

Alaska Earthquake Information Center

University of Alaska Fairbanks

Delivery of Earthquake Notification Systems to Emergency Managers in Alaska

AEIC Internal Report 2008-04

*by* Glenn Thompson, Roger Hansen, Josh Stachnik and Mitch Robinson

August 2008

Suggested citation:

Thompson, G., Hansen, R., Stachnik, J., Robinson, M., 2008, Delivery of Earthquake Notification Systems to Emergency Managers in Alaska, University of Alaska Fairbanks, AEIC Internal Report 2008-04.

This version was last revised: August 2008

This document, any updates to it, and any additional information are available at:  
<http://www.aeic.alaska.edu/AEIC/internal/report/2008-04/>

The Alaska Earthquake Information Center is a cooperative program between the Geophysical Institute of the University of Alaska and the U. S. Geological Survey with support from the Earthquake Hazards Programme.

DISCLAIMER

This report has not been edited or reviewed for conformity with U. S. Geological Survey and State of Alaska standards and nomenclature. The data in this report are preliminary and subject to revision. This report is released on the condition that neither the U. S. Geological Survey, nor the Geophysical Institute, University of Alaska Fairbanks, may be held liable for damages resulting from its authorized or unauthorized use.

Table of Contents

1.Heading1 1

Heading 1.1 1

1. The Development System 4

1.1 Directory structure 4

1.2 ORBSERVER CONFIGURATIONS 5

1.2.1 Orbserver sgms:6510 (summary) 5

1.2.2 Orbserver sgms:6511 (eocserver) 6

1.2.3 Orbserver sgms:6512 (eocmirror) 7

1.3 Other key programs and aliases 8

2INTRODUCTION 12

Hardware 12

3SOFTWARE 17

1.1 DBEVENTS (ANTELOPE) 17

4References 27

5Computers: 29

6Data import modules: 29

7Processes: 29

8Cronjobs: 29

9Data export modules: 29

# An Antelope-based system for delivering near-real-time seismic data to Emergency Operations Centers in the state of Alaska

Project coordination, main contacts:

Roger Hansen, [roger@giseis.alaska.edu](mailto:roger@giseis.alaska.edu)

Mark Roberts, [mark.roberts@alaska.gov](mailto:mark.roberts@alaska.gov)

# Computer Networking

Prior to installation of a computer at each remote EOC it was necessary to configure the network connections appropriately. Had this not been done, its very unlikely that data flow could have been established in the short time available at each EOC, so a return visit may well have been necessary. The first piece of information required was the IP address the remote EOC intended to assign the computer (when installed). Paul Delys, the systems administrator in charge of the firewall at the University of Alaska Fairbanks Geophysical Institute (GI), had to then allow port 6510 from this remote IP address to access inverse.giseis.alaska.edu on port 6510. To enable Antelope to download data onto the remote computer, the systems administrator at the EOC also had to allow port 6510 through the EOC firewall. The other ports required through the EOC firewall were 22 (for ssh access), and 39977 and 39988 (for CISN\_Display).

The other information required prior to the organization of an installation visit was the private static ip address the EOC intended to assign to the computer and the corresponding subnet mask, gateway and DNS servers. These values needed to be programmed into the appropriate computer prior to boxing it up and shipping it.

Providing all this information had been received and the configuration had been completed, the computer would work as soon as it was switched on and was provided with an ethernet connection.

Table 1: TCP/IP information.

| Identifier | Public IP address | Public subnet mask | Machine name | Data direction |
| --- | --- | --- | --- | --- |
| Fairbanks EOC | 216.67.105.44 | [255.255.255.255](mailto:David%20Gibbs%20%3cdgibbs@co.fairbanks.ak.us%3e) | AEIC\_Fairbanks\_EOC | Pull |
| Fort Richardson | 209.165.165.93 | 255.255.255.224 | AEIC\_FortRich\_EOC | Pull |
| Anchorage Muni EOC | 209.193.41.116 | 255.255.255.224 | AEIC\_AnchorageMuni\_EOC | Pull |
| Seward EOC | 24.237.136.125 | 192.168.1.22 255.255.255.0 | AEIC\_Seward\_EOC |  |
| Soldotna EOC | 209.193.25.111 | 255.255.255.0 (internal), 255.255.255.192 public  192.168.210.233 internal, 192.168.210.1 gateway | AEIC\_Soldotna\_EOC | Soldotna EOC |
| Valdez EOC | 209.161.163.48 | 10.10.10.1 | AEIC\_Valdez\_EOC |  |
| Kodiak EOC | 208.155.82.110 | 255.255.255.240 | AEIC\_Kodiak\_EOC |  |

To configure the orbserver on inverse.giseis.alaska.edu to accept connections from these remote computers, it was necessary to edit the file pf/eocserver.pf and insert the public IP address and subnet mask inside the valid\_ip\_address hash.

For troubleshooting purposes it may be necessary to remotely login to these systems, or transfer files.

To remote login, use the following command:

ssh [eoc@ip-address](mailto:eoc@ip-address)

To transfer files, use the following command:

sftp [eoc@ip-address](mailto:eoc@ip-address)

where ip-address is given in Table 1.

The GI firewall and inverse.giseis.alaska.edu are also configured to receive connections from 137.229.29.0 (the GI wireless network, for testing purposes), and from 137.229.51.99 (an iMac in Riechardt building purchased by Bill Witte).

EOC contact information:

