

ATLAS Production System

...

F. Barreiro, M. Borodin, I. Glushkov, D. Golubkov, A. Klimentov,
T. Korchuganova, T. Maeno, S. Padolski



Outline

- Production System and Distributed Computing Highlights - Alexei
 - Introduction to resources allocation - Tadashi
- Production System Technicalities - Misha
- Monte-Carlo Production Request (procedure and technicalities) - Junichi/Doug
- How to control Production System and JEDI - Dima
 - Tasks and jobs brokerage - Tadashi
- BigPanDA monitoring overview - Siarhei
- Production System Operations - Ivan
- Live demo - Misha
- Q&A



ATLAS Distributed Computing (ADC)

- **ADC Projects**

- *Workflow Management Software*
 - *Production System*
 - *PanDA*
 - *Harvester*
 - Event Service
 - Pilot
- Distributed Data Management (DDM) - Rucio
- Metadata : ATLAS Meta-Data Interface - AMI
- ATLAS Grid Information System - AGIS
- Analytics
- Non-standard resources : High-Performance Computers, Clouds, Volunteer Computing

- **ADC Activities**

- *Distributed Production and Analysis*
- Infrastructure and Facilities
- Integration and Commissioning
- DDM Operations
- Central Services
- Tier-0
- *MC and Group Production*
- *Shifts and Communication*

Topics to be covered today



Workflow Management Software Project

The project will bring together developers from different projects and areas. It is aiming to:

- *propose and implement a coherent WFM strategy for various workflows*
- *simplify the operational burden*
- *avoid duplication and overlaps in software development*

The project will also address issues with monitoring, packaging and WMF software releases

Production System is a part of Workflow Management Software Project



Workflow Management Software Documentation and Links

- Database schemas, components communication, workflow diagrams are available on ProdSys2 and JEDI Twiki
- ProdSys2 Twiki : <https://twiki.cern.ch/twiki/bin/view/AtlasComputing/ProdSys>
- PanDA/JEDI Twik <https://twiki.cern.ch/twiki/bin/view/PanDA/PandaJEDI>
- BigPanDA monitoring: <http://bigpanda.cern.ch/help/>
 - <https://twiki.cern.ch/twiki/bin/view/AtlasComputing/PandaPilot>
 - <https://twiki.cern.ch/twiki/bin/view/PanDA/GenericPanDAPilot>
 - <https://twiki.cern.ch/twiki/bin/view/PanDA/Pilot2> (placeholder)
- <http://news.pandawms.org>
- <https://twiki.cern.ch/twiki/bin/view/PanDA/PanDA>
- ATLAS Dataset Nomenclature : <https://cds.cern.ch/record/1070318?ln=en>
- ATLAS Distributed Computing : <https://twiki.cern.ch/twiki/bin/view/AtlasComputing/AtlasDistributedComputing>
- AMI - ATLAS Metadata Interface : <https://ami.in2p3.fr/>
- Rucio - ATLAS Distributed Data Management system : <http://rucio.cern.ch/>
- ATLAS Workflow Management System Overview :
 - https://docs.google.com/presentation/d/12I677tWxcSIibTTjc1MFX95RJnxSys_-RO46nzqVCk/edit#slide=id.g1407ec7d3d_0_0



ProdSys2 Team

- Brookhaven National Laboratory
- University Texas at Arlington
- U Iowa
- NRC “Kurchatov Institute” (Moscow and Protvino)
- Moscow Engineering Physics Institute
- Tomsk Polytechnic U



My Incomplete List of Abbreviations

ADC - ATLAS Distributed Computing

AGIS - ATLAS Grid Information System

AMI - ATLAS Meta-data Interface

ARC - Advanced Resource Connector (Nordugrid middleware)

ATLAS - a toroidal apparatus for LHC

CE - Computing Element

DB - database

DDM - Distributed Data Management

DEFT - Database Engine for Tasks - the second layer of ProdSys2

EGI/EGEE - European Grid Initiative (one of three LHC grid flavours)

JEDI - Jobs Execution and Definition Interface - the third layer of ProdSys2

HPC - High-Performance Computing (=supercomputers)

HTC - High-Throughput Computing (=Grid)

NDGF, Nordugrid - one of three LHC grid flavors

OSG - Open Science Grid (one of three LHC grid flavours)

PanDA - Production and Distributed Analysis WMS - the bottom layer of ProdSys2

ProdSys2 - the second generation of ATLAS Production System

pyAMI - python AMI (I/F to access AMI database(s))

Rucio - The second generation of ATLAS DDM

SE - Storage Element

TID - Task ID, unique task identifier

WFM - Workflow Management

WMS - Workload Management System



Basic Definitions

- Request - high level layer for Production managers ('reprocess 2016 PeriodA data')
 - ProdSys2 translates request to basket of tasks or task chain
 - Chain : event generation -> simulation -> reconstruction -> derivation
- Task : group of associated jobs, it is formed according to request
 - With the same production Tag (aka amiTag)
 - Production step
 - SW release
 - May have input(s) - dataset(s) or/and container(s)
 - Produce outputs - datasets
 - Current scale 2M tasks / year
- Job : basic unit of work
 - Executed on a CPU resource/slot
 - May have inputs (files)
 - Produces outputs (files)
 - Current scale 365+M jobs /year
- Pilot job
 - Lightweight execution environment to prepare Computing Element (CE), request actual payload, execute payload and clean up
- Dataset - group of files taken/produced under the same conditions
- Container - group of datasets

Task states :

Waiting : the task information is inserted to the DEFT task table (t_production_task) and task is waiting to be processed by JEDI
Registered : the task information is inserted to the JEDI task tables
Assigning : the task brokerage is assigning the task to a cloud
Submitting : the task is running scouts jobs
Running : the task is running jobs
Exhausted: task can go to the exhausted state from running if all attempts have been used, but not all jobs are done, usually it means that some task parameters (for instance, ram count) should be tuned. From the exhausted state task can go to final state : finished, aborted, failed or number of attempts can be manually increased and task can go to running state.
Done : all jobs are successfully finished
Finished : some inputs of task are not finished (or not executed), but task is considered as finished
Broken : task cannot be executed, task definition has problems
Failed : task failed in execution time and it should be aborted
Aborted : the task is killed, all outputs will be erased
Obsolete : the task is obsolete and all outputs will be erased



Task, Dataset, Container Nomenclature

ATLAS Dataset Nomenclature : <https://cds.cern.ch/record/1070318?ln=en>

- Task (has unique ID):
 - MC : Project.datasetID.PhysicsShort.productionStep.version
 - DATA : Project.runNumber.streamType.productionStep.version
- Dataset has unique name
 - DATA : scope:Project.runNumber.streamType.productionStep.dataType.version_TID
 - MC : scope:Project.datasetID.PhysicsShort.productionStep.dataType.version_TID
- Container has unique name

Dataset w/o TID and with '/' at the end (in ProdSys2, DDM supports containers and container of containers, '/' - in container name is kept ProdSys2, as it was done for Run1 DDM and Production)

Task `mc15_13TeV 404741 PythiaRhad_AUET2BCTEQ6L1_gen_gluino_p1_2400_qq_1800_1ns evgen e5881`

project

datasetID

Physics short

Production Step

Version (aka AMItag)

