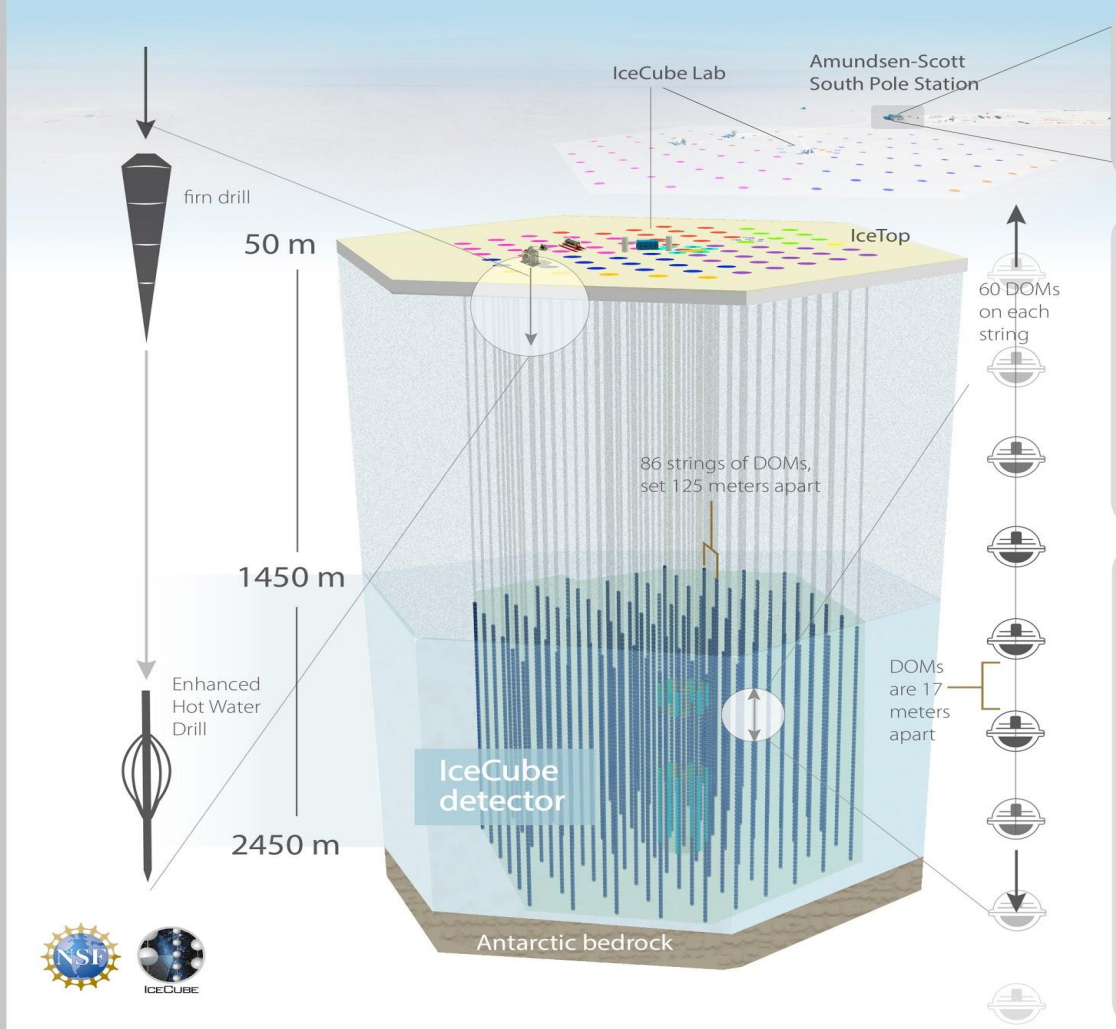
IceProd - A dataset management system for IceCube: Update

David Schultz, Juan Carlos Díaz Vélez

WIPAC, UW-Madison

The IceCube Neutrino Observatory

Design and construction

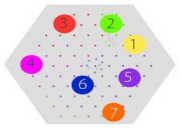


Detector Design

- 1 gigaton of instrumented ice
- 5,160 light sensors, or digital optical modules (DOMs), digitize and time-stamp signals
- 1 square kilometer surface array, IceTop, with 324 DOMs
- 2 nanosecond time resolution
- IceCube Lab (ICL) houses data processing and storage and sends 100 GB of data north by satellite daily

