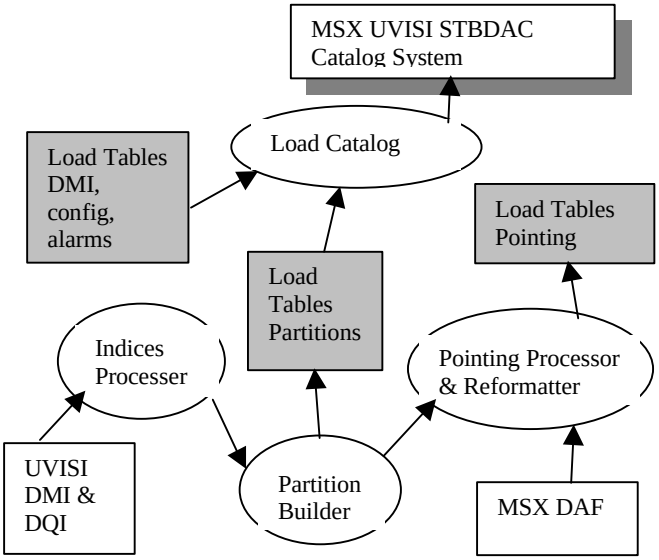

Figure 4. UVISI STBDAC Catalog Load Process Flow for Science Data content Indices

NEAR

NEAR SDC Overview

The NEAR ground system is composed of three segments: 1) Mission Operations Center (MOC), 2) Science Data Center (SDC) and 3) Science Teams. The NEAR science catalog system is found in the SDC. The SDC serves as the central processing site which combines the spacecraft and instrument telemetry data with the navigation data. This processing allows the SDC to provide science products in a format that the science teams can then analyze on their desktops. The NEAR science catalog system aids the science teams in locating data products which are pertinent to their particular studies.

Figure 5 shows the product flow via the external interfaces to the SDC. The NEAR MOC delivers telemetry data to the SDC. The JPL Navigation Team delivers the navigation data via data sets known as SPICE kernels. The primary SPICE data sets contain Spacecraft ephemeris, Planet, satellite, comet or asteroid ephemerides, Instrument description kernel, C-matrix (pointing kernel), and Events kernel together with spacecraft clock kernel (SCLK) and leap seconds kernel (LSK). The five remotely located science teams have access to the NEAR SDC products and the NEAR SDC catalog system via a World Wide Web interface.

The Catalog is one of five components in the SDC. Those five components are:
1) *Telemetry Server Process* - The TSP reads transfer frames from the level1 telemetry data files, sorts the data according to time, splits them into packets, removes duplicate packets, and moves them into the telemetry archive.
2) *Science Data Archive* - The SDA contains instrument Hierarchical Data Format (HDF) files that are created from the cleaned and merged telemetry archive, and associated SPICE kernels. The SDA is the source storage of science data to produce science products, to archive to the Planetary Data System and to load the SDC catalog.
3) *Product Generator* - The Product Generator creates each product requested by an internal or external user and is described in a Product Description Form. It also generates the data files needed to load the science catalog.
4) *Product Server* - The Product Server schedules and submits the product generators either at specified times or upon request.
5) *SDC Catalog* - The Catalog facilitates access to the SDC archive and subsets the archive based on science team criteria. The NEAR SDC data flows are shown in Figure 6.

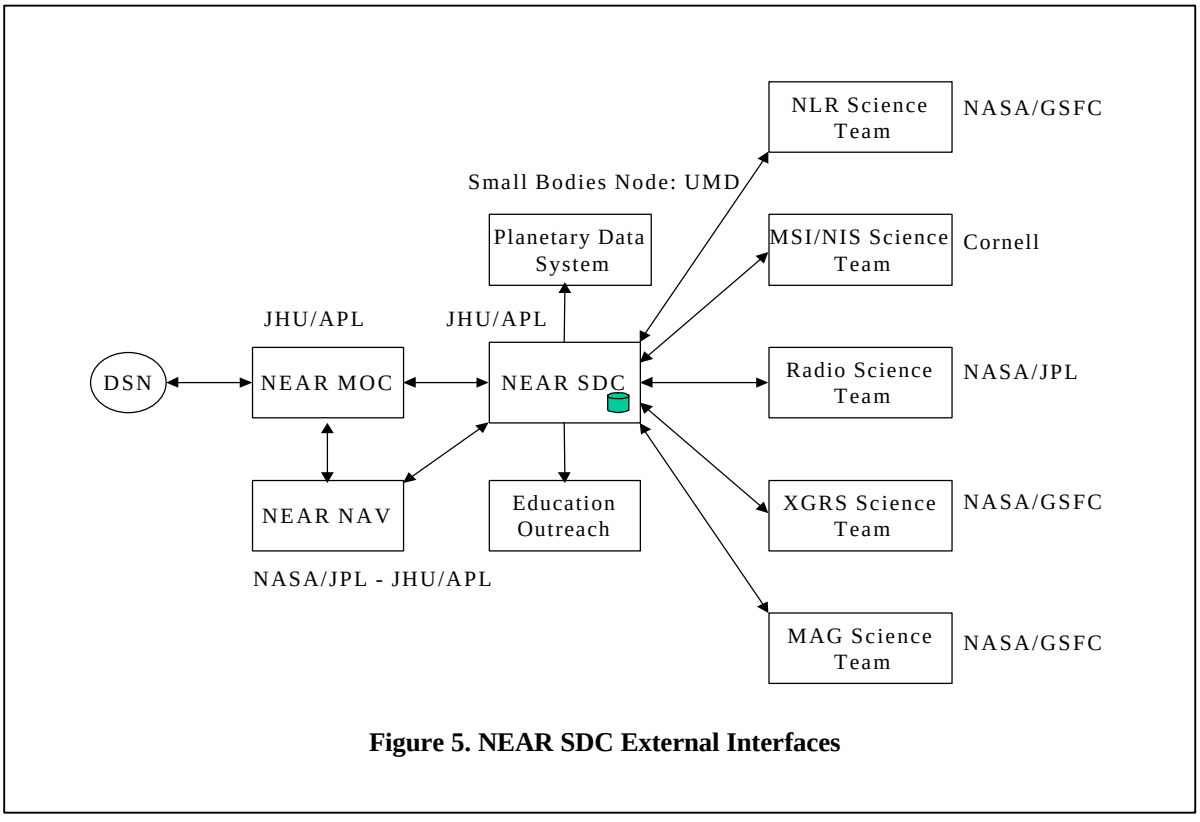

Figure 5. NEAR SDC External Interfaces

NEAR SDC Catalog Overview

The SDC archive file systems are organized to provide easy navigation for sequential time access. However, as the amount of data archived and products generated in the SDC archive grow, it becomes harder to locate specific products of interest, correlate one product to another, or share them among different instrument teams without downloading and reading all data files. The Catalog has been developed using a commercial Relational Database Management System to aid such searching capabilities through a WWW query interface.

