ADVANCES IN APPLYING DATA WAREHOUSES TO SEISMIC CALIBRATION RESEARCH

Julio C. Aguilar-Chang, Richard J. Stead, and Michael L. Begnaud

Los Alamos National Laboratory

Sponsored by National Nuclear Security Administration
Office of Nonproliferation Research and Engineering
Office of Defense Nuclear Nonproliferation

Contract No. W-7405-ENG-36

ABSTRACT

The Ground-based Nuclear Explosion Monitoring Research & Engineering (GNEM R&E) program has made recent advances in applying data warehouses to seismic calibration research. Some of the most challenging tasks of maintaining functional data warehouses are the development of software to automate the continuous population of the database, the quality control needed to resolve data conflicts, the synchronization of database tables between unclassified and classified warehouses, and the integration of all data sources into a cohesive database for delivery to the Knowledge Base.

Calibration efforts by Los Alamos National Laboratory (LANL) researchers require working with three separate data warehouses: unclassified at LANL, classified at LANL, and classified at the Air Force Technical Applications Center (AFTAC). It is therefore necessary to maintain three data warehouses that are physically unable to communicate with each other. Unclassified data are common to all the data warehouses, and some classified data are common only between the two classified warehouses. While it is relatively simple to add new data to all warehouses, it is difficult to capture changes made in one and then propagate them to the other two. We have recently developed a procedure based on database triggers to capture these changes. These triggers capture all update, insert, and delete operations against a predefined set of tables. Periodically, the information captured by these triggers is moved to the other environments and executed, thus keeping the warehouses synchronized.

One important challenge in using these large data warehouses is the simple and efficient access to the vast holdings in them. Web-based tools are important assets that address this problem. We have developed new web-based tools that enable researchers to do tasks such as track the progress of seismic analysis, access information about stations, handle logistical tasks (i.e., assignment of unique identifiers, and tracking the description and resolution of data problems identified through quality controls), and fast access to database metadata (e.g., schema descriptions).

Advances in easy access to metadata are supporting many of the higher-level efforts in quality control, automation, and web access. The first of these is the documentation of the seismic calibration schema using a database schema. This schema is designed to represent all of the detailed table and field information that, up until recently, has been available only in text-based documents. Such information in database form has immediate application to a wide variety of efforts involving the database, including those mentioned above, as well as future applications (see presentation at this meeting by Ballard, *et al*., on the subject of DBTools). Another advance in using metadata, with a more narrow application, has been the creation of bulletin descriptive tables. These tables describe the sources of bulletin data that have been imported into the data warehouse, as well as providing a means to track individual data elements to the corresponding lines of text in the original document.

As data become more voluminous and complex, quality control (QC) has become an increasingly visible and difficult issue regarding the Knowledge Base. Improvements in QC procedures are helping researchers and data managers to more readily identify complex quality problems. The outcome is improved research products resulting from improved data upon which those products are based. The improvements GNEM R&E has made in this area have even had modest impact outside the program; for example, in helping the International Seismological Centre (ISC) identify some problems with their data products. As we understand the QC problem in more detail, we have begun to automate the process of applying QC to large datasets.

OBJECTIVE

The Ground-based Nuclear Explosion Monitoring Research & Engineering (GNEM R&E) program has made recent advances in applying data warehouses to seismic calibration research. Some of the most challenging tasks of maintaining functional data warehouses are the development of software to easily access the contents of the data warehouse, the quality control needed to resolve data conflicts, the synchronization of database tables between unclassified and classified warehouses, and the integration of all data sources into a cohesive database for delivery to the Knowledge Base. This paper is a brief introduction to the wide range of data management technical issues that we face everyday, the progress we have made so far, and the future work needed to fully address all aspects of managing and handling vast amounts of data in a data warehouse that is used in explosion monitoring research.