| Identifier | Physical address for deployment | Primary contacts | Computer networking |
| --- | --- | --- | --- |
| Fairbanks EOC | Fairbanks North-Star Borough Emergency Operations Center, 3175 Peger Road, Fairbanks, AK 99709-5499 | Barry Jennings  David Gibbs  [<dgibbs@co.fairbanks.ak.us>](mailto:David%20Gibbs%20%3cdgibbs@co.fairbanks.ak.us%3e) | Steve Smith <[ssmith@co.fairbanks.ak.us](mailto:ssmith@co.fairbanks.ak.us)>  Don Logan <[dlogan@co.fairbanks.ak.us](mailto:Don%20Logan%20%3cdlogan@co.fairbanks.ak.us%3e)> |
| Fort Richardson | State of Alaska Division of Homeland Security and Emergency Management State Emergency Coordination Center, National Guard Armory, Fort Richardson, Anchorage. | Bryan Fisher  <[b.fisher@alaska.gov](mailto:)> | David Lee <[david.lee@alaska.gov](mailto:)>  Mark Merchant <[mark.merchant@alaska.gov>](mailto:mark.merchant@alaska.gov) |
| Anchorage Muni EOC | Municipality of Anchorage Emergency Operations Center, 1305 E Street, Anchorage, 99501. | Vince McCoy  [<McCoyVG@ci.anchorage.ak.us>](mailto:McCoyVG@ci.anchorage.ak.us)  Kattaryna Stiles, Acting Director, (343 1407) | [Jo](mailto:)hn Roberts,  <[RobertsJC@ci.anchorage.ak.us](mailto:RobertsJC@ci.anchorage.ak.us)> |
| Valdez EOC | Valdez City Hall, 212 Chenea Ave. Valdez | George Keeney  (835-4560)  <gkeeney@ci.valdez.ak.us>  [Eric Phillips](mailto:gkeeney@ci.valdez.ak.us)  [<ephillips@ci.valdez.ak.us](mailto:ephillips@ci.valdez.ak.us)> | Chris Farmer  <cfarmer@ci.valdez.ak.us>(835 4313) |
| Kodiak EOC | Kodiak Island Borough Police Department, 217 Lower Mill Bay Road, Kodiak. | Bud Cassidy (486-9363), <[bcassidy@kib.co.kodiak.ak.us>](mailto:bcassidy@kib.co.kodiak.ak.us) | Kris Brewster <[kbrewster@city.kodiak.ak.us](mailto:)>, Information Systems Administrator, City of Kodiak (907-486-8668) |
| Seward EOC | Seward Fire Department (City Emergency Operations Center / Dispatch), 316 4th Avenue, Seward. | David Squires, (224-3445), <[dsquires@cityofseward.net>](mailto:dsquires@cityofseward.net)  Jan Melvin | Mike Meeks, IT Head, <[mmeeks@cityofseward.net>](mailto:mmeeks@cityofseward.net) (907 362 1855)  Phillip Oates <[poates@cityofseward.net](mailto:Phillip%20Oates%20%3cpoates@cityofseward.net%3e)> |
| Soldotna EOC | The Kenai Peninsula Borough Emergency Operations Center, 253 Wilson Lane, Soldotna, 99669. | Scott Walden  (262-2097), <[swalden@borough.kenai.ak.us](mailto:swalden@borough.kenai.ak.us)>  Bonnie Hanson | Ben Hanson <[BenHanson@borough.kenai.ak.us](mailto:BenHanson@borough.kenai.ak.us)>  Bob Jones <[Bjones@borough.kenai.ak.us](mailto:Bjones@borough.kenai.ak.us)>, 714-2110 |
|  |  |  |  |

The CISN\_Display usernames & passwords used are:

| Identifier | Installation date | Other information |
| --- | --- | --- |
| Fairbanks EOC | 12/11/07 |  |
| Fort Richardson | 04/30/08 |  |
| Anchorage Muni EOC | 01/05/08 |  |
| Valdez EOC | 08/25/08 |  |
| Kodiak EOC | 08/27/08 |  |
| Seward EOC | 08/04/08 |  |
| Soldotna EOC | 08/05/08 |  |
| Bill Witte's iMac | 01/11/08 |  |

*Software overview:*

*Server:*

*System diagram*

*Directory structure*

*Programs developed (dbevents\_aeic, delete\_event, dbsubset2orb, orbsegment, parseShakeMapArchive, rename\_webmaps)*

*Client:*

*System diagram*

*Directory structure*

*Programs developed (watch\_for\_deletes, dbevents\_aeic)*

*Project repository (Mac software)*

# 1. The Development System

## 1.1 Directory structure

The development system is located at /home/glenn/dev/ = ($DEV). Under here are several directories including:

| bin/ | Where executable code is stored |
| --- | --- |
| cache/ | Where webmaps and shakemaps are stored are “cached” on disk |
| db/ | Where databases are stored |
| lib/ | Where any code libraries are stored |
| man/ | Where man pages are stored (documentation) |
| orb/ | Where each orbserver is configured |
| pf/ | Where parameter files are stored |
| src/ | Where source code is stored |

Three orbservers are configured in the $DEV/orb directory:

| Alias | Orbserver | Description |
| --- | --- | --- |
| summary | sgms3:6510 | Creates a mirror or the live summary database at $DEV/db/dbsum/dbsum |
| eocserver | sgms3:6511 | Furnishes an orbserver with all the data that need sending to a remote EOC |
| eocmirror | sgms3:6512 | Replicates the operation of an EOC client at a remote location |

Each orbserver has the following directory structure:

| rtexec.pf | The parameter file for rtexec |
| --- | --- |
| bin/ | Contains aliases to executables needed in $DEV/bin |
| db@ | An alias to $DEV/db |
| logs/ | Logfiles from processes on this orbserver |
| orb/ | Files containing current contents of this orb |
| pf/ | Contains aliases to parameter files needed in $DEV/pf |
| rtsys/ | Contains a database of the startup and shutdown times of processes on this orbserver |
| state/ | Files containing state information needed for some processes |

## 1.2 ORBSERVER CONFIGURATIONS

## 1.2.1 Orbserver sgms:6510 (summary)

Description:

* The purpose of this orb is to produce a mirror of the live summary event database “/iwrun/sum/db/dbsum/dbsum”, which can then be modified without any damage to the live systems being caused. This developmental summary database is then stored at “$DEV/db/dbsum/dbsum”.
* origin2orbpf.pl is used to place /pf/orb2dbt packets corresponding to origins (which match certain criteria) in the live summary database on $ORB.
* orb2dbt then processes these packets, creating the developmental summary database.

Processes running:

* orbserver -p $ORB pf/summary.pf
* bin/origin2orbpf.pl -s 60 -n 4 -m 0.0 -t 60 $DBSUM $ORB logs/last\_lddate
* orb2dbt -select '(/pf/orb2dbt)' -v -overwrite -state state/orb2dbt $ORB $DB

Directories:

bin/

* origin2orbpf.pl -> /home/glenn/dev/bin/origin2orbpf.pl\*

pf/

* orb2dbt.pf -> ../../../pf/orb2dbt.pf
* summary.pf -> ../../../pf/orbserver.pf

## 1.2.2 Orbserver sgms:6511 (eocserver)

Description:

* The purpose of this orb is to act as a clearing house for iMacs at remote EOCs to connect to to download seismic data.
* origin2orbpf.pl is used to place /pf/orb2dbt packets corresponding to origins (which match certain criteria) in the developmental summary database on $ORB.
* orbdbt2orb.pl processes these /pf/orb2dbt packets and segments corresponding waveform data from $ORBWFSEG and places those data on $ORB.
* rename\_recenteq\_gifs.pl monitors the recenteqs website for new events, copies the corresponding webmaps (gif files) and places these into $DEV/cache/webmaps and send/webmaps.
* Events deleted by the Duty Seismologist using dbevents\_aeic will also result in a corresponding (event\_time).del message file being posted in send/delete.
* orbxfer2 processes new webmaps found in send/webmaps and delete messages placed in send/delete and places these on $ORB.