The Catalog does not store in the database any science data such as images and spectra. It is a metadata catalog. This is because each instrument team is responsible for its own data analysis, and the SDC supports this architecture by keeping the science data archive entirely online at **http://near.jhuapl.edu/**. Without duplicate storage of science data in the database, the Catalog can provide rapid search and index services on the products in the SDC archive. The types of metadata stored in the Catalog have been selected based on the common search interests of the instrument teams. In addition, since most of the science data will be referenced to a shape (or plate) model of Eros, the Catalog also utilizes a plate model.

The types of data populating the Catalog are the following:

- Core metadata - instrument state and condition
- Associated metadata - plate index tables, mission phase, times of interest for each instrument team
- Spatial data - instrument pointing, range, spacecraft position
- Special metadata - plate model

Most of these data can be correlated based on the Mission Elapsed Time (MET). The MET of each instrument data measurement serves as an index to the products in the SDC. Hence, using the Catalog as a search engine, the users can quickly query for applicable science data and if available, rapidly access the products that satisfy the search criteria. Typical search criteria are time, instrument conditions, pointing, spacecraft position, mission phase, instrument team's times of interest and landmarks on Eros.

Catalog Implementation

The Catalog has been implemented using Oracle 7 Relational Database Management System on an HP

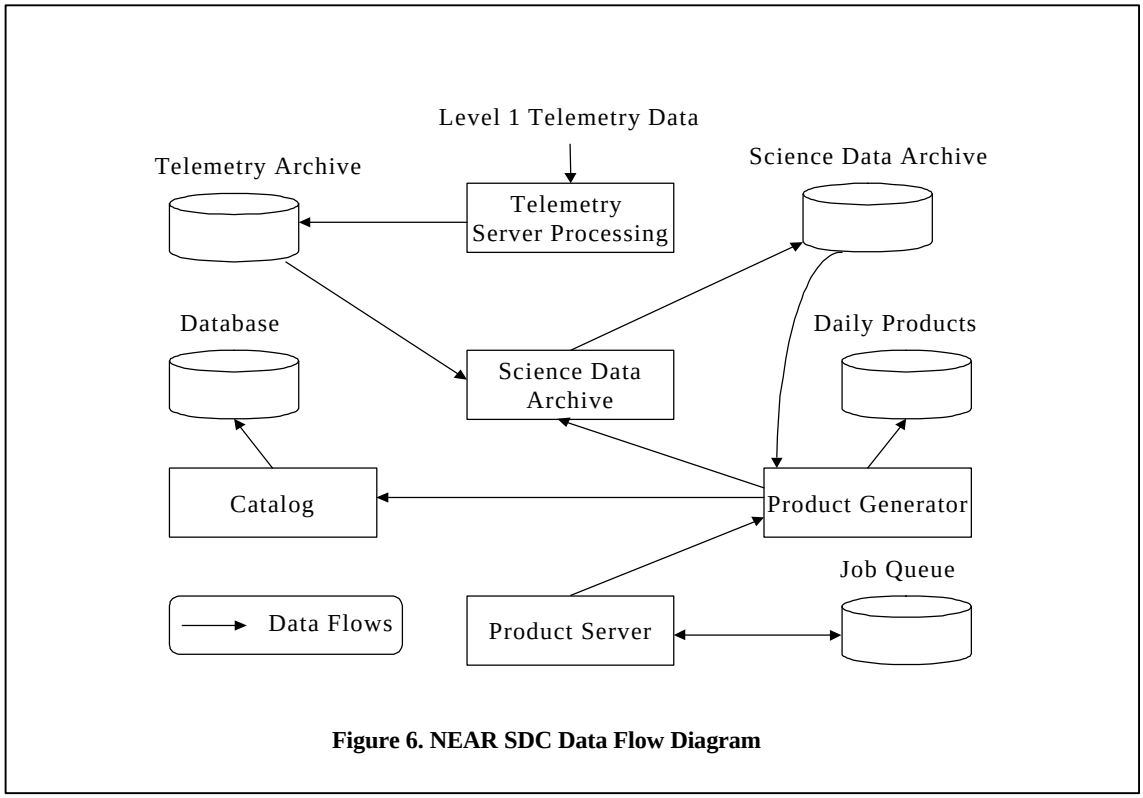

Figure 6. NEAR SDC Data Flow Diagram

9000/770 system running HP-UX 10.20. Primary access to the Catalog is through the WWW query interface utilizing Oracle's Web Server/PL/SQL Agent. The incoming request for generating dynamic documents can be processed through the Web Request Broker or Common Gateway Interface. The WRB, main component of the Web Server, is an asynchronous request handler and the PL/SQL is one of the WRB cartridges. Since the WRB is designed to establish persistent database connection between the transactions, it is expected to provide better performance than CGI.

Conceptual View as Tables

In a relational database, the data is stored and represented to the users as tables. The tables for the Catalog have been specifically designed to facilitate the types of queries most expected from the users. Furthermore, proper indexing on data fields also speeds up the search. All the data in the tables can be correlated by time except for the tables that overall describe a plate model implemented at a specific time in the Catalog. The plate model may be drastically changed as NEAR gets closer to Eros. The Catalog is designed to allow the entire system to be rebuilt using the most current plate model of Eros if necessary.