RESEARCH ACCOMPLISHED

Database synchronization - capturing data changes

Calibration efforts by Los Alamos National Laboratory (LANL) researchers require working with three separate data warehouses: unclassified at LANL, classified at LANL, and classified at the Air Force Technical Applications Center (AFTAC). It is therefore necessary to maintain three data warehouses that are physically unable to communicate with each other. Unclassified data are common to all the data warehouses, and some classified data are common only between the two classified warehouses. While it is relatively simple to add new data to all warehouses, it is difficult to capture changes made in one and then propagate them to the other two.
We have recently developed a procedure based on database triggers to capture certain changes made to registered database tables. This procedure has been implemented at the beta-testing level and is currently being used in our production database. Results to-date have been satisfactory.

The changes currently being monitored on a pre-defined set of tables are data inserts, updates, and deletes. This process is referred to as Capture Data Changes (CDC). The acronym, as well as the fundamental idea, is similar to Oracle's Change Data Capture method of implementing the incremental recording of data changes. The main difference between Oracle's implementation and ours is that our method does not depend on a particular version of the Oracle Relational Database Management System (RDBMS). Oracle's CDC method is directly tied to a specific application that must be installed, configured, and run against an Oracle 9i database. Our procedure is entirely based on database triggers that are available on any version of the Oracle RDBMS, thus our implementation is not tied to any particular version of the Oracle database.

The concept is simple. A database table called SYNC_TABLES is created with the list of tables to be monitored. This table contains the schema the table belongs to and the table name. A pre-defined set of SQL scripts is run once at the beginning of the set-up process to create several objects. The first set of objects are the insert/update and the delete triggers on the monitored tables defined in the table SYNC_TABLES. These triggers capture unique information about a row being inserted, deleted, or updated. Another SQL script reads from the SYNC_TABLES table to create tables called CDC_*tablename*_DELETES. These tables record unique information about the rows that are being deleted from the corresponding monitored source table. For example, the unique information needed to identify which row has been deleted from an ORIGIN table is different than that needed to identify a row being deleted from an ARRIVAL table. We are not saving the entire row being deleted, but rather only its unique information is stored in the database, which will later be used during the synchronization operation to identify the row to be deleted from the database being updated. The space savings are significant with our implementation. As an example, the average row length in the CDC_ARRIVAL_DELETES table is 24 bytes, while the average length of an entire deleted row from a typical ARRIVAL table is 110 bytes.

A table called CDC_UPSERTS is also created only once during the set-up process. This table contains basic information such as the schema name, the table name, and the *rowid* for each row that has been either inserted or updated in the monitored table. The *rowid* is a built-in Oracle identifier that uniquely identifies a row in a table. With this information we know exactly which rows have changed over the monitoring time period, allowing us to recreate and apply those changes to the target tables being synchronized.

It is important to note that both types of tables, the CDC_UPSERTS and CDC_*tablename*_DELETES, also capture the modification date and the name of the database user who made the modification to the monitored tables. This

information, together with the standard database auditing capabilities that can be enabled at the table level, can serve a second purpose of providing a security audit trail to be able to answer questions regarding changes made to critical production data.

The synchronization operation between the unclassified (source) database and the classified (target) databases is a manual process at this time. The synchronization operation starts after a pre-determined number of changes have occurred in the source database tables. The source tables are put in read-only mode for a brief period of time, usually around 15 to 30 minutes. A group of tables called CDC_*tablename*_UPSERTS are created during this time period. The term **upsert** is commonly used among Oracle users to refer to **up**date and in**sert** operations together. These are the tables that contain the actual table structure and changed rows of data from the source tables that will be used to replace the out-dated data in the target database. The CDC_*tablename*_UPSERTS and CDC_*tablename*_ DELETES tables are moved from the unclassified system to the classified environment and used to synchronize the databases. At this point in time, the CDC_UPSERTS and CDC_*tablename*_ DELETES tables in the source database are truncated (i.e. their contents deleted) so that they can be reused, and the CDC_*tablename*_UPSERTS tables are dropped from the source database in order to reclaim valuable disk space.

