AutoPyFactory (APF) 2.0
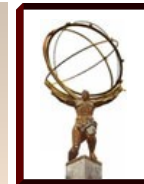

John Hover, BNL jhover@bnl.gov

Outline

- Rationales

- Design, Internals, Plugins, Features

- Wrappers

- Monitor

- Current use

- Future Plans

Rationales

Why migrate from Autopilot?

- cron-based, installed via tarball → tricky to set up.
- Some stability issues.
- one process per queue -> *condor_q* delays.

Benefits of APF v. 1.X?

- single process for all queues
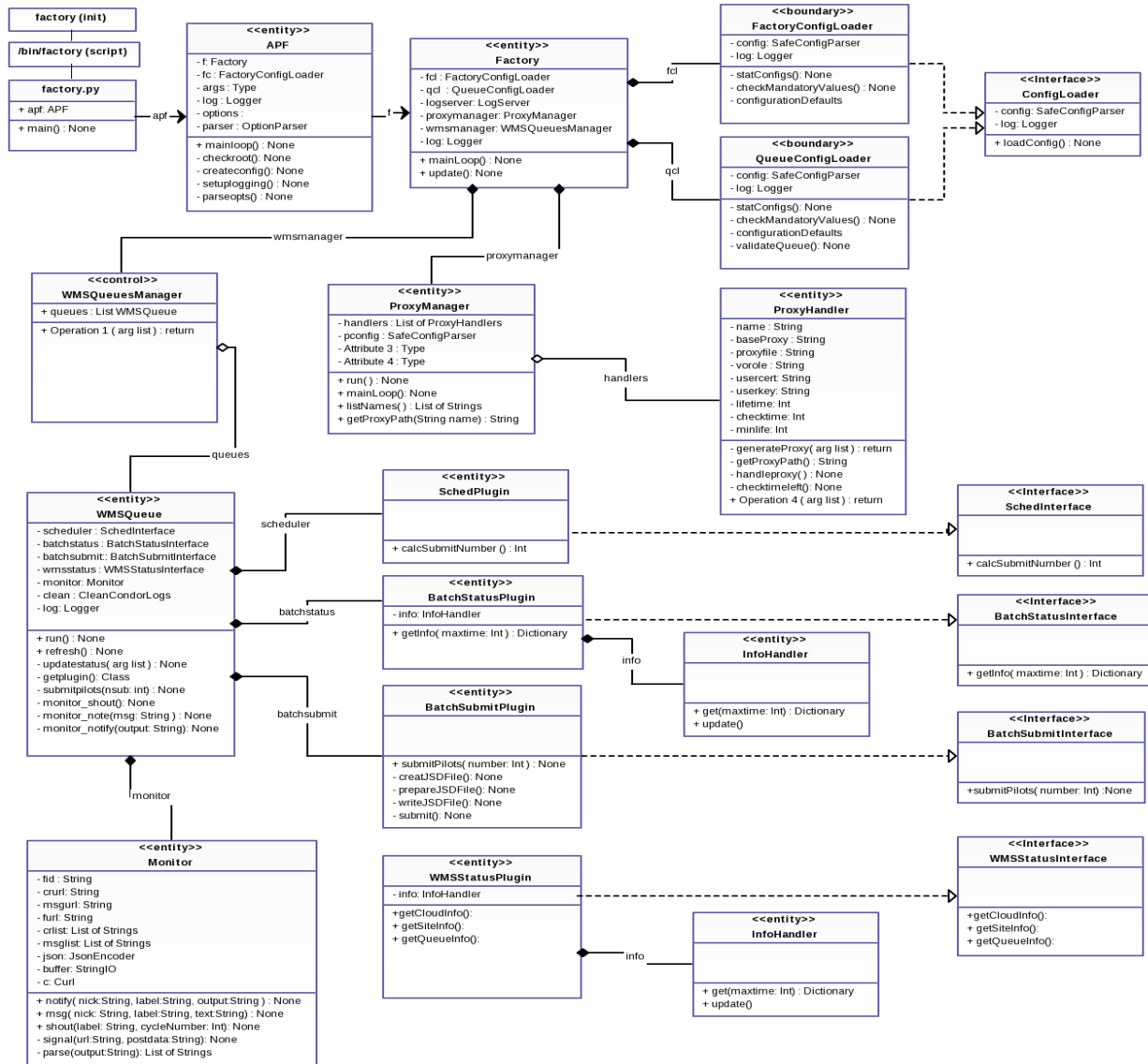- configured via files
- installed via RPM

Rationales

Why APF v. 2.X?