- *Instrument Tables* - For each instrument there is one main data table that contains time, housekeeping, pointing, and range information. Each row in this table represents one full science record and is keyed by the MET of the measurement.
- *Plate Model Information Tables* - A plate model for the shape of Eros is used to locate certain landmarks or to have positional reference on Eros. A plate is described by its three surrounding vertices, with the (*x, y, z*) coordinates of each vertex.
- *Plate Index Tables* - Each index table is instrument-specific and is used to relate each instrument measurement to a set of plates that were partially or totally in the instrument's FOV when the measurement was taken.
- *Spacecraft Position Table* - This table contains the sampled time (in MET and ET), the spacecraft position in J2000 and Asteroid Body Fixed referential frames, and solar and emission angles when the sampling was taken.
- *Mission Phase Table* - This table is a collection of all mission phases. Each entry describes a different mission phase with its start and end time.
- *Times of Interest Tables* - There is one table for each instrument team. Each entry describes an instrument-specific event of interest with its start and end time.

Catalog User Interface

The Catalog provides a user-friendly query interface through a WWW browser like Netscape or Microsoft Internet Explorer. It allows users to 1) browse tables and data in the database, 2) search for applicable science data with iterative refinement on criteria, or 3) submit a request for science data products satisfying the query result. Typically, WWW links (URLs) to the product files available in the SDC are returned. In most cases, the query results are presented in a plain tabular format for two reasons. First, occasionally the query result may contain a quite large amount of data. In this case, returning the data in HTML tabular format can be time consuming. Second, the query result saved as a text file can be easily imported into other applications such as spreadsheet or plotting packages.

User Types vs. Query Types

For any types of users - novice or expert, predefined queries with most common search criteria are provided for easy access. Predefined queries include the following.

Query Criteria	Query Result
Time	Selected instrument data taken during a specified time range. Time can be MET, ET, or YEARDAY.
Range	Selected instrument data where the S/C was within a specified distance range.
Pointing	Selected instrument data when the pointing was within a specified range.
Conditions	Selected instrument data when the condition was within a specified range. Condition can be any field in the instrument table accepting the search criteria.
Plate	Selected instrument data that contain specified plate IDs
Mission Phase	Selected instrument data taken during a specified mission phase
TOI	Selected instrument data taken during a specified time of interest

The users running these predefined queries do not need any previous knowledge of SQL commands. However, they are expected to provide reasonable input values for the user selectable parameters. The query results can be repeatedly refined or pruned before submitting a request for data products.

For each of these queries, a criteria string that is internally built during the query execution is displayed with the query result on the returned page. By referencing this string and on-line documentation on tables defined in the Catalog, experienced users can submit custom queries through the ad-hoc query submission Web page. The result of an ad-hoc query is a custom report. A request for science products based on the ad-hoc query result is not currently supported.

Web Security

Access to the Catalog is primarily controlled by the IP address of the client. Requests for product files are protected by an additional username-password authentication. In addition, username-password and role-based securities for data access are enforced within the database. All the access to and transactions on the database are carefully logged. The philosophy behind this logging was to provide information for optimizing and enhancing the search capabilities provided through the Catalog. For example, by logging the types of most frequently submitted ad-hoc queries, we can identify those to be included as predefined queries.

Query Example (Predefined Query)

INPUT: Instrument type: MSI, Time: 1998024 (as YEARDAY), filter wheel number in (5, 6). See Figure 7.
OUTPUT: MSI instrument data on Day 1998024 when the filter wheel number was 5 or 6. See Figure 8.

Catalog Loading

For each table defined in the Catalog there is a corresponding product generator module that creates load data files. All load data files are generated via the HDF readers from the HDF files, SPICE kernels and other data files in the Science Data Archive. The product generators for the Catalog update have been rapidly created because of the large number of reusable software components from the science product generators. These product generators also adopted the SDC software development standards like software version control, standard argument parsing, and event and error logging. The Catalog load scripts, that utilize the ORACLE SQL*Loader, are used to update the database in a batch mode with the load data files. The data accuracy is reinforced through the integrity constraints. The Catalog is routinely updated with a predefined time window to collect new instrument data records. The Catalog is also capable of rebuilding the database segment within any specified time range.

