International Hydrographic Review, Monaco, LXXVI(2), September 1999

BUILDING A CHS BATHYMETRIC DATA WAREHOUSE

S.R. FORBES, R.G. BURKE, C.E. DAY and H.P. VARMA [1]

Paper presented at the Canadian Hydrographic Conference, 10-12 March 1998, Victoria, British Colombia.

Abstract

The CHS is designing and implementing a Source Database for bathymetric data. To store and manage terabytes of bathymetric information for producing CHS products and providing services to clients requires a fresh approach. The implementation of HHCode (Helical Hyperspatial Code) for storing, analyzing and accessing dense spatial data sets in an Oracle Relational Database Management System (RDBMS) is a key technology used to build the data warehouse.

The paper addresses the concepts and applications necessary to manage this data within the data repository. Specific areas addressed are aggregation, attribution, and partitioning of the spatial data. The implementation of an on-line, near-line and off-line model to minimize storage costs is described. The approach used to perform horizontal and vertical datum transformations is discussed.

INTRODUCTION

During the past decade the Canadian Hydrographic Service (CHS) has adopted new digital bathymetry data-collection systems. These systems provide the capability for 100% sea floor coverage. Systems such as the Simrad EM3000 Multi-beam Swath Sounder have the potential to gather approximately 2800 million depth measurements during a three-month survey. This translates to approximately 280 Gigabytes of information to manage. Five similar EM3000 surveys in one survey year can gather in excess of one terabyte of data to manage. This volume of data requires new techniques for the aggregation, management and query of spatial information.

[1] Canadian Hydrographic Service, Atlantic Region.

The CHS is addressing this problem by designing a Source Database to manage the bathymetric information using a Relational Database Management System (RDBMS). The data warehouse addresses requirements to aggregate, partition, and perform vertical and horizontal datum transforms for bathymetric information. The design of the database prototype and implementation described in this paper extends the Oracle Spatial Data Option (SDO) product developed by Oracle Corporation with some spatial cartridge components. These additional components are required to manage very large volume bathymetric data within a data warehouse environment.

The CHS has been investigating the management of spatial data in the RDBMS environment since the mid-1980s. The development of a spatial data type called HHCode[2] (Helical Hyperspatial Code) by the CHS during 1989 facilitated the encoding and manipulation of spatial data (multiple dimensions) using a single key. This technology was transferred to Oracle Corporation, Redwood, California. and incorporated in the Oracle RDBMS Version 7.1.3. The product was released in March 1996 and is marketed as Oracle SDO.

BUILDING BLOCKS FOR THE CHS DATA WAREHOUSE

The design and implementation of the CHS data warehouse addresses the following primary components[3] for managing the spatial data:

1. Acquisition — Includes the applications required to support data acquisition, data conditioning, data import and data export.
2. Data Warehouse — The physical database using the Oracle RDBMS and the infrastructure required to manage multi-terabytes of bathymetry.
3. Access — The suite of applications required to query and use the information stored in the warehouse to meet CHS business needs.

Specific building blocks to support these components include data conditioning, data aggregation, data partitioning, data loading/data archiving, warehouse-backup and data query. This includes the proprietary data structure developed for the storage of spatial data and attributes defined as the Spatial Data Structure (SDS). These fundamental components are necessary to build the data warehouse.

[2] VARMA, H.P., H BOUDREAU, and PRIME, W., A Data Structure for Spatio-Temporal Databases, IHO Review Monaco LXVII(1), January 1990.

[3] MATTISON, Rob, Data Warehousing Strategies, Technologies and Techniques, McGraw-Hill,1996, page 9.

DATA CONDITIONING

The CHS has a large collection of historical bathymetric information that is in analog form. Data conditioning for non-digital information requires locating all required meta-data, associated data and capture of this data into digital form. Data verification, datum transformation (if necessary) and conversion to the SDS are carried out in preparation for database entry.

Conditioning digital information requires data extraction from legacy systems. Data cleaning, verification, datum transformation (if necessary) and conversion to the SDS structure are required for entry into the database. All relevant meta-information regarding the data sets must also be located. Mandatory data attributes, for example, accuracy and quality, must be computed or estimated.

