Database Interaction in AQMS

- ✓ <u>Read-only database programs</u>
  - Dbselect
  - STP
  - Catalog search pages on NCEDC/SCEDC websites

- ✓ <u>Read-write database programs</u>
  - Jiggle
  - SIS and station metadata
  - Perl post-processing utilities (tpp scripts)

- ✓ <u>Build your own</u>
  - Perl/Perl-cgi (catalog search pages)
  - Java (jasi objects)
  - Php
  - C++ (RT processes)
  - Pro*C (dbselect)
  - Web services (QuakeML/StationXML)

- ✓ <u>Oracle RDBMS & schemas</u>
  - Version
  - Critical Patch Update
  - Backup
  - CISN Schemas
  - CISN Schema Change Working Group

- ✓ <u>Database monitoring</u>
  - Server state of health & Oracle errors
  - Storage
  - Replication
  - Resources

Dbselect

- ✓ Command line tool to retrieve parametric information from the database.

- ✓ Origin, magnitude, arrival, amplitude, coda information:
  - ASCII formats.
  - Hypoinverse formats (summary, phase).

- ✓ Monent tensor and fault plane solution:
  - FPFIT, psvelomeca.
  - Psmeca, CMT standard, UCB.

- ✓ Input parameters:
  - Time, event Id, magnitude, latitude/longitude/depth, errors.
  - Annulus (ring), polygon.

STP

- ✓ STP (Seismogram Transfer Program) was originally developed by Rob Clayton and is run by Southern California and Northern California. STP provides waveform data in SAC, miniSEED, SEED, ASCII, and COSMOS V0 formats, amplitudes, event catalogs, and station and channel lists.

- ✓ The server program is written in C, using the Pro*C pre-compiler for database access. Each location runs servers on multiple machines for redundancy. When the STP client connects to a location, it will automatically attempt to connect to one server, and if that server is down, then it will try to connect to a second server, and if that fails, it will try a third.

- ✓ Command line clients exist for Linux, Unix, Mac, and Cygwin. The SCEDC also has a Java applet. The applet and a reference manual are available at http://www.data.scec.org/STP/stp.html. Clients can be downloaded at http://www.data.scec.org/research/downloads.html#stp..
  Northern California's client is available at http://www.ncedc.org/ftp/pub/programs/stp/. The command-line clients from both sites can connect to either southern or northern California servers.

Catalog search
- ✓ The catalog search pages are CGI scripts that provide read-only access to events in our database.
- ✓ The Southern California catalog search provides several search formats (by date/mag/location or by event ID, polygon and radius searches) and output formats (SCEDC, Hypoinverse, QuakeML, KML, Google Map, and more). The Northern California catalog search provides output in NCSN, Hypoinverse, KML, or CSV.
- ✓ The Southern California catalog search pages are located at http://www.data.scec.org/catalog_search/date_mag_loc.php and the Northern California catalog search pages are at http://www.ncedc.org/ncedc/catalog-search.html.

<u>Jiggle</u>

- ✓ Java GUI for earthquake analysis.
- ✓ Interfaces with the database via JASI (Java Abstract Seismic Interface).

<u>SIS</u>
- ✓ Interface for the SIS (Station Information System) metadata database.

<u>CISN System Utilities Scripts</u>

Command	Arguments	Function
Event Info		
cattail	-[ht] [-d dbase] <hours-back> (default = 24)	dump events since <hours-back>
trigtail	-[ht] [-d dbase] <hours-back> (default = 24)	dump triggers since <hours-back>
eventhist	[-d dbase] <evid>	show the event's history
catone	[-d dbase] <evid>	output summary line for one event
eventAge	[-d dbase] <evid>	output event age in seconds
eventMag	[-d dbase] <evid>	output event mag
showalarms	[-d dbase] <evid>	show alarms sent for event
Event Parameter Info		
ampdump	-[ht] [-d dbase] <event-id>	dump amps contributing to mag (AssocAmM)
ampdumpo	-[ht] [-d dbase] <event-id>	dump amps associated with origin (AssocAmO)
checkwavefiles	-[ht] [-d dbase] <event-id>	show dir and list archived waveform files for event (requires ssh)
wavedump	-[ht] [-d dbase] <event-id>	dump list of this event's waveforms
Station/Channel Info		
hypostalist		dump station list in Hypo200 format
makeadhoc		dump station list in 'adhoc' format
stainfo	-[aht] [-d dbase] [-f type] <station string>	dump station info - string like "NC.SJE.HHZ.–", "CE.%.hn_.0%" types: "hypo", "kml", "gmt" or default
stadist	-[h] [-d dbase] <lat.dec> <lon.dec>	sort current stations from point - also: <lat.dec> <lon.dec>
Duty Review Page Support		
acceptEvent	[-d dbase] <evid>	mark event as reviewed
acceptTrigger	[-d dbase] <evid>	mark trigger as reviewed
deleteEvent	[-d dbase] <evid>	delete event or trigger
undeleteEvent	[-d dbase] <evid>	undelete event or trigger
alarmsend	[-d dbase] <evid>	send alarms
alarmcancel	[-d dbase] <evid>	cancel alarms
finalizeEvent	[-d dbase] <evid>	finalize the event