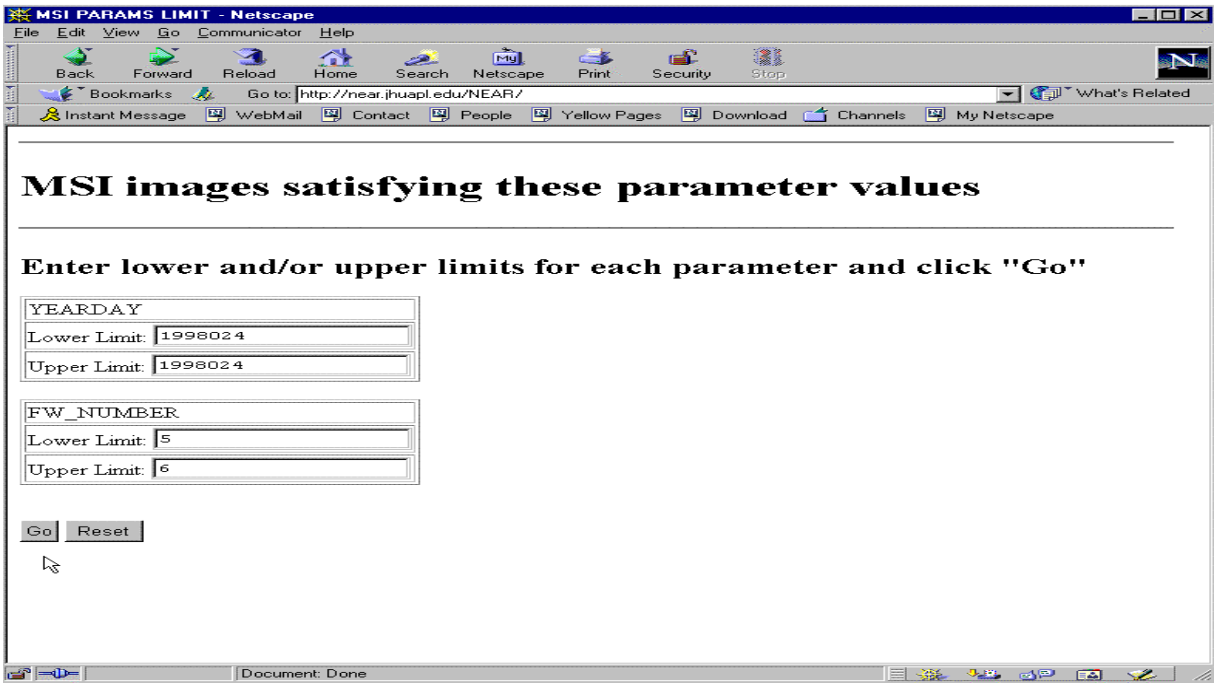

Figure 7. Query Example (INPUT)

The Catalog supports three types of updates - daily update, partial rebuild and total rebuild.
1) *Daily Update* - is used for daily additions of new data records down-linked within last 10 days. This 10-day window is configurable to balance the system load and the currency and availability of the Catalog.
2) *Partial Rebuild* - is used when a database segment that is outside the daily update window needs to be updated.
3) *Total Rebuild* - is used for significant changes to the current Catalog data due to updates of the SPICE kernels, especially the plate model, or reprocessing of the data in the SDA.
The Catalog update is scheduled through the cron-like Product Server.

Because updating the Catalog based on each change to the SPICE kernels affecting the data fields could be time consuming, the update will be based on certain criteria. This means that there may be a time when the Catalog is not in an up-to-date accurate state for the SPICE-related data. However, the changes in data should be minimal in such case. It is important to emphasize that the SDC Catalog is an indexing tool to subset the science data, not a data analysis tool. The science teams are expected to do science analysis using the data sets identified by queries to the Catalog.

Catalog Maintenance

The Catalog will maintain only the rendezvous data online. Once NEAR enters the orbiting phase in February 2000, the data collected during the cruise phase will be separately archived in a format that can be easily imported into desktop applications. Estimated size of the Catalog during the Eros orbiting phase is approximately 2 GB. A simple database backup and recovery method - a full offline backup of a database that has been shut down (cold backup) once per week is currently scheduled for the following few reasons:

- Moderate changes in a plate model of Eros could make a previous backup of data invalid and force the Catalog to be rebuilt for the sake of data currency.
- The Catalog is not required to be available at all times (24 hours per day, 7 days per week).
- The Catalog can be easily rebuilt or updated within any time range as a one time requested job through the Product Server.

With a weekly cold backup, the Catalog can increase the throughput during normal operational hours because extra I/Os needed for archiving (by the database process) are avoided. When a media failure occurs, the data can be quickly recovered by restoring the most recent full backup and reprocessing the lost segment, which would include at

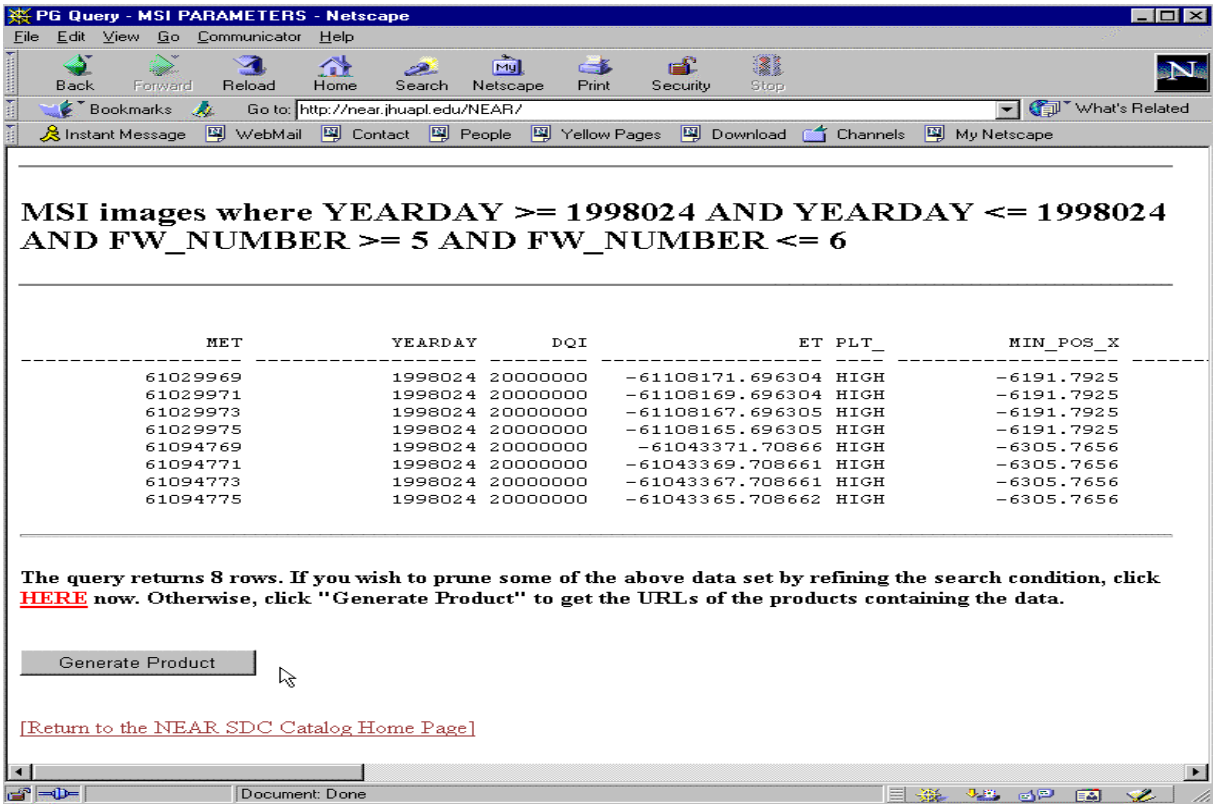

Figure 8. Query Example (OUTPUT)

most a week-worth of data.

