



reimagining how the Alaska
Earthquake Center catalog is
generated (part 2)

Alexandra Farrell

January 2023

Antelope User Group Meeting



Photo credit: Shila Cotton, AEC



The purpose of this project is to achieve two primary goals:

- 1) Ensure that Earthquake Center staff, researchers, and stakeholders always have straightforward access to the best available Alaska earthquake catalog**
- 2) Streamline the workflow from the real-time system to catalog QC to ensure that staff resources are being used effectively**



- Use a unified mechanism for posting catalog information and downstream products
- Provide analysts with the most up-to-date information available at the time of processing
- **Support delayed review of earthquakes**
- **Retain the ability to edit all aspects of the catalog at future points in time**
- Provide a mechanism for querying the best possible information at any point in time

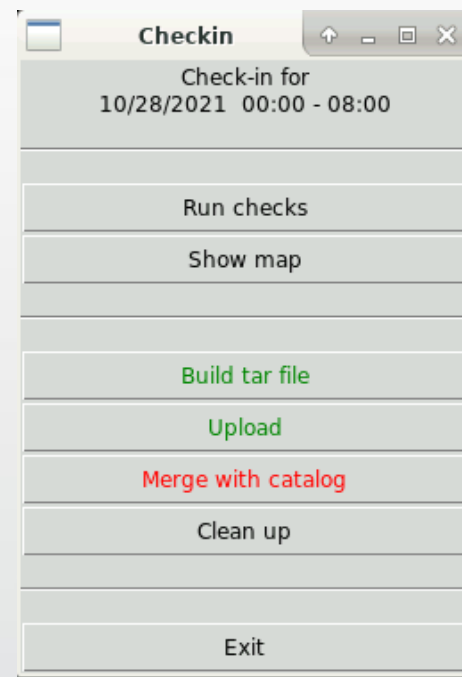
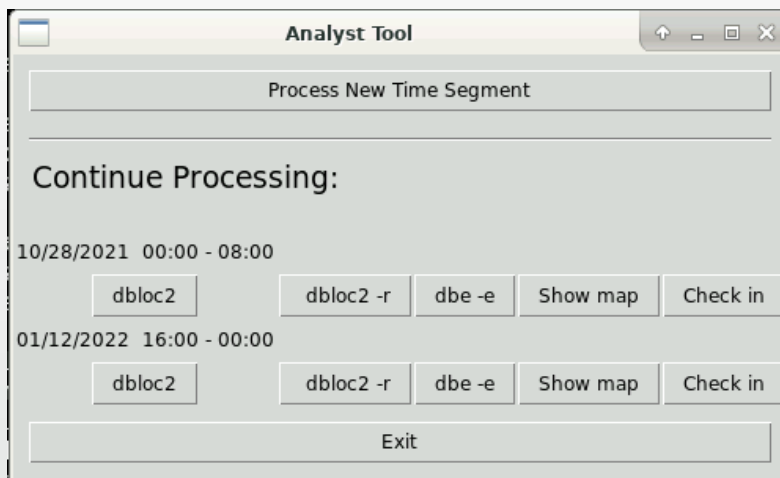


Requests:

- VPN+screen share is a notable bottleneck to processing remotely
- time segments of less than 1 day?
- easily see basic statistics (number of events, magnitude range) in segment
- map?
- **improve error checker**
- **rewrite GUIs into sustainable architecture**