scope

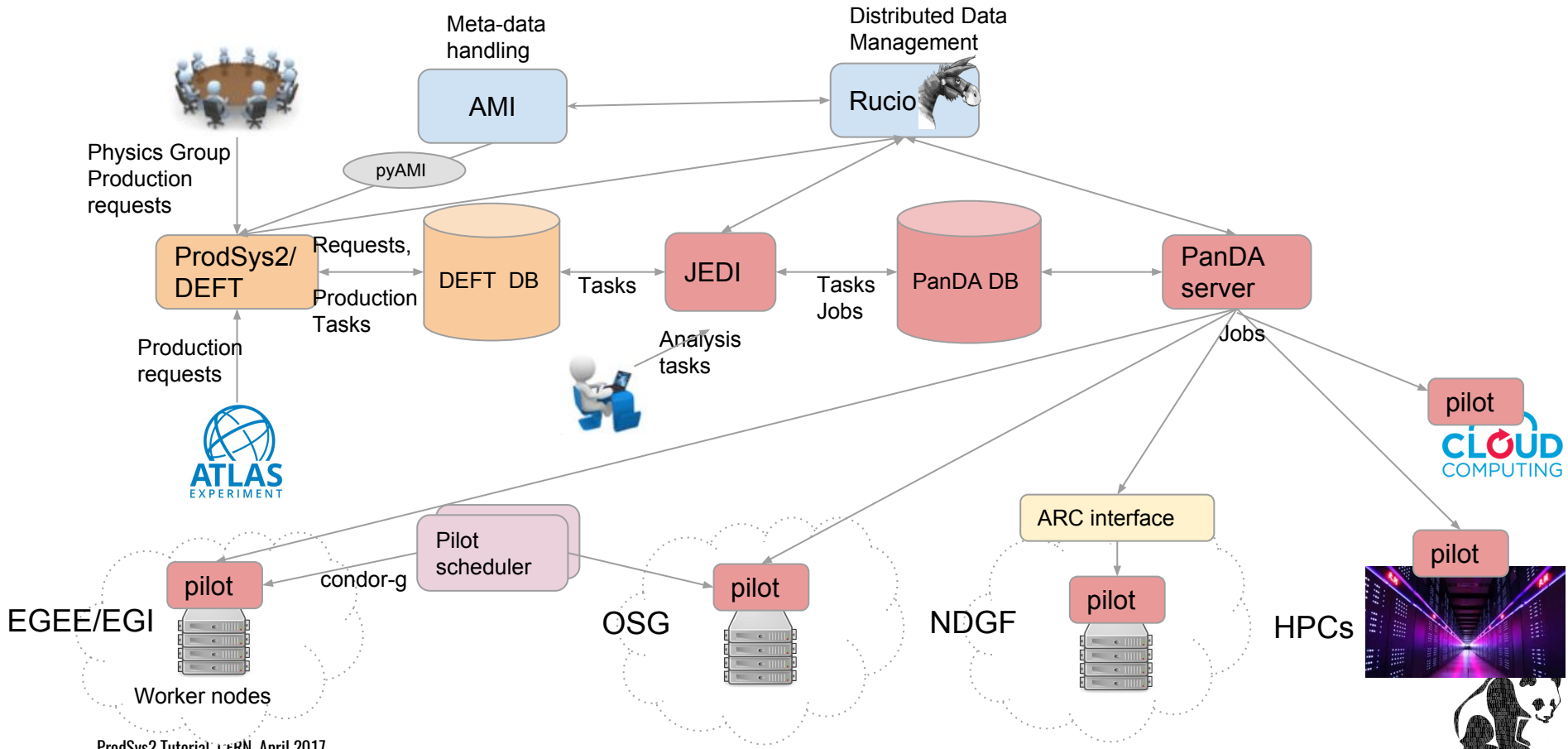
Data type

Task ID

`mc15_13TeV: mc15_13TeV.404741.PythiaRhad_AUET2BCTEQ6L1_gen_gluino_p1_2400_qq_1800_1ns.evgen.EVNT.e5881_tid10997644_00`



ATLAS Workflow Management schematic



Production System. Core ideas

Make hundreds of distributed sites appear as local

Provide central queue for users – similar to local batch systems

Reduce site related errors and latency

Build a pilot job system – late transfer of user payloads

Crucial for distributed infrastructure maintained by local experts

Hide middleware while supporting diversity and evolution

WMS interacts with middleware – users see high level workflow

Automation engines built into PanDA, not exposed to users

Hide variations in infrastructure

WMS presents uniform ‘job’ slots to user (with minimal sub-types)

Easy to integrate grid sites, clouds, HPC sites

Use the same system for Monte-Carlo production, data processing, group and derivation production and users analysis

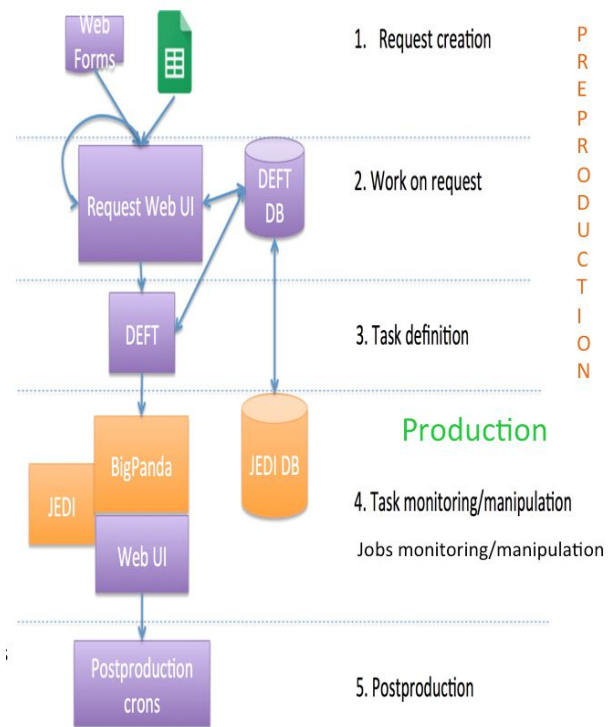


ProdSys2 Main Features.

- Keep key ProdSys1 features/concepts
 - Tasks, TID datasets, containers, database schemas for many tables
- Redesign system architecture.
- Grant more rights to customers (Production Managers). New Web UI for Production Managers

Modular design

- **DEFT : Requests**
 - Tasks definition
 - **JEDI : Tasks brokerage and execution**
 - Jobs definition and execution
 - **PostProduction : tasks, datasets handling**
 - Add upper layer : *Request* to group tasks
 - Add task layer to group user analysis jobs
 - Delegate task execution and jobs definition to lower layer (JEDI)
 - More flexibility in task/load handling and resources utilization
 - The first step to dynamic resource usage
- New Monitoring (BigPanDA)



Main workflows

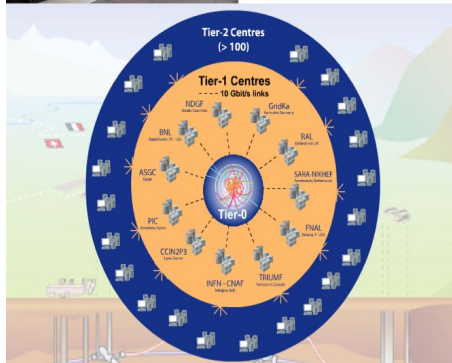
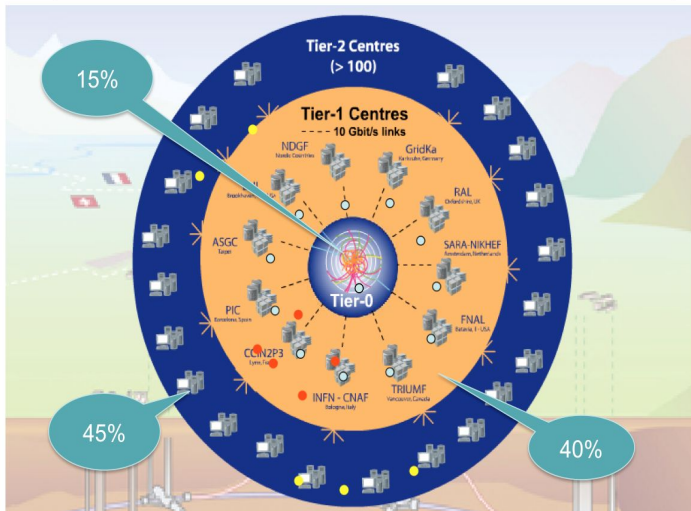
- Monte-Carlo Production
- Data Reprocessing
- High Level Trigger
- Tier-0 spill-over
- SW Validation
- Physics groups production
- Derivation production in trains
- Open-ended production
- Users Analysis

Resources are shared according to scientific goals between ATLAS & Physics Groups & Physicists