Lessons Learned

- Getting more feedback and interest from the science teams during earlier stage of the software development was desired. However, since the Catalog is an indexing tool, it will become more valued and appreciated during the orbiting phase, the period that may not have enough funds for new software development.
- SPICE updates affect the data currency and accuracy of the Catalog. Some changes require a dynamic rebuild of the system, which in turn determines the type of database backup and recovery scheme.
- Rollback on the spacecraft clock will cause duplicate METs, which violate the uniqueness constraint in the database.
- Plan for frequent version updates of the commercial products.
- Earlier versions of Oracle Web Server supported a limited number of pre-built cartridges (plug-in applications); with recent versions, it may worthwhile to closely investigate the performance on WRB vs. CGI for Java, Perl, or other cartridges.

TIMED Case Study

TIMED Science Data System (SDS) Overview

The TIMED Mission Data Catalog is part of the TIMED Science Data System. In order to achieve the goals of the TIMED mission, data collected, generated, analyzed as part of the TIMED program must be shared among TIMED investigators, the scientific community and the general public. The TIMED Science Data System (SDS) will provide the mechanism through which this sharing will take place. The SDS is responsible for the generation, acquisition, distribution, and archival of science data necessary to support the TIMED mission. In this capacity, the SDS will provide useful data products to the TIMED program elements, the scientific community, K-12 educators and the general public.

The TIMED SDS is a distributed system with elements that are part of several different facilities. The central facility is the Mission Data Center (MDC) which distributes and archives raw telemetry and maintains and manages cataloging and distribution of Science Data Products. The remaining elements of the SDS are the data processing facilities which produce Science Data Products from raw instrument data. The TIMED system context diagram is shown in Figure 9.

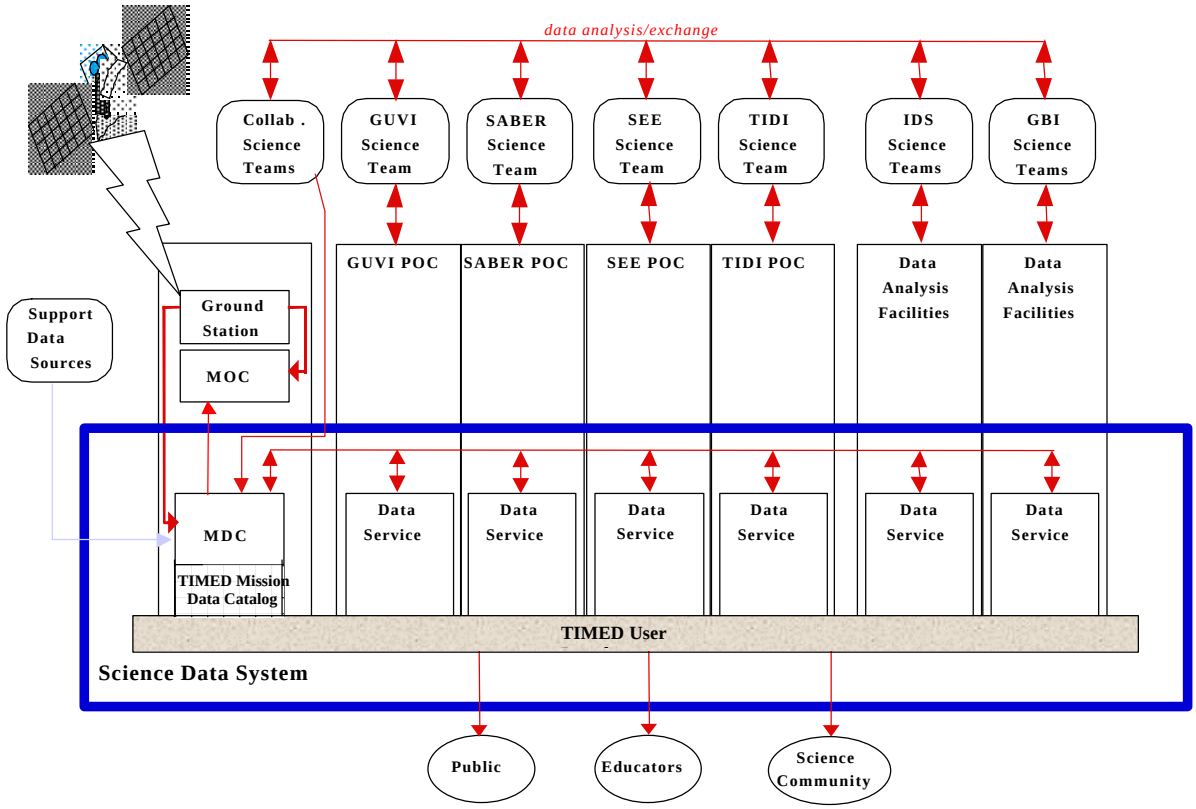

Figure 9. TIMED Science Data System

TIMED Mission Data Center (MDC) Overview

The TIMED Science Data Catalog is one of the four components of the MDC. The four components are:

Telemetry Server captures the telemetry from the Ground Station, builds a telemetry archive, and serves telemetry over network connections (both near real-time and play back from archive).

Data Product Production produces important planning, support and summary products.

Mission Data Cataloging and Distribution creates and maintains a TIMED Mission Data Catalog; supports the distribution of the data product information; archives all TIMED mission data; and performs administrative services.

Mission Publications provides the user interface for access to the data held at the TIMED MDC. It is implemented as a World Wide Web site for the distribution and coordination of TIMED data. It is the means of communication chosen to meet the mission objectives in the area of public and educational outreach. The TIMED home page **(http://www.timed.jhuapl.edu)** provides the user with access to all the Mission Publications features: news, mission information, instrument and science team information, planning tools, TIMED data, and educational information about the TIMED project.