Processes running:

* orbserver -p $ORB pf/eocserver.pf
* bin/origin2orbpf.pl -s 60 -n 4 -m 0.0 -t 60 $DBSUM $ORB logs/last\_lddate
* bin/orbdbt2orb.pl -v -p pf/orbdbt2orb.pf $ORB $ORBWFSEG $ORB
* orbxfer2 -w send $ORB

Cronjobs running:

* bin/rename\_recenteq\_gifs.pl /usr/local/mosaic/Seis/recenteqs\_sub/quakes /$DEV/cache/webmaps send/webmaps (every 5 minutes)

Directories:

bin/

* orbdbt2orb.pl -> ../../../bin/orbdbt2orb.pl\*
* origin2orbpf.pl -> ../../../bin/origin2orbpf.pl\*
* rename\_recenteq\_gifs.pl -> ../../../bin/rename\_recenteq\_gifs.pl\*

pf/

* eocserver.pf -> ../../../pf/orbserver.pf
* orbdbt2orb.pf -> ../../../pf/orbdbt2orb.pf2dbt.pf -> ../../../pf/orb2dbt.pf

send/

delete/

webmaps/

## 1.2.3 Orbserver sgms:6512 (eocmirror)

Description:

* The purpose of this orb is to mirror the system that will be available on iMac computers at various EOCs around the state of Alaska.
* orb2orb copies all data from the eocserver (sgms3:6511) orb, including /pf/orb2dbt packets, waveform data, webmaps and delete messages.
* orb2dbt builds a parametric earthquake database from the /pf/orb2dbt packets which should match the developmental (and live) summary databases, subsetted according to criteria set on the eocserver. The output database is db/eocdb/eocdb.
* orb2db builds the corresponding waveform tables in the same database.
* orbxfer2 saves the webmaps into received/webmaps (for use by dbevents\_aeic) and the delete messages to received/delete.
* watch\_for\_deletes monitors received/delete, and acts on delete messages it sees there by running delete\_event to delete the corresponding event from the eoc database. It also removes any corresponding webmap (from received/webmaps/) and shakemap (from shakemaps/).

Processes running:

* orbserver -p $ORB pf/eocmirror.pf
* orb2orb $ORBCH $ORB
* orb2dbt -v -select '(/pf/.\*|/db/.\*)' -overwrite -state state/orb2dbt $ORB $DB
* orb2db -p pf/orb2db.pf -S state/orb2db\_data $ORB $DATADB
* orbxfer2 -p pf/orbxfer2.pf -S state/orbxfer2 $ORB

Cronjobs running:

* bin/watch\_for\_deletes -p pf/watch\_for\_deletes\_eocmirror.pf $DB received/delete

Directories:

bin/

* delete\_event -> ../../../bin/delete\_event\*
* watch\_for\_deletes -> ../../../bin/watch\_for\_deletes\*orbdbt2orb.pl -> ../../../bin/orbdbt2orb.pl\*

pf/

* eocmirror.pf -> ../../../pf/orbserver.pf
* orb2db.pf -> ../../../pf/orb2db.pf
* orb2dbt.pf -> ../../../pf/orb2dbt.pf
* orbxfer2.pf -> ../../../pf/orbxfer2.pf
* watch\_for\_events\_eocmirror.pf -> ../../../pf/watch\_for\_deletes\_eocmirror.pf

received/

delete/

webmaps/

cache/

webmaps -> ../received/webmaps

shakemaps/

## 1.3 Other key programs and aliases

***dbevents\_aeic*** (in $DEV/bin/) is used to examine event databases. Most commonly it will be run on the live summary database, the developmental summary database or the EOC database.

The alias **duty\_dbevents** is used to run dbevents\_aeic on the live summary database, in which case the parameter file $DEV/pf/dbevents\_aeic\_duty.pf is used. This has yet to be implemented – the current system wide alias points to an old version of aeic\_dbevents.

The alias **dev\_dbevents** is used to run dbevents\_aeic on the developmental summary database $DEV/db/dbsum/dbsum, and again the parameter file $DEV/pf/dbevents\_aeic\_duty.pf is used. This gives the user the permission to delete events from the database, (old) QDDS system, Shakemap system and EOC system. To handle the deletes, the script ***delete\_event*** (in $DEV/bin/) is called. This does 6 things:

1. The event is deleted from the developmental summary database.
2. A delete message is placed into $DEV/orb/eocserver/send/delete, to initiate the process of deleting the event from the EOC system.
3. The 'cancel' program is called to delete a ShakeMap corresponding to the event (if it exists)
4. The QDDS delete script is called to delete the event from QDDS. Note, this uses the old Antelope-QDDS interface. Mitch is developing a new one which will use an ignore database, and the delete\_event program will then need to create a record in this ignore database to initiate an event deletion from QDDS.
5. If it exists, the corresponding webmap is deleted from the cache at $DEV/cache/webmaps.
6. If it exists, the corresponding ShakeMap is deleted from the cache at $DEV/cache/shakemaps.

The alias **eoc\_dbevents** is used to run dbevents\_aeic on the eoc mirror database $DEV/db/eocdb/eocdb. The parameter file $DEV/pf/dbevents\_aeic\_eocmirror.pf is used.

Note that ***delete\_event*** is also run by the program ***watch\_for\_deletes*** (which runs as a cronjob) on eocmirror. In this case it does 3 things:

1. The event is deleted from the eoc mirror database, $DEV/db/eocdb/eocdb.
2. If it exists, the corresponding webmap is deleted from the local cache at $DEV/orb/eocmirror/cache/webmaps.
3. If it exists, the corresponding ShakeMap is deleted from the local cache at $DEV/orb/eocmirror/cache/shakemaps.