Detector Construction

7 seasons of construction, 2004-2011



- 28,000 person-days to complete construction, or 77 years of continuous work
- 4.7 million pounds of cargo shipped, 1.2 million of which was the drill
- 48 hours to drill and 11 hours to deploy sensors per hole
- 4.7 megawatts of drill thermal power with 200 gallons of water per minute delivered at 88 °C and1,000 psi

What is IceProd

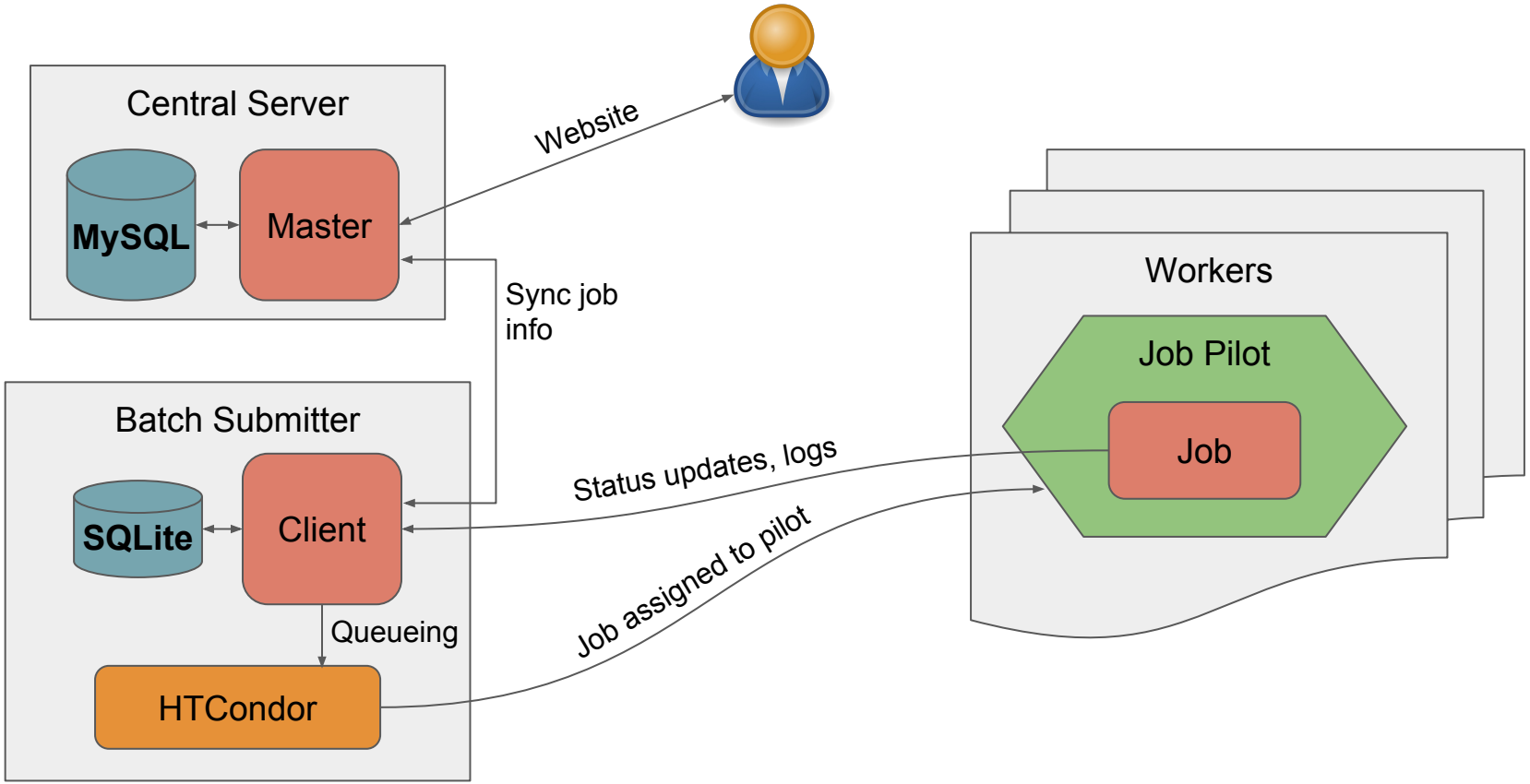
Data provenance

- Configuration - which software, what versions, when/where it ran, …

Dataset submission

- Monitor job status, resource usage
- Retry failed jobs - resubmit with different requirements

