WORKING WITH AND AROUND THE NNSA DATABASE SCHEMA

Charles A. Langston, Ivan Rabak, Christy Chiu, Chuntao Liang

University of Memphis

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

Contract No. DE-FC04-2002AL68280

ABSTRACT

We are implementing modifications of the National Nuclear Security Administration (NNSA) database schema and the Sandia National Laboratories (SNL) MatSeis program to handle waveform manipulation needed in modeling observed seismograms with synthetics. Functions include writing of windowed, processed data waveforms and waveform inversion for source and earth structure parameters. Our objective is to work within the existing NNSA schema in as far as possible without adding many additional tables or additional keys to relate the tables while preserving the original data within an event database. A modification of MatSeis allows writing of processed waveforms with a text file that describes the processing history. A single channel of data may have up to 99 "generations" of processed versions of itself that become input to other Matlab modules for modeling the processed data. Additional Matlab modules are constructed for earthquake focal mechanisms, moment tensor inversion, surface wave dispersion analysis and inversion, and regional waveform phase-time inversion. The ideal solution, which is beyond our work here, is to extend the NNSA database schema to add keys to the wfdisc table or to have additional wfdisc-type tables with keys for relations between waveforms.

OBJECTIVES

Objectives for this research include the following:

1) Determine waveform characteristics of earthquakes and explosions over a broad frequency band that can be used to locate and identify small seismic sources in central Asia
2) Construct a waveform and source parameter database to be used to understand local/regional wave propagation in central Asia and to identify sources
3) Construct an earth structure database for models of the crust and upper mantle that accurately predicts the waveforms for seismic sources in central Asia
4) Construct a knowledge base system that links data, data-derived source models, data-derived structure models, and educational tutorials for understanding sources, structure, and wave propagation in central Asia.

In this paper we report on problems encountered in our attempt to use the NNSA database schema as the basis of a seismic analysis processing system for treatment of global earthquake data. Our objective for this report is to suggest that some effort be made to fully incorporate relationships between data waveforms, derived processed waveforms, and synthetic waveforms in the NNSA database schema to fully integrate seismological waveform studies with previous parameter studies.

RESEARCH ACCOMPLISHED

Introduction

New tables for parameter data are continually being developed to extend the database schema as new data processing packages come online.

In conjunction with the use of the NNSA database schema, there have been ongoing software development efforts to incorporate seismic databases directly into seismic modeling codes. The Seismic Analysis Code (SAC), for example, can have the capability for making database queries and accessing waveforms if an Oracle database is available on the network. Of more interest to this work is the MatSeis system written at Sandia National Laboratories. This program has many of the attributes of SAC but with access to the functionality of the full Matlab system that it runs under.

Routinely handling waveform data from many events at many stations would seem to require a database system. Unfortunately, the industry standard, Oracle, is very expensive for a small research laboratory. MySQL is an alternative as are other open software systems, but it was decided that a simple flat file system based on the NNSA schema would be simplest. A flat file system consists of simple text files, can be constructed using a variety of programming languages and existing database systems, and can be easily accessed as well. Using the NNSA schema would also allow consistency with DOE laboratory efforts in seismic data handling.

The current practice in the graduate program at the University of Memphis is to encourage the use of the Matlab programming language in graduate courses and research. Matlab is a very extensive programming language with many useful mathematical, scientific, graphical, etc., software libraries. It also contains tools for building elaborate graphical user interfaces (GUIs). Thus, it seemed a straightforward decision to choose Sandia National Laboratories MatSeis system as the basis for modeling the seismic data in this project. MatSeis can be configured to query an Oracle database, an NNSA (CSS3.0) flat file system, or a “generic” database flat file system. Waveforms can be read into the system and then analyzed, modifying the existing event database, and a new database saved to disk. Once a database has been read into MatSeis, all information (waveforms, parameters) can be accessed by internal MatSeis routines and used in other applications run under Matlab but external to MatSeis. Thus, MatSeis seemed like the natural tool to use that would incorporate NNSA database flat file structures and serve as a software platform to build other waveform analysis tools not currently found in MatSeis through Matlab.

The Problem

The NNSA schema is good for storing derived parameter data like earthquake magnitudes, location, and the like but does not express relationships between waveforms. An ideal attribute for a seismological database is to collect pertinent information about the data (metadata) and the waveform data themselves and then to append all derived parameters, earth models, modeling methods, processed data seismograms, and synthetic seismograms. It should be unnecessary to duplicate database tables, for example, in a processing cycle of analysis.