Distributed Computing Resources

LHC Grid Computing – W:CG



ATLAS Grid would be around #30 from Top100

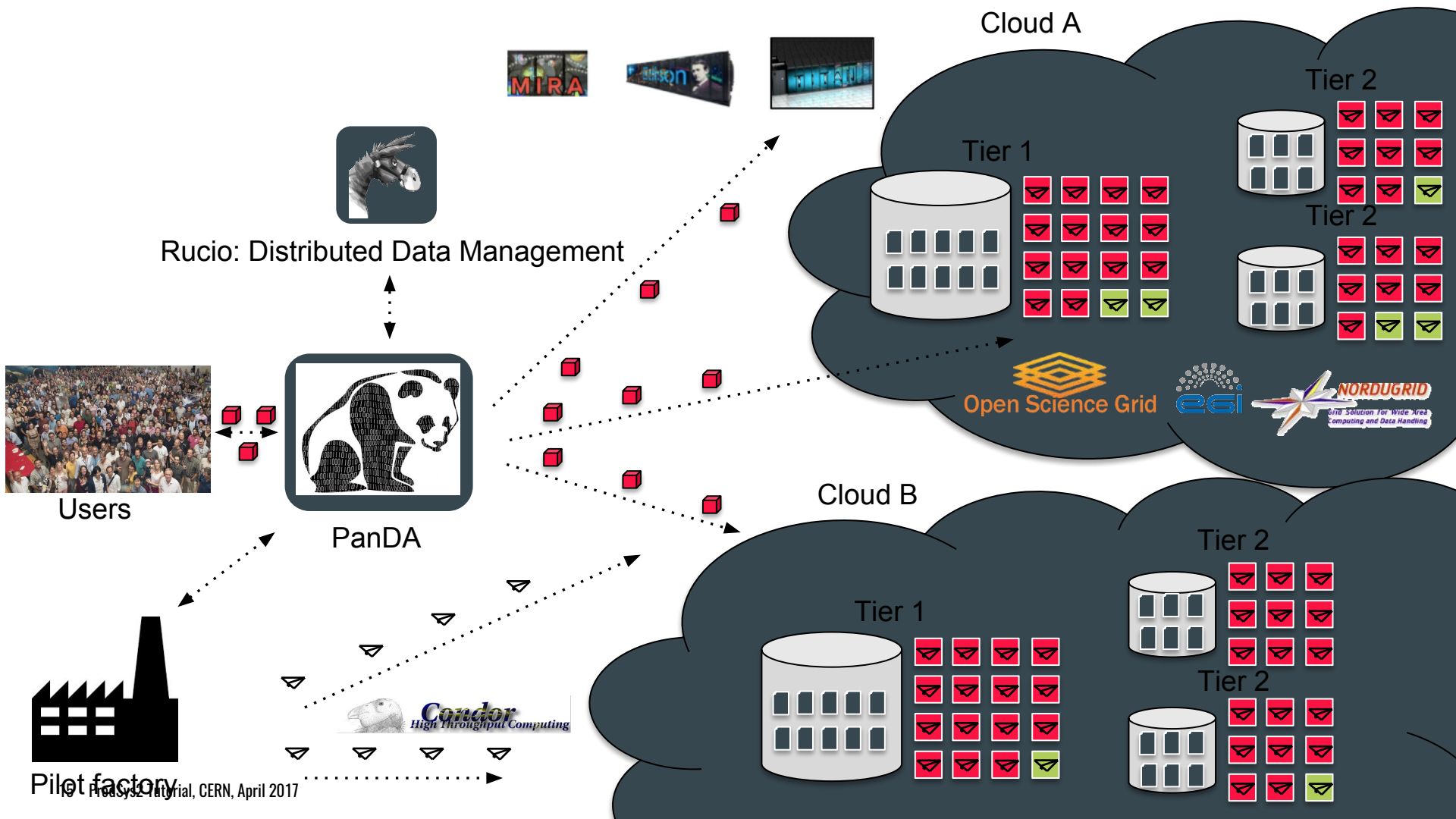


| | | | |
|------------------|--|-------------|------------|
| Peak Performance | 27.1 PF 16,688 compute nodes | 24.5 PF GPU | 2.6 PF CPU |
| System memory | 710 TB total memory | | |
| Interconnect | Gemini High Speed Interconnect | 3D Torus | |
| Storage | Lustre Filesystem | 32 PB | |
| Archive | High-Performance Storage System (HPSS) | 29 PB | |
| I/O Nodes | 512 Service and I/O nodes | | |



The ATLAS collaboration has broad range of R&D programs on these machines and on many others...



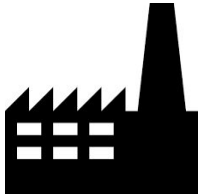


Rucio: Distributed Data Management



PanDA

Users



Pilot factory



Condor
High Throughput Computing

Cloud A

Tier 1

Tier 2

Tier 2

Cloud B

Tier 1

Tier 2

Tier 2

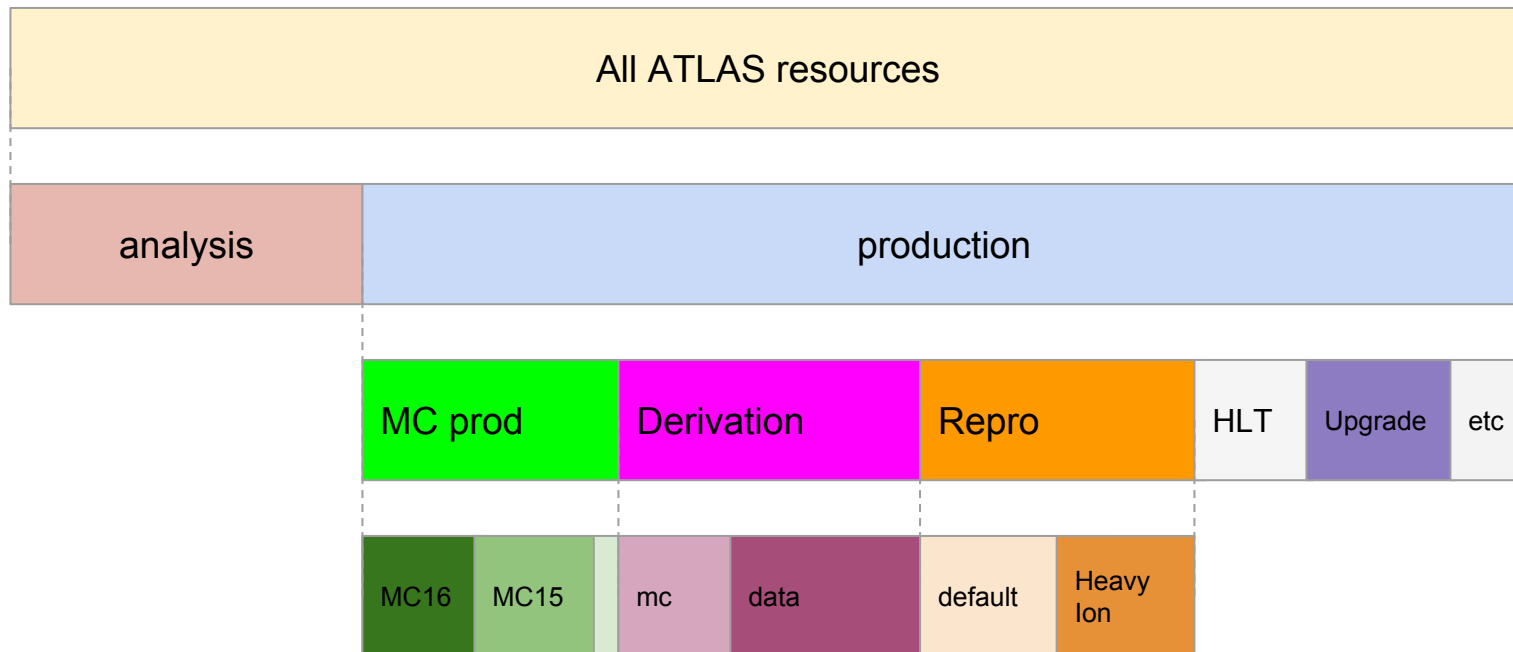


Resource allocation

- Static partitioning between Production and Analysis
 - One credential for each activity which is mapped to a separate uid in the underlying batch systems
- Production
 - Dynamic partitioning by Global share mechanism
 - Shares and allocation defined by DPA based on physics needs
 - E.g., large allocation to a working group before a conference
- Analysis
 - Normal user analysis using personal certificate and group analysis using group production role
 - The same allocation for all users and groups
 - No priority boost for groups by default
 - Higher priorities to a user and/or group if requested by Physics Coordination



Resource allocation (cntd)



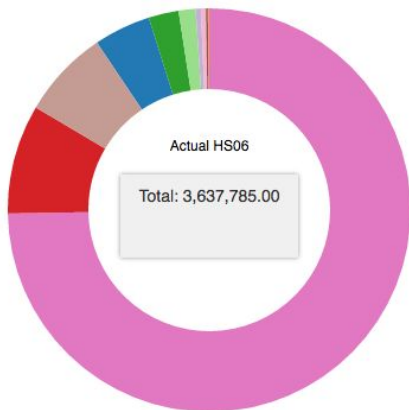
Global Shares: hierarchical fair share mechanism

- Global Shares were introduced Jan 2017
- Used to split processing resources on the grid between activities
 - E.g to allocate 20% of overall CPUs to reprocessing
- Measured in currently used HS06 ($=ncores \times corepower$)
 - It is not a quota system, i.e. we do not keep the history
- Shares are nestable: they will use the sibling's unused share
- Shares are assigned to a task at creation time and propagated to jobs
 - Rules based on prodsourcelabel, working group, campaign and processingtype
- They are restricted within certain limits and can not always be fully satisfied
 - We are working on improving the system and reduce the boundaries



Global Shares: overview

<http://bigpanda.cern.ch/globalshares/>



- Upgrade
- Reprocessing default
- Data Derivations
- Event Index
- MC production
- MC Derivations
- Analysis
- HLT Reprocessing
- Heavy Ion
- Test
- Group production
- Validation
- MC Default
- MC 16

| % of assigned resources | | | Used vs allocated HS06 | | | Queued HS06 (activated jobs) |
|-------------------------|-------------------------|-----------------------------|------------------------|--------------|----------|------------------------------|
| L1 Share | L2 Share | L3 Share | Actual HS06 | Target HS06 | Ratio | Queued |
| Analysis [20.0%] | | | 318,459.68 | 727,558.50 | 43.77 % | 1,775,300.91 |
| Production [75.0%] | | | 3,304,224.48 | 2,728,344.39 | 121.11 % | 5,257,441.15 |
| | MC root [40.8%] | | 2,821,232.06 | 1,485,732.09 | 189.89 % | 4,372,853.09 |
| | | MC production [13.6%] | 87,802.01 | 495,244.03 | 17.73 % | 0.00 |
| | | MC 16 [13.6%] | 14,604.76 | 495,244.03 | 2.95 % | 23,685.15 |
| | | MC Default [13.6%] | 2,718,825.30 | 495,244.03 | 548.99 % | 4,349,167.93 |
| | Derivations [14.9%] | | 52,863.71 | 540,266.22 | 9.78 % | 22,259.43 |
| | | MC Derivations [4.5%] | 49,447.38 | 162,079.86 | 30.51 % | 21,243.99 |
| | | Data Derivations [10.4%] | 3,416.33 | 378,186.35 | 0.90 % | 1,015.43 |
| | Reprocessing [3.7%] | | 316.98 | 135,066.55 | 0.23 % | 0.00 |
| | | Reprocessing default [3.0%] | 316.98 | 108,053.24 | 0.29 % | 0.00 |
| | | Heavy Ion [0.7%] | 0.00 | 27,013.31 | --- | 0.00 |
| | Group production [3.7%] | | 5,474.59 | 135,066.55 | 4.05 % | 26,823.11 |
| | Upgrade [3.7%] | | 165,804.89 | 135,066.55 | 122.76 % | 668,993.61 |
| | HLT Reprocessing [3.7%] | | 10.91 | 135,066.55 | 0.01 % | 0.00 |
| | Validation [3.7%] | | 258,145.59 | 135,066.55 | 191.12 % | 166,251.73 |
| | Event Index [0.7%] | | 375.75 | 27,013.31 | 1.39 % | 260.18 |
| Test [5.0%] | | | 15,108.37 | 181,889.63 | 8.31 % | 69,729.36 |



ATLAS Production System

Production chain from Task Request to Derivation Production and User's Analysis

...