– Heavy multi-threading: no coupling between processing for different sites-> greater reliability (so far)

– object-orientation: threading becomes easy, plugins natural

– plugins: different behavior per APF queue

– Increase UNIX integration/deployability

– Clean install/upgrade via YUM

Design

UNIX Integration

- **Full Linux system integration:**

    - Standard service init script in /etc/init.d, configs in /etc/apf, /etc/syconfig, daemonization with dropped privileges.

    - Standard packaging (RPM) with clean upgrade path.

    - Runs as single-process daemon (no cron necessary)

    - Server log rollover handled by UNIX logrotate service.

- **Repository-based deployment**

    - Development, testing, and production YUM repositories at BNL for RPMs.

    - RPMs are signed for security.

Internals

- **Heavily Multi-threaded**
  - Failures/timeouts in one section should not affect others.

- **Fully modularized.**
  - Each function handled by self-contained object with defined interface for usage by other objects.
  - Overall system now candidate for embedding, i.e. we run it as a daemon from init now, but it could also be instantiated as a web service with a web GUI.

- **Plugin architecture**
  - Allows easy configuration, extension, and customization.

Plugins

- **WMS Status (Panda) Plugin**
  - Queries WMS for its queue config and current state, e.g. how many jobs activated? Running?

- **Batch Status Plugin**
  - Queries local batch system (e.g. Condor-G or Condor) for submitted job state info (pending, failed, submitted, etc.)

- **Batch Submit Plugin**
  - Creates the submit file and issues batch submit command(s).
  - E.g. CondorGT2, CondorLocal

- **Sched Plugins**
  - Decides exactly how many pilots to submit each cycle.
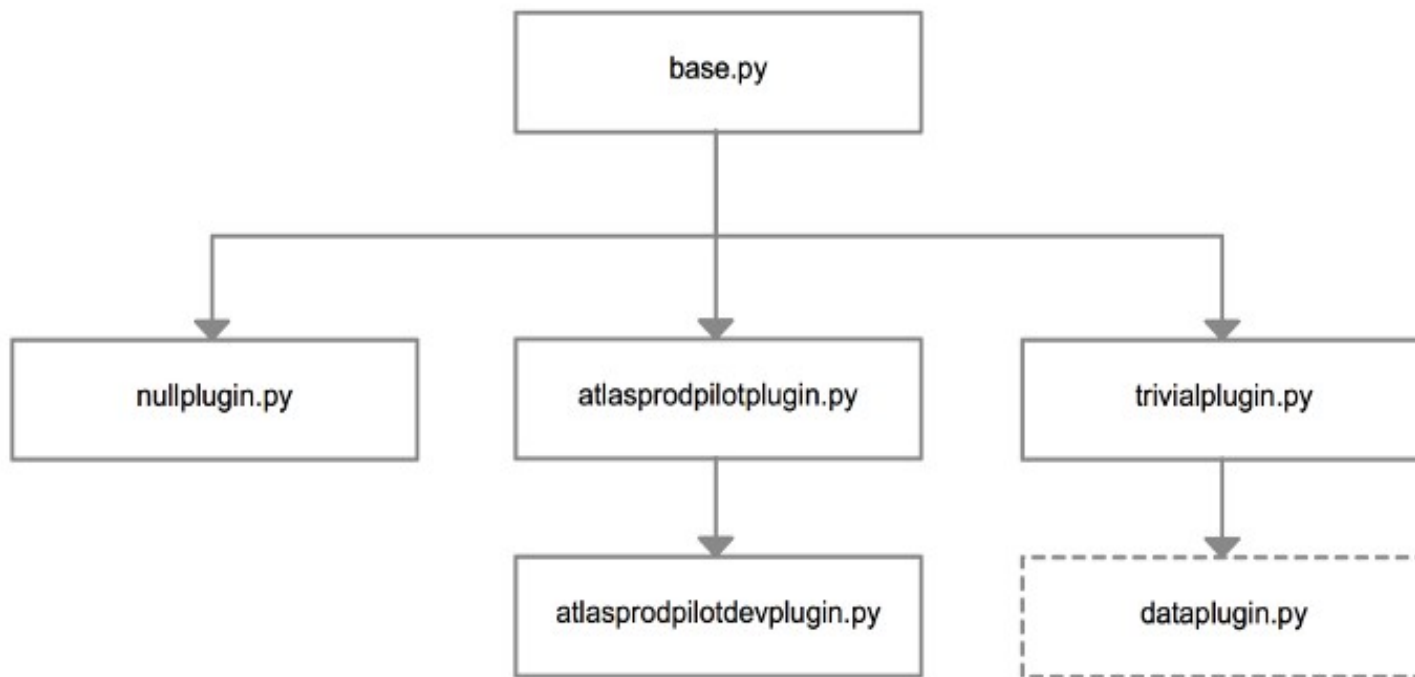 - E.g. Fixed, Activated, NQueue