The final step of the synchronization process is done in the classified environment, where the target database resides. There are only two steps involved here: first we delete rows from the target tables, and then we insert rows into the target tables. Information about the rows to be deleted comes from the CDC_*tablename*_ DELETES and CDC_*tablename*_UPSERTS tables. The CDC_*tablename*_ DELETES contains unique information about the rows that were deleted from the source tables, thus the same rows need to be deleted from the target tables. The CDC_*tablename*_UPSERTS tables contain the rows that have been inserted as new rows, or rows whose data have been updated. In either case, the procedure is to delete the rows from the target tables containing out-dated data, and then inserting the entire contents of the CDC_*tablename*_UPSERTS into the target tables. In this step, the decision was made to replace the entire row of data where an update occurred, rather than try to make a column-by-column comparison to update only a specific column. The latter choice is costlier in terms of computer resources.
The steps defined above to synchronize the unclassified and classified databases at LANL are repeated to synchronize the LANL classified database and the classified LANL research database at AFTAC.

Web access to data

One important challenge in using these large data warehouses is the simple and efficient access to the vast holdings in them. Web-based tools are important assets that address this problem. We have developed a web interface that allows researchers to do tasks such as track the progress of seismic analysis, access information about stations, handle logistical tasks (i.e., assignment of unique identifiers, and tracking the description and resolution of data problems identified through quality controls), and fast access to database metadata (e.g., schema descriptions).

The new web interface being used in LANL's GNEM R&E program has been in operations only a few months, but it has already proven to be an extremely efficient tool that researchers and computer technicians involved in the management of our vast data holdings are using more each day.

This web interface is used to manage two types of information: static and dynamic information. Both the static and dynamic pages can be accessed by anyone authorized to do so from anywhere inside or outside of LANL through the use of an internet connection and a LANL-issued crypto card. All pages are protected through password and the HTTPS protocol.

Static information consists of reference documents that rarely change. Examples of such documents are: database schema descriptions, coordination schedules for upcoming Knowledge Base (KB) deliveries, and general database usage instructions. These documents are generally presented in PDF format in order to eliminate operating system platform dependencies.
Dynamic information is generated by generally issuing queries to our data warehouse to retrieve the most up-to-date information. Below is a brief summary of the different types of data that can be accessed by our researchers.

Database Operational-Health System (DOHS
The primary purpose of this application is to give database users an easy and flexible way to report problems found in the data. This application is a recent implementation and will most likely become the back bone of the data

warehouse quality control procedures being developed in our program. Users reporting a problem have the choice to enter both a short and a free-format detailed description of the problem. Information such as the date the problem was reported and who reported the problem is automatically generated by the application. LANL GNEM R&E personnel involved in data management are alerted of new problems, and, as problems are resolved, the status of the reported issue is changed to reflect the current status of resolution. The information shown in the web page is stored in and retrieved as needed from custom database tables. Storing this information in tables in the data warehouse has many advantages. We use the daily backup routines employed to backup research data to save this information. The optimization features implemented in the database that allow fast access to these data are also employed.

Lastid table

This application was designed to help researchers coordinate updating the CSS3.0 database table LASTID, which keeps track of the maximum value assigned to a unique column identifier (such as *evid, orid, arid, magid*, etc.). The LASTID table is a critical table in the data warehouse, since unique column identifiers in tables such as CSS3.0-style EVENT, ORIGIN, ARRIVAL, and others, are required for every record that are used in our daily research activities. This application allows two or more researchers generating new data to see quickly the maximum identifier value for a particular column, but more importantly the application shows them if the identifier that they are interested in is currently being updated by another user, thus avoiding the common and extremely damaging problem of producing duplicate identifiers for the same table. This application is a critical tool in the quality control effort currently underway in LANL's GNEM R&E research data warehouse.