Outline

- Goal
- Creating request
 - Interface
 - Clone
 - Spreadsheet
- Working with request
 - Changing steps parameters
 - Fixing tasks
- Submitting
- Working with tasks
- Working with metadata
- Authentication



Goal and general points of the Production System web ui

- Goal is to automate task submission and manipulation
 - Complex workflow
 - Full control
 - Less error
 - Faster result
- General implementation details
 - Data model(request-slice-step-task) is the core of development
 - Modular structure which provides unified but at the time flexible interface
 - Evolved as new requirements appear

Requests ▼

Tasks ▼

Jobs ▼

Train ▼

Datasets ▼

Meta ▼

PanDA Config ▼



Concept of request, slice, step

| Request ID: | Description: | Reference: | Manager: | Physic group: | Project: | Status: |
|---------------|---|------------|------------------------|---------------|--------------|----------------------|
| menu(≡) 10675 | Data reprocessing validation request, run 284285, release 21.0.14 | | dsouth Me | REPR | data15_13TeV | working |

Add # HashTags:

last comment: - New comment

[Show/hide long description](#)

Total input: 22, from them approved: 22

Select All Filter by: slice data Filter: Filter by status: all Sort: slice ID

Select pattern Show Bind

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| | | | | | | | | | | |

Replace empty

0 + data15_13TeV.00284285.physics_Main.daq.RAW
Slice outputs: AOD.HIST.DAOD_IDTIDE.DRAW_ZMUMU.DRAW_EGZ.DRAW_TAUMUH.DRAW_EM...

| | | | | | | | | | | |
|---------|--|--|--|--|--|--|--|--|--|--|
| r9042 | | | | | | | | | | |
| T: none | | | | | | | | | | |

submitted edit (saved)

1 + data15_13TeV.00284285.physics_Main.daq.RAW
Slice outputs: HIST

| | | | | | | | | | | |
|---------|-------|-------|-------|--|--|--|--|--|--|--|
| r9042 | p2985 | p2985 | p2985 | | | | | | | |
| T: none | none | none | none | | | | | | | |

submitted edit (saved)

Request

Slice

Task

Step



Slice and step parameters

- Step - template for task
- Parameters which construct the task name :
 - AMI tag
 - Input
 - Output
- Dozens parameters to optimize task execution

Prev Slice: 0 Next

Dataset:
valid1:valid1.423001.ParticleGun_single_photon_egammaET.simul.HITS.e5112_s3091/

job options:
MC15_423001.ParticleGun_single_photon_egammaET.py

Input events:
-1

Comment:
(Fullsim)

Evgen Simul Merge Digi Reco Rec Merge Rec TAG Attfast Attf Merge Attf TAG Deriv

AMI tag: e5112

Step is skipped

Events per Input file: 5000

Events per job: 5000

Total events: -1

Files per job: [dropdown]

nGB per job: [dropdown]

max failure attempt: 10

Output formats: EVNT

Input format: [dropdown]

project mode: spacetoken=ATLASDATADISK

Priority: 899

Destination: [dropdown]

JEDI internal merging:

merging tag: [dropdown]

nFilesPerMergeJob: [dropdown]

nGBPerMergeJob: [dropdown]

nEventsPerMergeJob: [dropdown]

nMaxFilesPerMergeJob: [dropdown]

Previous tasks: [dropdown]

Save



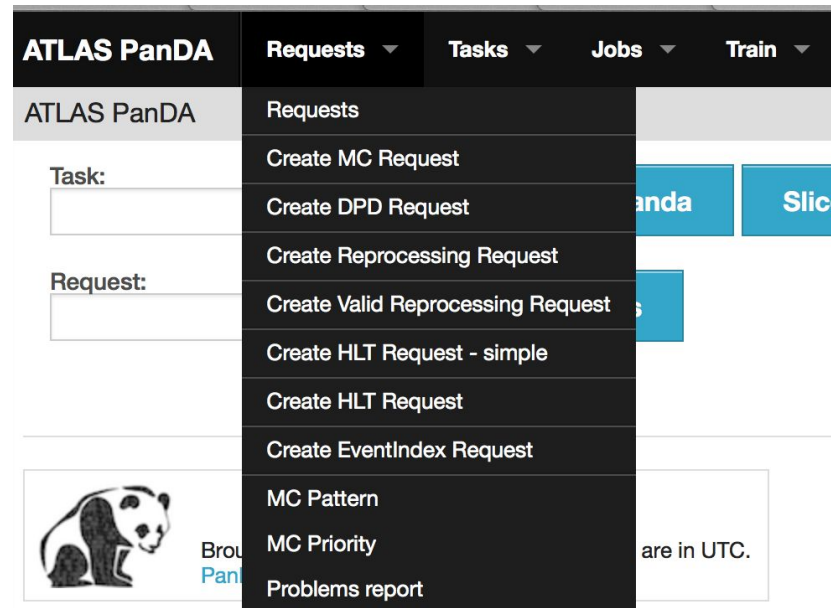
Request types

- System supports request with different topology
 - MC
 - Derivation
 - Reprocessing
 - HLT
 - Validation
 - Event index
 - Tier0 spillover
- Data model and interface are the same for all types



Request creation

- Using spreadsheet
- Using interface
 - General interface
 - Special interface
 - HLT
 - Validation
 - Derivation train
- Clone from existed request
 - Clone slices
 - Clone as a pattern



The screenshot displays the ATLAS PanDA web interface. At the top, there is a navigation bar with the following items: **ATLAS PanDA**, **Requests** (with a dropdown arrow), **Tasks** (with a dropdown arrow), **Jobs** (with a dropdown arrow), and **Train** (with a dropdown arrow). Below the navigation bar, the main content area is visible. On the left, there are input fields for **Task:** and **Request:**. On the right, there are buttons for **anda** and **Slice**. A dropdown menu is open under the **Requests** tab, listing the following options: **Requests**, **Create MC Request**, **Create DPD Request**, **Create Reprocessing Request**, **Create Valid Reprocessing Request**, **Create HLT Request - simple**, **Create HLT Request**, **Create EventIndex Request**, **MC Pattern**, **MC Priority**, and **Problems report**. At the bottom left of the interface, there is a panda logo and the text **Brou** and **Pan**. At the bottom right, there is a text box containing the text **are in UTC.**



Request parameters

- Each request has set of parameters which are common for all task in requests
 - Description
 - Energy
 - Manager
 - Project, campaign
- Parameters are used for bookkeeping

| | | |
|--------------------|---|--|
| Manager: | <input type="text" value="mborodin"/> | |
| | • This field is required. | |
| Short description: | <input type="text" value="MC16a HPC-minlow"/> | Short description of request (will become request title) |
| Ref link: | <input type="text" value="https://its.cern.ch/jira/browse/ATLMCPROD-4456"/> | Link to JIRA ticket for request |
| Provenance: | <input type="text" value="AP"/> | |
| Request type: | <input type="text" value="MC"/> | |
| Campaign: | <input type="text" value="MC16"/> | |
| Subcampaign: | <input type="text" value="MC16a"/> | |
| Phys group: | <input type="text" value="BPHY"/> | Physics Analysis/Combined Performance group |
| Energy gev: | <input type="text" value="8000"/> | Collision centre-of-mass energy in GeV |
| | • This field is required. | |
| Project: | <input type="text" value="mc16_13TeV"/> | Dataset project name, e.g mc15_13TeV |
| Long description: | <input type="text"/> | This will be the body of the mail that will be sent to PMS/PC for approval (and read by production managers), it should contain a summary and technical details of the request. As a minimum it should contain the justification, requested events (per samples and in total) and priorities. Additional technical information should also be provided if the samples require 25 ns and/or 50 ns reconstruction, if a specific release is required for evgen, if RDOs or ESDs should be kept for reconstruction, pileup profile (only 2015 mu profile, no pileup etc), other special options (please write 'Please check the JIRA ticket for ...') |
| Cc: | <input type="text"/> | Include email address(es) of relevant people, e.g. Group convenors |
| Need approve: | <input type="text" value="No"/> | |
| Train: | <input type="text" value="-----"/> | Pattern request for automatic derivation creation |



Request creation - Interface(Reprocessing, Derivation, EI)

- Interface to create tree-like workflow

Add slice

Slice #0 step #0

Datasets list:

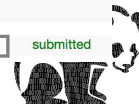
Or search datasets in ddm/prodsys by filter:

Find datasets

| <input type="checkbox"/> | Dataset Name | events | <input type="checkbox"/> |
|--|--|---|--|
| <input type="text" value="AMI tag"/> | <input type="text" value="Output formats (e.g. AOD.ESD)"/> | <input type="text" value="Events per job"/> 5000 | <input type="text" value="Total events"/> -1 |
| <input type="text" value="cmtconfig"/> default | <input type="text" value="project mode"/> | <input type="text" value="Priority"/> 520 | <input type="text" value="ram"/> |
| <input type="text" value="Files per job"/> | <input type="text" value="GB per job"/> | <input type="text" value="Max failure attempt"/> 15 | |
| <input type="text" value="Destination token"/> | | | |
| <input type="checkbox"/> JEDI internal merging | | | |