- User runs analyst_tool



- Allows interaction with dbloc and dbec
- One unified GUI for processing, error checking, and checking in



```
Removed teleseism evid=23, orid=41.
Removed acevent entry for deleted event 0224qlxnlb.
Removed acevent entry for deleted event 0224qn7pkq.
Removed acevent entry for deleted event 0224qnddw8.
Removed acevent entry for deleted event 0224qnqrs0.
Removed acevent entry for deleted event 0224qnvlc0.
Removed acevent entry for deleted event 0224qoeden.
Removed acevent entry for deleted event 0224qojz8k.
Removed acevent entry for deleted event 0224rzwbvr.

-----
Errors for event 2:
2022/04/13 16:06:43.234 65.0232 -147.1314 d=0.0 M=0.77 auth=AK:natasha northa etype=Q : NORTHERN ALASKA
orid=159: There are fewer than 3 S-wave picks, only 2.

-----
Errors for event 8:
2022/04/13 16:51:18.417 61.0107 -148.5698 d=0.0 M=1.23 auth=AK:natasha northa etype=G : SOUTHERN ALASKA
orid=175: There are fewer than 3 S-wave picks, only 2.
orid=175: Low location quality: station gap is 195 degrees between stations GLI and SSN.

-----
Errors for event 74:
2022/04/13 17:02:50.664 57.1125 -135.6558 d=7.2497 M=1.41 auth=AK:natasha northa etype=- : SOUTHEASTERN ALASKA
orid=194: Low location quality: station gap is 216 degrees between stations U33K and S31K.

-----
Errors for event 76:
2022/04/13 17:25:21.077 61.3575 -141.8336 d=0.0 M=0.62 auth=AK:natasha northa etype=X : SOUTHERN ALASKA
orid=202: There are fewer than 3 S-wave picks, only 0.

-----
Errors for event 91:
2022/04/13 20:18:59.840 58.3628 -142.6867 d=10.0 M=1.87 auth=AK:natasha gulfak etype=- : GULF OF ALASKA
orid=212: There are fewer than 3 S-wave picks, only 0.
orid=212: Low location quality: station gap is 212 degrees between stations GAMMA and MURKIN

Exit
```




How to

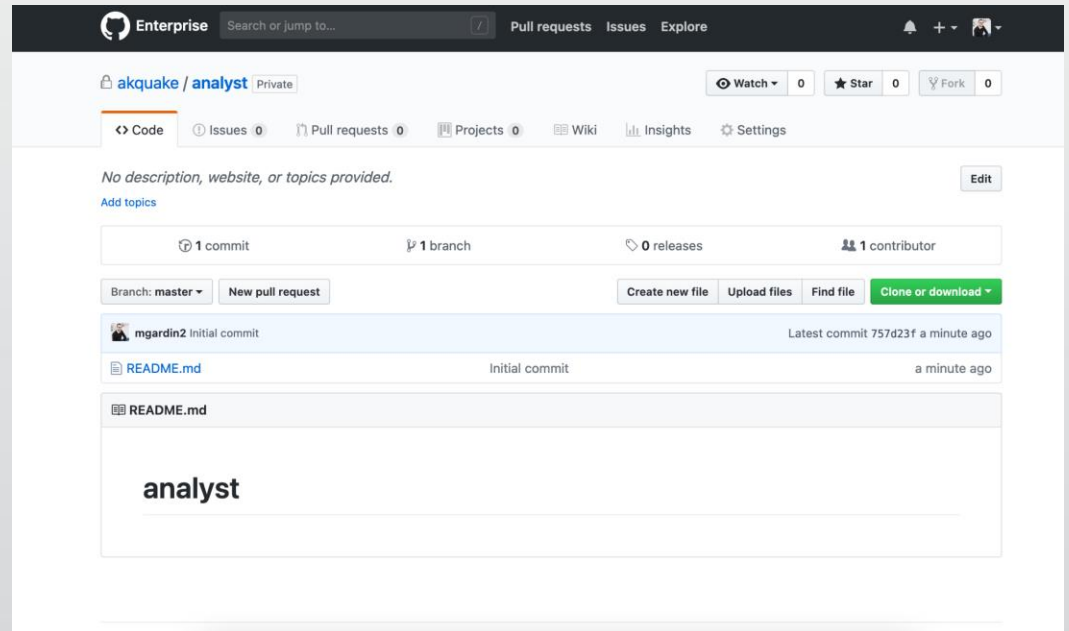
- merge rows from a local database (processing) to a database on a remote file system?
 - on workstations, handled by NFS mounts but that doesn't work over WiFi/VPN
- ensure that event updates are properly associated with existing origins?
- ensure that event additions or updates are propagated to external (website, ComCat) sources?
- ensure that events deleted by analyst are propagated to external (website, ComCat) sources?