What is IceProd



Successes in the last 1.5 years

Switch from IceProd v1 to v2 in late 2016

Pilot job infrastructure

- Run multiple tasks sequentially and in parallel
  - Reduces startup overhead, connection costs with server
- Resource monitoring in real-time
  - cpu, gpu, memory, disk usage, time

Scaling bottlenecks

Scaling bottleneck at ~4k nodes

- Database is not responsive enough
  - Queuing tasks is a complex operation

Scaling bottleneck with many datasets processing

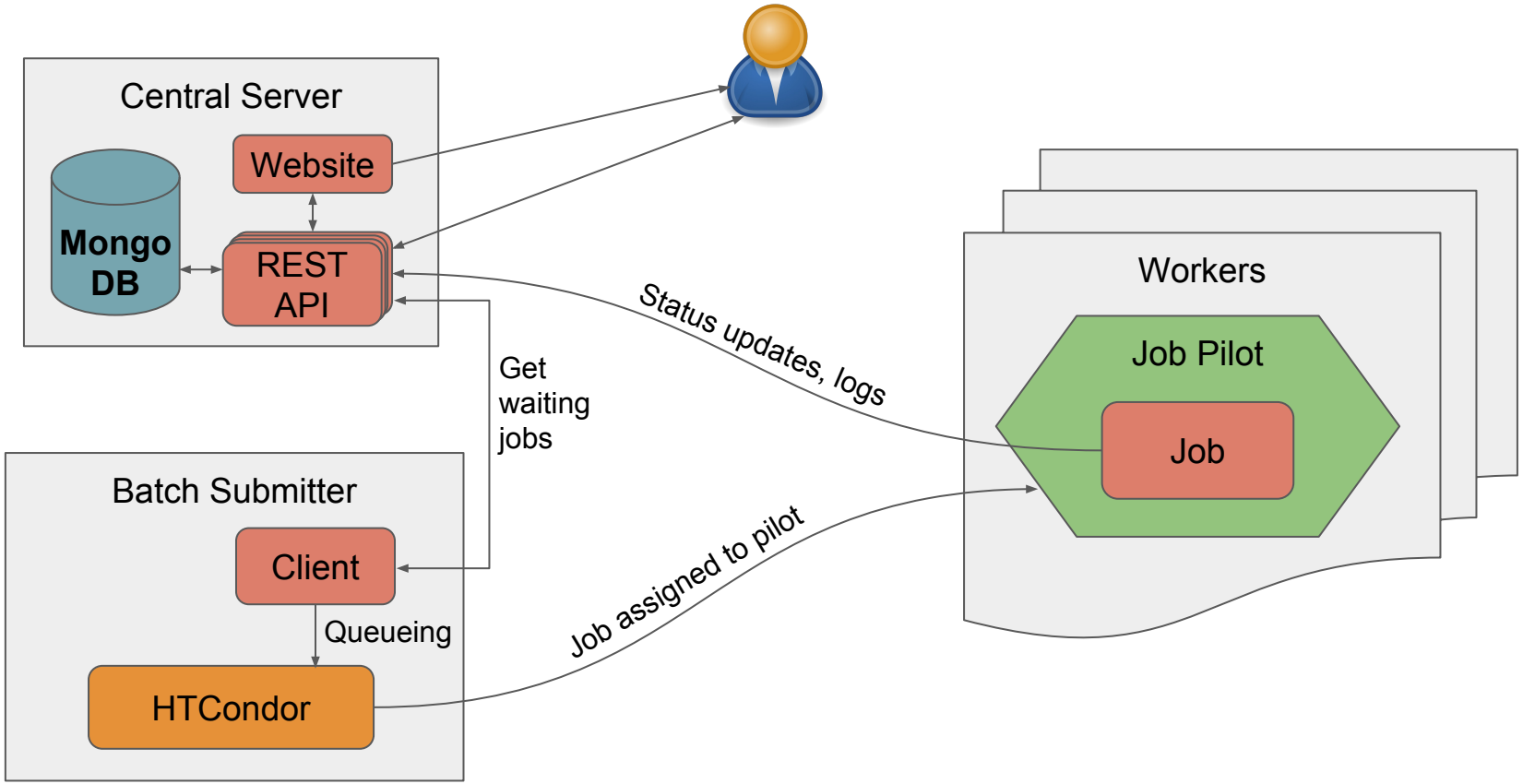
- Split / distributed database partially at fault
  - Design choice from years ago, no longer relevant
  - We only run one central instance now