DATA AGGREGATION

CHS has investigated and developed algorithms based on HHCodes to aggregate[4] dense bathymetric data sets. An example of a candidate data set for aggregation is one acquired using the Simrad EM3000. Aggregation of bathymetric data sets involves creating a single HHCode cell that contains multiple smaller cells. The criteria for the single HHCode cell (tile) is based on a user defined criteria. For bathymetric information this is the maximum tolerance defined for a given depth.

i.e. Maximum depth - Minimum depth $\leqq$ User defined tolerance

If the tolerance is not met then the tile is subdivided until the tolerance criteria is met. This produces tiles of varying size that can be viewed as a Digital Terrain Model (DTM) of two-dimensional tiles.

This technique aggregates the data set, preserves user defined depth resolution and minimizes the data that must be loaded or queried. Preliminary tests using this algorithm aggregated seven million depths from an EM3000 survey in six minutes and reduced the data storage requirement by a factor of 9.33.

DATA PARTITIONING

Management of large volume spatial data requires new techniques for storage and retrieval of the information. A poorly predicted growth rate using a traditional data management approach may require the database to be re-

[4] Application developed for the CHS by M. McConnell Information Technologies, Lunenburg, Nova Scotia.

engineered and the data periodically reloaded. Data partitioning[5] alleviates the requirement to accurately predict the data growth rate. Oracle SDO uses this technique to store the data in multiple partitions (tables) that dynamically and automatically subdivide when the data is loaded or updated.

The number of partitions created is based on the data density. The partition size is defined by the maximum table size that is determined by specifying a High Water Mark (HWM)[6]. This HWM limits the number of records contained in a partition before it sub-divides into another set of partitions. The optimum partition size is dependent on the data density and the required query response time.

Two-dimensional partitioning is illustrated in Figure 1 as a quadtree structure. Three-dimensional partitioning is represented by an octree structure shown in Figure 2.

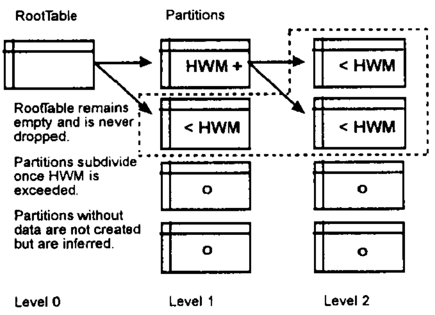
RootTable
Partitions
HWM +
< HWM
< HWM
< HWM
RootTable remains empty and is never dropped.
Partitions subdivide once HWM is exceeded.
Partitions without data are not created but are inferred.
o
o
o
o
Level 0
Level 1
Level 2

FIG. 1.- Two Dimensional Partitions.

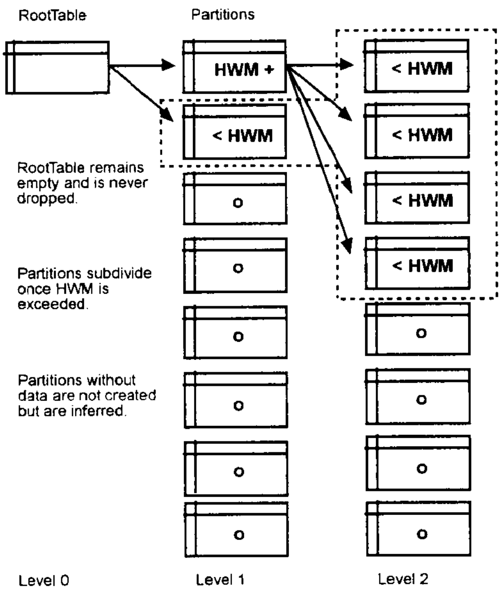
RootTable
Partitions
HWM +
< HWM
< HWM
< HWM
< HWM
< HWM
RootTable remains empty and is never dropped.
Partitions subdivide once HWM is exceeded.
Partitions without data are not created but are inferred.
o
o
o
o
o
o
o
o
o
Level 0
Level 1
Level 2

FIG. 2.- Three Dimensional Partitions.