How to

- merge rows from a local database (processing) to a database on a remote file system?

Use UA GitHub as an intermediary

- accessible from off-site
- secure
 - UA user + SSH keys
- provides some “backup”

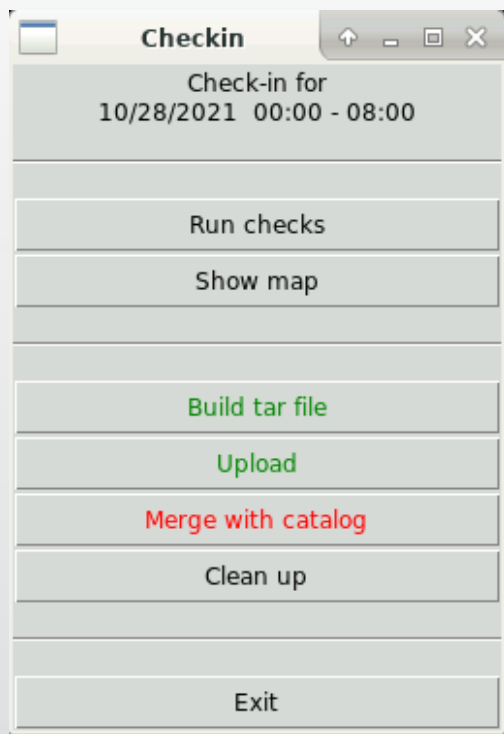


The screenshot shows the GitHub interface for a repository named 'akquake/analyst'. The repository is private and has 0 watches, 0 stars, and 0 forks. It contains 1 commit, 1 branch (master), and 0 releases. The latest commit is by 'mgardin2' and is titled 'Initial commit', made 'a minute ago'. The repository contains a file named 'README.md', which is also the latest commit. The content of the README.md file is displayed as 'analyst'.

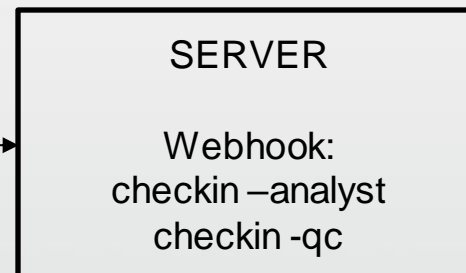


Webhooks

- a way for processes to communicate with each other
- allows information to be sent from one process to another when an event occurs
 - Example: when analyst uploads a database to GitHub, trigger the server to download and process the database
- simple web server - exposed as URL that accepts a specific data format (payload)
 - send JSON payload to URL that contains database name and if this is an 'analyst' or 'qc' database
- parse the payload and run a defined process with the options specified



webhook



- downloads from GitHub
- run update_archive
- saves added/removed event information
- removes downloaded files
- removes from GitHub



eventname

a 10-character alphanumeric string that uniquely identifies an earthquake

022333u81m

<https://earthquake.alaska.edu/event/022333u81m>

<https://earthquake.usgs.gov/earthquakes/eventpage/ak022333u81m/executive>

evid

an 8-character integer that serves as a database primary key for an event

21156180

orid

an 8-character integer that serves as a database key for a particular earthquake solution

32941259



How to

- ensure that event updates are properly associated with existing origins?
 - ensure that event additions or updates are propagated to external (website, ComCat) sources?
-
- eventnames are assigned when event is first created (real-time) or after processing is complete but before check-in (analyst review)
 - all downstream products use the eventname as the identifier
 - based on initial origin time, but associated with evid
 - will not change unless origin does not “associate” with an existing event (via orb2dbt [real-time] or dbloc2 [analyst review])



Merging



- **update_archive**
- a single program that handles database upkeep and production of external products
- two modes:
 - “delete” - removes origins from database
 - “merge” - add or update origins in database
- can run any number of external processes for each event added/updated/deleted