Picking status

Through this application researchers have instant access to view the current status of phase picking at LANL. At any one time, there can be several researchers and computer technicians making phase picks on different stations and creating arrival information for given events. Researchers who need the phase arrival information use this application to see the status of the stations and events of interest, and also view general comments and warnings noted while making the phase picks. This information is of high value to researchers involved in location and event identification efforts, to name a few. Two of many examples of highly valuable information available on this page are: "Station X very low S/N" – valuable information to know when making amplitude measurements for event identification; and "Station Y timing error, phase appears 4 mins before predicted" – valuable data for event relocation purposes. These comments are presented to the researcher in a clear and easy to read format, and each comment is tied to a specific event, thus giving the researcher the information needed to manually access the data warehouse and mine the necessary information to look at the problem first-hand.

Station location information

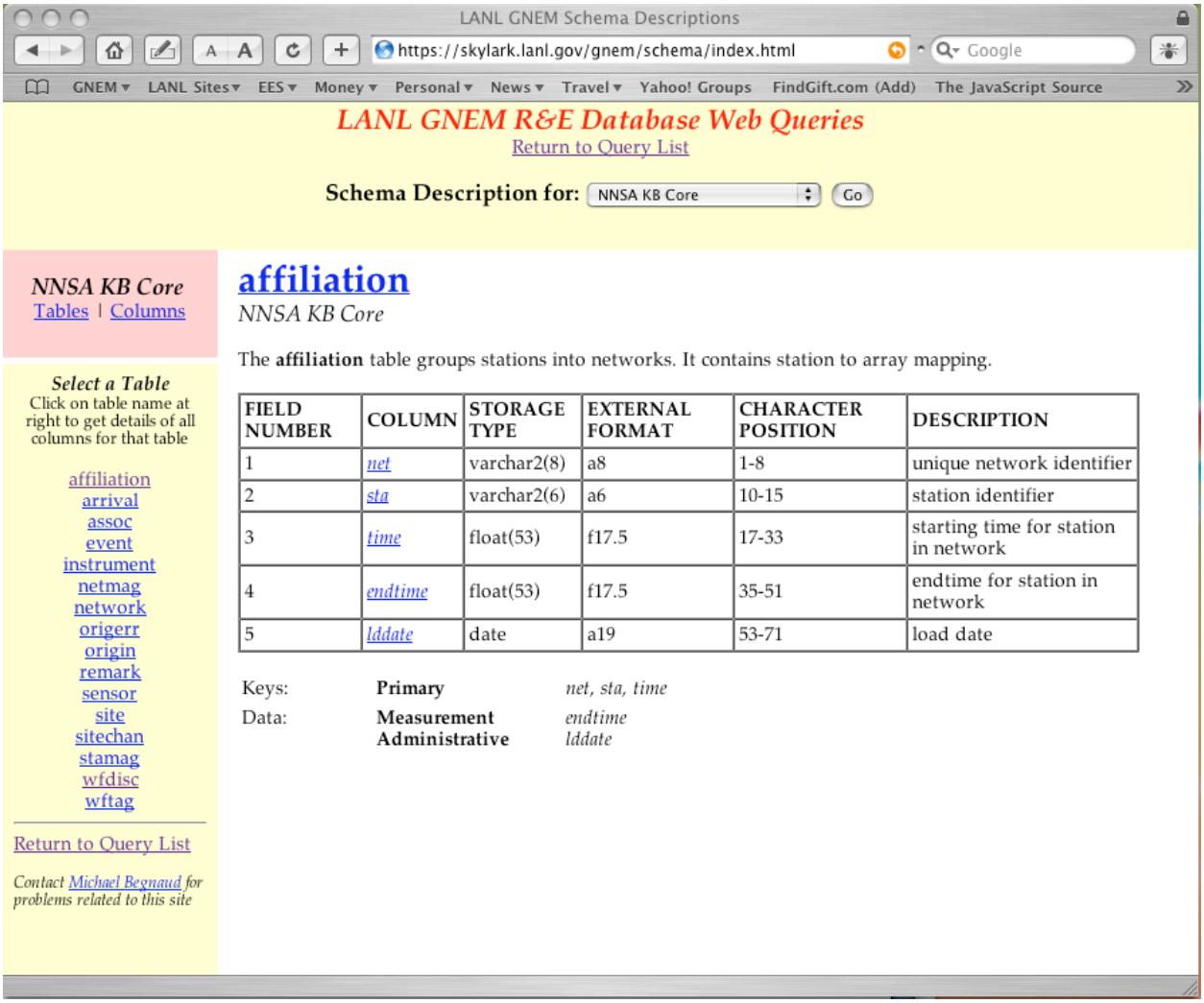
A quick interface was created to allow users to search for station location information easily. Users can enter a station or reference station into the query page and receive full CSS3.0 site information (i.e., latitude/longitude, elevation, ondate/offdate, dnorth/deast) about that station or station that belongs to the entered reference station (e.g. array elements).