Add step **Remove step** **Fork**

| | | |
|---|--|--|
| 5 | + data16_13TeV.00297730.physics_Main.daq.RAW | <input checked="" type="checkbox"/> |
| Slice outputs: AOD.HIST.DAOD_IDTIDE.DRAW_ZMUMU.DRAW_EGZ.DRAW_TAUMUH.DRAW_EM... | | |
| | <input type="text" value="r9264"/> | <input type="text" value="submitted"/> |
| | T: done | |
| 36 | + data16_13TeV.00297730.physics_Main.daq.RAW | <input checked="" type="checkbox"/> |
| Slice outputs: HIST | | |
| | <input type="text" value="r9264 p3084 p3084 p3084"/> | <input type="text" value="submitted"/> |
| | T: done done done | |
| 67 | + data16_13TeV.00297730.physics_Main.daq.RAW | <input checked="" type="checkbox"/> |
| Slice outputs: AOD | | |
| | <input type="text" value="r9264 p3083"/> | <input type="text" value="submitted"/> |
| | T: done | |
| 98 | + data16_13TeV.00297730.physics_Main.daq.RAW | <input checked="" type="checkbox"/> |
| Slice outputs: DAOD_IDTIDE | | |
| | <input type="text" value="r9264 p3083 p3083"/> | <input type="text" value="submitted"/> |
| | T: done done | |
| 129 | + data16_13TeV.00297730.physics_Main.daq.RAW | <input checked="" type="checkbox"/> |
| Slice outputs: DESDM_SGLEL | | |
| | <input type="text" value="r9264 p3082 p3082"/> | <input type="text" value="submitted"/> |
| | T: done done | |
| 160 | + data16_13TeV.00297730.physics_Main.daq.RAW | <input checked="" type="checkbox"/> |
| Slice outputs: DESDM_SLTTMU | | |
| | <input type="text" value="r9264 p3082 p3082"/> | <input type="text" value="submitted"/> |
| | T: done done | |
| 191 | + data16_13TeV.00297730.physics_Main.daq.RAW | <input checked="" type="checkbox"/> |
| Slice outputs: DESDM_MCP | | |
| | <input type="text" value="r9264 p3082 p3082"/> | <input type="text" value="submitted"/> |



Request creation - existing request is a pattern

- Existed request could be used as a pattern
 - Select one tree (same run)
 - Click 'Create base on this'. It'll fill the request creation interface with existed structure
 - Modify and add more inputs

The screenshot displays a web interface for managing requests. At the top, a table lists existing requests with columns for Request ID, Description, Reference, Manager, Physic group, Project, and Status. A context menu is open over the first request (ID: 11398), with the option 'Create base on this' highlighted. Below the table, there are filters for 'Filter by:' (set to 'slice data') and 'Filter by status:' (set to 'all'). A 'Select All' button is visible. The 'Sort:' dropdown is set to 'dataset name'. Below these are 'Show' and 'Bind' buttons. A 'Select pattern' dropdown is also present. At the bottom, a request creation form is shown with a checked checkbox, a '+' icon, and a text input field containing '+ data16_13TeV.00304008.physics_Main.daq_RAW'. Below this, 'Slice outputs:' are listed as 'AOD.HIST.DAOD_IDTIDE.DRAW_ZMUMU.DRAW_EGZ.DRAW_TAUMUH.DRAW_EM...'. A table with 11 columns (0-10) is shown, with the first cell containing 'r9232' and the last cell containing 'not_submitted edit (saved)'. A 'Replace empty' checkbox is also visible.

| Request ID: | Description: | Reference: | Manager: | Physic group: | Project: | Status: |
|-------------|---|------------|-----------|---------------|--------------|---------|
| 11398 | Data reprocessing validation request, run 304008, release 21.0.19, test of new DRAW | | dsouth Me | REPR | data16_13TeV | waiting |

Filter by: slice data Filter: Filter by status: all Sort: dataset name

Select pattern Show Bind

0 1 2 3 4 5 6 7 8 9 10 Replace empty

+ data16_13TeV.00304008.physics_Main.daq_RAW

Slice outputs: AOD.HIST.DAOD_IDTIDE.DRAW_ZMUMU.DRAW_EGZ.DRAW_TAUMUH.DRAW_EM...

r9232 not_submitted edit (saved)

+ data16_13TeV.00304008.physics_Main.daq_RAW



Request creation - predefined forms

- If workflow is well defined special interface could be created
 - HLT
 - Data validation

| + data16_13TeV00309640.physics_EnhancedBias.recon.RAW.r9295 | |
|---|--|
| NTUP_TRIGCOST | Slice outputs: HIST_HLTMON.RAW.NTUP_TRIGRATE.NTUP_TRIGCOST |
| r9318 | submitted edit (saved) |
| 1 | + data16_13TeV00309640.physics_EnhancedBias.merge.RAW |
| | Slice outputs: NTUP_TRIGCOST |
| r9318 p3094 | submitted edit (saved) |
| 2 | + data16_13TeV00309640.physics_EnhancedBias.merge.RAW |
| | Slice outputs: NTUP_TRIGRATE |
| r9318 p3094 p3094 | submitted edit (saved) |
| 3 | + data16_13TeV00309640.physics_EnhancedBias.merge.RAW |
| | Slice outputs: HIST_HLTMON |
| r9318 p3093 p3093 | submitted edit (saved) |
| 4 | + data16_13TeV00309640.physics_EnhancedBias.merge.RAW |
| | Slice outputs: ESD.AOD.HIST |
| r9318 r9319 | submitted edit (saved) |
| 5 | + data16_13TeV00309640.physics_EnhancedBias.merge.RAW |
| | Slice outputs: HIST |
| r9319 p3093 p3093 | submitted edit (saved) |
| 6 | + data16_13TeV00309640.physics_EnhancedBias.merge.RAW |
| | Slice outputs: AOD |
| r9319 p3092 | submitted edit (saved) |

Create hlt request Built 15:21

Dataset:

Short description(request title):

Link to JIRA ticket:

Two step reco:

Outputs:

- AOD
- ESD
- HIST
- HIST_HLTMON
- NTUP_TRIGCOST
- NTUP_TRIGRATE

Reco tag:

HIST* merge tag:

AOD merge tag:

NTUP_* merge tag:

Reco ramcount:

HIST* ramcount:

AOD ramcount:

NTUP_* ram:

Priority:

Sites:

Common project mode:

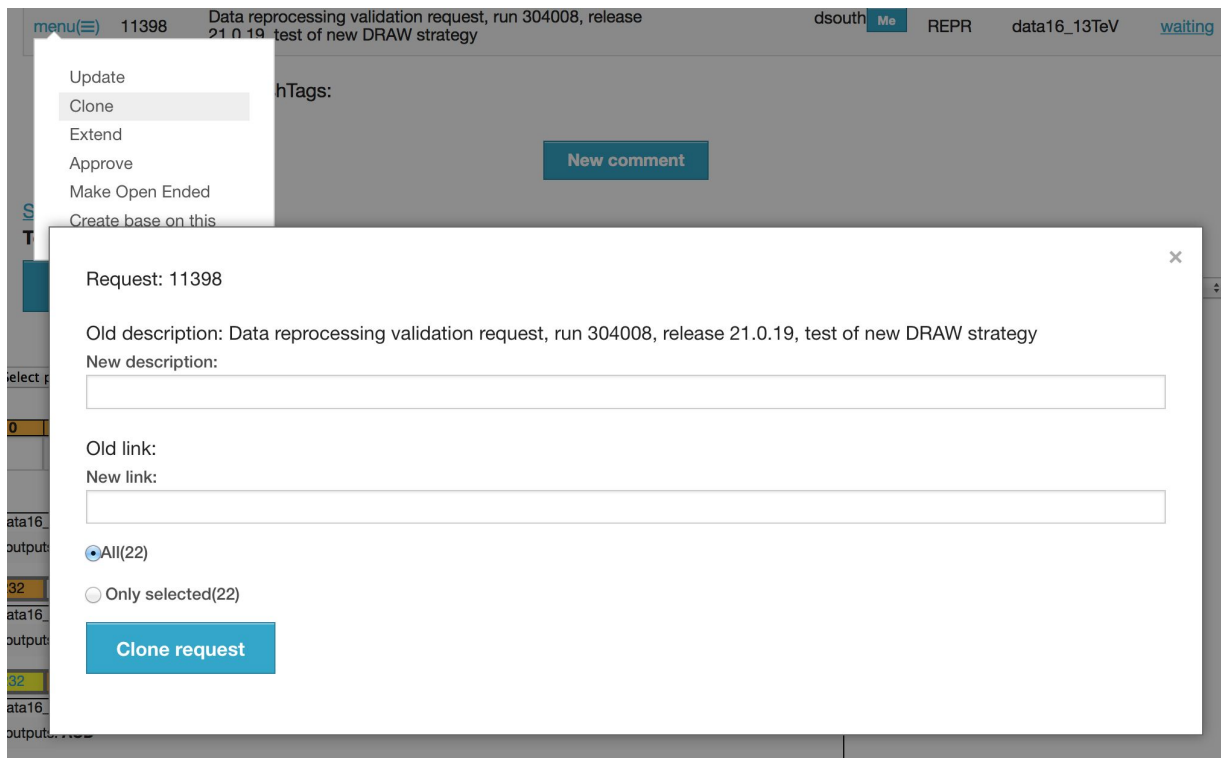
Reco specific project mode addition:

Proceed



Request creation - clone existing request

- Whole or a part of the existed request could be cloned to a new one



The screenshot shows a web interface for managing requests. A context menu is open over a request with ID 11398. The menu options are: Update, Clone (highlighted), Extend, Approve, Make Open Ended, and Create base on this. A 'New comment' button is visible in the background. A modal dialog box is displayed in the foreground, titled 'Request: 11398'. The dialog contains the following fields and options:

- Request: 11398
- Old description: Data reprocessing validation request, run 304008, release 21.0.19, test of new DRAW strategy
- New description: [Text input field]
- Old link: [Text input field]
- New link: [Text input field]
- All(22)
- Only selected(22)
- Clone request [Button]



Request creation - MC production, spreadsheet

- MC production uses spreadsheet as the main source of an input request data
 - Structure is predefined
 - Depends on the campaign
 - Spreadsheet parsed and translated to slice and steps
 - Fields are checked to make sense
 - Medium term plans is to move spreadsheet to the system



Request creation - MC production, spreadsheet(2)

MC Request

Requests

Create MC Request

Create DPD Request

description:

sheet-HPC-minlow.xlsx

File Edit View Insert Format Data Tools Add-ons Help

View only

Brief description

| | A | B | C | D | E | F | G | H |
|---|-------------------|---|------------------|---------------------|------------------|--------------------|----------|----------------|
| 1 | Brief description | JobOptions | CoM energy [GeV] | Events (Evgen-only) | Events (FullSim) | Events (Atfast II) | Priority | Output formats |
| 2 | W+ (->enu) +jet | MC15.363776.PowhegPythia8EvtGen_AZNLO_Wjplusenu.py | 13000 | | -1 | | 1 | |
| 3 | W- (->enu) +jet | MC15.363775.PowhegPythia8EvtGen_AZNLO_Wjminusenu.py | 13000 | | -1 | | 1 | |
| 4 | Z (->ee) +jet | MC15.363777.PowhegPythia8EvtGen_AZNLO_Zjee.py | 13000 | | -1 | | 1 | |
| 5 | | | | | | | | |

This will be the body of the mail that will be sent to PMC/PC for approval (and read by production managers), it should contain a summary and technical details of the request. As a minimum it should contain the justification, requested events (per samples and in total) and priorities. Additional technical information should also be provided e.g. if the samples require 25 ns and/or 50 ns reconstruction, if a specific release is required for evgen, if RDOs or ESDs should be kept for reconstruction, pileup profile (only 2015 mu profile and pileup etc), other special options (please write "Please check the JIRA ticket for ...")

Includes email address(es) of relevant people, e.g. Group convenors

Enter the "shareable" link to the google doc spreadsheet

Submit



Request creation - MC production, spreadsheet(3)

Long description:

This will be the body of the mail that will be sent to PMG/PC for approval (and read by production managers). It should contain a summary and technical details of the request. As a minimum it should contain the **justification, requested events (per samples and in total) and priorities**. Additional technical information should also be provided if the samples require 25 ns and/or 50 ns reconstruction, if a specific release is required for evgen, if RDOs or ESDs should be kept for reconstruction, pileup profile (only 2015 mu profile, no pileup etc), other special options (please write 'Please check the JIRA ticket for ...')

Cc:

Need approve:

Train:

Include email address(es) of relevant people, e.g. Group convenors

Pattern request for automatic derivation creation

- Check slices were read correctly
- Fill parameters
- Click submit

Submit

Input List

| # | Dataset: | Job Option: | Brief: | Tags: | Events |
|---|--|------------------|------------------|-------|----------|
| 0 | MC15.364253.Sherpa_222_NNPDF30NNLO_llv.py | Sherpa 2.2.2 llv | Sherpa 2.2.2 llv | | 16150000 |
| 1 | MC15.364254.Sherpa_222_NNPDF30NNLO_lvv.py | Sherpa 2.2.2 llv | Sherpa 2.2.2 llv | | 15000000 |
| 2 | MC15.364255.Sherpa_222_NNPDF30NNLO_lvuv.py | Sherpa 2.2.2 llv | Sherpa 2.2.2 llv | | 6000000 |



Request creation - MC production, spreadsheet(4)

- Spreadsheet could be added to existed request

| | Request ID: | Description: | Reference: |
|---------|-------------|------------------|------------------------------|
| menu(☰) | 11557 | MC16a HPC-minlow | ATLMCPROD-44 |

hTags:

- Update
- Clone
- Extend
- Approve
- Make Open Ended

S

Request was approved for processing by scientist at Oct 1



Approval policy

- Different approval policies could be applied for created requests
 - MC requests from groups should be approved by PMG first

[Show/hide long description](#)

we would like to request aMcAtNloHerwig NLO 2HDM H->hh->bbaubtau samples with lephad and hadhad filters and for different mass points. The JOs are set to have 2000 events per job after parton shower.

Concerning the number of events, we would like to request a total of 2.64M fastsim events with priority 1.

Concerning the cache to be used, we would like to use MCProd 19.2.5.19.1 in order to have fully compatibility with the samples requested by the other channels. (These samples will be used for di-Higgs combination)

For the MC production team, in summary:

- 1) MCProd 19.2.4.19.1
- 2) MC15c, 25ns, sim+dig+reco
- 3) 2.64M events fastsim, priority 1
- 4) No special configurations

PMG approval

Cancel

10 slices with priority:

1

Change priority



Request creation - train(Derivation)

- ‘Train’ is an approach to run derivation
 - Many outputs for the same input
 - Only some outputs is required for some group
- To create train request:
 - Derivation coordinator creates pattern request in a usual way
 - Group contact fills requested outputs based on pattern request
 - Similar requests merged in one(in testing)

| | | | |
|---|---|---|----------------------------|
| 0 | + | mc15_13TeV.410000.PowhegPythiaEvtGen_P2012_ttbar_hdamp172p5_nonallhad.merge.AOD.e3698_s2608_s2183_r7725_r7676 | Cloned |
| ✓ | | Slice outputs: DAOD_EXOT5.DAOD_EXOT7.DAOD_EXOT9.DAOD_EXOT11.DAOD_EXOT12.DAOD_EXOT14.DAOD_EXOT19.DAOD_EXOT20 | |
| | | p2952 | not_submitted edit (saved) |
| 1 | + | mc15_13TeV.410000.PowhegPythiaEvtGen_P2012_ttbar_hdamp172p5_nonallhad.merge.AOD.e3698_s2608_s2183_r7725_r7676 | |
| ✓ | | Slice outputs: DAOD_HIGG2D1.DAOD_HIGG5D1.DAOD_HIGG5D2.DAOD_HIGG5D3 | |
| | | p2952 | not_submitted edit (saved) |
| 2 | + | mc15_13TeV.410000.PowhegPythiaEvtGen_P2012_ttbar_hdamp172p5_nonallhad.merge.AOD.e3698_s2608_s2183_r7725_r7676 | |
| ✓ | | Slice outputs: DAOD_HIGG2D2.DAOD_HIGG2D4.DAOD_HIGG3D1 | |
| | | p2952 | not_submitted edit (saved) |
| 3 | + | mc15_13TeV.410000.PowhegPythiaEvtGen_P2012_ttbar_hdamp172p5_nonallhad.merge.AOD.e3698_s2608_s2183_r7725_r7676 | |
| ✓ | | Slice outputs: DAOD_EGAM2.DAOD_EGAM3.DAOD_EGAM4 | |
| | | p2952 | not_submitted edit (saved) |
| 4 | + | mc15_13TeV.410000.PowhegPythiaEvtGen_P2012_ttbar_hdamp172p5_nonallhad.merge.AOD.e3698_s2608_s2183_r7725_r7676 | |
| ✓ | | Slice outputs: DAOD_EXOT0.DAOD_EXOT6.DAOD_EXOT17.DAOD_EXOT18 | |
| | | p2952 | not_submitted edit (saved) |
| 5 | + | mc15_13TeV.410000.PowhegPythiaEvtGen_P2012_ttbar_hdamp172p5_nonallhad.merge.AOD.e3698_s2608_s2183_r7725_r7676 | |
| ✓ | | Slice outputs: DAOD_JETM3.DAOD_JETM10.DAOD_JETM11 | |
| | | p2952 | not_submitted edit (saved) |
| 6 | + | mc15_13TeV.410000.PowhegPythiaEvtGen_P2012_ttbar_hdamp172p5_nonallhad.merge.AOD.e3698_s2608_s2183_r7725_r7676 | |
| ✓ | | Slice outputs: DAOD_JETM2.DAOD_JETM4.DAOD_JETM6.DAOD_JETM7 | |
| | | p2952 | not_submitted edit (saved) |



Request creation - train(2)

Train: EXOT12 20.7.8.2 w/ TRT fix skim | Status: loading | pattern request: 9748 | Departure:

Assemble | Close

New load | Show trains

Load #0

Group: EXOT Submitted by: rnayyar Modify: [edit](#)
Outputs: [[1,["DAOD_EXOT12"]]]
Datasets:

- mc15_13TeV:mc15_13TeV.306734.aMcAtNloPy8EG_A14N30NLO_LQ1_M1500.merge.AOD.e5521_a766_a821_r7676/
- mc15_13TeV:mc15_13TeV.306747.aMcAtNloPy8EG_A14N30NLO_LQ2_M450.merge.AOD.e5521_a766_a821_r7676/
- mc15_13TeV:mc15_13TeV.306711.aMcAtNloPy8EG_A14N30NLO_LQ1_M350.merge.AOD.e5521_a766_a821_r7676/
- mc15_13TeV:mc15_13TeV.306749.aMcAtNloPy8EG_A14N30NLO_LQ2_M550.merge.AOD.e5521_a766_a821_r7676/
- mc15_13TeV:mc15_13TeV.306766.aMcAtNloPy8EG_A14N30NLO_LQ2_M1400.merge.AOD.e5521_a766_a821_r7676/
- mc15_13TeV:mc15_13TeV.306746.aMcAtNloPy8EG_A14N30NLO_LQ2_M400.merge.AOD.e5521_a766_a821_r7676/

- DAOD_SUSY1 DAOD_SUSY4 DAOD_SUSY5 DAOD_SUSY6 DAOD_SUSY7 DAOD_SUSY8 DAOD_SUSY10 DAOD_SUSY11
- DAOD_SUSY2 DAOD_SUSY3 DAOD_SUSY12 DAOD_SUSY13 DAOD_SUSY14
- DAOD_FTAG1 DAOD_FTAG2 DAOD_FTAG3 DAOD_FTAG4 DAOD_FTAG5
- DAOD_EXOT8
- DAOD_TOPQ1 DAOD_TOPQ2 DAOD_TOPQ3 DAOD_TOPQ4
- DAOD_TAUP2
- DAOD_BPHYZ
- DAOD_STDM6
- DAOD_STDM6
- DAOD_STDM9
- DAOD_SUSY15
- DAOD_SUSY16
- DAOD_EXOT16
- DAOD_SUSY17

Datasets:

Cancel | Save



Request creation - train(3)

- ‘Train’ could be created as a child for MC or reprocessing task
 - Same interface as for train creation
 - If parent task is redefined child is redefined automatically

Create trains:

Slices:12

Parent step:
EVENT

Pattern:
(3470)train definition for 20.1.5.1 - skim

Make trains

Submit only steps before: Sim Trains Split



Open ended (EI, Derivation)

- Request is been created as usual but input for slices is an 'open' container
 - Steps on container slices play template role
 - As soon as dataset appears in container new slices are being created

Request ID: 8013 Description: open-ended derivation production on Main stream with 20.7.6.5 Reference: atlas-dpd-production Management: atlas-dpd-production

Update
Clone
Extend
Approve
Make Open Ended

New comment

Total input: 1752, from them approved: 804 and hidden: 150

Select All Filter by: slice data Filter:

Select pattern Show

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|----|
| | | | | | | | | | | |

Replace empty

0 + data16_13TeV:data16_13TeV.physics_Main.OpenEnded.AOD.t0prod20/ Cloned
Slice outputs: DAOD_EGAM4.DAOD_EGAM3.DAOD_EGAM2.DAOD_EGAM9.DAOD_EGAM8
p2689 not_submitted edit (sa)

1 + data16_13TeV:data16_13TeV.physics_Main.OpenEnded.AOD.t0prod20/
Slice outputs: DAOD_EGAM7.DAOD_EGAM6.DAOD_EGAM5.DAOD_EGAM1
p2689 not_submitted edit (sa)



Step and slice modification

- Step and slice parameters could be changed individually:

The screenshot displays a workflow management interface. At the top, a task is identified as '0 + MC15.424101.Pythia8B_A14_CTEQ6L1_Jpsimu2mu2.py' with '(Fullsim)Jpsi muon filter (pT>2GeV)' and 'events: 5000000'. Below this, a row of slice identifiers is shown: 'e3982', 's2608', 's2183', 'r7772', and 'not_submitted_original'. The 'not_submitted_original' slice is highlighted in red, and a blue arrow points to the 'edit (saved)' link next to it. Below the slice row is a 'Select All' button. Underneath, the text 'Work with selected slices:' is followed by a row of buttons: 'Save', 'Submit', 'Clone', 'Fix', 'Reject', 'Hide', and 'Find input'. Further down, there are buttons for 'Submit only steps before:' (with a 'Sin' dropdown), 'Retry', 'Trains', and 'Split'. At the bottom, there are three buttons: 'Bulk steps modification', 'Bulk slice modification', and 'Manage tasks'.



Step and slice modification

- Step and slice parameters could be changed in bulk:

The screenshot shows a workflow management interface. At the top, a task is identified as '0 + MC15.424101.Pythia8B_A14_CTEQ6L1_Jpsimu2mu2.py' with '(Fullsim)Jpsi muon filter (pT>2GeV)' and 'events: 5000000'. Below this, a row of slice identifiers is shown: 'e3982', 's2608', 's2183', 'r7772', and 'not_submitted_original'. A 'Select All' button is positioned below the slice row. A section titled 'Work with selected slices:' contains buttons for 'Save', 'Submit', 'Clone', 'Fix', 'Reject', 'Hide', and 'Find input'. Further down, there are buttons for 'Submit only steps before:', a 'Sin' dropdown menu, 'Retry', 'Trains', and 'Split'. At the bottom, three buttons are visible: 'Bulk steps modification', 'Bulk slice modification', and 'Manage tasks'. Two blue arrows point from the bottom towards the 'Bulk steps modification' and 'Bulk slice modification' buttons.



Bulk step and slice modification

Steps approved: 0 Steps to change (not approved): 6

AMI Tag: Step: Slices:

| | | | | |
|--|----------------------------------|----------------------------------|---------------------------------|-------------------------------|
| AMI tag | Events per Input file | Events per job | Total events | |
| <input type="text" value="s2726"/> | <input type="text" value=""/> | <input type="text" value="100"/> | <input type="text" value="-1"/> | |
| Files per job | nGB per job | | max failure attempt | |
| <input type="text" value=""/> | <input type="text" value=""/> | | <input type="text" value="10"/> | |
| Output formats | Input format | | | |
| <input type="text" value="HITS"/> | <input type="text" value=""/> | | | |
| project mode | Priority | | | |
| <input type="text" value="spacetime=ATLASDATADISK"/> | <input type="text" value="320"/> | | | |
| Destination | <input type="text" value=""/> | | | |
| JEDI internal merging: | nFilesPerMergeJob | nGBPerMergeJob | nEventsPerMergeJob | nMaxFilesPerMergeJob |
| merging tag | <input type="text" value=""/> | <input type="text" value=""/> | <input type="text" value=""/> | <input type="text" value=""/> |



Creating new step and changing AMI tag

- To create new step or using another AMI tag
 - Pattern should be chosen.
Patterns are created by MC coordination
 - AMI tag for step should be filled
 - Pattern should be applied(bind) for selected slices

FS MC15c 25ns (pre, no pileup) Show Bind

| Evgen | Simul | Merge | Digi | Reco | Rec Merge | Rec TAG | Attfast | Attf Merge | Attf TAG | Deriv |
|-------|-------|-------|------|-------|-----------|---------|---------|------------|----------|-------|
| | s2726 | | | r7661 | r7648 | | | | | |

Replace empty

0 + MC15.308264.Pythia8EvtGen_A14NNPDF23LO_LLzprimemumu_m100t100.py
(Fullsim)DAOD_RPVLL from existing HITS using r8788 and r8535 events: 20000

| | | | | | | | | | | |
|-------|-------|--|--|-------|-------|--|--|--|--|---|
| e5818 | s2726 | | | r9999 | p9999 | | | | | not_submitted_original edit saved |
|-------|-------|--|--|-------|-------|--|--|--|--|---|

1 + MC15.308265.Pythia8EvtGen_A14NNPDF23LO_LLzprimemumu_m100t250.py
(Fullsim)DAOD_RPVLL from existing HITS using r8788 and r8535 events: 20000

| | | | | | | | | | | |
|-------|-------|--|--|-------|-------|--|--|--|--|---|
| e5818 | s2726 | | | r9999 | p9999 | | | | | not_submitted_original edit saved |
|-------|-------|--|--|-------|-------|--|--|--|--|---|



Filtering and rearranging slices in request

- For easy of steps management slices could be filtered or sorted
 - Slices could be rearranged by input dataset/joboptions

Total input: 4, from them approved: 4

Select All

Filter by:

slice data ↕

Filter:

Filter by status:

all ↕

Sort:

slice ID ↕



Submitting(Approving) steps

- After steps are ready slices should be submitted(approved)
 - All or only part of steps could be submitted
 - DEFT asynchronously creates task or generate errors for approved steps