TIMED MDC Catalog Overview

The TIMED Mission Data Catalog provides the users with a summary and index of data products, a description of spacecraft and instrument configuration and performance information, and other information that is necessary to locate, understand and use TIMED data. There is no science data actually stored in the catalog database. It is a metadata catalog. It is a catalog of data related to the science data products. The catalog will aid the users in determining the data products they need and provide a URL to the actual file(s). The producers of the science data products are responsible for the storage and availability of all of their data products.

The types of data populated in the Mission Data Catalog are the following:

- Core metadata – Data Product Type information
- Associated metadata – Data Product Version and File information
- Spatial data – Spacecraft Position and Time, Orbit Number
- Special metadata – Timeline File data

All of these data types are correlated based on Universal Time (UT). Therefore, the user may query on any combination of the database fields to determine the data product(s) they desire.

TIMED MDC Catalog Implementation

An Oracle Relational Database Management System (RDBMS) is being used to store the TIMED Mission Data Catalog and support the distribution of the data product information. The Oracle RDBMS offers easy access to all data, flexibility in data modeling, reduced data storage and redundancy, and a high-level data manipulation language (SQL). In addition, Oracle offers design, development and reporting tools.

The TIMED Mission Data Catalog was designed using Oracle's CASE tool, Oracle Designer version 2.1.1. Oracle Designer is a Rapid Application Development-like, intuitive, task-oriented environment to model and generate server Data Definition Language, client/server and Web-based applications. The Entity-Relationship diagram, the logical view of the database, is shown in Figure 10. The Entity-Relationship diagram is used to define the information needs of the business. The diagram identifies the things of importance in an organization (entities), the properties of those things (attributes), and how they relate to one another (relationships).

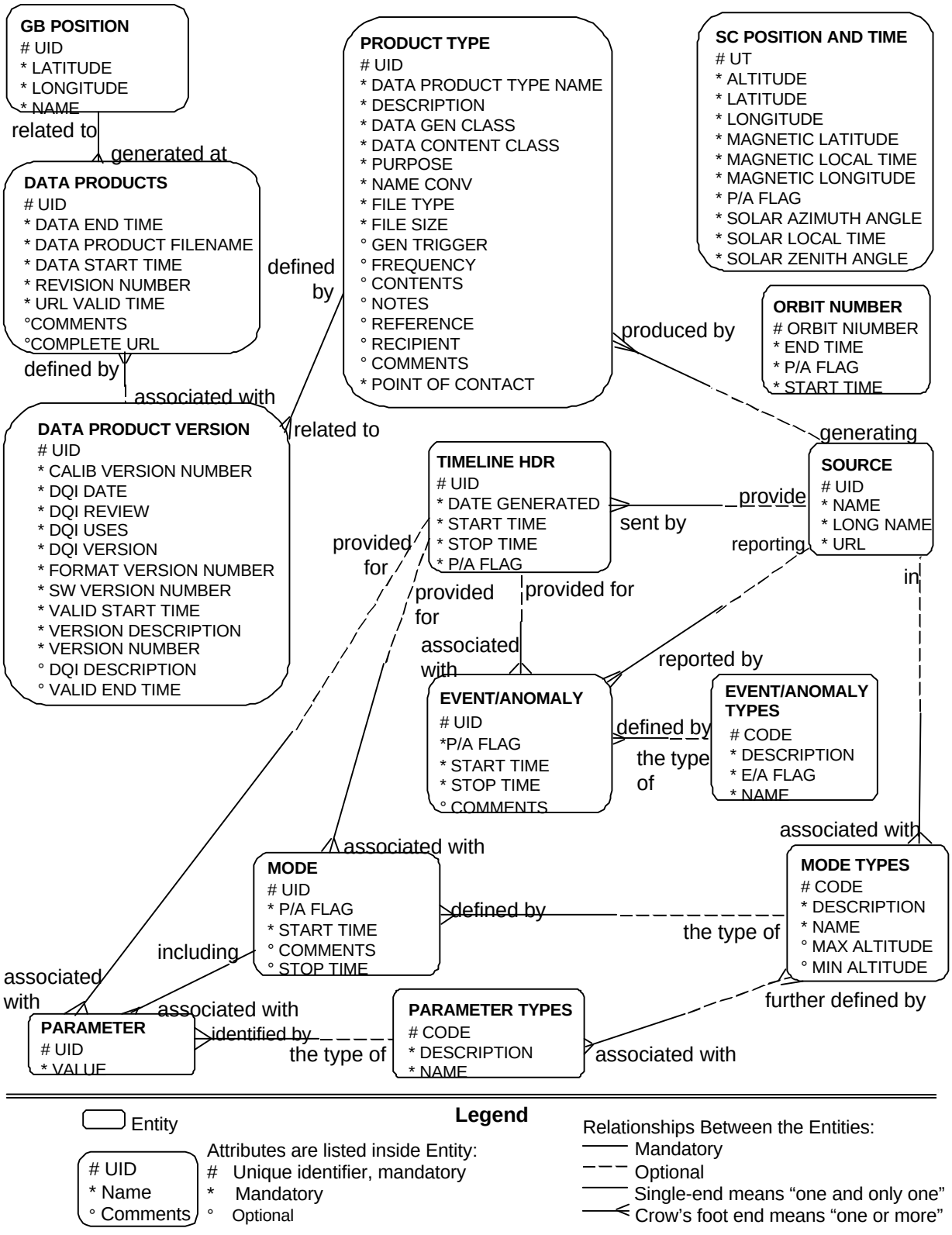

Figure 10. TIMED Mission Data Catalog Entity Relationship Diagram

Conceptual View as Tables

Oracle Designer was used to produce a normalized database design from the entity relationship model. In this particular design, each entity represents a single table. Within each table, the attributes identify the columns. And, the relationships determine the constraints for each table. The Oracle database allows the developer to identify key constraints (primary, foreign, and unique) and check constraints. These constraints enable the developer to define rules for data entry ensuring data integrity.

The TIMED Mission Data Catalog can be divided into three groups: Spatial Data Tables, Core and Associated Metadata Tables, and the Special Metadata Tables.

