PanDA Workload Management System

Tania Korchuganova (Pitt) on behalf of PanDA team

CEPC workshop, 5-10 Nov 2025

PanDA = Production ANd Distributed Analysis system

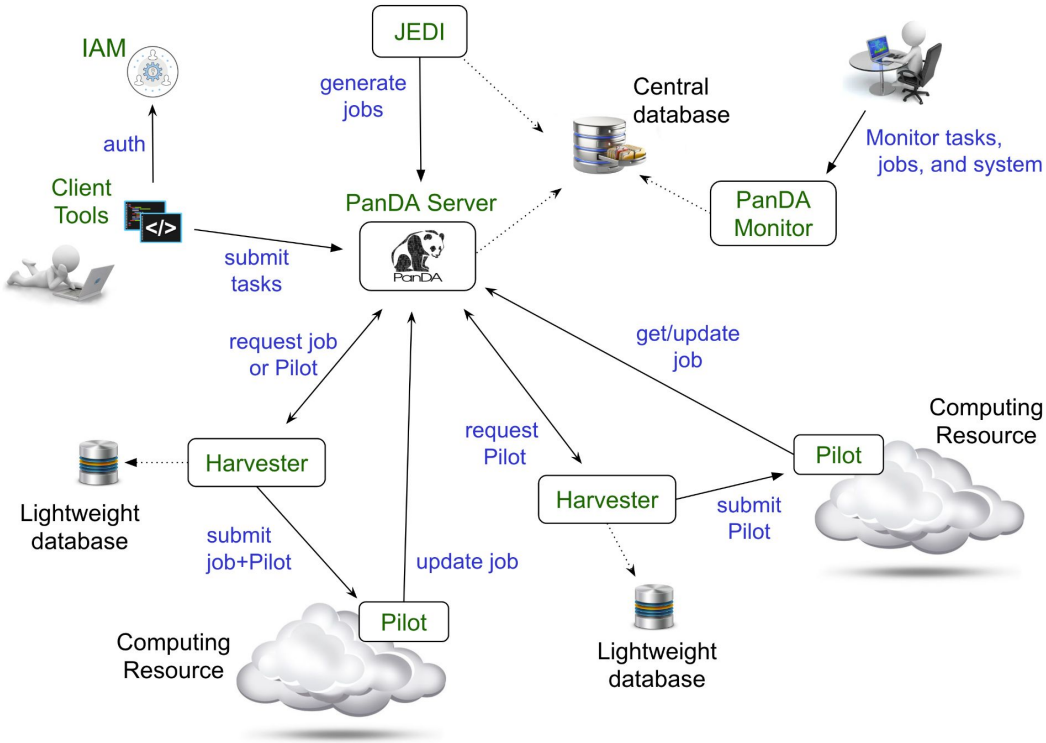
- **Data-driven workload management system** for large-scale scientific computing
- Originated in **the ATLAS experiment**, but now is being adopted **beyond HEP/LHC**:
  - Vera C. Rubin Observatory - in production;
  - DarkSide-20K, Electron-Ion Collider (EIC), Square Kilometre Array Observatory (SKA) - under evaluation
- Executes **data processing and analysis workflows at the exabyte scale**
- Integrates with **Grid, HPC, and Cloud** infrastructure
- Provides **VO-aware workload orchestration**, monitoring, and data movement

Key features

- **>1M jobs/day** across **~200 sites** globally ➔ **billions of CPU hours** for **thousands of scientists** in diverse VOs annually
- **Dynamic prioritisation**, **resource allocation**, and **automatic retries**
- **Tight integration with Rucio** for **data locality optimisation**
- **Modular and extensible design** - adaptable to a VO's global and evolving compute needs
- **Plugin system** enables **customisation** for resources, jobs, and data handling