Schema information

This page was designed to show all of the detailed table and column information that, up until recently, has been available only in text-based documents. This information has been compiled in custom database tables and is updated when necessary, thus giving the user dynamic access to the most up-to-date information about table design and column definition (Fig. 1). This tool eliminates the need for a researcher to keep bulky paper copies of potentially out-dated material. The specific use of this application is detailed below, in the **Metadata** section.

Metadata

A major new effort in making database metadata available is the documentation of the seismic calibration schema using a database schema. This schema is designed to represent all of the detailed table and field information that, up until recently, has been available only in text-based documents. These documents include various versions of the NNSA Knowledge Base (KB) core schema, NNSA KB custom schema, and the United States National Data Center (USNDC) schema documents. The portions that are most needed as readily available metadata are also the portions most amenable to adaptation into database tables themselves: the table descriptions and the column description. The schema designed for this purpose is still preliminary at the time of this writing. It includes four tables. The first is also the simplest, the table description table, TABDESCRIPT (see Table 1).



FIELD NUMBER	COLUMN	STORAGE TYPE	EXTERNAL FORMAT	CHARACTER POSITION	DESCRIPTION
1	net	varchar2(8)	a8	1-8	unique network identifier
2	sta	varchar2(6)	a6	10-15	station identifier
3	time	float(53)	f17.5	17-33	starting time for station in network
4	endtime	float(53)	f17.5	35-51	endtime for station in network
5	lddate	date	a19	53-71	load date

Figure 1. View of web tool view of the NNSA KB Core table AFFILIATION. Users may navigate between table or column view for a particular schema.

Table 1. TABDESCRIPT

field number	column	storage type	description
1	table_name	varchar2(30)	name of table
2	descript	varchar2(1024)	full text description of table
3	schema	varchar2(30)	name of schema
4	auth	varchar2(15)	author
5	lddate	date	date of entry

TABDESCRIPT provides a basic description of the table, identifies that table with a particular documented schema, and provides a reference in the database to connect the fields that may be associated with the table.

The second table is the COLDESCRIPT table (see Table2). This table is where most of the practical information is stored, and thus it is a somewhat more complicated table.

Table 2. COLDESCRIPT

field number	column	storage type	description
1	column_name	varchar2(30)	name of column
2	internal_format	varchar2(30)	format within relational database (same as storage type above)
3	external_format	varchar2(30)	format for programmatic reads and writes
4	external_width	number(8)	width of external text format in characters
5	na_value	varchar2(80)	value to use when field is not set
6	units	varchar2(80)	physical units, if any
7	range	varchar2(80)	full text description of range
8	rangetype	varchar2(30)	range type (numeric, defined, finite set, reference set, any integer, any float, any string, etc.)
9	nmin	number(53)	minimum for numeric rangetypes, if any
10	nminop	varchar2(2)	operator corresponding to nmin
11	nmax	number(53)	maximum for numeric rangetypes, if any
12	nmaxop	varchar2(2)	operator corresponding to nmax
13	reftab	varchar2(30)	reference table name for rangetype='reference set'
14	short_descript	varchar2(80)	one-line brief description of column
15	long_descript	varchar2(1024)	full text description of column
16	schema	varchar2(30)	name of schema
17	auth	varchar2(15)	author
18	lddate	date	date of entry