The primary benefit of data partitioning is to ensure consistent and predictable query performance for a very large database (VLDB) where growth rate is not easily predicted.

The partitioning of data sets facilitates database access by localizing the data. It also allows users to place partitioned tables off-line in files when the data is inactive. Only data that is active remains on-line, minimizing the disk storage required. For very large data warehouses it is advantageous to place information

[5] Varma, H., Boudreau, H., Prime, W., Galluchon, M., Dahlgren, G., MacDonald, L., Spatio-Temporal Database Implementation and Functionality using HHCode, 1991 Proceedings US Hydrographic Conference.

[6] Oracle7 MultiDimension User's Guide, Version 1.3.2, May, 1995.

off-line on lower cost storage media when the data has not been queried or updated for a specific period of time.

The partitioning of the database is implemented by the use of virtual tables. The virtual table is composed of a union of all virtual partitioned tables as shown in Figure 2. In a three dimension partition schema the area and time is represented as a space time cube.

When a partitioned table (three dimensions) approaches saturation level (the user defined HWM) then eight new temporal cubes are created. The data from the original partitioned table is inserted into the eight new tables and the old table is dropped. The information about the partitions is stored in a root table and only partitions that actually hold data are created.

Each of the new tables can hold data up to the HWM. If a new partitioned table exceeds the HWM, it subdivides into eight new tables. The dictionary only maintains the virtual partition names that have data in them. The partition names with zero elements are computed by an inference routine from the existing partition names. In Figure 2 the dashed box indicates the partitions that exist after loading is complete.

The partitioned tables can be viewed as space-time buckets, containing the data and times of collection. The partition names, data paths and other relevant information are maintained by the RDBMS data dictionary.

DATA LOADING/DATA ARCHIVING

The purpose of the Archiver module is to store data in directories as standard files. The Archiver supports tiled and non-tiled bathymetric information. These files consist of partitioned SDS files using partitioning techniques similar to those implemented in the Oracle SDO product. If the data is not required on-line, the loading process becomes a registration process of data partitions in the data dictionary and the archived data is placed on-line when needed. The Archive Load module supports the capability to read the SDS files and consolidate the incoming data with the archived data. The module generates new partitions or tiled partitions to be placed in the archive. It also maintains the dictionary for data partitioning and archiving of very large data sets. The Archive software module is composed of two sub-modules, the Archive Loader and the Archive Import/ Export sub-module.

The Archive Loader process is shown in Figure 3. The data is tessellated (tiled using data aggregation), if required, consolidated with the existing data and partitioned. The partitioned SDS file is registered in the Oracle SDO database and the data is stored in a user defined directory. The non-tiled SDS files reside in one directory and the tiled SDS files are stored in a separate directory.