- Database generalization
- Input substitution

```
1  idserver :5555
2
3  # Arrays of programs to be run if an event is added/updated (external_update_programs),
4  # or deleted (external_delete_programs). Note that external_delete_programs are called
5  # PRIOR to actual deletion from the database, in case the external program needs the
6  # database entries for some reason.
7
8  # Currently, the following variables are allowed for substitution:
9  # {dbout} - database given from "-dbout" command line input
10 # {eventname} - eventname from 'aecevent' table being processed by program
11 # {evid} - event id being processed by program (unavailable for external_delete_programs)
12
13 database /aec/db/catalogs/final/%Y/catalog_%Y
14 dir_for_qc /home/aecrt/dbs_for_qc/
15
16 external_update_programs &Tbl{
17     usgs_query -db {dbout} -event {eventname}
18     db2quakeml -db {dbout} -pf /home/aecrt/run/pf/db2quakeml.pf -ev {evid} -o -a -m > /usr/local/PDL/
19     polldir/{eventname}.quakeml
20     db2mysqlpy database -d {dbout} -s {evid} -e {evid} -p /home/aecrt/run/pf/db2mysql.pf
21     usgs_query -event {eventname} -db {dbout} -dyfi -mag -mt
22     db2trigger -pf /home/aecrt/run/pf/db2trigger.pf -db {dbout} -orb 10.23.201.149:6510 -event {eventname}
23 }
24
25 external_delete_programs &Tbl{
26     db2quakeml -db {dbout} -pf /home/aecrt/run/pf/db2quakeml.pf -ev {eventname} -d > /usr/local/PDL/
27     polldir/{eventname}_delete.quakeml
28     db2mysqlpy eventname -n {eventname} -p /home/aecrt/run/pf/db2mysql.pf
29     make_aecdelete -eventname {eventname} -evid -1 -orbout 10.23.201.149:6510
30 }
```



update_archive - merge

- for each event:
 - sets output database name based on `epoch2str`
 - `database /aec/db/catalogs/final/yearly/catalog_%Y -> /aec/db/catalogs/final/yearly/catalog_2023`
 - checks if eventname is in database
 - yes: delete event and associated rows (arrivals, etc.)
 - merges input database into output database
 - runs external programs defined in parameter file