The Spatial Data Tables provide the predicted and actual spacecraft position and time data and the predicted and actual orbit numbers. The Core and Associated Metadata tables contain the metadata about the data products, both for the data product types and the specific data product files. The URL of each data product file is included to allow the users to retrieve the actual file if necessary. The Special Metadata Tables contain all of the timeline data. The timeline data identifies all of the planned and as flown modes, events as well as anomalies reported by the spacecraft and the instruments.

TIMED Mission Data Catalog User Interface

The users can access the TIMED Mission Data Catalog by one of three methods: online direct access, online pre-defined queries, and online ad hoc queries.

The users may directly access the data products produced by the MDC by entering the exact filename of the data product they desire. This will initiate a query of the TIMED Mission Data Catalog. If the file exists, the user will be presented with the URL of the requested file.

The users may access the TIMED Mission Data Catalog via pre-defined queries. All of the pre-defined queries are implemented using HTML forms for the web user interface, with Perl CGI scripts to connect to the database.
The pre-defined queries are:

1) Find a Data Product
2) Find Information by Time
3) Find Information by Location
4) Timeline Query
5) Instrument Parameter Query
6) Instrument and Time Query

All of these pre-defined queries allow the user to query the TIMED mission data product catalog using a GUI interface. The user enters specific values, clicks on radio buttons, and selects from menu lists. The first pre-defined query, Find a Data Product, and the associated results page is described below. The other pre-defined queries are similar.

Query Example

Find a Data Product – The user selection criteria includes the following items: Time, Location, Solar Zenith Angle, Solar Azimuth Angle, Timeline data, Product Type data, and Product Version data.

The results of this query are two tables of information and a review of the user's query definition. The first table contains information about each data product file that matched the user's selection criteria. The second table lists the specific time range covered by a particular data product file. The data product results table includes the following items, presented either as an HTML table or as a plain text table, as specified by the user: data product file URL, data product description, range of UT covered by the data product file, the composite data product version number, revision number, data quality version, data quality review, data quality uses, orbit numbers covered by the data product file, and the estimated maximum file size.

The users may access the TIMED Mission Data Catalog using ad hoc queries. The ad hoc capability is restricted to those users who submit a request to the SDS manager. Users who have been granted this access write their own SQL query. The results of their query will be shown in a table format.

Catalog Loading

The TIMED Mission Data Catalog will be populated from different sources and methods as illustrated in Figure 11. The data for the Spatial Data Tables are extracted from the Predicted and Actual PVAT and Orbit Number files that the MDC produces daily. The PVAT files contain spacecraft position data for every second; however, the spacecraft position data in the database will be at a minute resolution. This resolution satisfies the users' requirements in searching for data products and will significantly reduce the amount of data stored in the database. The data for three of the Core and Associated Metadata tables (Product Type, Data Product Version and Source tables) is entered and updated by the SDS

