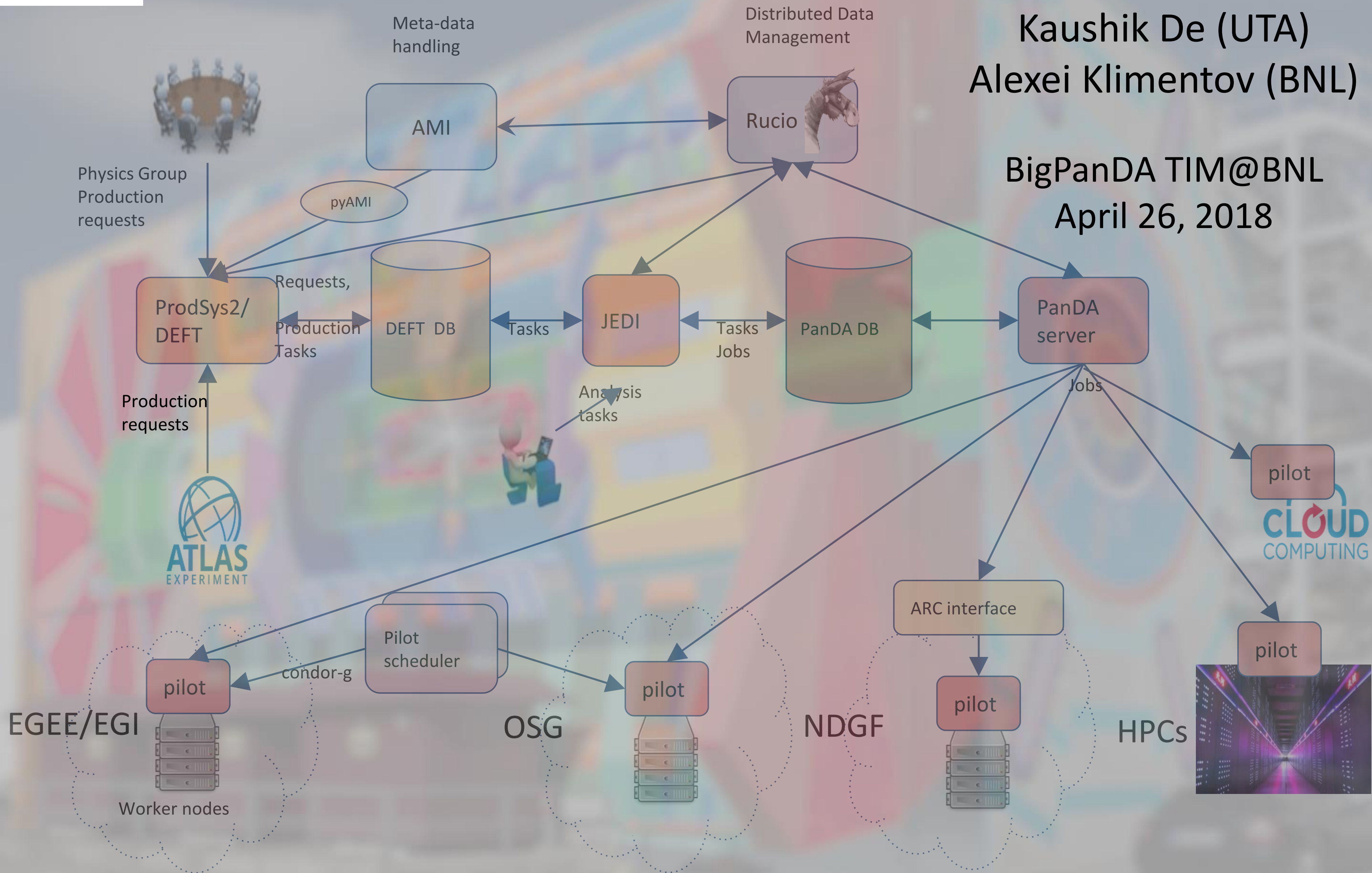
Workflow Management Software

Kaushik De (UTA)
Alexei Klimentov (BNL)

BigPanDA TIM@BNL
April 26, 2018



WLCG – HSF. *Personal* view

- LHC experiments 25+ years old
- World LHC Computing Grid : ~18 years old
  - The first WS in ~2001 (Marceille)
  - The first public understanding of the problem in 2009 (CHEP in Prague, L.Robertson talk – “landscape is changed”)
- HEP Software Foundation : ~5 years old