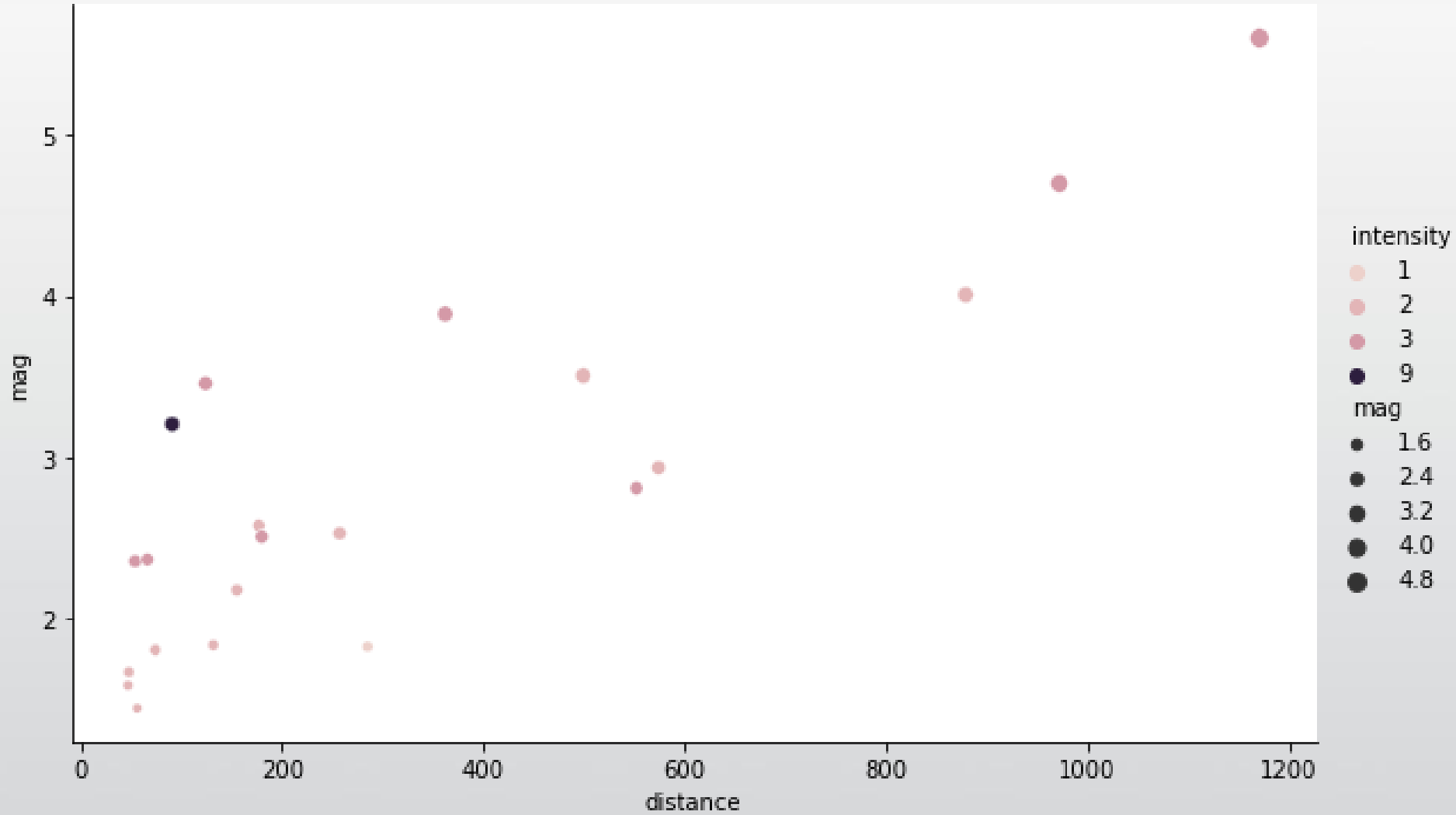
update_archive - delete

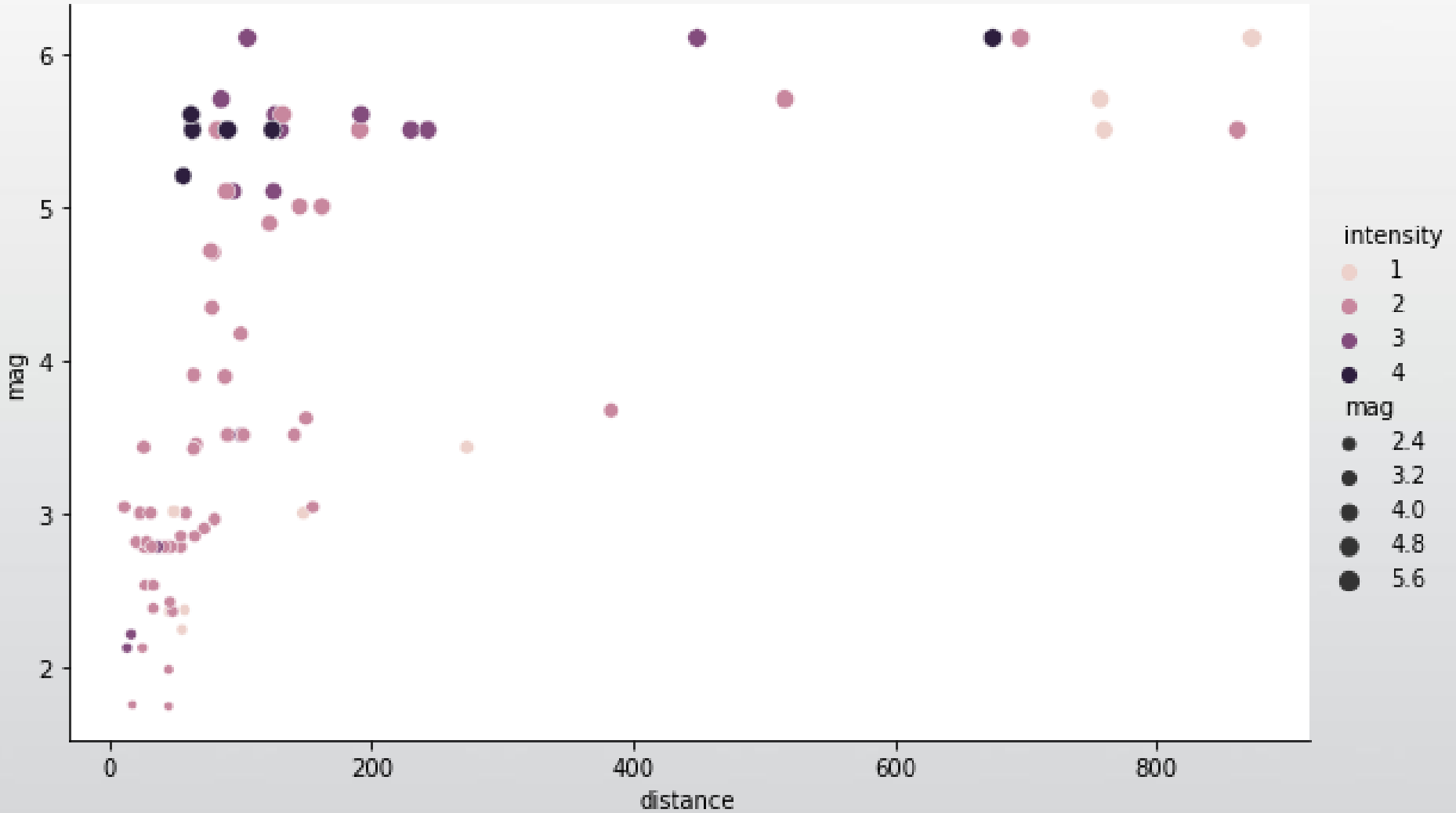
- limited to a single event (input: eventname)
 - removes event and all associated rows (arrivals, etc.) from database
 - runs external programs defined in parameter file