For example, in a teleseismic moment tensor inversion, raw waveform data must be corrected for instrument response, filtered in some manner to avoid noise, time-windowed to obtain P or S wave body phases, possibly correlated with synthetic waveforms, time-shifted, and then modeled using a variety of assumptions about the source, the source-area earth structure, and choice of input waveform data. The NNSA schema has the facility for different source origins (orid) that can key into different moment tensor results, but it does not have appropriate keys to relate different waveform files to the raw data waveforms. In other words, a database key that relates many different processed waveforms back to the original data waveform is necessary. There also should be a key that relates many different synthetic waveforms back to processed waveforms and the data waveforms. In the current schema, this would mean being able to define many different "wfdisc" tables that relate to entries in other wfdisc tables. In addition to relations between wfdisc tables, it would be useful to have additional tables that define the processing steps that relate the tables together.

MatSeis and the present implementation of the NNSA schema allow a particular event database to be read in. Some processing can occur, and some database tables can be modified as a result of the processing. For example, first motions, times, and amplitudes can be picked using measuretool and the arrival table modified. MatSeis then has the facility to write a new set of database tables to disc. In effect, different processing steps on the data require a repetition of database tables and a need to manipulate the resulting multiple databases outside of MatSeis with other database software.

A larger problem, however, is that MatSeis does not allow a user to save processed waveforms to disc, and the NNSA database schema does not allow relations between waveforms, as discussed above. These two problems were not recognized at the beginning of the decision-making process for streamlining a waveform modeling system. The problems are significant enough and the possible solutions important enough for future database and seismic analysis work that we have spent considerable effort trying to come up with solutions that are most closely consistent with the idea of using a single-event database in and as part of the modeling process.

Approach

MatSeis is a software package intended to help process waveforms. It is GUI driven relying on Matlab to handle graphics. Furthermore, it is tied with Matlab in terms of a toolbox giving all the functionality of Matlab-based programs. This advantage is exploited to expand MatSeis capabilities by writing custom Matlab code.
One of the goals in expanding MatSeis is to allow saving processed waveforms.

The current version of MatSeis (1.7 devel), here at the University of Memphis, and the new version (1.8) coming from Sandia National Laboratories do not support writing waveforms. Myriads of seismic data formats are used by various geophysical groups, so choosing the right format for writing waveforms is somewhat a matter of institutional preference. The University of Memphis widely uses the SAC format so we decided to use it as the only supported format for writing waveforms at this time. Additional data formats can be bundled into the solution and implemented as Matlab function/scripts.

MatSeis allows access to Custom Menus through the main GUI. We use this option to build a set of scripts to perform various database related tasks. The database schema is a modified NNSA type. The modification includes creating additional tables to reflect the existence of processed and saved waveforms and accompanying information on the processing steps. Figure 1 shows a custom table *wfdisc_s* that contains saved waveform information.

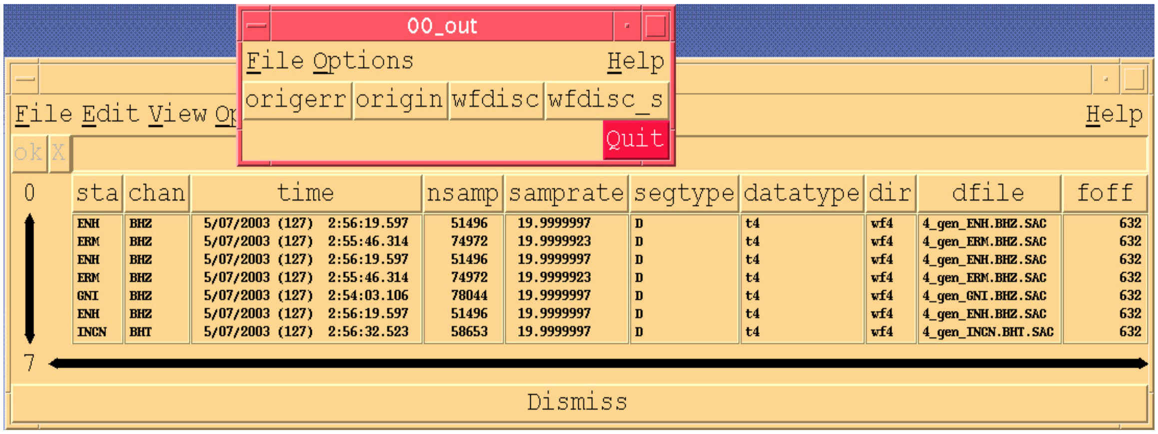

Figure 1. An example of a *wfdisc_s* table filled with the fourth generation of waveforms.