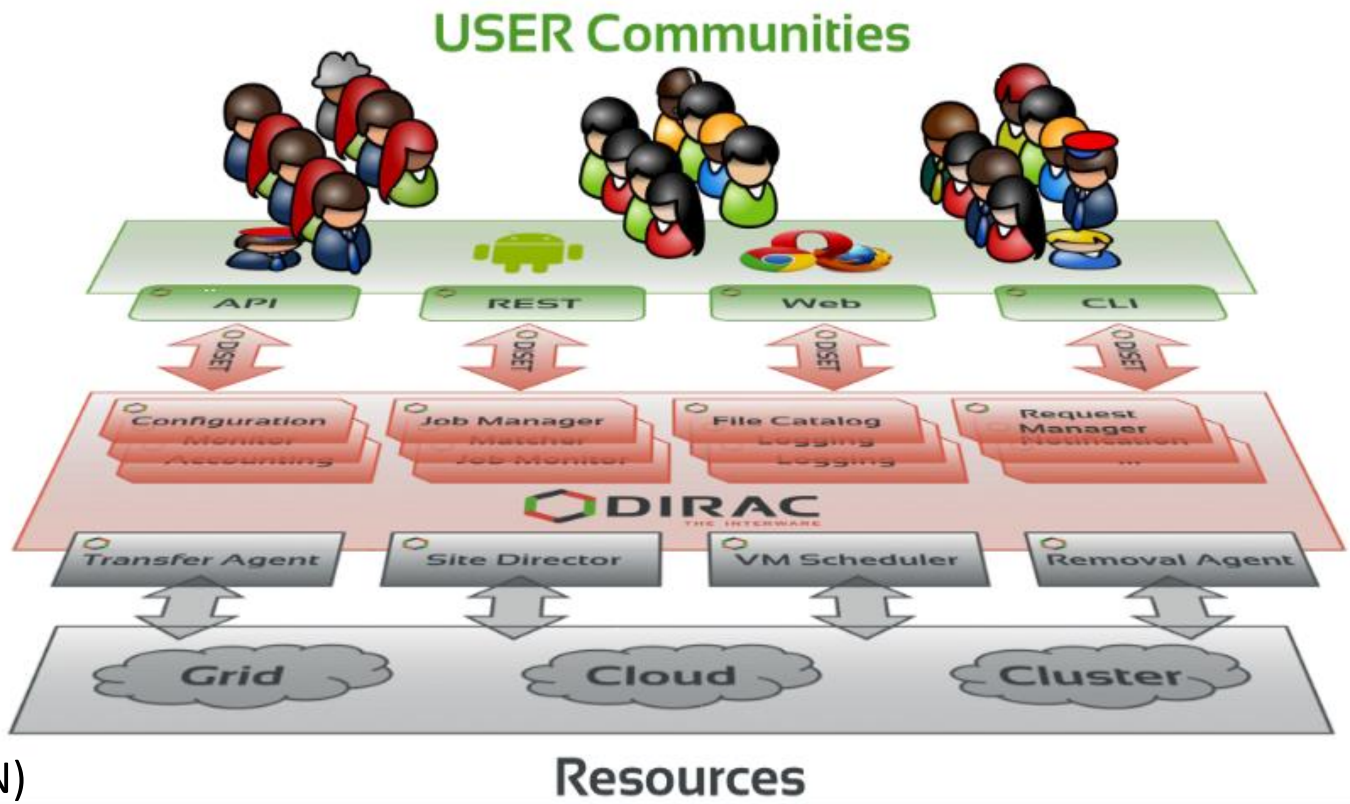
- 2014 reincarnation
  - identify inter-experiments R&D projects, to have SW&C ready for Run3/4
- 2018 reincarnation
  - Triggered by
    - Dramatic changes in computing model (thanks to BigPanDA project)
    - Community White Paper
    - S2I2 initiative in US
    - New members of coordination (potentially next project leaders)
  - Another attempt to identify common R&D topics
    - Or at least to identify common components/modules
  - “data lake” as one of main topics of today
    - One of key points is SW technology to be used for sites federation
    - Rucio is one of key components in Data Management SW stack
    - ATLAS R&D “data ocean” project
  - Requirements to WFM is not as hot as for Data Management
    - Thanks to PanDA/DIRAC
      - It still to be demonstrated for ALICE
  - ALICE view (more cooperation between ALICE and FAIR, than ALICE within WLCG) – O^2
    - many historical reasons
  - ATLAS and CMS
  - LHCb
  - Non-LHC experiments (“small” experiments)