QDDSdelete	[-d dbase] <evid>	send a QDDS delete message for event
QDDSsend	[-d dbase] <evid>	send a QDDS message for event
cubeMessage	[-d dbase] <evid> [<version>]	return a Cubic message for this evid

Process Control System (PCS)

puttrans	[-d dbase] <oldGroup> <oldSource> <oldState> <oldResult> [<newGroup> <newSource> <newState> <newRank> <newResult>]	create transition definition
deltrans	[-d dbase] <oldGroup> [<oldSource>] [<oldState>] [<oldResult>] [<newGroup>] [<newSource>] [<newState>] [<newRank>] [<newResult>]	delete a transition definition
gettrans	[-d dbase] [<group>] [<table>] [<stateold>]	show transition definitions
getstates	[-d dbase] <group> [<table> <state> <rank> <result>]	show posted events
post	[-d dbase] <id> <group> <source> <state> <rank> [<result>]	post event to a state
postFromList	[-d dbase] <filename> <group> <source> <state> <rank> [<result>]	post events listed in a file to a state
unpost	[-d dbase] <evid>	remove an event from all states
delstates	[-d dbase] <group> [<source> <state> <rank> <result>]	Delete all events with the given signature
next	[-d dbase] [-s secs] <group> <source> <state>	Get the next event in this state
result	[-d dbase] <id> <group> <table> <state> <result>	set event's result value

Post Processing Tasks

ampgenpp	<evid> <properties_file>	create amps and write to base
Rcgone	[-d dbase] <evid>	Generate request cards for one event

Miscellaneous Utilities

masterrt		output current master RT system name
masterdb		output current master database name
showlocks	[-d dbase]	show events locked by Jiggle
unlockevent	[-d dbase] <evid>	unlock an event locked by Jiggle
unlockall	[-d dbase]	unlock all events locked by Jiggle
epoch		show system time in epoch seconds
hypolocate	-[h][-P] [-d dbase] -u URL -p port <evid>	locate this event with a SolServer

Building your own database access applications

- ✓ Any language that can access Oracle can be used to build applications for reading from or writing to the database. The languages that we use include Perl, PHP, Java, C, C++, and Python. Most of these languages require downloading additional packages or libraries for Oracle access. In post-processing C and C++ applications we use the Pro*C pre-compiler, which allows programmers to embed SQL statements in C or C++ source code. The RT C++ applications use OTL (Oracle Template Library). For Python we use cx_Oracle, and for Perl we use DBI and DBD::Oracle.

Web Services – QuakeML

- ✓ Enables users to retrieve event parametric data from SCEDC.
- ✓ QuakeML V1.1.
- ✓ Hierarchical QuakeML produced for picks and amplitudes.
- ✓ Web services can be queried by :
  - 10 most recent earthquakes
  - Event id
  - Start and end dates
  - Date, magnitude and location (latitude and longitude)
  - Polygon
  - Radius

1.- Database Maintenance & Schemas:

* CISN and RSN's are currently running 10g Release 2 (10.2.0.4 in NC).
Plans are underway in NC to migrate all servers to Oracle 11g Release 2.