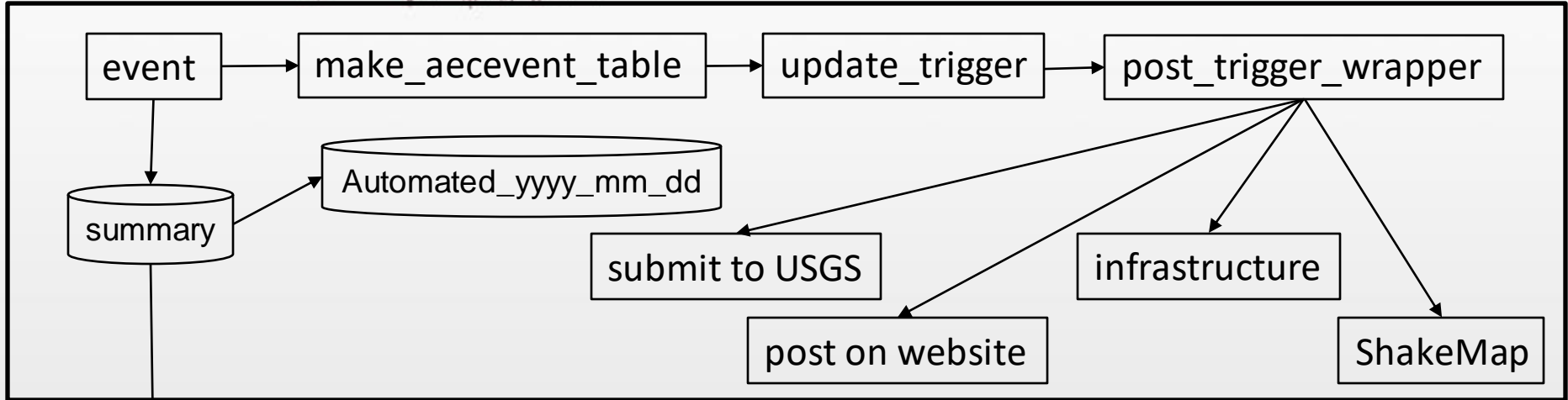


external programs

- `usgs_query` - downloads from USGS ComCat magnitudes (Mww), assigns if event was felt or not from DYFI? reports, and adds moment tensor information into 'mt' table
- `db2quakeml` - produces quakeML file (how origins are submitted to USGS ComCat system)
- `db2mysqlpy` - adds/updates/deletes events from our website database
- `db2trigger` - creates a /pf/trigging packet (used to trigger ShakeMaps)
- `make_aecdelete` - creates a /db/aecdelete packet (used to delete ShakeMaps)