*NEAR-REAL TIME DISPLAY OF EARTHQUAKE LOCATIONS AND MAGNITUDES AT EMERGENCY OPERATIONS CENTERS*

*Two different solutions – which is better?*

***Questions:***

1. ***What type of computer (PC, Mac, Sun) can your organization support?***
2. ***What operating system (Windows, MacOS, Linux, Solaris) can your organization support?***
3. ***Do you care about seeing waveforms?***
4. ***How many minutes “latency” is acceptable?***

Current problems diagnosed from dbevents:

1. There are no waveforms for EHZ in any of the databases.
2. There are no arrivals for EHZ on travel:6510.
3. aeic\_dbevents does not run on MacOS.
4. dbevents does not have audible voice, so can’t use this (however, a geographic region & map can be configured).
5. when connecting to database on travel remotely, get a “can’t find ‘elev’” problem. This is not a problem when running from travel. Have to use database on inverse instead!
6. aeic\_dbevents somehow needs merging with dbevents.
7. orbdbt2orb segments waveform data multiple times. How can this be eliminated?
8. how do I import shapefiles into quakewatch and get them to work?
9. how to create waveform data images for quakewatch?
10. probably need to setup a quakewatch server if we choose this solution.
11. discrepancies between what shows up on AEIC wall, duty\_dbevents, and what is seen on Mac & travel
12. probably bigger discrepancies with what is seen via qdds
13. what about NEIC solutions and associated magnitude discrepancies?
14. how to

|  |  |
| --- | --- |
| “Antelope”  FEATURES:   * Used by AEIC (events as we see them) * Complicated, but powerful * Commercial software * Runs on MacOS, Linux and Solaris but not Windows (currently a key program only runs on Solaris) * Takes hours to install / configure * Less latency   DATA:   * Earthquake locations * Town names * Waveforms (optional) * Detection stations (optional) * Topography * Voice announcement of earthquakes | “QuakeWatch”  FEATURES:   * Designed for emergency managers (tailored) * Simple, but limited * Free (subscription required) * Runs on Windows, MacOS, Linux and Solaris * Takes < 15 minutes to install and configure * More latency * Based on an open-source GIS / easy to add new layers   DATA:   * Earthquake locations * Town names * No waveforms (yet) * All stations (optional) * No topography (yet) * No voice, just audible alarm * Links to other data: Shakemaps, Felt reports, Focal Mechanisms, Aftershock Forecasts, Tsunami Warnings * Can be filtered for a specific geographic region and magnitude threshold |

* cisn\_display took 30 minutes for 23:46:43
* antelope took 4 minutes

# Computers:

# Data import modules:

# Processes:

# Cronjobs:

# Data export modules:

|  |  |
| --- | --- |
| orb2orb | copies data to another orb |
| orb2db -r (stationlist) | creates waveform database for all stations not listed (r=reject). These data go to directories like /iwrun/op/run/db/archive. |
| orb2db\_diag -m (stationlist) | creates waveform database for diagnostic stations. These go to directories like /iwrun/op/run/dbdiag/diagnostic. |
| orb2dbt | creates triggered waveform database and writes to directories like /iwrun/op/run/dbseg/quakes. |
| orb2db\_avo -m (stationlist) | no manpage, but matches (m) the station list and writes them to directories like /iwrun/op/archive\_wf/avobaddata/avobaddata. |
| orbptrigger | triggers tdmt\_launch to be run whenever specified packets appear on the orb. |
| orb2ew | Export to an Earthworm server (for AVO stations) |
| orb2dbt\_web | Seems to create database /iwrun/bak/run/webquakes/quakes. This seems to be linked to the dbrecenteqs cronjob. |
| orb2vdl | to nsn8.cr.usgs.gov (no manpage). |

|  |  |
| --- | --- |
| checkhomerdata | ? |
| make\_archive\_dblinks | links wfdisc table to a day-volume in the archive |
| split\_archive\_database | splits out old days from an rt1.0 archive waveform database being written by orb2db |
| archive\_status\_plot | makes plots summarising the archived continuous database |
| rtreport | useful statistics on network performance / data flow |
| rtdbclean | remove old waveforms and associated wfdisc rows from real time system |
| dbsplitcron | ? |
| remove\_old\_archive\_waveforms | removes daily archived waveform directories older than 80 days. Run in conjunction with split\_archive\_database, which archives current data to daily directories after 2 days. |
| dbrecenteqs | watches a real-time db of hypocenters and generates maps and related XHTML content. |
| update\_finger | ? |
| rtdbclean | removes old waveforms and corresponding wfdisc runs from dbdisplay/dbdisplay real time database. |
| Touch\_wfdisc | ? |

|  |  |
| --- | --- |
| orbdetect | a multi-frequency STA:LTA detector |
| orbtrigger | real-time network trigger |
| orbassoc | spatial real-time associator/locator. Produces arrival, assoc, event and origin records (which are written to database by orb2dbt) |
| orbmag | real time Richter/local magnitude computation. Takes packets written by orbassoc and modifies origin packets. Optionally produces netmag and stamag packets too. |
| orbampmag (mb) | similar to orbmag |
| orbampmag (ms) | similar to orbmag |
| orbgenloc | generates a generalized gauss-newton location in real time. |
| orbwfmeas | this looks for '/db/detection' packets, performs the measurement specified, and then writes new '/db/wfmeas' packets back to the orb. |
| dbgme | seems to generate a grid of intensity or ground motion for hazard maps. Seems like data goes to /home/shake/run/db/quakes. |

|  |  |
| --- | --- |
| orb2orb | copies data packets from one orb to another |
| grf2orb | ? |
| ida2orb | imports data from an ida hub |
| q3302orb | from Quanterra 330 dataloggers |
| k22orb | from Kinemetrics Altus digitizers |
| guralp2orb | from Guralp digitizers acquired via Scream! |
| liss2orb | acquire Live Internet Seismic Server data (miniSEED?) |
| adsend2orb | important analog data from Earthworm |
| dbt2pf | seems to bring in initial releases from /Seis/catalogs/releases, but no manpage. |
| orb2dbt\_web | seems to be a database linker to data from 6511 |
| orb2dbt2orb | importing data from ice, but no manpage |
| orbwatch |  |

|  |  |  |
| --- | --- | --- |
| fk | data concentrator | 137.229.32.207 |
| earlybird | operational system (/iwrun/op) | 137.229.32.250 |
| energy | migrational system (/iwrun/mig, which is actually the modern development system) | 137.229.32.60 |
| ice | backup system (/iwrun/bak) | 137.229.32.103 |
| nordic | development system (/iwrun/dev) | 137.229.32.109 |
| inverse | exports eqs to Menlo Park & EOCs | 137.229.32.208 |
| tele | alarm response processes | 137.229.32.? |
| moment | Nagios server/display server for wall (Nagios is a popular open source computer system for network monitoring). | 137.229.32.40 |
| cdvaeic | for ShakeMaps | 137.229.32.142 |
| aeicpipe | for pipeline alarm system | 137.229.32.106 |

**EOC Project**

**BACKGROUND**

Sometime in 2004-5, Josh had set up a system that AEIC used to send database tables and waveforms corresponding to events to an EOC at Fort Richardson near Anchorage. Ft. Richardson. At that EOC an orbserver and aeic\_dbevents were run. At some point the software stopped working and was no longer maintained. We now need to get it running again, and extend it to other EOC's around Alaska.