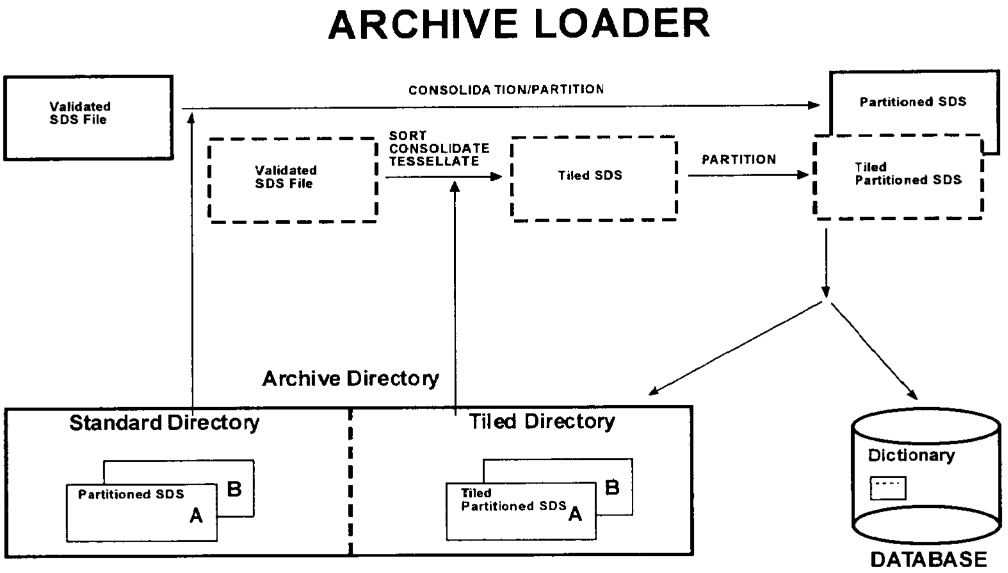

FIG. 3.- The Archive Loader Process.

The Archive Import/Export module places partitions on-line when users require data access. It moves partitions to near-line or off-line storage when on-line data access is no longer needed. The Archive Import/Export process is shown in Figure 4. The Archive Export module extracts data from the database and places the data in the archive directory. Conversely the Archive Import module extracts data from the archive directory and places the data in the database.

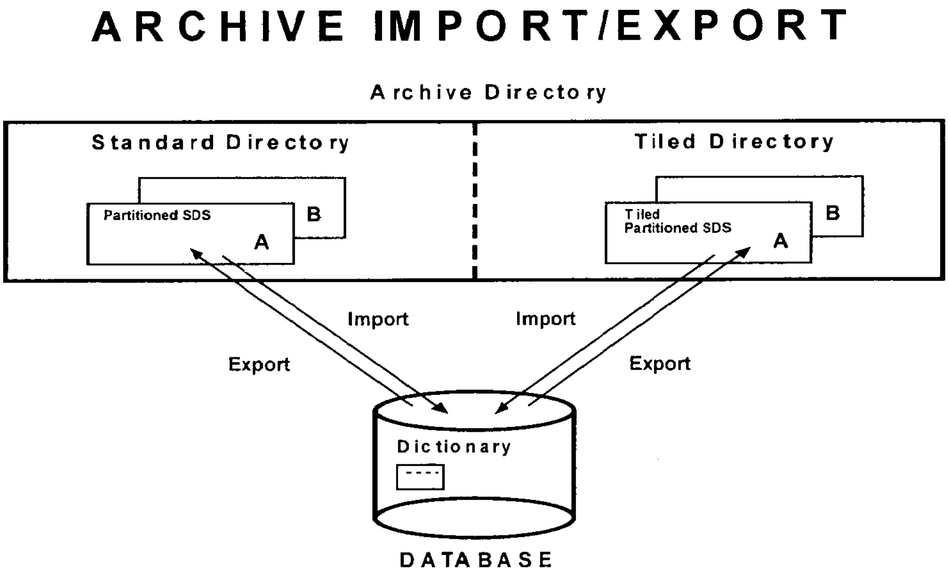

FIG. 4.- The Archive Import/Export Process.

There are three states of data, on-line, near-line and off-line. The expected query response time is defined as follows:

Data Category	Location	Expected Response Time
On-line	Oracle database	Less than 1 minute
Near-line	Files on magnetic storage, Files on magnetic/optical storage silos	Less than 2 hours
Off-line	Magnetic tape or optical archives	Less than 24 hours

The partitioned tables are stored on disks, optical disks and robotic data servers using optical disk or magnetic tape storage media. A spatial query by a user causes those tables referenced in the data dictionary to be brought on-line automatically. If the data is off-line, the system prompts the operator for the appropriate tapes, optical disks or other storage media. If the data is near-line (on disk or jukebox), the system automatically loads it into the database, since the path to the data is also maintained in the dictionary. When the data is on-line, the system creates the appropriate view for the user.

In theory, the database may be empty until the user queries the system and then the appropriate tables are brought on-line. If users do not require data access the tables (SDS files) can be stored near-line or off-line in the archive.