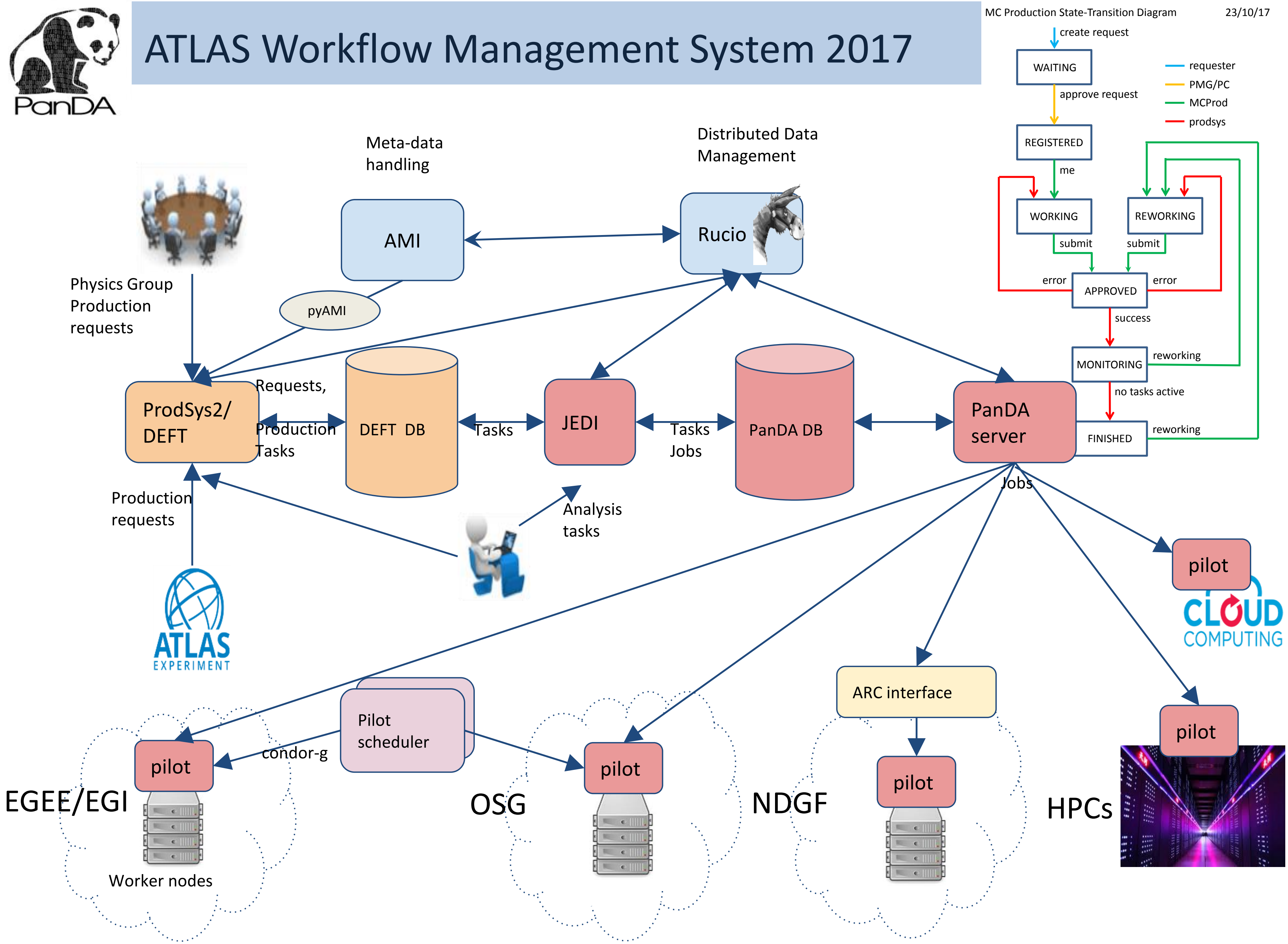
WMS for LHC Philosophy. Retrospective

- Design goals
  - Achieve high level of automation to reduce operational effort for large collaboration
  - Flexibility in adapting to evolving hardware, middleware and network configurations
  - Insulate user from hardware, middleware, and all other complexities of the underlying system
  - Unified system for data (re)processing, MC production, physics groups and user analysis
  - Incremental and adaptive software development
- Key features
  - Central job queue
    - Unified treatment of distributed resources
    - SQL DB keeps state of all workloads
  - Pilot based job execution system
    - Payload is sent only after execution begins on CE
    - Minimize latency, reduce error rates
  - Fairshare or policy driven priorities for thousands of users at hundreds of resources
  - Automatic error handling and recovery
  - Extensive monitoring
  - Modular design

DIRAC: the interware