Josh has provided the following comments regarding the system that was operational in the past:

*“ [there] was an operational system at Fort Richardson for quite some*

*time. I think it ran well for the most part. Most of the issues were*

*from poor network connections and nobody was really certain what the*

*system down there was supposed to accomplish. They weren't really sure*

*of what they wanted so I wasn't sure how to cater the software to their*

*needs. I think this is a big part of the EOC push. Hopefully each EOC*

*won't have different system goals. I believe there are some people*

*coming up to Fairbanks to work this out. I went down to Ft. Rich*

*several times to work with them. Very nice people, but nobody was*

*really sure what it's real purpose was. i think the ball is rolling now*

*with some real funding. ”*

**GOALS**

Essentially the EOCs need a map showing the locations and magnitudes of events, and their distance from towns and cities. They might also want event waveform data, and maybe continuous data too (good way to see if data are being acquired).

We will probably have 6 EOCs to send data to. And we want to mirror the system we install at each here. So whenever we walk by the screen we can do a visual “its working”. Josh was great a writing software. What we also need is to document the software and monitor it day by day.

The strategy suggested by Roger was:

1. Get it the old system running again, which should only take a day or so.
2. Set up a virtual EOC in the lab, on the AEIC wall.
3. Test on a Macintosh computer with 2 monitors, as this would be easier to maintain than Sun/Solaris at an EOC.
4. Have this system ready to show at an EOC meeting at Fairbanks in February 2007.
5. Install at an EOC in Fairbanks first, and maintain this till stable.
6. Extend to other EOC's.
7. If the Antelope-based solution looks difficult, CISN or some kind of webserver based delivery of data could be the answer. CISN pushes data. It might be more practical than running Antelope at lots of EOCs. Natasha and Mitch seem to favour a non-Antelope approach, but there might be a need to develop a lot of new software to deliver such a solution.

**ANALYSIS OF THE OLD SYSTEM**

This was made difficult by the lack of information concerning the old system:

* no report concerning the history, the design or the maintenance of the system
* no information about which programs constituted the system, or where the source or binary code was located
* no man pages

The learning curve was made somewhat steeper still since it was not clear how the system interfaced with the Antelope software, existing AEIC databases, or AEIC's complex real-time system. So steps had to be taken to learn about all of these simultaneously.

The key to understanding the real-time system is to examine the rtexec.pf file on a particular computer. This file is like a more complex form of a crontab file, but contains information about background processes to initiate and monitor, as well as environment variables and cronjobs. It is run by the the Antelope program 'rtexec', which is an executive for running real-time processes.

By looking at rtexec.pf in the $FEMA\_RUN directory (where FEMA\_RUN = /net/inverse/export/inverse/fema/run) it could be seen that the program orbdbt2orb was being run, but was pointing to a non-existent parameter file, so it could not have been doing anything. Since Josh was probably the last person to change this, its likely orbdbt2orb had not therefore run successfully for at least 1 year.

orbdbt2orb watches an orb and sends segmented waveforms to another orb. orbdbt2orb was set up to detect orbdbt2orb.pf packets from ice:6513, and then pull data from ice:6513 and write it out to inverse:6510 by invoking orb2orb. Other processes (orb2dbt and orb2db) were running, with the intention of extracting the packets placed on the inverse:6510 by orbdbt2orb, and writing them to a database along with the corresponding waveform data. Its not clear if orbdbt2orb copies the waveform data, or whether a separate orb2orb process is required to copy continuous data onto inverse:6510, so that orb2db can access it.

orbdbt2orb is looking for any /pf/orb2dbt packets. These exist on many other orbs,

depending on which solutions are desired to be processed. Now, there's a small caveat to orbdbt2orb. It simply looks for /pf/orb2dbt packets. The new and improved associator (orbassoc) spits out lots of these. As more arrivals are picked for a single event, new associations are made and /pf/orb2dbt packets created. orbassoc can be tuned to make an event association after 5,10,15...however many you want, picks are made. SO, in its current

state, orbdbt2orb will blindly process all of these /pf/orb2dbt packets. This isn't a real big deal except that a lot of extra network traffic is created. The receiving orb will be running orb2db and shouldn't duplicate waveforms in the wfdisc. This is just something to be aware of and I don't have a fix for it...yet.

Josh has provided an example packet, orb2dbtexample.pf. He also mentions that dbt2pf2.pl (see “Master Events database') reads the same packets.

**DESIGN**

This diagram gives an indication of how the new system should look:

initial\_releasesdb <--(dbt2pf2)-->ORB1 -- (orb2dbt) --> webquakes\_db

/ /

/ /

/ /

(orbdbt2orb)--> EOCserver

\

\

\

eocquakes1\_db <-- (orb2dbt) – EOC1 -- (orb2orb)

Wherever orb2dbt is shown in this diagram, orb2db is also implied.

So dbt2pf2 reads new entries on the initial releases database, and places them on ORB1 (currently ice:6513). orb2dbt then merges these data packets with an output database (currently webquakes\_db) AND also copies these packets onto EOCserver (currently inverse:6510). This is the 'clearing house' for EOC data. orbdbt2orb also copies /pf/orb2dbt packets from ORB1 and places them on EOCserver.

EOCserver would be a dedicated orb at AEIC (currently inverse:6510) containing all data necessary for the EOCs. This dedicated orb would have:

-segmented waveforms from orbdbt2orb

-database packets for all events(/db/origin, /db/assoc, ...)

orb2orb processes are then run on a computer at each EOC to copy data from EOCserver. orb2db and orb2dbt are run at each EOC to build event databases out of these packets, which can then be viewed with aeic\_dbevents. The remote EOC rtexec would run:

* orb2orb EOCserver LOCALORB
* orb2db (to archive the segmented wfs)
* orb2dbt (to archive the database tables)

Then a database exists at the EOC for aeic\_dbevents (or any other display software) to run off of.

The local (EOC) orb should be running orb2db, which will archive the waveforms in an output database wfdisc. The segmented waveforms from orbdbt2orb end up in a database which can be displayed through aeic\_dbevents. The segmented waveforms are a way to avoid streaming tons of continuous data for no good reason. aeic\_dbevents solely reads from database tables. orb2dbt writes out the database tables that deal with origins (event, origin, arrival, assoc, detection, etc..). These tables are what aeic\_dbevents monitors for incoming events. The waveforms that are shown with aeic\_dbevents should be in the wfdisc of the same database as origin, arrival...etc.