The on-line database size is constrained by the physical table space (including rollback segments) available on disk. The physical size of the database warehouse is constrained only by the physical limitations of the combined on-line, near-line and off-line storage. A typical initial on-line size of 60 Gigabytes of disk, near-line size of 1 to 5 Terabytes and unlimited off-line storage allows the database to grow as required. As the near-line and off-line data growth continues the on-line storage may be upgraded to facilitate query response and/or the number of clients.

Near-line robotic storage is not a mandatory requirement and the warehouse design does not constrain implementation of a warehouse that uses only on-line storage. The objective is to organize the data for cost effective data storage, efficient access and to prevent overloading of limited resources.

DATA BACKUP

The design of the Archive module allows the database to be backed up in a manner similar to a traditional server environment. Full and incremental backups may be performed on all the data contained in the archive. If there is no requirement for data mirroring, the backups may be retained off-line. In addition the on-line database may be backed up frequently (weekly) by the near-line storage to provide a snapshot backup.

The optimum formula for determining the full versus incremental backup is dependent on the following factors:

1) Growth rate of the data in the warehouse.
2) The update frequency the business requires for the data held in the warehouse.
3) The minimum acceptable time required to restore the data archive from the backup.

Device shadowing or mirroring are alternative backup solutions for data warehouse implementations which require fault tolerance. This approach will double the required on-line storage.

DATA QUERY

The warehouse access will support three categories of products and tools:

1) Ad hoc query interfaces	to include query managers, data browsers and report generators.
2) Custom applications	Spatial interfaces specific to the discipline of hydrography for querying temporal spatial data.
3) Autonomous applications	existing applications within the CHS or supplied by third party vendors i.e. Computer Aided Resource and Information System (CARIS), Geographic Information System (GIS).

IMPLEMENTATION OF TIME LINES

Each sounding or tile in the data warehouse has a temporal reference. The management of bathymetric information requires the implementation of time lines to address the temporal nature of the information. The information may become obsolete for many reasons but the most common in the hydrographic discipline include:

1) A physical change in the sea floor topography due to natural phenomenon such as ice scouring, silting, erosion, etc.
2) A physical change in the sea floor or land sea interface due to man made influences such as dredging, building of structures, etc.
3) Periodic resurvey of critical marine areas such as harbours, with improved technology i.e. 100% bottom coverage acoustic data acquisition systems and the Global Positioning System (GPS).
4) The Hydrographic Office is notified of a hazard to navigation.

The sounding attributes include the database entry time, the database update time and the database termination time. The entry time confirms the validity of the data. The update time indicates an update to an instance or instances of the bathymetry. The termination time ends the validity of a single or multiple instances of bathymetry. This time also declares instances of bathymetry no longer valid without explicitly removing these data from the data warehouse.

These time lines facilitate analysis and query of data in the warehouse. The specific functions that may be used are:

1) Comparison of data in the same spatial area in the warehouse for data verification purposes.
2) Extraction of the most recent bathymetric information.
3) Extraction of bathymetric information in the same spatial area from different epochs to support trend and prediction analysis. For example, the silting rate in an estuary may be established from several bathymetric data sets collected in successive surveys. Based on this analysis a predictive model may be developed to determine when the area is resurveyed and dredged.

IMPLEMENTATION OF HORIZONTAL AND VERTICAL DATUM TRANSFORMS

Spatial information, such as bathymetry, which requires a vertical and horizontal datum reference, may constrain the flexibility of the warehouse for data entry and data delivery to the client because:

1) A decision must be made to normalize all data to one horizontal and vertical datum prior to entry into the data warehouse.
2) Delivery of data to the client is limited to these normalized datums and all data transformations are done external to the data warehouse.

The disadvantage to this approach is all the information must be referenced to one horizontal and vertical datum. The clients querying the spatial information may require the data to be referenced to a different datum than was referenced to store the source information. For example, to compare and verify historical bathymetric data with a modern survey would require the historical data to be transformed to the datums on which the new survey is referenced. As the warehouse grows to terabytes of data it is impractical to unload, transform, and reload the data into the warehouse to address the new datum references required to satisfy the majority of client queries. Also, an error in the datum transformation application would compromise the integrity of the source bathymetric data.