- A software framework for distributed computing
- A **complete** solution to one (or more) user community
- Builds a layer between users and resources

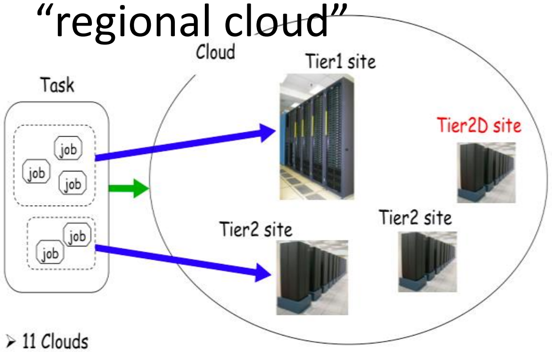



ATLAS Workflow Management System 2017
PanDA
MC Production State-Transition Diagram
23/10/17
create request
WAITING
approve request
REGISTERED
me
WORKING
REWORKING
submit
submit
error
APPROVED
error
success
MONITORING
reworking
no tasks active
FINISHED
reworking
requester
PMG/PC
MCProd
prodsys
Meta-data handling
Distributed Data Management
AMI
Rucio
pyAMI
Physics Group Production requests
ProdSys2/ DEFT
Requests, Production Tasks
DEFT DB
Tasks
JEDI
Tasks Jobs
PanDA DB
PanDA server
Production requests
Analysis tasks
Jobs
pilot
CLOUD COMPUTING
ATLAS EXPERIMENT
ARC interface
Pilot scheduler
condor-g
pilot
pilot
pilot
pilot
EGEE/EGI
OSG
NDGF
HPCs
Worker nodes

From "regional" to "world" cloud

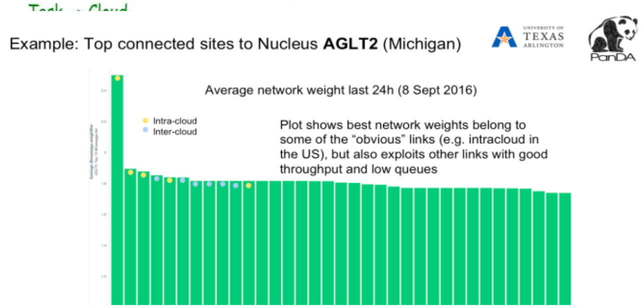
ATLAS Computing Model 2012

"regional cloud"