Some additional notes on the current setup:

* webquakes\_db is the nearest AEIC presently has to a master event database, as far as I can tell. Its not clear if ice:6511 receives only released events (via dbt2pf2) or if it also receives automatically detected events (probably via orb2orb from ice:6510). The latter seems likely.
* This would mean that orb2dbt places all events (that would constitute a master event database) on inverse:6510 at present. And orbdbt2orb would grab all /pf/orb2dbt packets on the real-time system, and copy those onto ice:6510.
* Something else that is unclear is the role of orb2db – from where does it get the segmented waveform data? Does orbdbt2orb copy this off inverse:6513? Or does orb2db need an orb2orb to be running to access the data? Probably the former.
* There is an orb2orb -m (stationlist) process running on inverse:6510, pulling data from ice:6510. This exists to send continuous datastreams for a good sample of stations from around the state. The EOCs wanted to see continuous datastreams for stations close to important features in Alaska.
* There is a second orb2orb pulling data from ice:6513. This just passes along database tables from ice:6513, produced by orb2dbt. This is how db/quakedb and db/quakedb\_data are getting updated, even with orbdbt2orb not running.

Changes to the current setup:

* it seems smart to modify rtexec.pf so that orbdbt2orb reads data from ice:6511 rather than ice:6513, as it would be desirable to reduce dependencies (data are copied from ice:6511 to ice:6512, and from there to ice:6513, creating 2 extra point of potential failure). Josh believes that each orb can handle a lot more traffic, and that /pf/orb2dbt packets exist on many other orbs.
* All code should end up in the CVS repository. Code for the old system is rather disorganised. The main directory is /net/inverse/export/inverse/fema/run/bin at present.
* Code should be properly commented, and summarised with man pages. Presently there is little or otherwise inadequate commenting, and no man pages.
* Josh has noted that orbdbt2orb could be rebuilt based on orbstaseg. However, he feels that this would require multiple instances of orbstaseg – one per station. And it monitors detections rather than arrivals.
* Josh feels that for robustness reasons, the EOC project should be moved to Anchorage, to avoid complications with the state microwave system, fiber-optic cables etc.
* As much as possible, use Antelope components, as these will be maintained by BRTT. Antelope evolves, and we don't want our systems to break as a result. Also search for contributed software.
* Where there isn't an Antelope module to do the job, try to wrap or augment an existing program, or at least use a similar program as a base. And share any useful programs with the Antelope Users Group.
* Need to identify a suitable Macintosh computer to run Antelope on. This needs to be able to drive two monitors.
* Josh feels orbdbt2orb should be revamped and renamed. orbstaseg would be a good basis for both. This is for a single station only. Perhaps something like orbnetseg or orbarrseg?

**CODING**

orbdbt2orb.pl has been modified:

* the code was more logically organised so that related commands were grouped together
* the directives 'use strict' and 'use warnings' were added to make sure that all variables were used in a consistent manner. Several variables clashed, and these problems were eliminated. Other variables were found to be unused, and were removed. Variables were then given global or local scope as necessary with the 'our' and 'my' commands.
* variables were declared and described at the start of the script. Previously there was no description.
* Unsupported switches were removed.
* Each command was commented and simplified where possible.
* Code was indented in a consistent manner.
* An extensive header was added, to explain the purpose of the program and which will be used to document further changes to the program.
* Repeated code was cleaned up by delegation to a new subroutine.
* Usage information was updated.
* Extensive logging was added, to aid maintenance / troubleshooting.

The new version of orbdbt2orb.pl is named orbdbt2orb\_new.pl.

**TESTING**

Currently orbdbt2orb.pl and orbdbt2orb\_new.pl are running in parallel to compare performance. Both are running in manual mode on travel, and looking at packets on ice:6511. The former is outputting to travel:6510, and the latter to travel:6511. Orb2db and orb2dbt are then setup (an instance of each for each orb) to write data to a database (db1 and db2 respectively). The path for the rtexec directory on travel is /export/travel1/run.

While it appears both programs are processing the same data packets in the same way, it is not possible to prove this, as currently no database tables are being produced. This suggests an error with the way orb2db (and orb2dbt?) is setup for each orb. It might also help to switch to automated mode for orbdbt2orb.pl & orbdbt2orb\_new.pl.

**Other important software components:**

Orbmondb: there is no manpage!

**Orbdbt2orb**

runs on inverse

but references a parameter file in a directory that doesn't exist!

So it can't possible be working!

i.e. /export/inverse/fema/pf

when it should be

/export/inverse/fema/run/pf

parameter file doesn't say much:

mode =0

pretime, posttime = 360

channels BH[ZNE],SH[ZNE],H[ZNE]

It attempts to pull data from ice:6513 and put it on inverse:6510

Do those input packets on ice:6513 exist on other orbs too? Could they come from elsewhere?

And what happens after output packets are placed on inverse:6510? Does aeic\_dbevents just pick them up from the orb? Is there a module that writes a database that aeic\_dbevents uses?

I am assuming inverse:6510 is just being used for testing purposes. In rtexec.pf there is a reference to a FEMA ip address, but this isn't used. I'm guessing this is a computer at Fort Richardson.

I've hardened/documented orbdbt2orb and called it orbdbt2orb\_new

Question is how do we want to set this up?

Presently its designed only to send data to one output orb

But if we're planning to send same data to multiple orbs, which is better:

* modify this program so it uses multiple output orbs?
* Run many instances of this program, each writing to a single orb?

Does orbdbt2orb create orbdbt2orb packets, or does it reap them?

Seems dbt2pf also reaps these packets, so presume something else is creating them. But what?

But that makes little sense, since dbt2pf clearly reads a database, not an orb.

**Dbt2pf**

This program is not running.

Rtexec is trying to run dbt2pf.pl, but the only program is dbt2pf2.pl

This is the program Josh now thinks should be called origin2orbpf.pl

Appendix 1

DHS Emergency Management Project: Progress to July 2007

Purpose:

The aim of this project is to deliver near-real-time earthquake alert capability to emergency operations centres around the State of Alaska. What this means is that within about 1 to 5 minutes of an earthquake being recorded, an audible alarm will occur at the emergency operations centre(s), and information such as the location and magnitude of the earthquake will be displayed on a computer monitor. This will automatically provide emergency operations centres with crucial earthquake information.

As noted in the previous report (May 2007) the Alaska Earthquake Information Center (AEIC) has identified two software solutions (*Antelope* and *CISN\_Display*) for delivering near-real-time earthquake alerts to an emergency operations center (EOC). Each EOC will be equipped with an iMac computer running both applications. Further background information can be found in that report.