The COLDESCRIPT table not only provides the description of a column, but also provides metadata such as NA values, units and ranges in useful forms. In fact, it is planned that the "range" column will eventually be retired, when the information contained in this column is fully represented. Currently, "range" stands as a go-between to bring the text information on range from the documents into the database. Similarly, units and na_value may change somewhat when the information is properly distilled. Currently most NA values are just a single value represented as a text string, which is the goal for all entries in this column. Most numeric ranges have been properly translated into *nmin*, *nminop*, *nmax*, and *nmaxop*. Each operation is relative to the value in the column; that is, 'column *nminop nmin'* and 'column *nmaxop nmax'*. If both are set, then both must apply (implied 'and'). In general, one is a minimum and the other a maximum. To represent it as a range, the sense of *nminop* must be reversed, '*nmin* reverse(*nminop*) column *nmaxop nmax'*. A range type of 'defined' means the value of the column is limited to a short set of pre-defined values. A 'finite set' is a limited but long or not pre-defined set of values. A 'reference set' is limited to the values in a particular table (as given in *reftab*). Using these fields properly can completely and precisely define the various field ranges.

The next table is COLASSOC (see Table 3). This table provides the necessary links between TABDESCRIPT and COLDESCRIPT.

Table 3. COLASSOC

field number	column	storage type	description
1	table_name	varchar2(30)	name of table
2	column_name	varchar2(30)	name of column
3	column_type	varchar2(30)	function of column
4	column_position	number(8)	column ordering position within table
5	na_allowed	varchar2(1)	is NA allowed for the column
6	key	varchar2(30)	type of key, if column is a key
7	keyschema	varchar2(30)	schema to which key's reference table belongs
8	schema	varchar2(30)	name of schema
9	auth	varchar2(15)	author
10	lddate	date	date of entry

Each table will have multiple columns (column positions 1 through the total number, in order), and columns can appear in multiple tables. The column type will define the basic function of the field in the database (i.e. primary key, unique key, descriptive data, measurement data, administrative data). An NA value should always be given for a column in COLDESCRIPT, but na_allowed will indicate if that NA value can be used in a particular table or not. Key allows key columns to be identified with respect to the table: the reference table for the key, or a table in which the key is foreign. These keys are primarily numerical identifiers. Keyschema is used when the reference table is part of a separate schema.

The remaining table is the GLOSSARY table (see Table 4). This table serves two purposes: the first is to simply define generic strings used in various description fields, primarily acronyms and abbreviations. The second is to serve as the reference table for 'defined' range types and 'finite set' range types.

Table 4. GLOSSARY

field number	column	storage type	description
1	glossid	number(8)	unique numerical identifier
2	lineno	number(8)	numerical order if definition is multiple lines
3	name	varchar2(30)	name to be defined
4	column_name	varchar2(30)	column name if specific to a subset of columns
5	table_name	varchar2(30)	table name, if specific to a subset of tables
6	owner	varchar2(30)	owner of table in database, if specific to a subset of owners
7	definition	varchar2(80)	full text definition of name
16	schema	varchar2(30)	name of schema, if specific to a subset of schemas
17	auth	varchar2(15)	author
18	lddate	date	date of entry

A given definition can apply in all circumstances (column_name, table_name, owner and schema are all not set), which is a generic definition, or it can apply to increasingly selective subsets of columns, tables, owners and schemas. For 'defined' and 'reference set' range types in COLDESCRIPT, the complete permitted set of values will be found in GLOSSARY, one entry for each value (values are found in the name column), and column_name will always be set for these.