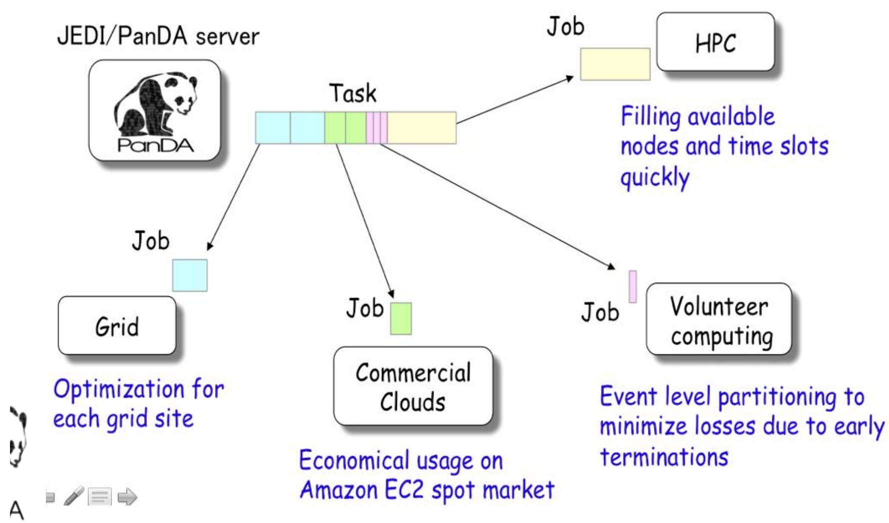


- 11 Clouds
  10 T1s + 1 T0 (CERN)
  Cloud = T1 + T2s + T2Ds (except CERN)
  T2D = multi-cloud T2 sites
- 2-16 T2s in each Cloud

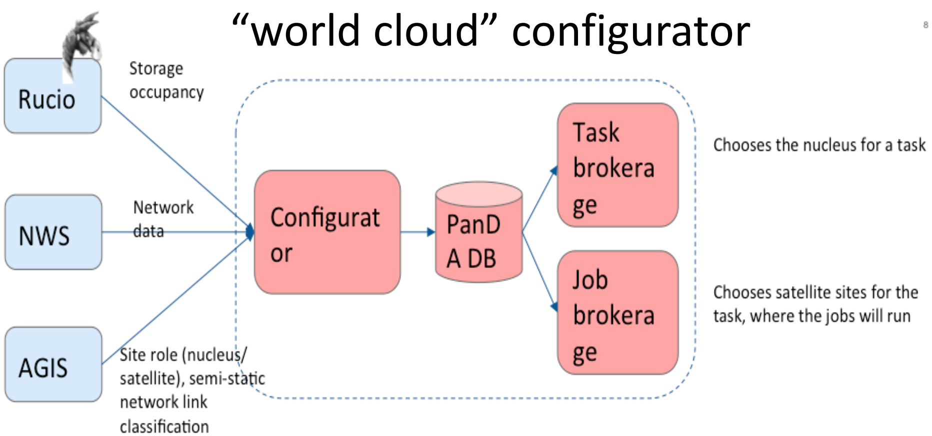
After 2012 we relaxed Tiers hierarchy and started dynamic resources configuration and dynamic workload partitioning