**Update:**

All of the iMacs and extra (23") monitors were received. Three-button mice on are order - these are required for the Antelope software. The iMacs were configured so that following a power outage, there is an automatic login and restart of Antelope and CISN\_Display (dbevents needs starting manually though - full instructions will be provided).

During recent weeks, AEIC installed a new version of the Antelope software on its real time systems. This version has also been installed on all the iMacs for best compatibility. Extensive testing and reconfiguration was required. Moreover a new version of the Antelope program *dbevents* was developed, with all of the extensions required by AEIC for this project. These extensions include the ability to see zoomed-in maps of an earthquake epicenter, to add a voice announcing each event as it is detected, and to display a list of towns and other landmarks and their distance from the epicenter.

AEIC has also documented its interface between Antelope and the USGS QDDS system which feeds CISN\_Display. It has become apparent that there is often a significant delay (tens of minutes) between AEIC submitting events to QDDS and them showing up in CISN\_Display, which is unacceptable for emergency management purposes.

Further work:

There is still an issue with segmenting waveform data within Antelope which means that waveform data for all seismic stations that registered an event may not be displayed.

There are a couple of issues remaining from upgrade of dbevents which need to be resolved.

There is part of the real-time system feeding data to the iMacs which is not currently on a UPS or configured to automatic restart following a power outage.

*Glenn Thompson*

*AEIC Seismologist*

*17 July 2007*

Appendix 2

DHS Emergency Management Project: Progress to 30 September 2007

Purpose:

The aim of this project is to deliver near-real-time earthquake alert capability to emergency operations centres around the State of Alaska. What this means is that within about 1 to 5 minutes of an earthquake being recorded, an audible alarm will occur at the emergency operations centre(s), and information such as the location and magnitude of the earthquake will be displayed on a computer monitor. This will automatically provide emergency operations centres with crucial earthquake information.

As noted in the May 2007 report the Alaska Earthquake Information Center (AEIC) has identified two software solutions (*Antelope* and *CISN\_Display*) for delivering near-real-time earthquake alerts to an emergency operations center (EOC). Each EOC will be equipped with an iMac computer running both applications. Further background information can be found in that report.

**Update:**

During this reporting period substantial changes to AEIC infrastructure were made to migrate all data processing systems to a single summary event database. These changes will allow much of the data processing software and data management to be streamlined, making it easier to maintain, troubleshoot, and develop. These changes impact all downstream systems, include the system that has been under development for this Emergency Management Project.

Software is also being developed to allow the AEIC Duty Seismologist to delete bogus events from AEIC's summary event database, and for these deletes to be reflected at remote Emergency Management Centers, in Antelope and CISN\_.

Some further modifications were made to dbevents, and other programs developed, to enable AEIC-generated ShakeMaps (coloured contour maps of maximum ground acceleration) to be displayed at remote EOCs. This work is nearing completion.

In the previous report it was noted that there was a significant delay between events being detected at AEIC, and these events appearing in CISN\_Display. AEIC has undertaken a major redesign of its QDDS interface, though largely driven by other infrastructural changes at AEIC, which may have eliminated this problem or will otherwise enable AEIC to better monitor and resolve this problem.

Some components of the software systems mentioned above have now been documented in man pages for AEIC staff. This effort will continue, as such documentation greatly assists AEIC in being able to keep the software needed for the EOC project operational.

Further work:

In addition to ongoing work mentioned above:

* There is still an issue with segmenting waveform data within Antelope which means that waveform data for all seismic stations that registered an event may not be displayed.
* There is part of the real-time system feeding data to the iMacs which is not currently on a UPS or configured to automatic restart following a power outage.
* Documentation needs to be provided for Emergency Managers to understand the software installed on the iMac, and the data analysis available through that software.

*Glenn Thompson*

*AEIC Seismologist*

*15 October 2007*

Appendix 3

DHS Emergency Management Project: Progress to 31 December 2007

Purpose:

The aim of this project is to deliver near-real-time earthquake alert capability to emergency operations centres around the State of Alaska. What this means is that within about 1 to 5 minutes of an earthquake being recorded, an audible alarm will occur at the emergency operations centre(s), and information such as the location and magnitude of the earthquake will be displayed on a computer monitor. This will automatically provide emergency operations centres with crucial earthquake information.

As noted in the May 2007 report the Alaska Earthquake Information Center (AEIC) has identified two software solutions (*Antelope* and *CISN\_Display*) for delivering near-real-time earthquake alerts to an emergency operations center (EOC). Each EOC will be equipped with an iMac computer running both applications. Further background information can be found in that report.

**Update:**

The waveform segmentation problem, mentioned in the previous 2 reports, was finally resolved in November. Since November 20th we have been confident that it is performing correctly.

The first iMac was deployed at Barry Jennings at the Fairbanks North-Star Borough Emergency Management office at the Department of Transport on Peger Road, Fairbanks on December 11th. Firewall access had been preconfigured by Steve Smith (Statewide) and Paul Delys (UAF/GI).

This system exhibited no problems with the high data volume rate in the hours following the magnitude 7.2 earthquake at 0930 on December 19th in the Andreanof Islands when more than 40 aftershocks were recorded.

The iMac apparently was found 'off' a few days later, following a power outage. I have recommended that a UPS be purchased for each iMac that is deployed, which will allow the system to ride though outages of a few minutes. Beyond this, unless a generator automatically kicks-in, the iMac will shut down, and somebody on-site will have to press the start button to boot the Mac up following the restoration of power. All software restarts automatically.

Manpages for all programs were completed.

**Further work:**

* There is part of the real-time system feeding data to the iMacs which is not currently on a UPS or configured to automatic restart following a power outage. At some stage it would be prudent to switch from inverse to kobuk.
* We have offered to provide Barry Jennings any documentation or other training he feels may be necessary for staff at his office to operate this system effectively. Thus far, no requests have been received. We will make same offer to emergency managers at other sites where iMacs are deployed.
* We anticipate deploying 2 more iMacs in the first quarter of 2008.

*Glenn Thompson*

*AEIC Seismologist*

*31 December 2007*

Appendix 4

DHS Emergency Management Project: Progress to 31 March 2008

Purpose:

The aim of this project is to deliver near-real-time earthquake alert capability to emergency operations centres around the State of Alaska. What this means is that within about 1 to 5 minutes of an earthquake being recorded, an audible alarm will occur at the emergency operations centre(s), and information such as the location and magnitude of the earthquake will be displayed on a computer monitor. This will automatically provide emergency operations centres with crucial earthquake information.