As with any other new application, there are issues associated with our implementation of capturing metadata information and making it readily available through the power of a relational database. These issues have appeared as more people use the tables we have designed. We are working successfully with application developers at Sandia National Laboratories in identifying areas where improvements to this process can be made. To-date, we consider this approach to be in beta-testing mode, but will work hard to elevate it to full production mode within one year. Having such information in database form has immediate application to a wide variety of efforts involving the database, both specific to Los Alamos, and across NNSA (see presentation at this meeting by Ballard, et al., on the subject of DBTools). This schema for table description metadata is a key contribution to the web-based database documentation discussed in this paper. It is also the foundation for an automated quality control (QC) effort at LANL. As a rather fundamental tool, it is foreseeable that a wide variety of other uses may arise with time. It is hoped, for example, that this schema can contribute to the maintenance of the schema descriptions, such that they are maintained in the database, where they can be easily checked for errors, quality, and completeness, and that the corresponding section of the documents could be generated automatically, directly from the database.

Another advance in using metadata, with a more narrow application, has been the creation of bulletin descriptive tables. These tables describe the sources of bulletin data that have been imported into the data warehouse, as well as providing a means to track individual data elements to the corresponding lines of text in the original document. There are two tables involved: BULLETIN and BULLASSOC. The BULLETIN table contains one entry for each individual bulletin, with columns *dir* and *dfile* pointing to the text file on the system that contains the bulletin. It also has a *bullid* column that is unique to each bulletin. The other information in the table describes the bulletin, including format. The BULLASSOC table has one line for each data object extracted from the bulletin (origins, arrivals, magnitudes, etc.) It links the *bullid* and the *id* of the extracted object and provides the line number in the

bulletin corresponding to the object. These tables have two immediate uses. The first is to help in QC – problematic objects can be directly traced to the corresponding file and line number. The second is user interaction – an application can be created that allows the user to automatically view the source line and context for any bulletin-derived data in the database, which it is hoped will be helpful in understanding the nature of some bulletin data.

Quality control

As data become more voluminous and complex, quality control (QC) has become a challenging and extremely interesting issue to consider when developing content for the Knowledge Base (KB). In particular, because of recent advances in tools that access KB data, researchers have been able to make more efficient use of this large volume of data, but have also found problems in applying the data to their research efforts. Improvements in QC procedures are helping researchers and data managers to more readily identify complex quality problems. The outcome is improved research products resulting from improved data upon which those products are based. QC is handled in a wide variety of ways at the present, and much effort is being made to better structure this procedure and automate as much of it as is practical to do so. There are three main categories of QC: manual, tool-assisted, and automated. Improvements in manual QC are occurring constantly as a wider variety of issues are captured and understood. But this kind of QC is the least transportable and repeatable. The next step is to capture the tracking and resolution of QC problems in various simple tools, usually case-specific scripts. Tool-assisted QC is more transportable and repeatable, since the script serves as documentation of procedures, but it still requires case-by-case modification and application. Automated QC is preferred and cannot happen without there first being a fairly comprehensive understanding of the problem. The best approach is to document exactly what the database should be and reject anything that does not conform. This approach is why the database descriptive schema discussed above will be a valuable tool in automated QC of database data. However, not all QC issues are in the database. QC must be applied to all data from raw waveforms to final calibration parameters.

The improvements GNEM R&E has made in this area have even had modest impact outside the program; for example, in helping the International Seismological Centre (ISC) identify some problems with their data products. This opportunity arose during QC investigations into unusual problems linking origins and events. Using tool-assisted QC of the problem, we were able to identify problems in the original data files where duplicate information was being reported as different events. We could completely characterize the problem and send detailed descriptions to the ISC, which then worked to correct the problems we reported. Considering that the problems were unreported and dated back to bulletins that were released as many as five years earlier, and that the ISC produces a widely-used bulletin recognized by the seismological community as the final global bulletin, it is apparent that these errors are difficult to find in any research or operation context. In other words, QC is a hard problem and GNEM R&E is making new and unique contributions toward resolving the problem.

Applying data warehouses to event location research

Merged global and regional phase arrival and ground-truth source data are available in our research data warehouse to assist researchers in their event location efforts. The location effort has, in fact, created its own database tables by combining all available phase arrival information and manipulating it to create consistent arrivals and improve event relocations that lead to better correction surfaces. Researchers use the reconciled database tables that contain global and regional bulletin information, as well as LANL-generated phase arrival and waveform data, to create new tables specific for relocation purposes.