Oracle releases on a quarterly basis Critical Patch Updates (CPU). They contain
security fixes for Oracle products and should be installed promptly.
The next four dates are:

     19 April 2011

     19 July 2011

     18 October 2011

     17 January 2012

Daily backups should be performed for the DC databases. NC & SC use '**rman**'
(Recovery Manager) to perform the backups. NC applies the following schedule:

Sunday:	Full backup (level 0)
Monday:	Incremental backup (level 1)
Tuesday:	Incremental backup (level 2)
Wednesday:	Incremental backup (level 1)
Thursday:	Incremental backup (level 2)
Friday:	Incremental backup (level 1)
Saturday:	Incremental backup (level 2)

* CISN schemas are available at:

  http://www.ncedc.org/db/Documents/NewSchemas/schema.html

The CISN Schema Changes Working Group evaluates and make
recommendations on possible schema updates. Historic/new requests
can be consulted/submitted at:

  http://www.ncedc.org/db/schema_change_web/index.php

2.- Database Monitoring:

2.1.- Servers:

* In-house scripts are used to monitor the database state (up/down) in NC. This can be done by just trying a simple connection to the database. SC uses Big Brother.

The database server & listener can be started with:

```
$ORACLE_HOME/bin/dbstart
$ORACLE_HOME/bin/lsnrctl start
```

Likewise, the database server & listener can be shut down with:

```
$ORACLE_HOME/bin/dbshut
$ORACLE_HOME/bin/lsnrctl stop
```

* Oracle logs errors in the following file:

```
$ORACLE_BASE/admin/<SID>/bdump/alert_<SID>.log
```

For major issues, Oracle also creates a trace file in the above directory with more details concerning the error.

The utility '**oerr**' can be used to retrieve more information about an error message:

```
Usage: oerr facility error
Facility is identified by the prefix string in the error message.
For example, if you get ORA-7300, "ora" is the facility and "7300"
is the error.  So you should type "oerr ora 7300".
If you get LCD-111, type "oerr lcd 111", and so on.
```

Here are some common error messages:

ORA-12012: *error on auto execute of job XX*

An error was caught while doing an automatic execution of a job. Look at the accompanying errors for details on why the execute failed. If there are broken jobs, unschedule and reschedule those jobs.

ORA-04031: *unable to allocate XXXX bytes of shared memory ("XX","XX","XXX","XX")*

Database's shared pool has become too fragmented and cannot find enough contiguous space to load object. Best thing to do is stop and restart the database to flush the shared pool.

ORA-01653: *unable to extend table XXXX.XXXX by XXXX in tablespace XXXX*

Failed to allocate an extent of the required number of blocks for a table segment in the tablespace indicated. Use ALTER TABLESPACE ADD DATAFILE statement to add one or more files to the tablespace indicated.

ORA-01110: *data file XXXX: "XXXX"* and **ORA-01116:** *error in opening database file XXXX*
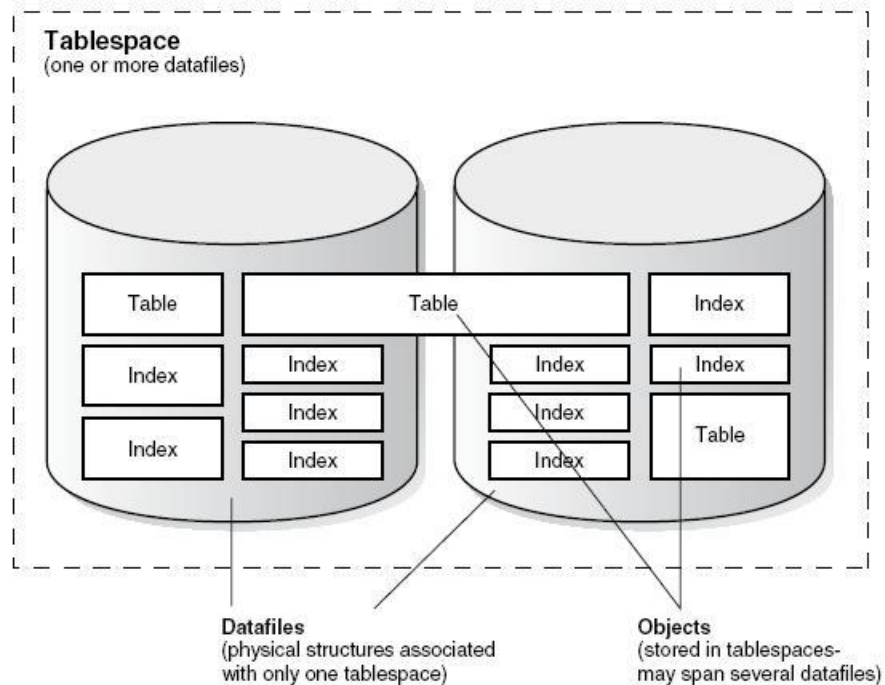
Usually the file is not accessible. Restore the database file. Consult DBA.