An alternative approach is shown in Figure 5. On data entry into the warehouse an instance of bathymetry is referenced to a defined horizontal and vertical datum. This permits the warehouse to store bathymetry referenced to different datums. The bathymetry is locked to the reference datums at data entry time and all datum transforms take place during data query.

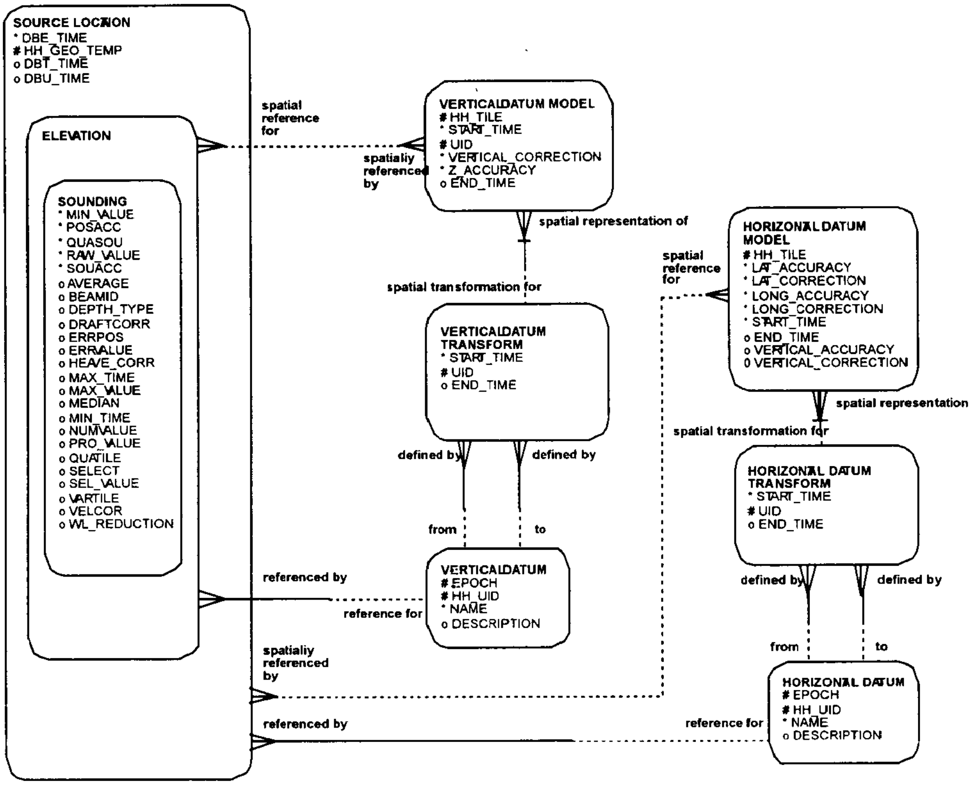

FIG. 5.- Vertical and Horizontal Datum Transform Model.

To query the database the client must specify the destination horizontal and vertical datum for the bathymetry. The data may be retrieved referenced to the datum established during data entry or may be transformed to a client-specified datum. To transform the data it is necessary that a valid datum transform is defined to transform *from* the source datum *to* the destination datum. In addition a valid datum transform model must also be defined. This transform model is a DTM with the appropriate transform coordinates.

An example of a vertical transform model is shown in Figure 6. The bathymetric DTM (*left*) with the bathymetry value in decimeters is shown for each area, and underneath this value is the unique area reference identifier. The datum adjustment DTM (*centre*) is illustrated with the vertical datum adjustment in decimeters and its unique area identifier. The adjusted bathymetry DTM (*right*) is derived by simply adding the bathymetric values and the datum adjustment model values. For example, area 12 with a depth value of 62 is added to datum adjustment model value of -3 producing a new depth value of 59 for area 12. Note in the figure, bathymetry area 13 has been subdivided into four new areas to match the areas´

defined for the datum adjustment model (*areas 130 to 133 inclusive*). If the bathymetry source area is larger than the datum adjustment model then the area must be subdivided until it matches the resolution of the datum adjustment model areas.