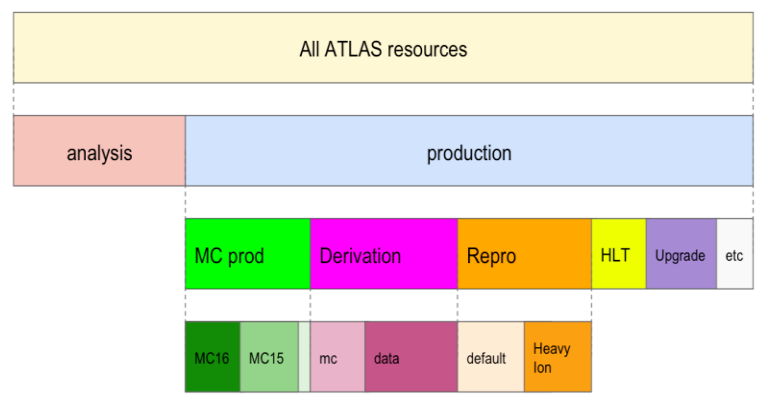


- Workload partitioning for traditional and opportunistic resources



"world cloud" configurator

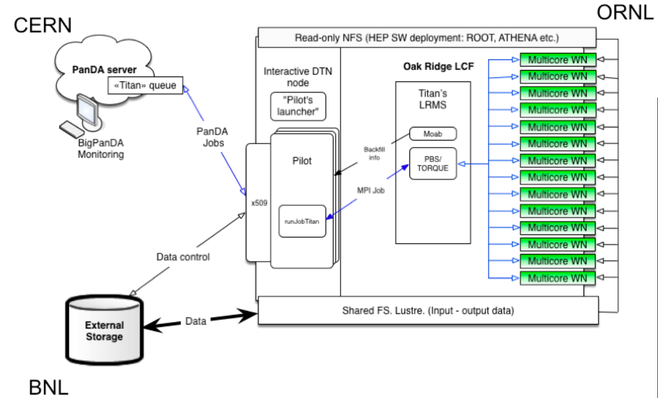


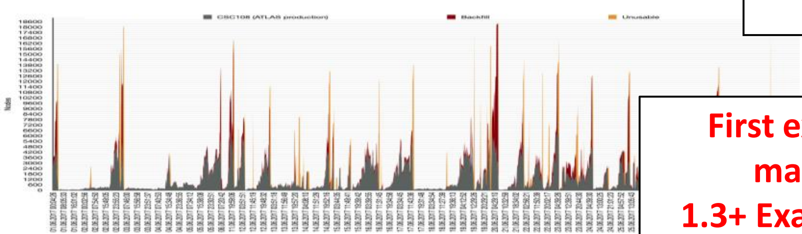


Resource allocation

Workflow Management. PanDA. **P**roduction **and** **D**istributed **A**nalysis System

https://twiki.cern.ch/twiki/bin/view/PanDA/PanDA





Global ATLAS operations

Up to ~800k concurrent jobs
25-30M jobs/month at >250 sites
~1400 ATLAS users

rly

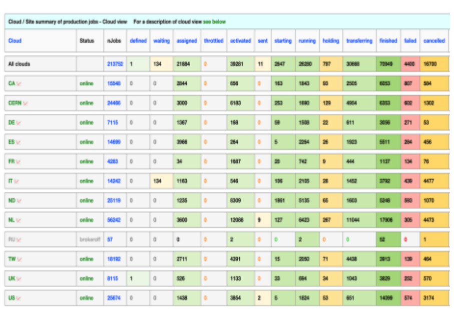
First exascale workload manager in HENP
1.3+ Exabytes processed in 2014 and in 2016
Exascale scientific data processing today

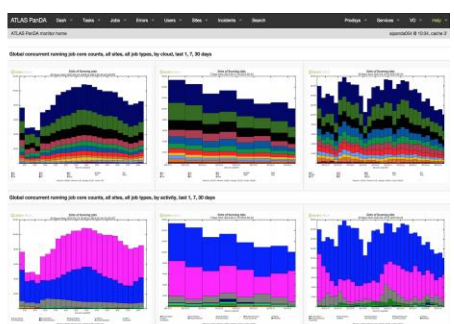
PanDA Brief Story

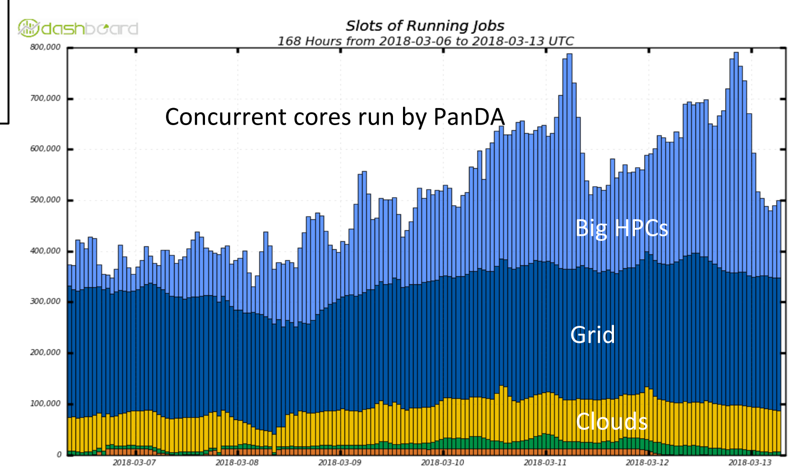
2005: Initiated for US ATLAS (BNL and UTA)
2006: Support for analysis
2008: Adopted ATLAS-wide
2009: First use beyond ATLAS
2011: Dynamic data caching based on usage and demand
2012: ASCR/HEP BigPanDA project
2014: Network-aware brokerage
2014 : Job Execution and Definition I/F (JEDI) adds complex task management and fine grained dynamic job management
2014: JEDI- based Event Service
2014:megaPanDA project supported by RF Ministry of Science and Education
2015: New ATLAS Production System, based on PanDA/JEDI
2015 :Manage Heterogeneous Computing Resources
2016: DOE ASCR BigPanDA@Titan project
2016:PanDA for bioinformatics
2017:COMPASS adopted PanDA , NICA (JINR)
PanDA beyond HEP : BlueBrain, IceCube, LQCD