2.2.- Storage:

     * Oracle stores its objects in tablespaces (composed of one or more datafiles). Tablespaces containing objects that are updated frequently (such as the parametric and waveforms tables) should be monitored constantly. Paul Friberg has implemented some SeisnetWatch scripts to perform the monitoring.
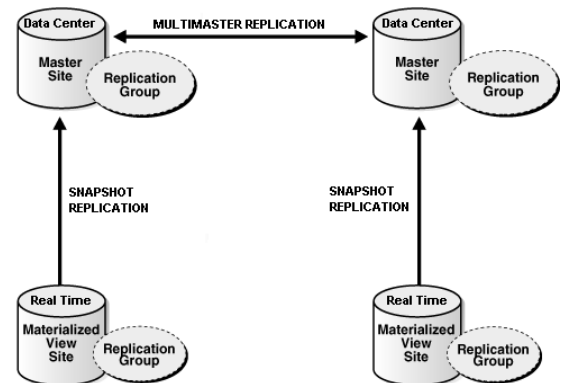
     * In the case where a tablespace is running out of space, Oracle will log an **ORA-01653** error in its alert log file. You can add more space to a tablespace by issuing the following commands:

```
% sqlplus /nolog
SQL> connect / as sysdba
SQL> ALTER TABLESPACE <tname> ADD DATAFILE </path/fname> SIZE <nb>M;
SQL> quit
```

Tablespace
(one or more datafiles)

Table	Table	Index
Index	Index	Index
	Index	Index
Index	Index	Index
	Index	Table

Datafiles
(physical structures associated with only one tablespace)

Objects
(stored in tablespaces-may span several datafiles)

2.3.- <u>Replication</u>:

Snapshot replication is used to propagate
information from the RT systems to the
DC systems. Data usually flow only from
the RT databases to the DC databases.
Snapshots (also called materialized views)
are defined on the RT databases based
on their corresponding master tables
that reside on the DC Databases.
A snapshot can only have one master;
however, multiple snapshots can be
defined from the same master table.

Multimaster replication is used to
propagate information between the
different DC systems.

It is critical to monitor the replication status as all times. In NC, in-house
scripts are used to perform the monitoring.

* Push jobs are scheduled on the different databases to propagate
transactions from one database to another. If an issue arises on the target
database (for example the database is down), the push job is stopped and
deferred transactions are being queued up on the source database
 (in the **DEFTRANDEST** view).
The following query will return information on broken jobs:

SELECT job, dblink FROM defschedule WHERE job IN
(SELECT job FROM dba_jobs WHERE broken='Y');

A broken job can be restarted with:

EXEC DBMS_JOB.RUN(<job number>);

* If a transaction cannot be applied on the target database (for example
because it is violating a constraint), Oracle will log a replication error in
the **DEFERROR** view. In that case:

✓   *Connect as replication administrator:*
     sqlplus repadmin@dbname

✓   *Find the affected object(s):*
     SELECT DISTINCT SCHEMANAME,PACKAGENAME,PROCNAME FROM defcall;

✓   *Select error(s) information:*
     SELECT DEFERRED_TRAN_ID, DESTINATION, ERROR_MSG FROM deferror;

✓   *Take care of the problem.*

✓   *Re-execute the deferred transaction(s):*
     EXEC DBMS_DEFER_SYS.EXECUTE_ERROR ('deferred_tran_id','destination');

2.4.- <u>Oracle Resources</u>:

 * Ellen Yu (eyu@gps.caltech.edu) and Stephane Zuzlewski
 (stephane@seismo.berkeley.edu) are the respective DBA's
 for SC & NC.

 * Online resources:

  ✓ Oracle Support: http://support.oracle.com
  ✓ Oracle Technology Network: http://otn.oracle.com
  ✓ Oracle 10g documentation: http://www.oracle.com/pls/db102
  ✓ Oracle 11g documentation: http://www.oracle.com/pls/db112