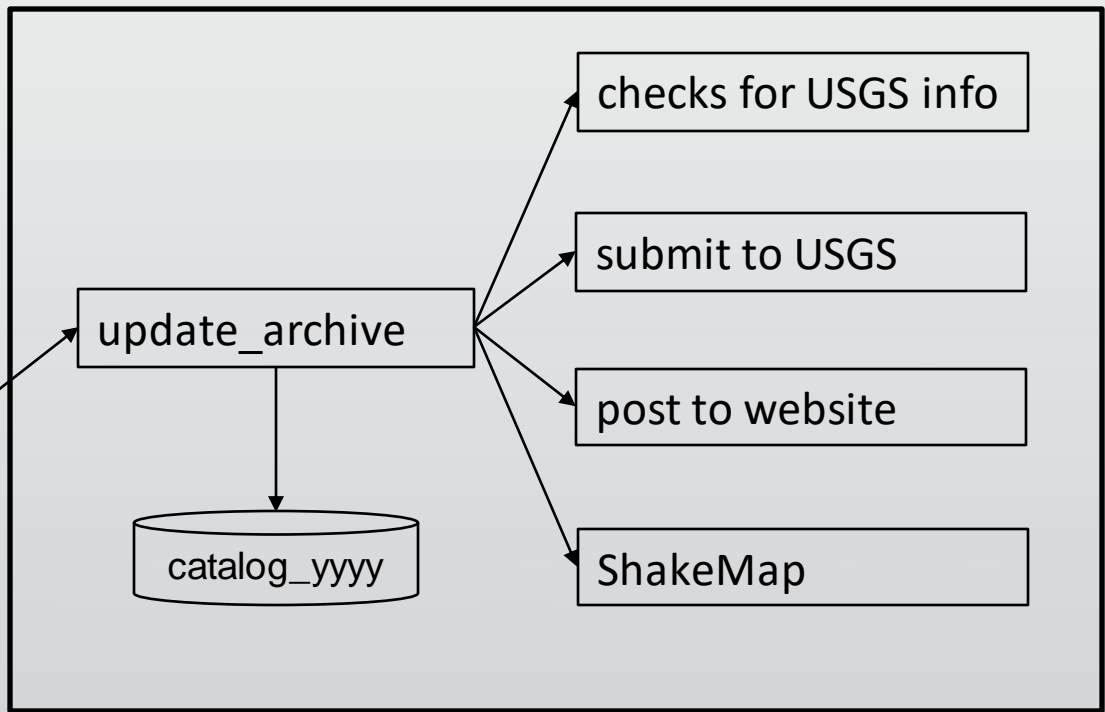




Analyst Checkout Website

Checklist	Date	Assignee	Status	Change	Url (if any)	Waveform (if any)
12/22/2012 0:00 - 0:00	12/22/2012 0:00 - 0:00	alaska	checked out	change		Download Waveform Data
12/24/2012 0:00 - 0:00	12/24/2012 0:00 - 0:00	alaska	checked out	change		Download Waveform Data
12/24/2012 0:00 - 0:00	12/24/2012 0:00 - 0:00	steve	available			
12/24/2012 0:00 - 0:00	12/24/2012 0:00 - 0:00	steve	available			
12/25/2012 0:00 - 0:00	12/25/2012 0:00 - 0:00	steve	available			
12/26/2012 0:00 - 0:00	12/26/2012 0:00 - 0:00	alaska	checked out	change		Download Waveform Data
12/26/2012 0:00 - 0:00	12/26/2012 0:00 - 0:00	steve	available			
12/26/2012 0:00 - 0:00	12/26/2012 0:00 - 0:00	steve	available			
12/26/2012 0:00 - 0:00	12/26/2012 0:00 - 0:00	steve	available			
12/27/2012 0:00 - 0:00	12/27/2012 0:00 - 0:00	steve	available			
12/27/2012 0:00 - 0:00	12/27/2012 0:00 - 0:00	steve	available			
12/28/2012 0:00 - 0:00	12/28/2012 0:00 - 0:00	steve	available			
12/28/2012 0:00 - 0:00	12/28/2012 0:00 - 0:00	steve	available			
12/28/2012 0:00 - 0:00	12/28/2012 0:00 - 0:00	alaska	checked out	change		Download Waveform Data
12/29/2012 0:00 - 0:00	12/29/2012 0:00 - 0:00	steve	available			
12/29/2012 0:00 - 0:00	12/29/2012 0:00 - 0:00	steve	available			
12/30/2012 0:00 - 0:00	12/30/2012 0:00 - 0:00	steve	available			
12/30/2012 0:00 - 0:00	12/30/2012 0:00 - 0:00	steve	available			

GitHub repository interface for 'alaska/analyst' showing a README.md file.





- Use a unified mechanism for posting catalog information and downstream products
- Provide analysts with the most up-to-date information available at the time of processing
- Allow for dataset processing from a remote location
- Support delayed review of earthquakes
- Retain the ability to edit all aspects of the catalog at future points in time
- **Provide a mechanism for querying the best possible information at any point in time**



how do I access the Earthquake Center catalog?

Antelope:

- reviewed events are in yearly databases
 - `/aec/db/catalogs/final/yearly/catalog_YYYY`
- automatic/duty reviewed events are in summary
 - `/aec/db/catalogs/summary/summary`
- ...and that's it.

Website:

- earthquake.alaska.edu
- basic event information from 1987-current is in MySQL database
- contains ALL reviewed + duty + automatic events

<https://earthquake.alaska.edu/fdsnws/ui>

points to existing
yearly databases
(1987-2023)

returns event
quakeML

can be easily
accessed via ObsPy

The screenshot displays the Swagger UI for the Antelope FDSN Server. The browser address bar shows the URL `https://earthquake.alaska.edu/fdsnws/ui/`. The page title is "Antelope FDSN Server" with a version indicator of "4.0 OAS3". Below the title, there are links for "Alaska Earthquake Center - Website" and "Send email to Alaska Earthquake Center". A "Servers" dropdown menu is set to "/fdsnws". The main content area is organized into three sections, each with a dropdown arrow:

- Station** Information on stationxml data
 - GET `/station/1/application.wadl`
 - GET `/station/1/query` :return:
 - GET `/station/1/version`
- Data Select** Information on dataselect data
 - GET `/dataselect/1/application.wadl`
 - GET `/dataselect/1/query` :return:
 - POST `/dataselect/1/query` :return:
 - GET `/dataselect/1/version`
- Event** Information on event data
 - GET `/event/1/application.wadl`
 - GET `/event/1/catalogs` :return:
 - GET `/event/1/catalogs` :return:



Questions?

Alexandra Farrell

akfarrell@alaska.edu

Matt Gardine

mgardin2@alaska.edu