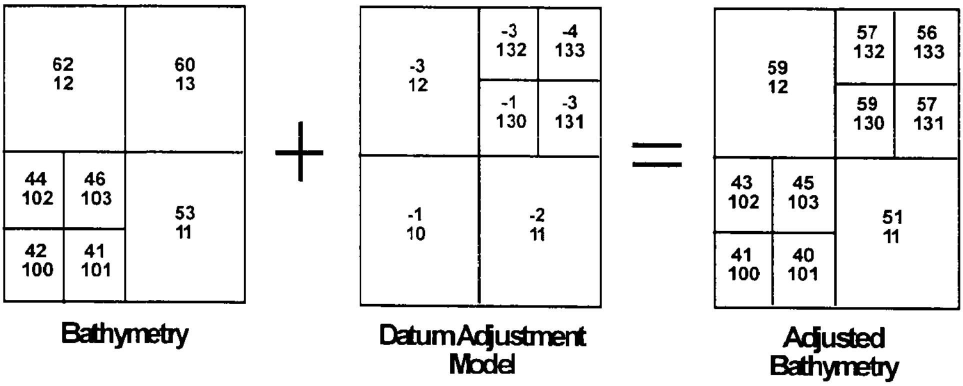

FIG. 6.- A Bathymetric Vertical Transform Model.

A similar approach is used for horizontal datum adjustments. The horizontal datum adjustment model has a datum adjustment value for latitude and longitude (x and y) for each area.

In conclusion this design supports the storage of bathymetry referenced to several datums with the query and extraction of the bathymetry on one or more client defined datums. The bathymetry is referenced to a known datum, verified and entered into the data warehouse. The datum transforms are performed when the client queries the data warehouse. This approach ensures the integrity of the source bathymetry.

SPATIAL DATA WAREHOUSE PROTOTYPE

A prototype[7] of the Spatial Data Warehouse was developed and tested in October 1997. The purpose of the prototype was to:

1) Verify the SDS data structures are appropriate and meet the performance requirements for very large spatial data sets.
2) Demonstrate fast access to the large volume of spatial data in the RDBMS, without the need for large-scale servers. The scalability of the spatial warehouse on a variety of classes of hardware architectures is an important design feature.

[7] Developed for the CHS by NDI Data Warehouse Technologies Inc., 200 Montcalm, Hull, Quebec.

3) Demonstrate that the data aggregation and area portrayal of the bathymetric data can be managed in the relational environment. The prototype demonstrated that the point data sets and the area aggregate data sets could co-exist in the data warehouse using the same spatial data architecture (SDS).
4) Verify the archive approach to minimize storage costs is a practical solution. It is important to show that the archive approach addresses minimizing the cost of data storage and that the length of time needed to be import the information into the data warehouse from the archive is acceptable.

The environment used to demonstrate the prototype included:

1) An Alpha Server with a single 300 MHz processor and 256 Mbytes of memory with OSF/1 Version 3.2 UNIX operating system.
2) Three Seagate 9 Gbyte disk drives with a wide Small Computer System Interface.
3) Oracle RDBMS Version 7.3.2 with the Spatial Data Option and spatial data cartridge components.

The prototype test successfully demonstrated the design criteria was easily met or exceeded. The complete data archive was represented by approximately 12 million data points. Specific tests and results for this environment included:

- 1,000,000 bathymetry data points were loaded in 3 minutes and 30 seconds.
- 125,000 bathymetric tiles representing the same area were loaded in 30 seconds.
- 1,000,000 bathymetric data points were queried and retrieved from the data warehouse in 50 seconds.
- 125,000 bathymetric tiles representing the same area as the referenced data points were queried and retrieved in 10 seconds.

The tests also indicated that the spatial load and query performance was linear for the load and access times versus the number of data points.