BigPanDA Monitor

http://bigpanda.cern.ch/



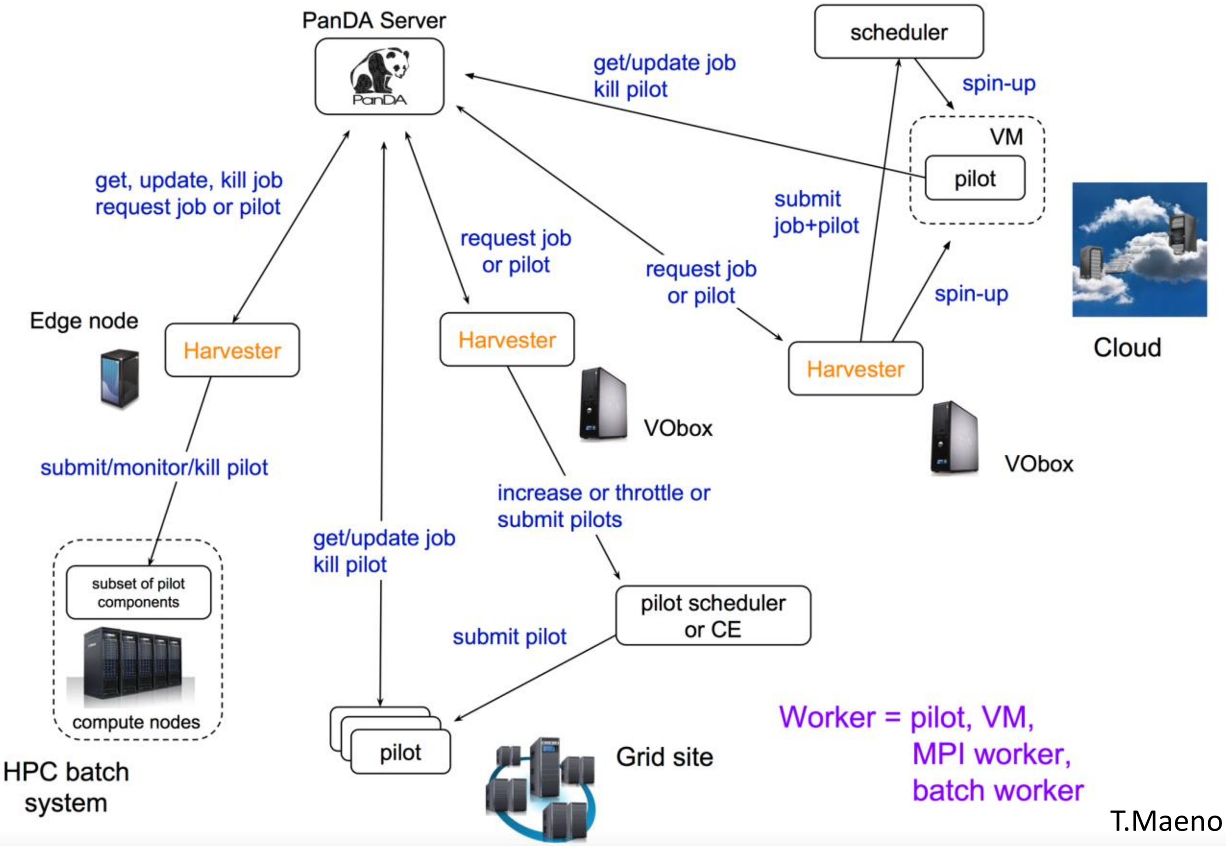




Lessons Learned

- WMS is designed by and serves the physics community
- WMS new features are driven by experiment operational needs
- Computing model and computing landscape in general has changed
  - Tiers hierarchy relaxed (~not exist)
  - Computing resources are becoming heterogeneous
    - Dedicated (grid) sites, HPCs, commercial and academic clouds …
    - HPCs and clouds are successfully integrated for Run 2/3
    - The mix of site capabilities and architectures
      - The mix will change with time - though all will be needed
- There are several systems with very well defined roles which are integrated for distributed computing : Information system (AGIS), DDM (Rucio), WMS (ProdSys2/PanDA), meta-data (AMI), and middleware (HTCondor, Globus…). We managed to have a good integration of all of them in ATLAS.
  - Combine all functionalities in one system or separate them between systems ?
    - Catalogs, layers, ….flexibility to add new features and to evaluate new technologies
- Monitoring and accounting are key components of Distributed SW
- Errors handling
- Scalability
  - WMS
  - Database technology
  - Monitoring
- WMS functionality is important as scalability
- Edge service is (should) be an additional layer to serve all heterogeneous resources