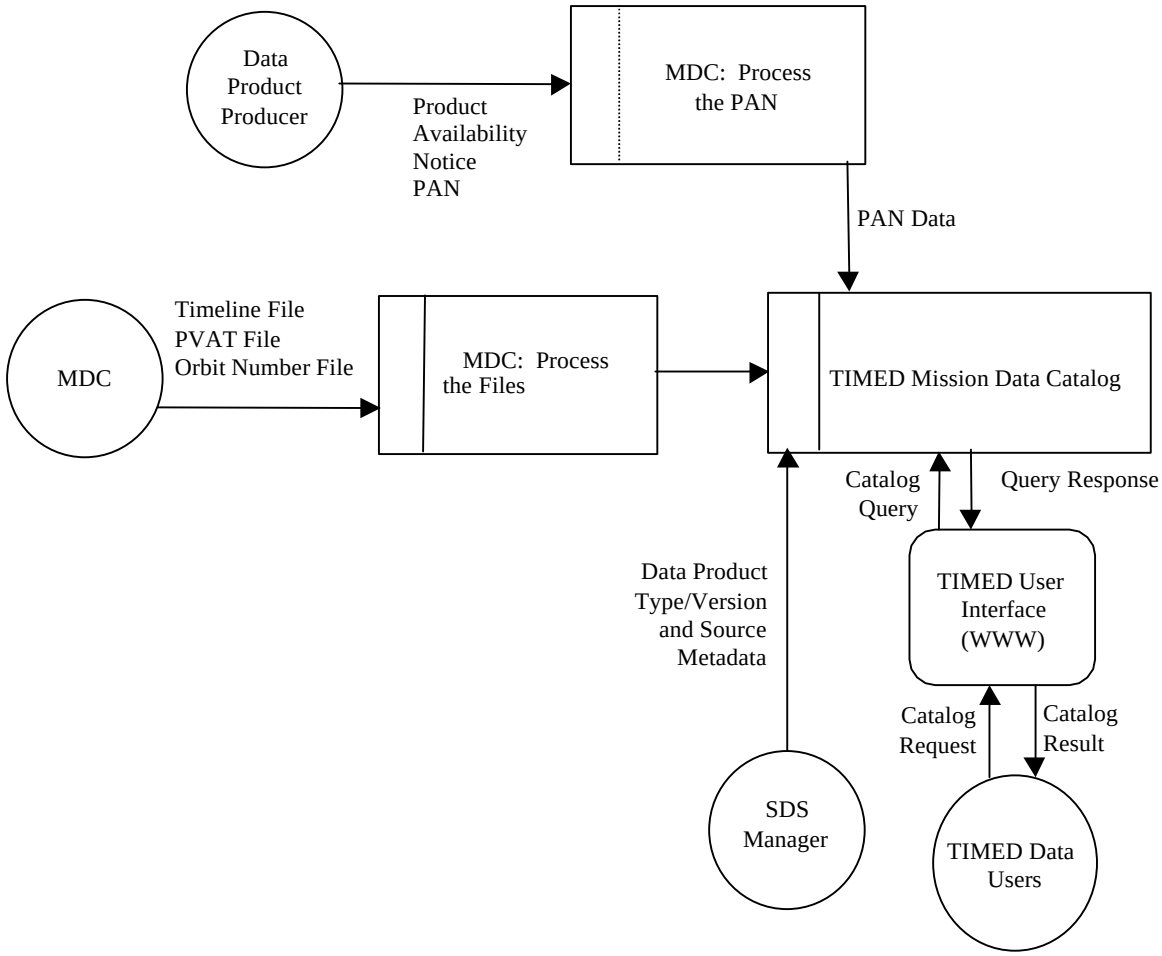

Figure 11. TIMED Mission Data Cataloging and Distribution

Manager. The SDS Manager is responsible for the validation of this data. The data for the other two tables (Data Products and GB Position tables) is entered automatically when the data product producers notify the MDC that a data product has been generated, updated, or deleted. The data product producers will ftp a Product Availability Notice (PAN) when a new data product has been generated, a data product has been updated, or a data product has been deleted/removed. The MDC software development team developed Perl scripts that are executed automatically when a PAN is sent to the MDC. The Perl scripts use the DBI module to interface with the Oracle database. The process for updating the Special Metadata Tables is similar. When the Mission Operation Center (MOC) or the Payload Operations Centers (POCs) generate and ftp a timeline file to the MDC, a Perl script is automatically executed to update the Special Metadata tables.

TIMED Mission Data Catalog Maintenance

The database engine for the TIMED Mission Data Catalog is the Oracle Server version 8.0.5, running on a Sun Sparc Enterprise 2 with the Solaris 2.6 operating system. The Oracle Server was installed with four drives, one for the Oracle software storage and three for the data storage.

The Oracle Recovery Manager Utility manages the backup, restore and recovery operations for the database. It is installed on a Sun Sparc Enterprise 450.

Lessons Learned

The MDC development team has successfully integrated the Mission Data Cataloging and Distribution file processing and the Mission Publication web query processing with the Oracle database. The user interface

for the query is complex; therefore, it offers a great deal of flexibility to a broad range of users.

Some of the lessons learned during the design and development of the Mission Data Catalog and User Interface include the following:

- Plan for the training of new languages, applications and tools.
- Plan for the upgrades to commercial products. This includes not only the time to perform the upgrade but also the time to learn the new features.
- Hold detailed design reviews and prototype demonstrations with the users. The system is being built for the users – it must satisfy their needs. Requirement documents are necessary, but the requirements will change and the design must be flexible to adapt to the modifications required.

Summary

The development of the databases for the MSX, NEAR and TIMED missions provided insights into the factors that determined their overall usefulness. To successfully create a catalog for a science data system the developers must achieve success in four key areas:

1. Translate the science goals into clear primary and secondary system requirements.

2. Develop a design which satisfies the requirements and is flexible enough to provide additional functionality as experience is gained.

3. Develop a user interface which provides easy access to the average user.

4. Produce a system which meets the goals and requirements within the performance and cost constraints of the mission.

Clear Goals to Requirements Translation - The users of a science data archive want to be able to quickly and easily find select mission data from a large, complex set of data products. Translating this goal into a clear set of system requirements is the responsibility of the catalog design team. While the primary or general requirements of the system are usually clear, the more detailed requirements are often not as well known, especially early in the design process. The system developer must capture as much of this information as possible early in the development process, then continuously refine and update the requirements. If user expectations and goals cannot be understood clearly enough by the development team, there is a high risk of user disappointment or frustration with the final system.

Flexible Design - Because not all the expectations of the science user can be translated into clear system requirements in the early stages of development, the database design must allow tailoring of the system to match the evolution of the detailed system requirements. The catalogs are often used as analysis tools whose capabilities must closely support the science user's research efforts in order to be successful. As the system developers gather a better understanding of the types of database queries and reports desired by the user, the database design must allow them to incorporate this additional functionality into the system. One risk is that changes in user requirements can force changes to the underlying database structures. These changes are costly and often not understood by the user. The database developer must creatively address these issues.

.

User Interface - User interfaces are critical to success. In all of these missions the development of the user interface was challenging. The full range of user queries is complex and the user is usually uncertain as to what specific queries will be required. The development of user interfaces can also be quite expensive. For the NEAR and TIMED missions the solution was the definition, via user input, of a limited set of most used queries plus development of a supporting WWW interface. This interface is satisfactory for the majority of accesses to the database. In other cases the need to make more complex, previously-undefined queries, was met by the development of an Ad-Hoc query interface. This interface makes a larger set of flexible, complex queries available to a limited set of users. Commercial database products are rapidly changing to provide better WWW interfaces. As these capabilities increase, future missions will have a wider range of options and less difficulty in custom user interface development.

Performance and Cost Constraints - The developer of the science catalog system must make trade-offs between the user requests for a highly functional, customized system with the imposed cost and performance constraints. To meet user goals the system must maximize access speed and accuracy, be kept as current as possible, and contain the maximum amount of science data. Accuracy is especially of concern in the representation and storage of time and ephemerides. To achieve this accuracy the system may be required to repeatedly update the ephemerides values. The system must then accommodate the accompanying reprocessing loading. The developer must use knowledge of the

research goals to limit the list of data items or query types in order to prevent the data catalogs from growing to the point of challenging system resources or delivering inadequate response. In these missions considerable time was spent in optimizing the system to decrease the time required for loading and updating the catalog. The developer must also understand the science requirements well enough to design a schema which maps the logical representation of the data into an internal database structure. This structure must support the science query requirements while limiting the size of the system to avoid increased maintenance costs and slower access speeds.

Conclusion

In all of these missions the databases were developed to aid the science users in finding select mission data from a large, complex set of data products in the mission archives. The databases catalog large amounts of complex information about the mission data in a way which allows users to flexibly answer their questions about what data is where. The data catalogs combine information about the science data with position, pointing information, descriptive data and data quality information. The catalogs give the user the ability to find the science data needed for research by linking the items above to the specific data products.

Acknowledgements

Thanks to Cheryl Hill, Carol Trimper , Marjorie O'Riordan, and Marghi Hopkins for manuscript support and review.

References

1. Midcourse Space Experiment: Overview, Johns Hopkins APL Technical Digest, January-March 1996, Volume 17, Numbers 1 & 2.

2. The NEAR Mission, Johns Hopkins APL Technical Digest, April-June 1998, Volume 19, Number 2.