Method

In the pull-down 'Custom' menu in MatSeis, a new option allows saving of manipulated waveforms in up to 99 generations (Figure 2). It is possible to save all the manipulated waveforms (not affecting raw read-in data that already exists on disc) or the selected ones with a standard suite of interactive 'Are you sure?' dialog boxes. The new waveforms can be read in MatSeis, manipulated again, and saved as a new generation.

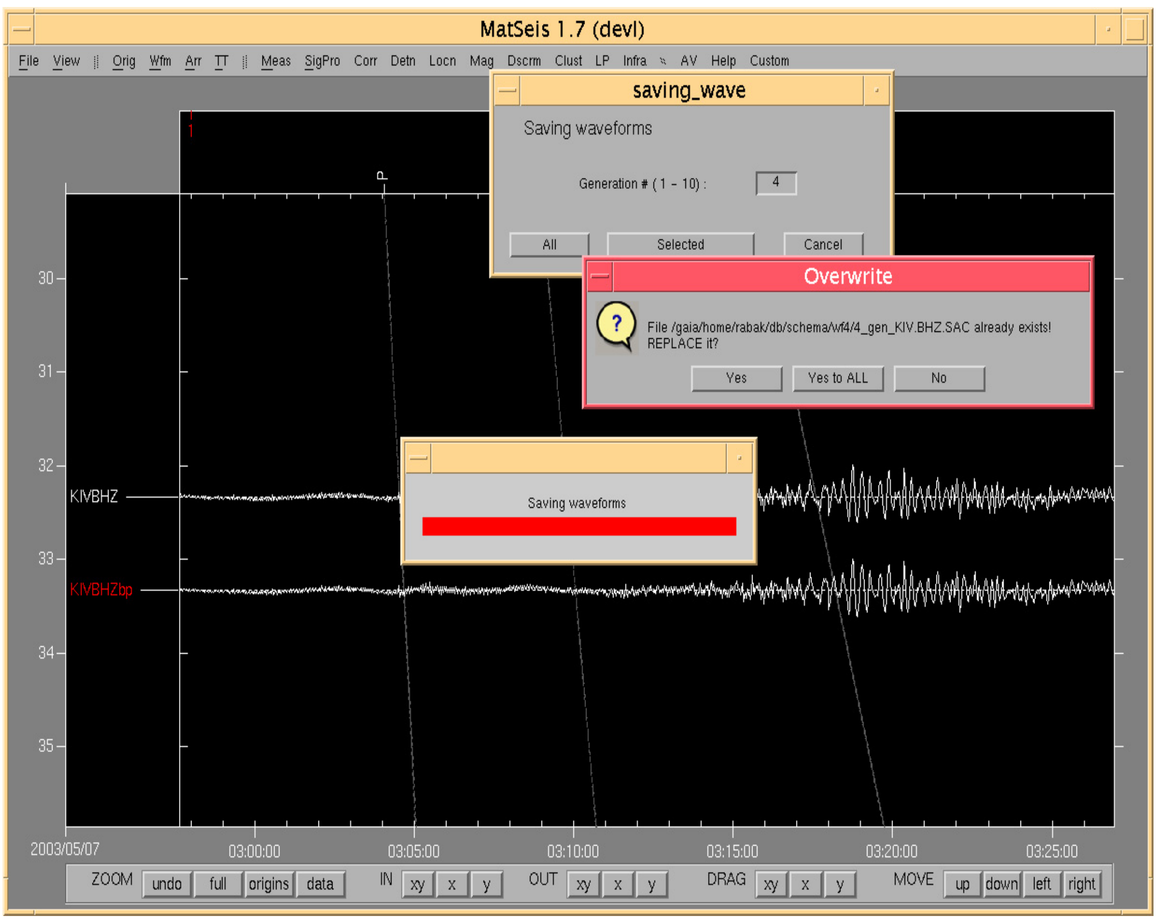

Figure 2. The 'saving waveforms' GUI, once custom -> write waveforms is selected from the pull-down menu.