As noted in the May 2007 report the Alaska Earthquake Information Center (AEIC) has identified two software solutions (*Antelope* and *CISN\_Display*) for delivering near-real-time earthquake alerts to an emergency operations center (EOC). Each EOC will be equipped with an iMac computer running both applications. Further background information can be found in that report.

**Update:**

The iMac at the Fairbanks EOC performed well, requiring just one intervention when ip addresses had been modified at the EOC without the AEIC being informed. The new ip address for the iMac simply had to be set on the Geophysical Institute firewall and on the data server (inverse) to rectify the problem.

Attempts were made to get ip addresses the AEIC needed for transmitting data from the Anchorage strong motion network.

The Natural Sciences Department at the University of Alaska purchased a similar iMac, and asked the AEIC to configure this for real-time earthquake notification. This has been running since January 11th.

**Further work:**

* There is part of the real-time system feeding data to the iMacs which is not currently on a UPS or configured to automatic restart following a power outage. At some stage it would be prudent to switch from inverse to kobuk.
* We have offered to provide Barry Jennings any documentation or other training he feels may be necessary for staff at his office to operate this system effectively. Thus far, no requests have been received. We will make same offer to emergency managers at other sites when iMacs are deployed.
* We will deploy two iMacs in Anchorage as soon as we have the necessary ip addresses.

*Glenn Thompson*

*AEIC Seismologist*

*31 March 2008*

Appendix 5

DHS Emergency Management Project: Progress to 30 June 2008

Purpose:

The aim of this project is to deliver near-real-time earthquake alert capability to emergency operations centres around the State of Alaska. What this means is that within about 1 to 5 minutes of an earthquake being recorded, an audible alarm will occur at the emergency operations centre(s), and information such as the location and magnitude of the earthquake will be displayed on a computer monitor. This will automatically provide emergency operations centres with crucial earthquake information.

As noted in the May 2007 report the Alaska Earthquake Information Center (AEIC) has identified two software solutions (*Antelope* and *CISN\_Display*) for delivering near-real-time earthquake alerts to an emergency operations center (EOC). Each EOC will be equipped with an iMac computer running both applications. Further background information can be found in that report.

**Update:**

An earthquake notification computer was successful installed at two EOCs in Anchorage this quarter. The first was installed at the National Guard Armory at Fort Richardson on April 30th. The second was installed at the Municipality of Anchorage EOC on May 1st. Each installation was comfortably completed within 1 work day. However, since the computers had to be driven down from Fairbanks, this was a 4 day trip.

In addition to these iMacs, there is are identical machines running on the 'AEIC wall' and in my office, on the giseis and GI wireless networks respectively. Moreover, there is Bill Witte's iMac in the Natural Sciences building. In total, 6 earthquake notification computers are running, and without difficulty.

On May 30th, Mark Roberts provided preliminary contact information for deployment of the four remaining earthquake notification computers. So far there has been no decision made by the individual EOCs regarding when and where to deploy these computers, nor have any ip addresses been provided.

As the previous quarterly report suggested, the software has been very stable, and consequently there has been a much reduced need for software development this quarter. With the software development phase behind, the main cause of delay in this project has been administratio n and management. Here are the necessary steps as a list:

1. Get the preliminary contact information.
2. Contact the Emergency Manager, explain the project, and ask for the address where the computer should be installed, and the contact information for the person in charge of computer networking. Convince them the computer is beneficial for them, as is not a burden in any way.
3. Contact the Head of Computer Networking. Explain the project, and the need for a static ip address and ssh access.
4. Agree a date for installation. Also determine how to get there, and make an appointment with someone who can provide access to the office in which the computer will be located.
5. Install Antelope, xtools, CISN\_Display, Perl modules and project software on the computer. Ensure it works while connected to the GI Wireless network (½ day).
6. Ask Paul Delys to allow a hole for the given ip address through the GI Firewall on ports 22 and 6510 (0 day).
7. Get a travel authorisation signed (¼ day)
8. Deploy and test (1 day).
9. Complete a travel claim (¼ day).

Experience has shown that the first 4 steps can take weeks or months.

The data server *inverse* has now been upgraded to a more powerful computer and is also now in the server room, so power should now automatically failover to UPS and a backup generator, providing greater reliability.

**Further work:**

* Determine why the program orbsegment sends more than 1 copy of the waveform data.
* Write Technical Report.
* Write User Guide, and distribute.
* Deploy the four remaining computers.

*Glenn Thompson*

*AEIC Seismologist*

*1 July 2008*

Appendix 6

Purpose:

The aim of this project is to deliver near-real-time earthquake alert capability to emergency operations centres around the State of Alaska. What this means is that within about 1 to 5 minutes of an earthquake being recorded, an audible alarm will occur at the emergency operations centre(s), and information such as the location and magnitude of the earthquake will be displayed on a computer monitor. This will automatically provide emergency operations centres with crucial earthquake information.

As noted in the May 2007 report the Alaska Earthquake Information Center (AEIC) has identified two software solutions (*Antelope* and *CISN\_Display*) for delivering near-real-time earthquake alerts to an emergency operations center (EOC). Each EOC will be equipped with an iMac computer running both applications. Further background information can be found in that report.

**Update:**

Earthquake notification computers were successful installed at four EOCs this quarter. Previously three computers had been installed in Fairbanks, Fort Richardson and Anchorage. The fourth was installed in Seward on August 4th. The fifth was installed in Soldotna the following day. Roger Hansen and Ervin Petty assisted with these installations. The sixth system went into Valdez on August 25th and the seventh (and final) system went into Kodiak on August 27th. The Valdez installation ran into difficulties as CISN\_Display would not work. Eventually we tracked this down to outgoing ports 39977 and 39988 which needed to be enabled at the EOC firewall. The other installations were comfortably completed within 1 work day.

In addition to these iMacs, there is an identical machine running as part of the AEIC display wall. Moreover, there is a similar-configured iMac in the Natural Sciences building. In total, nine earthquake notification computers are running, and without difficulty.

A user manual was deployed to each EOC iMac (via ssh) on September 12th, 2008. This is also available as AEIC Internal Report 2008-05. The instructions for configuring an iMac are also available as AEIC Internal Report 2008-06. A final project report is being written containing a summary of all the pertinent contact information as well as physical address where the computers were installed and TCP/IP information needed to reach them remotely.

**Further work:**

* There is no further work: the project is complete. However, if desired, AEIC could add additional features to the software in response to feedback by emergency managers, subject to funding being made available.

*Glenn Thompson*

*AEIC Seismologist*

*30 September 2008*