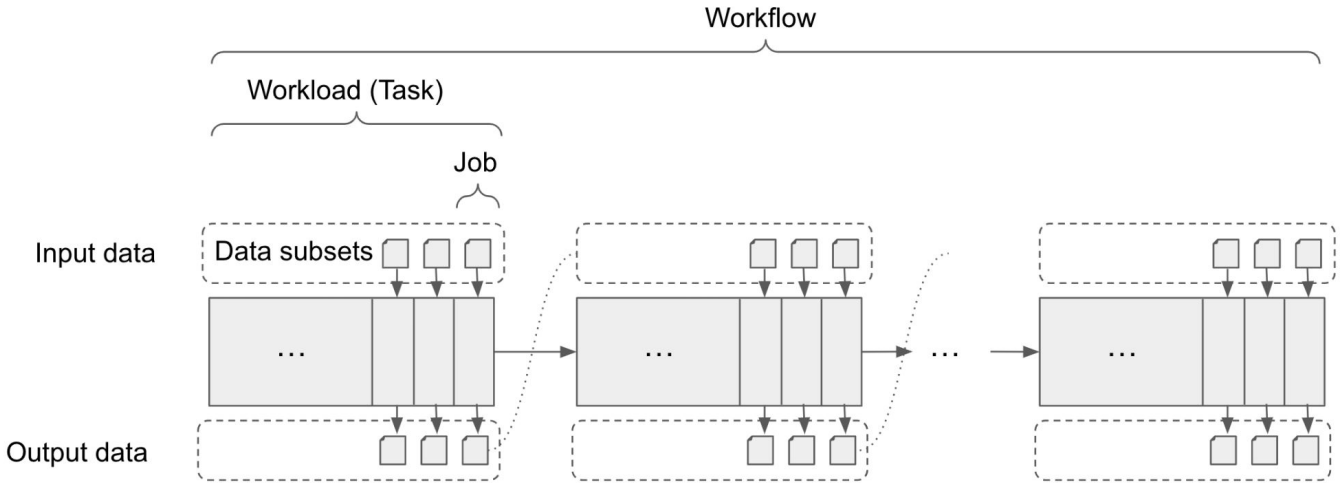
PanDA architecture

- **JEDI** – intelligent job generation and brokerage
  - dynamic sizing by **CPU, memory, disk, and walltime**
- **PanDA Server**: stateless central job pool with **REST API**
  - coordinates workload management across distributed resources
- **Harvester** – manages pilot provisioning per resource via pluggable Grid/HPC/Cloud integration
- **Pilot** – executes jobs in containers, manages & monitors payloads and errors
- **Monitor** – real-time observability and debugging with 70+ views and dashboards



Core concepts

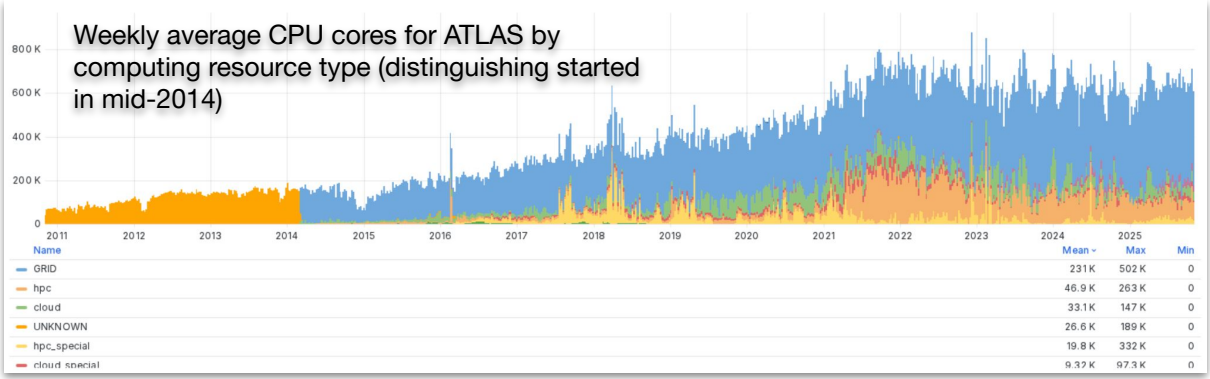
- Entities: Workflow ➔ **Workload** (Task) ➔ **Payload** (Job)
- **Workflows** define scientific objectives
- **Workloads** define **execution steps** and map to **job templates**
- Jobs process **subsets of input data**
- **Partial chaining**: downstream tasks can start before upstream ones finish
  - Enables **concurrent execution** and **efficient pipeline throughput**

