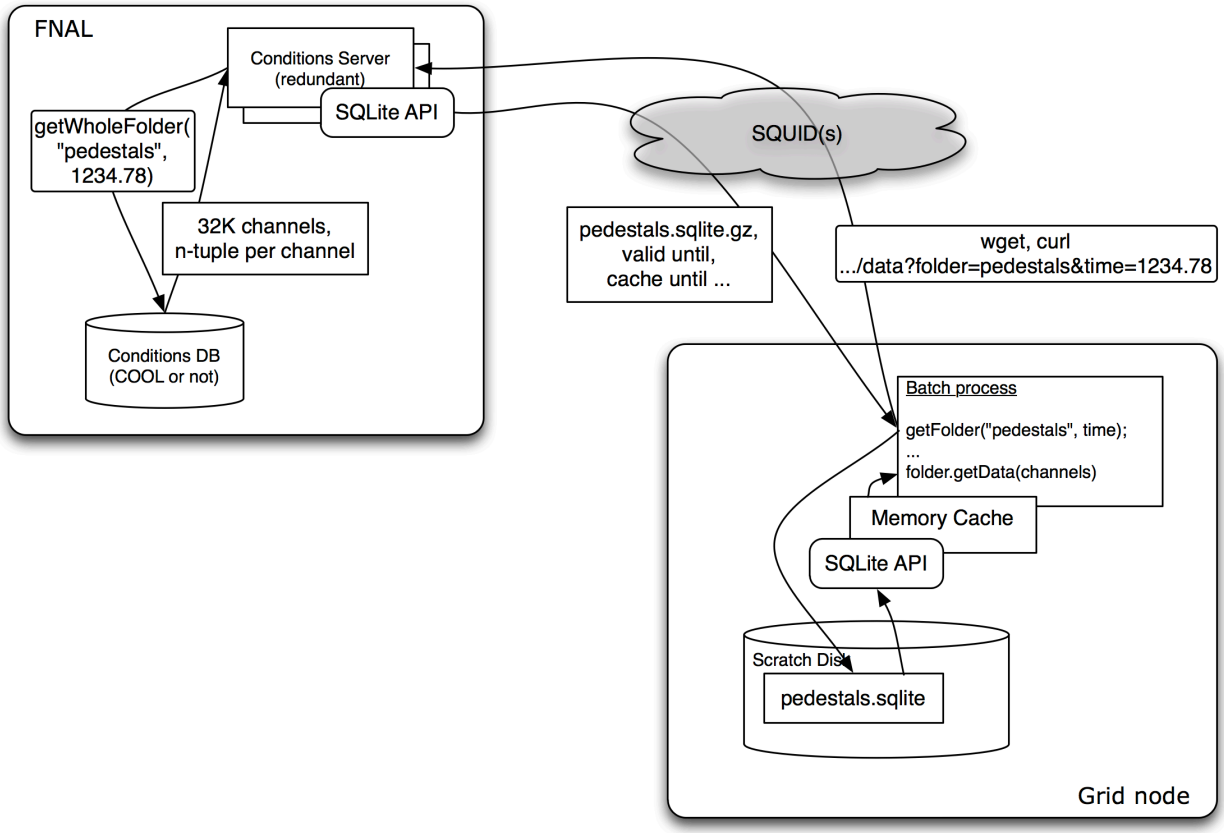
Conditions Data Delivery

web-based access to conditions data

- python general API
- HTTP Database service
- C++ client lookup library
- Demo code overview
- Performance

Architecture

MINERvA conditions data delivery



Database service

- implemented in python
- Two web requests
  1. find time boundaries (example)
  2. get data for time (example)

Client library

- Fetches all data for given time at once
- Caches based on IOV range
- Calls:
  - Folder f(name, service-url) // constructor
  - f.getChannelData(time, channel, &v1, &v2, ...)

Demo code

```
#include "uncool.h"
...
static double d[9];
lokuptime = 1290651324.1;
double halfhour = 30 * 60;
char *service_url = "http://rexdb01.fnal.gov:8088"
                    "/IOVServer/IOVServerApp.py";
...
Folder f("sample32k", service_url); // open folder
...
for(i = 1; i < 5; i++ ) {
   for (chan = 10; chan < 90; chan+=20) {

       f.getChannelData(            // get channel data
              lookuptime + halfhour * i,
              chan,
              &d[0], &d[1], &d[2], &d[3], &d[4],
              &d[5], &d[6], &d[7], &d[8], &n,  buf
          );
       ...
}
```

Performance

- test db
  - 32k channels x
  - (9 doubles + int + string) x
  - >600 entries / 1 hour apart
- Above demo -- 5 pedestal fetches, half hour apart
  - 3 misses, ~7 seconds apiece
  - 2 cache hits, free.
  - overall wall clock time 15.6s