SkyDriver
A SaaS Solution for Event Reconstruction

Ric Evans
Research Software Engineer
UW-Madison
IceCube / WIPAC

CHEP 2023
The IceCube Neutrino Observatory is a cubic kilometer neutrino telescope located at the geographic South Pole focused on the search for > 1 TeV astrophysical neutrinos.
A neutrino is detected by IceCube!

Where did it come from?

Where do we need to point other telescopes for immediate follow-up observations?

Real Example: Blazar TXS 0506+056 (2017-09-22)
We need to reconstruct a Sky Map

Most accurate and detailed directional reconstruction comes by scanning across the sky in varying granularity: $O(10k)$ pixels

- Split sky into constant surface area pieces, **pixels**
- Test each directional hypothesis against likelihood
- Create directional likelihood map
- Gives most probable direction and error
Skymap Scanner

The Starting Point
Skymap Scanner – 2 Pieces

➔ 1 Central Server - generates pixels, collects likelihood statistics on each pixel, and ultimately constructs a skymap

➔ N Client Workers - computes statistics on subset of pixels, one at a time
Skymap Scanner – Queueable Pixel Data

The reconstruction/statistical test of each pixel is computationally independent.

This allows any client worker to analyze any pixel, in any order.

But a local message queue (0MQ) & manual setup does not scale well…
The SkyDriver Journey

Building an Automated & Scalable Reconstruction-as-a-Service Solution
The Problem

CASE 1:

Real-time Scans

FAST & Resource Intensive -> High Priority

⇒ O(10k+) CPUs, spun up ASAP

CASE 2:

Historical Catalog & Simulation

Steady/Predictable -> Lower Priority

⇒ Varying # of CPUs, subject to availability
Step 1: User-Configurable

Modular Central Server and Client Workers

➔ Swappable sky map reconstruction algorithms at runtime

Containerized Releases

➔ Provides version control
Step 2: Remote-Accessible MQ Broker

A remote-accessible message broker transfers pixels/stats between server and client workers, with native load balancing – workers can be anywhere!

➔ Auto-retries, message broker native failsafes, and management tools

The OMS-MQClient supports interchangeable brokers:

➔ Apache Pulsar, NATS.io, RabbitMQ

pypi.org/project/oms-mqclient/
Step 3: Increased Scalability – 3 Vectors

Instance Scalability - Concurrent Scans
➔ Possible with globally unique queues
➔ More scans

Workforce Scalability
➔ Provided by HTCondor + OSG
➔ Faster Scans + Seamless Worker Failover

Pixel Resolution Scalability, # of Messages (Pixels)
➔ Enables Automated CI Testing
Step 4: Automated Orchestration

Central servers and client managers are hosted on Kubernetes cluster

→ Additional scalability and risk management

*Larger resource pool = Faster Computation*
Step 5: User Interface + DB

➔ Now accessible from any computer, no internal knowledge needed
➔ Allows automated scanning
➔ Stores metadata, progress stats, and scan results
SkyDriver
The Event Workflow Management System (EWMS) Journey

A Generalized Massively-Scalable Task Service
Generalizing the Solution - EWMS

Event Reconstructions via Messages

Tasks via Messages
EWMS - Example Workflows

- Astronomical observations (images)
- Cryogenic electron microscopy (cryo-EM) data
- Optical Character Recognition on pages in a book
- and more!
Acknowledgments

I’d like to acknowledge the following people for their contributions to SkyDriver and EWMS

- Benedikt Riedel
- David Schultz
- Massimiliano Lincetto
- Tianlu Yuan
- Miron Livny
- Claudio Kopper
- Erik Blaufuss
- Brian Bockelman
- Christina Lagunas
- Robert Stein

National Science Foundation Funding

- OAC #2103963
- OPP #2042807
SkyDriver – EWMS Overview

1. **SkyDriver - IceCube-specific application that talks to “EWMS”**
   a. An interface for a user (or automation service) that allows launching a new Skymap Scanner instance (a “scan”)

2. **Skymap Scanner Server - a prototype WMS**
   a. Distributes events and processes results

3. **SkyDriver “clientmanager” - a prototype TMS**
   a. Launches HTCondor jobs and removes jobs when scan completes

4. **Skymap Scanner Client - a prototype Task Pilot**
   a. Processes events with physics “scanner” code

5. **Message Broker Service - raw DDS message queue**
   a. A service for distributing atomized events, from server to client(s) and client(s) to server (currently rabbitmq)
SkyDriver – Worker / Scanner Client POV

1. TMS submits a worker to HTCondor and HTCondor schedules the computer resource.
2. Worker includes:
   - EWMS Pilot
   - Scripts to execute
   - Information to connect to DDS: Auth token, address, input & output topics