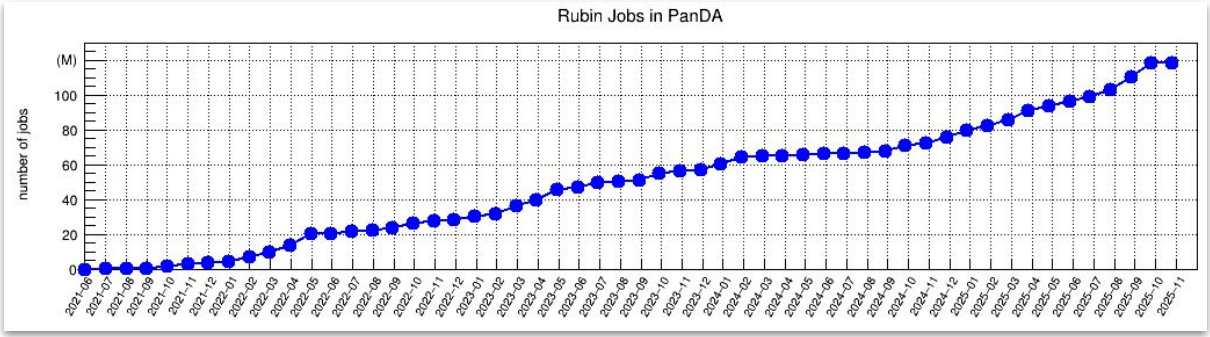


PanDA in production

- **ATLAS**: **1M+ jobs/day**, 85% multi-core usage, CERN VMs deployment with Puppet, Oracle DB backend
- **Rubin**: **>120M** jobs since May ‘21, K8s + Helm deployment on SLAC infrastructure, Postgres (CNPG) backend
- **EIC**: OKD-based deployment at BNL using HTCondor and Harvester, Postgres backend - AID2E is using it, ePIC is evaluating it

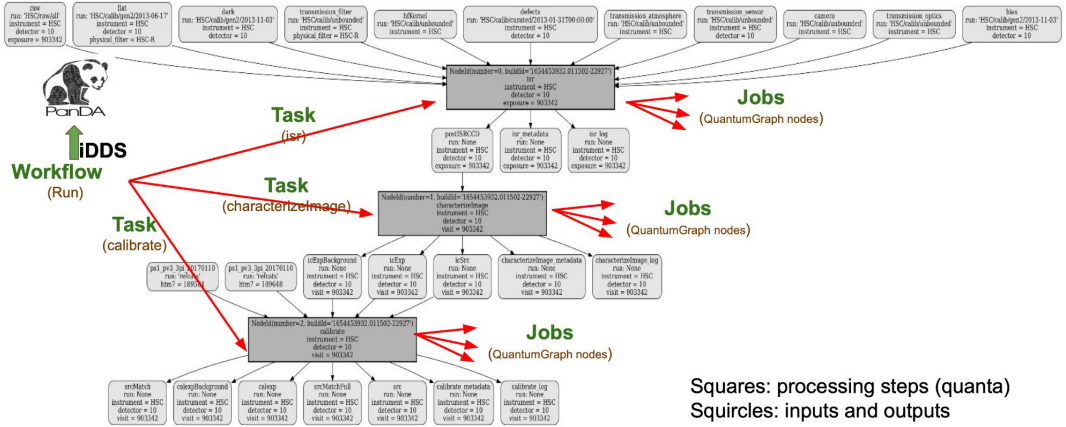
Elastic, scalable infrastructure under heavy load (200Hz req rate from pilots requesting/updating jobs)