SPATIAL DATA WAREHOUSE BENCHMARKS

To test the scalability of the data warehouse it is necessary to performance test the application on a suite of low to high end servers from a single manufacturer. The benchmarks will include a full warehouse configuration with data storage architecture. Data storage components include hard disk and near-line tape jukebox technology. It is anticipated the initial benchmark tests will be run in the second quarter of 1998 (April to June) on 64 bit architecture servers. These benchmarks will include performance tests for sorting, data aggregation, loading data into the data warehouse, querying and retrieving data from the data warehouse. The benchmark results will determine the scale of server CHS will choose for national implementation of the data warehouse.

SPATIAL DATA WAREHOUSE BENEFITS

The benefits derived from the warehouse are dependent on the value propositions. The value propositions are the proposals made to solve a business problem using the data warehouse. The value propositions for the bathymetric data warehouse are:

1) Integration of bathymetric source information from a variety of acquisition systems using proprietary formats into a common data structure to ensure the data is readily accessible to the client. The bathymetric data is independent of obsolete applications and data formats.
2) Spatial indexing (HHCode) of the bathymetric source data accelerates temporal spatial queries. This minimizes the time and compute resources required to satisfy the client's query.
3) The bathymetric data warehouse will facilitate the comparison of survey data sets that are collected in the same geographic area. The verification of the bathymetric data is enhanced.
4) The warehouse will facilitate the management of CHS source information. This will reduce the time required to produce CHS products and services.

The design of the warehouse is fundamental to the fulfillment of the value propositions. The warehouse design features to support the value propositions are:

1) Scalable The warehouse is designed to grow as the data storage and compute resources grow. The design is not compromised by inaccurate data growth rate estimates. The compute and storage resources are scaled to the data storage requirements without loss of initial infrastructure investment.
2) Data export The warehouse is designed to export data in the formats and data structures necessary to provide feedback information to legacy systems.
3) Integration The integration of the warehouse applications with existing applications under development is continuously addressed with the custodians and developers of the applications.
4) Data access The utilization of spatial indexing, data aggregation and data partitioning enhance the performance and minimize the on-line storage requirements for the data warehouse. The ability to predict the response time for data queries is an important design feature of the data warehouse.

5) Backup The backup of data warehouses using data storage mirroring is an expensive solution which is justified for applications which require fault tolerant systems. The bathymetric data warehouse design is flexible and will support fault tolerant environments or traditional server backup solutions.

The tangible benefits of the warehouse cannot be assessed and measured until the system is placed in production. The strategy to minimize risk is to build applications based on the value propositions in a serial manner. Each application must add a tangible benefit for the organization.

CONCLUSION

The CHS bathymetric warehouse is being developed using new techniques and technologies to convert, aggregate and manage bathymetric information to enable the CHS to produce products and provide services. A prototype has been developed and tested to resolve functional and operational issues and to test assumptions made during the development. A benchmark will be developed to determine the hardware and software necessary to meet the anticipated workload while taking into consideration the available resources and acceptable performance criteria. The benchmark will also be used to determine the scalability of the application on different server configurations. CHS Regional implementation of the bathymetric warehouse is planned for 1998.

References

BARKER , R., CASE*Method Tasks and Deliverables, Addison Wesley, 1990.

BARKER, R., CASE*Method Entity and Relationship Modelling, Addison Wesley, 1990.

BARKER, R., CASE*Method Function and Process Modelling, Addison Wesley, 1992.

FORBES, S., BURKE, R., DAY, C., VARMA, H., CHS Source Data Base Strategy Report, Volume I (Internal DFO Report), April, 1996.

FORBES, S., BURKE, R., DAY, C., VARMA, H., CHS Source Data Base Analysis Report, (Internal DFO Report), April 30, 1997.

MATTISON, Rob, Data Warehousing Strategies, Technologies and Techniques, McGraw-Hill,1996.

Oracle7 MultiDimension User's Guide, Version 1.3.2, May, 1995.

VARMA, H.P., H Boudreau, and Prime, W., A Data Structure for Spatio-Temporal Databases, IHO Review Monaco LXVII(1), January 1990.

VARMA, H., BOUDREAU, H., PRIME, W., GALLUCHON, M., DAHLGREN, G., MACDONALD, L., Spatio-Temporal Database Implementation and Functionality using HHCode. 1991 Proceedings US Hydrographic Conference.