Several Matlab scripts are being developed that deal in stages with the problem of saving waveforms. As much as possible, data are retrieved using existing MatSeis vocabulary, but at this stage, some hardwiring has been performed through editing of the original MatSeis function/scripts. The following scripts are called in the order shown. However, they are written as functions so they can be called by passing arguments listed through 'help *script_name'* command.

script 1 – the GUI that deals with the generation selection. It is created using a the Matlab GUI editor and saved as a Matlab figure. It is not created on-the-fly, with code, as most of MatSeis interactive windows are.

script 2 – retrieves all the information using existing MatSeis functions necessary to save waveforms such as current generation number, waveform station and channel names, directory locations, NNSA table fields, table formatting, currently selected waveforms, file name generation.

script 3 – writes SAC files. The algorithm uses the original SAC header for a processed waveform header because this information is not used by the NNSA schema once the tables are generated. This approach is subject to change.

script 4 – finally the existence of the new waveforms is reflected through a new *wfdisc*-like table. The existence of this table must be reflected in the NNSA schema text file. This step can be done automatically, but at the moment the change is done by editing the file with a text editor.

script 5 – makes sure that wfdisc_s table rows stay unique throughout other table updates.

The following diagrams give more detail about the scripts:

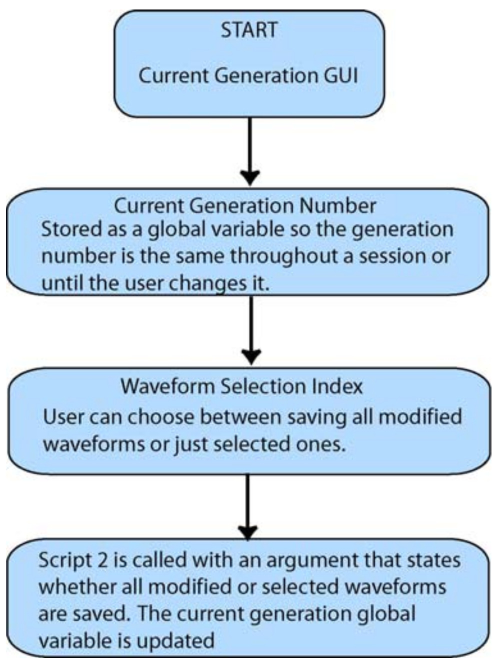

Figure 3. Flow chart of Script 1.

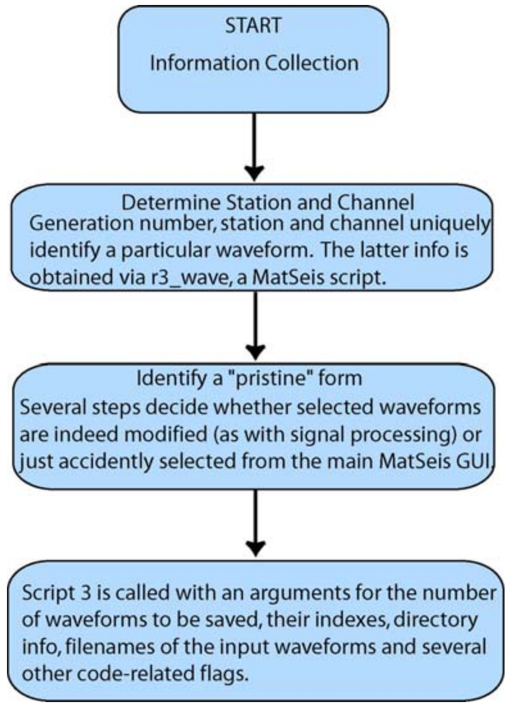

Figure 4. Flow chart of Script 2.