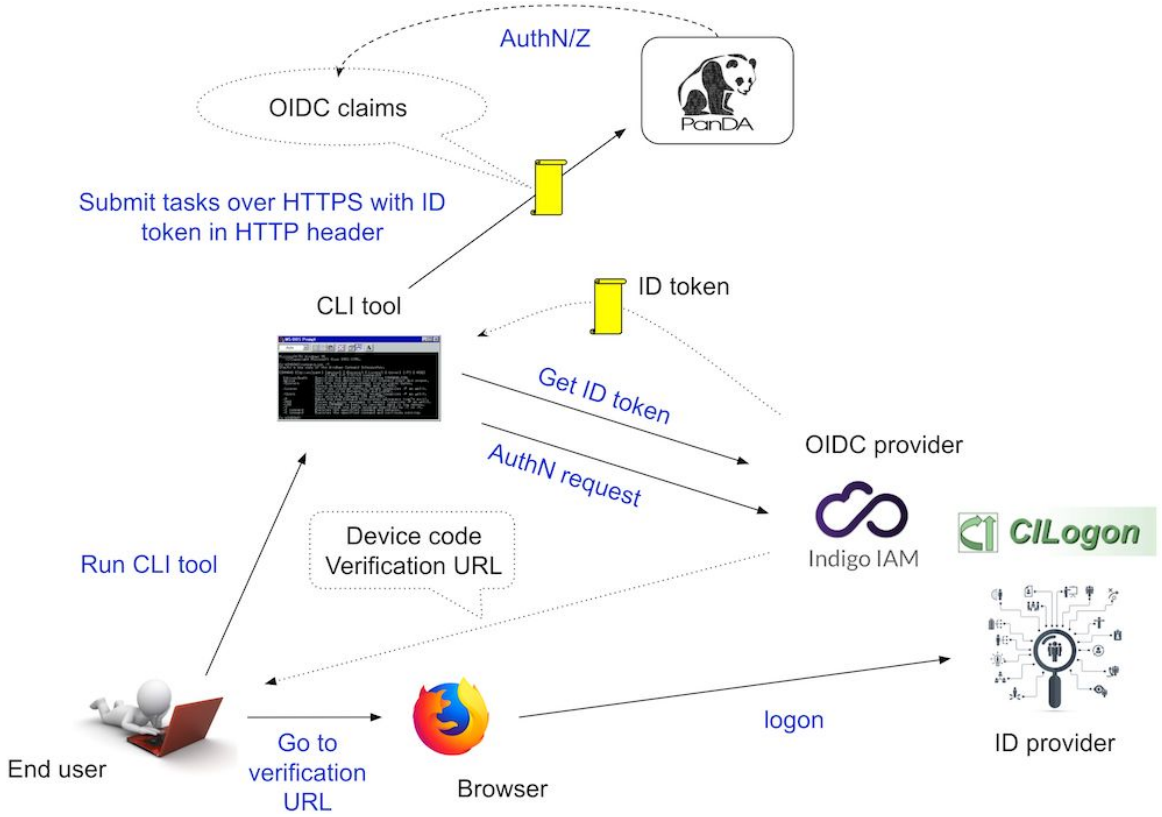
Ecosystem evolution

- **Data Carousel** - coordinates large-scale tape staging across distributed sites with **automated rule management**, **dynamic load balancing**, **real-time dashboards and alerts**, and **automated recovery** for common tape issues and downtimes.
- **iDDS** (Intelligent Distributed Dispatch Scheduling) integration for high-level data orchestration
  - Includes DAG support for complex workflow dependencies
  - Supports ML/AI workloads: HPO, active learning parameter scanning
  - Enables iterative pipelines using Bayesian optimization and Nevergrad
  - Emerging integration with Globus Compute (FaaS) for LCF/HPC workflows