Self-Contained

- **Built-in web server for batch log export**
  - No more separate Apache setup.

- **Built-in in-process batch log cleanup**
  - Configurable by time and/or disk % usage.

- **Integrated Proxy Management**
  - Allows for multiple proxy types, with each queue specifying which to use.
  - Allows for use of long-lived base vanilla proxy. No requirement for clear text passwords on system.

New Wrappers

- **Uses new wrapper system**
  - Hierarchical: Top-level wrapper never changes.
  - Maintainable/modular: Functionality split between layers of the wrapper system. Grid, VO, type, project.
  - General: Handles any type of job on any grid from any VO, (provided plugins exist for the combination).
  - Extensible: Plugin architecture allows admins and other VOs to customize functionality without re-invention.
  - Robust: Very little logic in bash, mostly in Python.
  - Better Logging: Due to near-total migration to Python.

- **Rationale**
  - Current copy-and-customize approach is unsustainable.
  - Current non-modular design requires new wrapper for every combination of characteristics for non-ATLAS users.
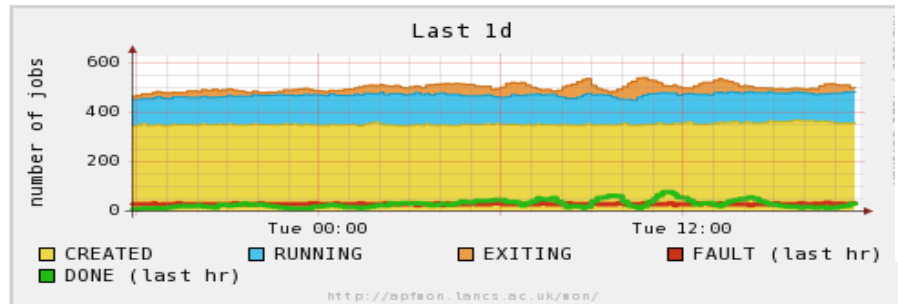
Wrapper Hierarchy

Monitor: Display

Current Experience

7 Active APF Queues

- 2 production (1 CVMFS)
- 1 test production
- 1 analysis
- 1 test analysis
- 1 non-ATLAS (OSG ITB)
- 1 trivial test

Run experience

- 120 simultaneous on CVMFS queue.
- Steady running for a week so far.

Wide Release v2.0

Very soon! (After S&C)

– We want to run at a large scale for a week at BNL

– Want to deal with any glitches in local submission at T3s.

– Want to finalize documentation based on testing.

– Still need feedback from OSG ITB test factory.

Future Plans

Better Panda Monitor Integration

- Currently posts pilot log links, but more would be nice.

More plugins

- (if needed or requested)

Allow usage with glideInWMS as WMS.

Web-based administrative interface.

Multiple configuration sources

- Queues.conf merged from exterior sources (e.g. URL backed by DB).

Reloading configuration without restart

Future Plans 2

Embedded persistence

– Allows cross-cycle comparisons and calculations.

Intelligent, instance-aware clustering

– allow multiple APF submit hosts to be aware of jobs submitted by other hosts for a particular Panda queue.

– This will allow one instance to "take up the slack" for an offline instance.

Experiment with local submission as Cloud-based VM job engine.