Future development. Harvester



T.Maeno

Primary objectives :

- To have a common machinery for diverse computing resources
- To provide a common layer in bringing coherence to different HPC implementations
- To optimize workflow executions for diverse site capabilities

- To address wide spectrum of computing resources/facilities available to ATLAS and experiments in general
- New model : PanDA server- harvester-pilot
- The project was launched in Dec 2016

Harvester Status

- Architecture designed and implemented
- Harvester for cloud
  - In production : CERN+Leibniz+Edinburgh resources (1.2k CPU cores
  - Work in progress : HLT farm @ LHC Point1, Google Cloud Platform
- Harvester for HPC
  - In production :
    - Theta/ALCF, Titan (OLCF)
    - ASGC (non-ATLAS Vos)
    - Cori+Edison / NERSC
    - KNL@BNL
- Harvester for Grid
  - Core SW is ready
  - Many scalability tests are planned in 2018 before commissioning
    - harvester is currently running at BNL (~800 jobs). Migration to full scale production is ongoing at BNL

Future Challenges

- New physics workflows
  - also new ways how Monte-Carlo campaigns are organized
- New strategies
  - “provisioning for peak”
- Integration with networks (via DDM, via IS and directly)
- Data popularity -> event popularity
- Address new computing model
- Address future complexities in workflow handling
  - Machine learning and Task Time To Complete prediction
  - Monitoring, analytics, accounting and visualization
  - Granularity and data streaming

Future Challenges. Cont'd

- Incorporating new architectures (like TPU, GPU, RISC, FPGA, ARM…)
- Adding new workflows (machine learning training, parallelization, vectorization…)
- Leveraging new technologies (containerization, no-SQL analysis models, high data reduction frameworks, tracking…)
- we have experience to enable large scale data projects for other communities, we are working through BigPanDA (DOE ASCR funded project)
  - Some components of WMS software stack could be used by others (i.e. *harvester)*
- Event Service and Event Streaming Service (see Torre's talk)
- WMS – DDM coupled optimizations
  - WMS will evolve to enable new data models
  - Data lakes, data ocean, caching services, SDN, DDN,…
  - Data carousel (more intensive tape usage, tape/disk data exchange)
  - Another level of granularity (from datasets to events)

Industry R&D Collaboration. Google Cloud Platform

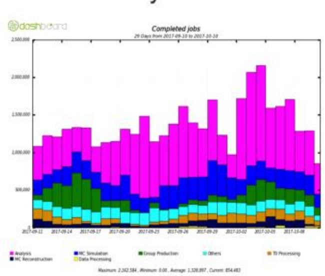
ATLAS DDM and WMS common R&D (+ CERN OpenLab +...)

- Integrate GCP(Storage and Compute) with ATLAS Distributed Computing
- Allow ATLAS to explore the use of different computing models to prepare for HL-LHC
- Allow ATLAS user analysis to benefit from the Google infrastructure
- Provide scientific use-case for Google product development and R&D
- Whitepaper : https://cds.cern.ch/record/2299146/files/ATL-SOFT-PUB-2017-002.pdf

Three initial ideas interesting to all partners :

- User analysis
  - Place copies of analysis output on GCP for reliable user access
  - Serves as cache with limited lifetime
- Data placement, replication, and popularity
  - Store the final derivation of MC and reprocessing data campaigns
  - Use Google Network to make data available globally (e.g., ingest in Europe but job reads from US)
  - Incorporate cloud access patterns into popularity measurements
- Data marshaling and streaming. Event streaming service
  - Evaluate necessary compute for generation of sub-file products (branches/events from ROOT files)
  - Job performance and network behavior for very small sample streaming

User Analysis



Google Cloud

Data Analysis, Replication and Placement

Data Streaming