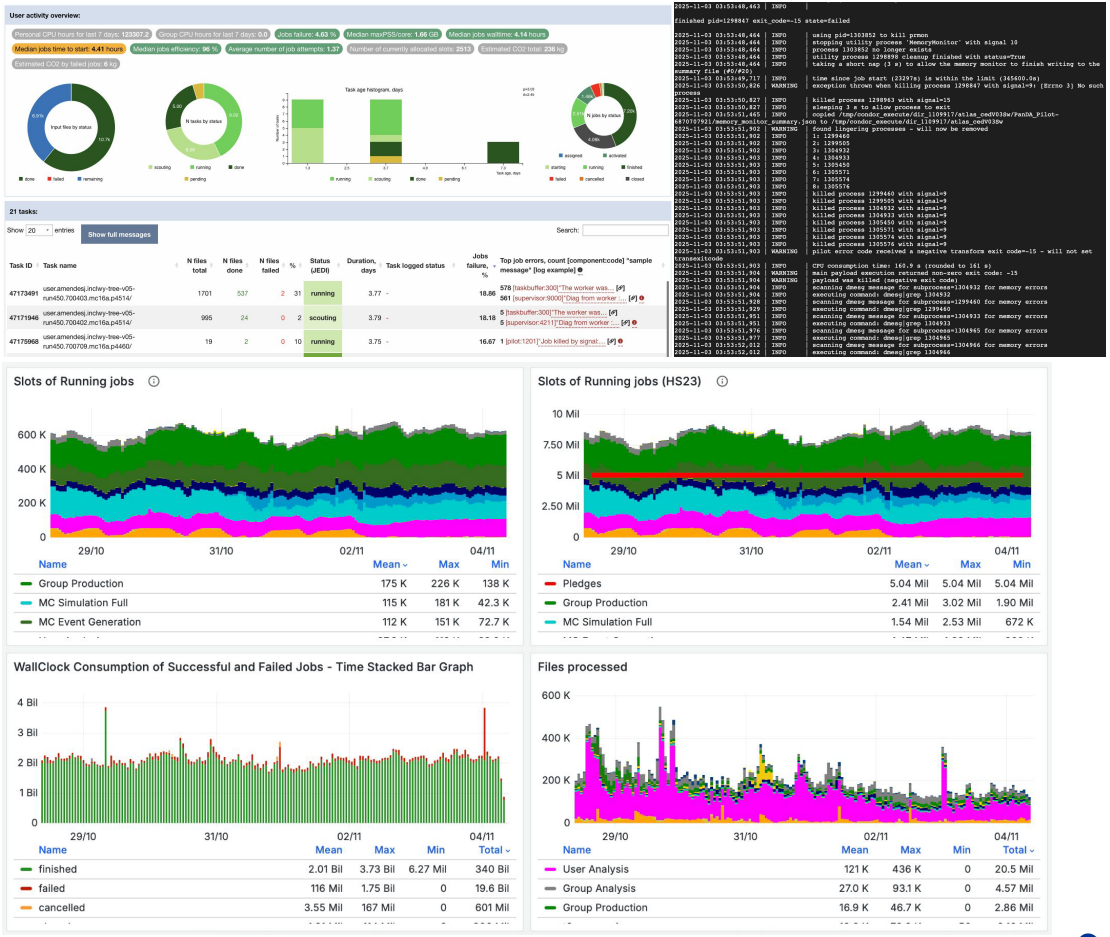
Authentication & authorization

- Supports modern identity standards: **OIDC/OAuth2.0** via **Indigo IAM**
- Integrates with **CILogon** for federated identity (CERN, universities, Google, ..)
- VO roles and group membership managed dynamically
- Supports **token-based AuthN** for pilots and users
- Legacy X.509 certificates also supported for backward compatibility



Monitoring & accounting

- Monitor: modular Django-based web UI
  - Custom dashboards for users, developers, operators, and site admins
  - Interactive drill-downs into jobs, tasks, queues, and resource usage
  - Live debugging tools with pilot logs, stdout/stderr, runtime metrics
- OpenSearch-driven monitoring & accounting
  - PanDA server/JEDI backend logs indexed in OpenSearch and visualized via Grafana
  - Accounting data exported to OpenSearch and used for VO-level reporting
  - JSON APIs and CERN IT MONIT integration for programmatic access



Multi-experiment sustainability

- PanDA server package started ~20 years ago (ATLAS)
- **Distributed Computing** and **technical requirements** have evolved significantly
- Some modules have become **complex** and **hard to maintain**
- Core architecture has proven **robustness** — it has stood the test of time ➔ we need **evolution**, **not revolution**
- **Modernisation** proceeds in the context of a **continuously operating production system**, supporting ATLAS, Rubin and EIC
- Ongoing **PanDA modernisation** is organized into **4 development streams**

1. Codebase optimization & modernization

- Code cleanup and modernization
- Enhancement of modularity
- DB simplification
- Explore CI/CD frameworks
- Standardization and auto-documentation of API

2. System optimization and enhancement

- General development
- Service metrics
- Speed up job generation
- Heterogeneity of resources
- PanDA queue and site objects

3. Outreach

- Documentation
- Community meetings
- Communication channels for announcements and issue tracking
- Video demos and tutorials
- Journal papers
- Revamp website

4. Interactive and dynamic workflow-oriented platform

- Technology choice and their scalability
- Integration of workflow execution engine
- Introducing dynamism in monitoring
- Unified and sophisticated interface
- Common authN/Z scheme

Summary

- PanDA (Production and Distributed Analysis) – data-driven workload management for large-scale scientific computing
- Originated in ATLAS, now used for Rubin Observatory, DarkSide-20K, and EIC
- Executes >1M jobs/day across ~200 sites, integrating Grid, HPC, and Cloud resources
- Core components: JEDI, PanDA Server, Harvester, Pilot, and Monitor
- Workflow → Workload → Job model enables concurrent execution and efficient pipelines
- Extended ecosystem with Data Carousel and iDDS for data orchestration, DAG workflows, and ML/AI pipelines
- Modern identity and monitoring stack: OIDC/OAuth2, OpenSearch, Grafana, JSON APIs
- Nearly 20 years of proven architecture – evolving through structured modernization for multi-experiment sustainability

Questions?