Near-term plans

IceProd 2.4 release next week

- Fix the scaling bottlenecks
  - Unified, more performant database
- Simple queryable REST API
- Multi-user + authentication
  - Allow non-production users to submit datasets

Near-term plans



Future directions

- Distributed storage support
  - Intermediary file storage at more than one location
- Supercomputer support
  - Need a less connected way to still submit and monitor jobs, handle site firewalls
- Better monitoring
- Finding new bottlenecks

Summary

- Switched to IceProd 2 at end of 2016
- Scaling bottlenecks identified and being addressed
- Opening up to analyzers soon
  - Tracking and catalog of private simulations, lower level analyses
- Plenty of future work to improve

Backup

Details on database issues

Currently have a split database

- MySQL master, SQLite clients
- These need to be synced, which can cause problems
  - Updates can get lost if they timeout, or if race conditions occur
- SQLite client isn't as performant
  - Probably asking too much of it, as it's a limited clone of the MySQL db

Client connection issues caused by SQLite slowness

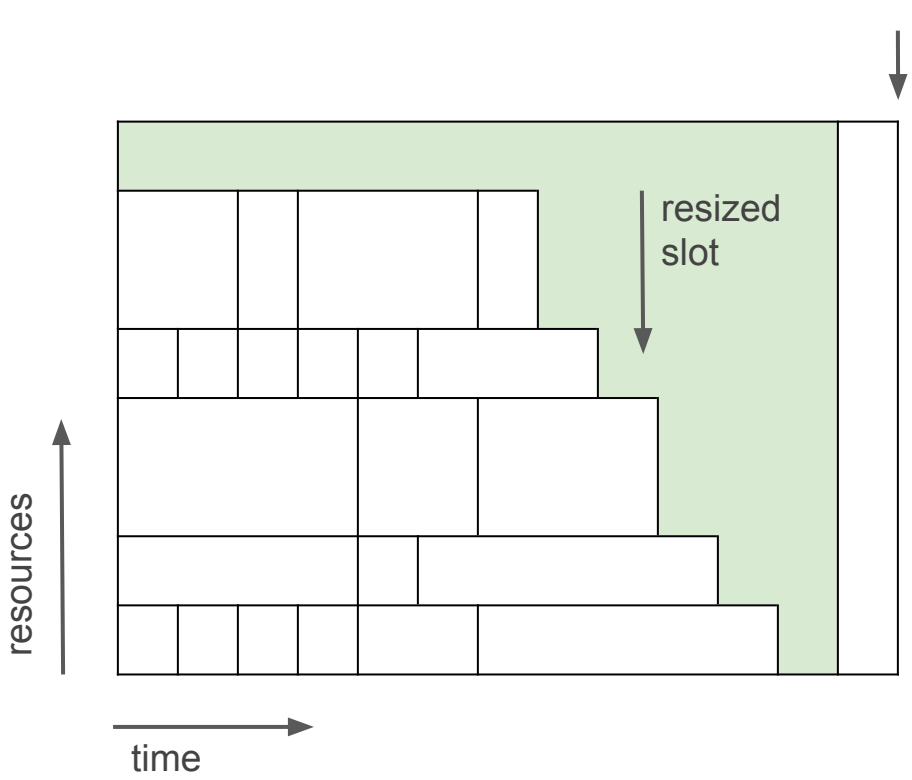
- Connection floods asking for a new task -> some succeed, most fail and retry again
- Causes 100% cpu usage, delay other calls
- Can "lose" task completions / resets

IceProd Pilots

We run a pilot inside the HTCondor job:

- Aggregate communications with the IceProd server
  - IceProd pilots are whole-node jobs: one communication link per node
- Resource monitoring in real-time
  - cpu, gpu, memory, disk usage, time
- Future: Asynchronous file transfer
  - stage in/out files for next/prev jobs while jobs execute
- Future: Dynamically resizable “jobs”

Dynamically resizable slots



give more resources to a long-running job to try and complete it before the end of the glidein