LANL researchers have created Arrival Sorter (ARsorter), a Perl-based program whose primary objective is to reconcile similar phases reported from many organizations, including our internal phase picks. The reconciliation of many reported similar phases recorded by different sources for a particular seismic station for a given event requires some phases to be renamed. The phase renaming process is necessary to provide the best distinct phase arrival times to the programs used in location. By doing this process, we are compiling a database with the best distinct seismic phase arrivals, so that for a given event at a given station there is only one particular phase (e.g., only one Pg, Pn, Sn).

Applying data warehouses to event discrimination and yield research

The event discrimination and yield research conducted at LANL benefits directly and indirectly from the data warehouse. The event discrimination work uses data stored in database tables directly as input for processing, while

event yield work uses the data warehouse indirectly by extracting the data contained in the headers of the SAC-format waveform files, which are populated using data from the database. The final products of the event discrimination work are MDAC (Magnitude and Distance Amplitude Corrections) parameters for given event/station paths and amplitude corrections for raw amplitude measurements. These data are also stored in the data warehouse in NNSA custom tables, which are used by tools such as EventID and CodaMag developed at Sandia National Laboratories (SNL). The final products of the yield research work are the computation of network and station coda moment magnitudes, also stored in CSS-style database tables.

CONCLUSIONS AND RECOMMENDATIONS

The newly created process of capturing data changes (CDC) shows great promise. It is expected that we have not encountered all possible use cases, where modifications to the process will need to be made and perhaps auxiliary tables or triggers will need to be created. This process has been in place for several months and it appears to get the work done. As mentioned earlier, many of the steps involved are manual and human intervention is needed throughout. The next step of this process is to automate many of the synchronization steps, and also incorporate appropriate quality control checks to ensure that the synchronization between databases was successful and that the data being inserted does not create internal conflicts and errors in the target database.

The use of web-based database querying/population tools has shown to be an efficient method to access database table information and track important research steps on any computer platform with a web browser. We are continually finding new reasons and ways to access the database through the web, including determination of outdated information based on newly loaded data (e.g., new/updated arrival picks) and quality control checks.

The newly acquired ability to parse and store in the database the contents of core documents such as the KB database table and column descriptions is proving to be extremely valuable. Having such information in database form has already had immediate impact on a wide variety of efforts. One specific example can be viewed at this meeting on the subject of DBTools, paper and poster by Ballard, et al. It is also the foundation for an automated QC effort at LANL that has recently started this summer. It is hoped, for example, that this schema can contribute to the maintenance of the schema descriptions, such that they are maintained in the database, where they can be easily checked for errors, quality, and completeness. Our vision for the future is that the corresponding section of the documents could be generated automatically, directly from the database.

Quality control (QC) has become an increasingly visible and difficult issue regarding the Knowledge Base (KB), as data have become more voluminous and complex. Improvements in QC procedures currently under way will help researchers and data managers to more readily identify complex quality problems. The outcome is improved research products resulting from improved data upon which those products are based. The improvements GNEM R&E has made in this area have even had modest impact outside the program; for example, in helping the International Seismological Centre (ISC) identify some problems with their data products. We hope that present and future quality control efforts will lead to continued collaboration with external organizations to benefit the entire seismic community.

ACKNOWLEDGEMENTS

The authors wish to acknowledge LANL personnel (below) who have made important contributions in the past, and those who continue to make vital contributions to the development and maintenance of the LANL research data warehouse:

Hans E. Hartse, Diane F. Baker, George E. Randall, James T. Rutledge, Steven R. Taylor, W. Scott Phillips, Marian D. Peters, and Thomas L. Riggs.

REFERENCES

Carr, D. (07/04), *National Nuclear Security Administration Knowledge Base Core Table Schema Document*, Sandia National Laboratories report SAND2002-3055, available at: https://www.nemre.nnsa.doe.gov/prod/nemre/fileshare/coretables.pdf