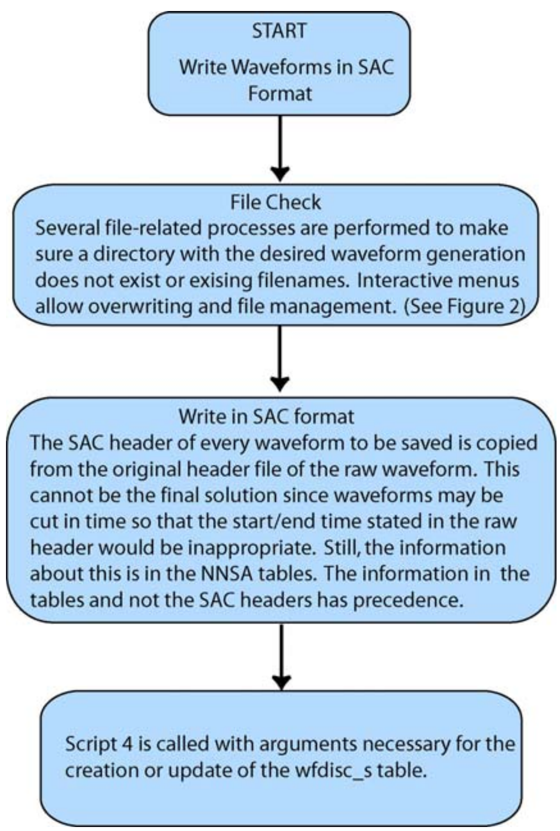

Figure 5. Flow chart of Script 3.

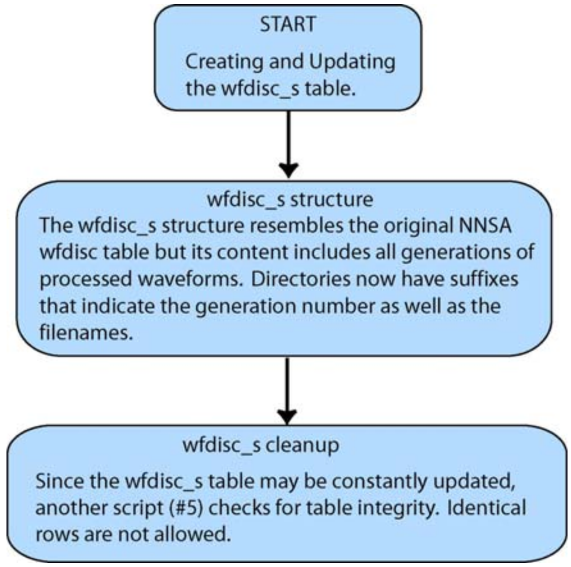

Figure 6. Flow chart of Scripts 4 & 5.

MatSeis obtains necessary information from its **.config* file for each event. Because processed waveforms can be read in and additionally processed and saved as another generation, a second *config* file must be generated. At this time, this must be done with a text editor by copying the raw *config* file and substituting the MS_IN_DB3_WFDISC variable with a new value: *wfdisc_s.*

Future steps should eliminate a need for hardwiring of the original MatSeis scripts. This will unfortunately duplicate tasks already performed by the original scripts, but it will create portability with future MatSeis releases. The lack of a MatSeis blueprint reduces the speed of building independent custom scripts to the programmer's ability to infer what and how various steps work by analyzing the Matlab and C code.

There are plans to develop an additional table and introduce it into the NNSA schema that would contain all the processing steps applied on a particular waveform, and would accelerate future processing because guesswork on what was done for a particular event would disappear.

An important additional step to complete the processing system is to include an additional wfdisc table for synthetic seismograms and a key to associate the synthetics with a particular generation of processed waveforms. Perhaps called wfdisc_aux, such a table would contain a variety of synthetic waveforms that include "complete" waveforms (constructed with all source and structure parameters) and Green's functions for computing complete waveforms.

CONCLUSIONS AND RECOMMENDATIONS

The global seismic data set of waveforms is constantly growing with the addition of new seismic stations and arrays. Considerable progress has been made on archiving these data and constructing catalogs of parameters derived from the waveform data. However, many modern seismological studies rely on waveform modeling to infer structure and source parameters. At the moment, no self-consistent database structure can incorporate either processed or synthetic data waveforms within the same database. Ideally, it is important to preserve processed and synthetic waveforms within an event database so future researchers can evaluate the accuracy of wave propagation codes used in constructing the synthetics in deriving parameters or to investigate other assumptions made in the modeling. Such a database structure would also help standardize seismogram modeling systems and software packages that incorporate it, improving collaboration between researchers.

Our principal recommendation is for the NNSA to develop an extension of the existing NNSA database schema to incorporate processed waveforms, synthetic waveforms, and processing history with the raw data waveforms.

Appropriate database keys should associate all tables. Such a modification would address an important need in the seismological community and could serve to standardize waveform data handling and modeling.

ACKNOWLEDGEMENTS

Work on these modifications and additions to MatSeis and issues with the database schema have benefited from suggestions and comments made by Darren Hart at Sandia National Laboratories.