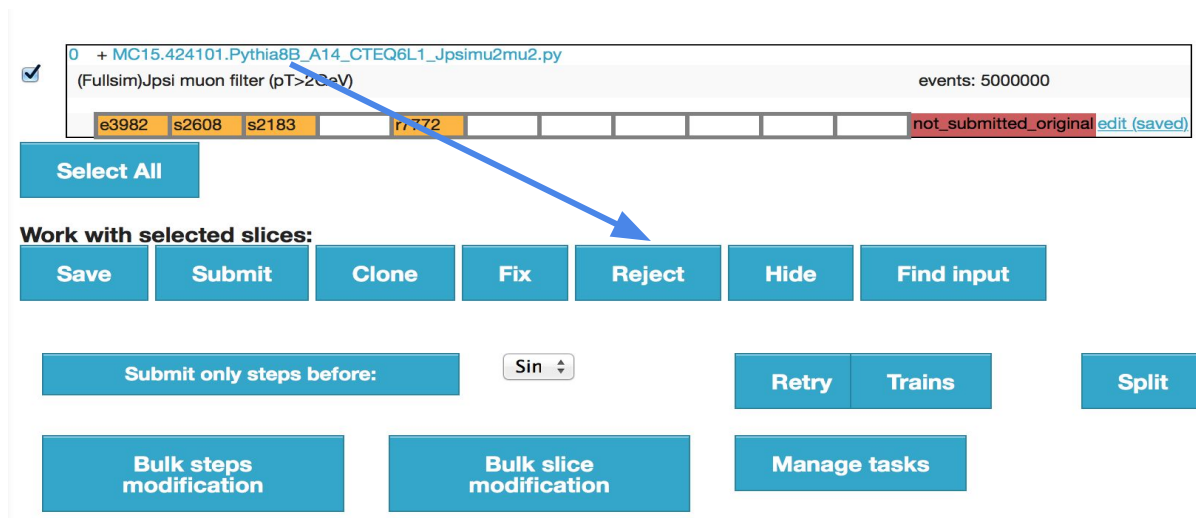
The screenshot displays a task interface with the following elements:

- Task Header:** A checked checkbox, a task ID (+ MC15.424101.Pythia8B_A14_CTEQ6L1_Jpsimu2mu2.py), a description (Fullsim)Jpsi muon filter (pT>2GeV), and event count (events: 5000000).
- Slices Table:** A table with columns for slice IDs (e3982, s2608, s2183, r7772) and a status column (not_submitted_original). A blue arrow points to the 'Submit' button below the table.
- Action Buttons:** A 'Select All' button, a 'Work with selected slices:' section containing 'Save', 'Submit', 'Clone', 'Fix', 'Reject', 'Hide', and 'Find input' buttons, and a 'Submit only steps before:' button with a 'Sin' dropdown menu. A blue arrow points to the 'Submit only steps before:' button.
- Additional Buttons:** 'Retry', 'Trains', 'Split', 'Bulk steps modification', 'Bulk slice modification', and 'Manage tasks' buttons.



Rejecting steps

- Only non-approved steps could be modified
 - 'Reject' button should be used if approved steps require modification



The screenshot shows a web interface for managing simulation steps. At the top, there is a table with a header row containing a checkbox, a step ID, a description, and an event count. Below the header is a row of colored buttons representing different steps. A blue arrow points from the 'Reject' button in the 'Work with selected slices:' section to the 'not_submitted_original' button in the table.

| <input checked="" type="checkbox"/> | 0 | + MC15.424101.Pythia8B_A14_CTEQ6L1_Jpsimu2mu2.py | (Fullsim)Jpsi muon filter (pT>20GeV) | events: 5000000 | | | | | | | |
|-------------------------------------|-------|--|--------------------------------------|-----------------|--|--|--|--|--|------------------------|--|
| | e3982 | s2608 | s2183 | r, 772 | | | | | | not_submitted_original | edit (saved) |

Select All

Work with selected slices:

Save **Submit** **Clone** **Fix** **Reject** **Hide** **Find input**

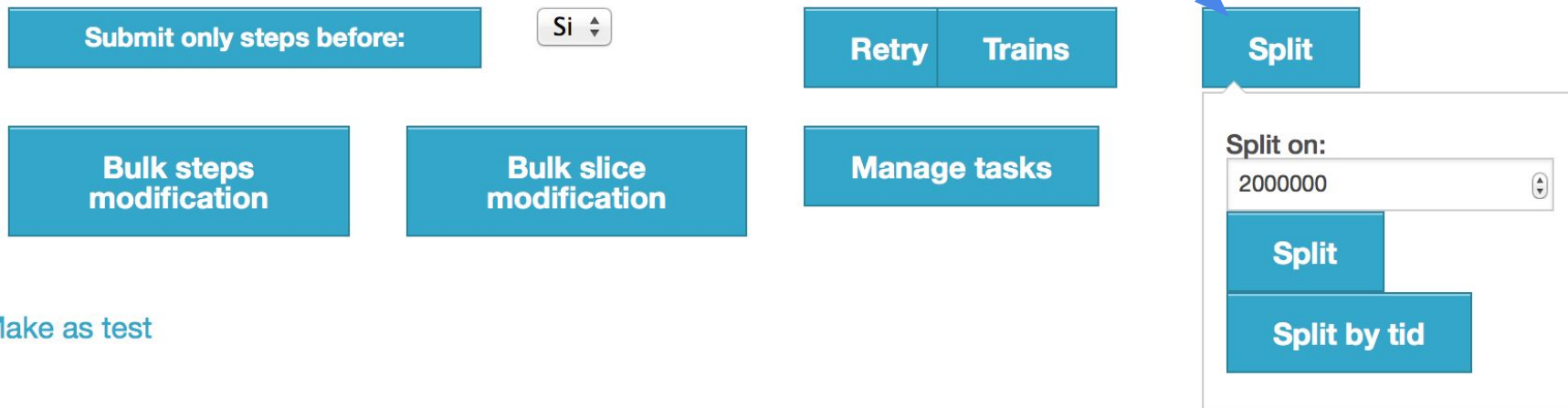
Submit only steps before: **Retry** **Trains** **Split**

Bulk steps modification **Bulk slice modification** **Manage tasks**



Splitting slices

- Slices could be splitted by number of events or by tid in containers
 - E.g. if some part of events should be run on HPC

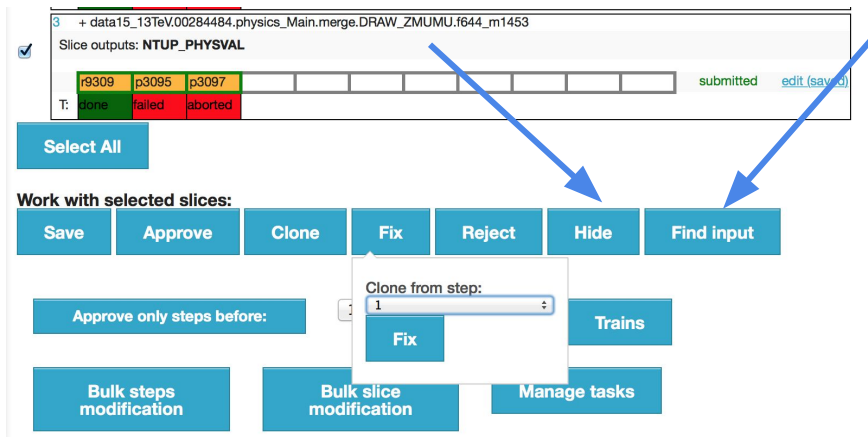


Make as test

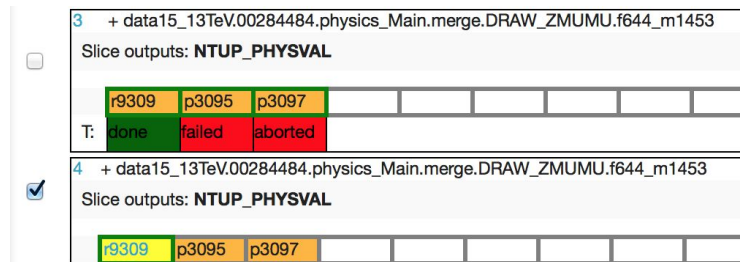


Fixing broken tasks

- If task is failed/broken a new task with the same parameters and fixes can be submitted by cloning steps
 - ‘Clone’ allows to clone all steps in slices
 - ‘Fix’ should be used if some part of slice succeed. In this case succeed task become parent for cloned slices. E.g. if first two tasks done, ‘fix’ should be used with option ‘clone from step 2’



The screenshot shows a task management interface. At the top, a task is listed with ID 3 and name '+ data15_13TeV.00284484.physics_Main.merge.DRAW_ZMUMU.f644_m1453'. Below the task name, the slice outputs are listed as 'NTUP_PHYSVAL'. A progress bar shows three steps: 'r9309' (green, done), 'p3095' (yellow, failed), and 'p3097' (red, aborted). Below the progress bar, there are buttons for 'Select All', 'Save', 'Approve', 'Clone', 'Fix', 'Reject', 'Hide', and 'Find input'. A 'Fix' button is highlighted with a blue arrow. A dropdown menu is open for the 'Fix' button, showing 'Clone from step:' with a value of '1' and a 'Fix' button. Other buttons include 'Approve only steps before:', 'Bulk steps modification', 'Bulk slice modification', and 'Manage tasks'.



The screenshot shows a task management interface with two tasks. Task 3 is at the top, with the same name and slice outputs as in the previous screenshot. Task 4 is below it, with the same name and slice outputs. The progress bar for task 4 shows three steps: 'r9309' (green, done), 'p3095' (yellow, failed), and 'p3097' (yellow, failed). A blue arrow points from the 'Fix' button in the previous screenshot to the 'Fix' button in this screenshot.



Task monitoring and actions

- BigPanda monitor is the first place to check task progress and status
- Many actions on tasks are available
 - Abort/Finish
 - Retry
 - Change priority
 - ...

| | |
|-----------------|---|
| Input dataset: | data16_13TeV.00307306.physics_Main.recon.DAOD_EMU.r9264_r9264_tid11079096_00 |
| Output dataset: | data16_13TeV.00307306.physics_Main.merge.log.r9264_r9264_p3083_tid11079252_00 data16_13TeV.00307306.physics_Main.merge.DAOD_EMU.r9264_r9264_p3083_tid11079252_00 |
| Physics tag: | |
| Hashtags: | |



Find request/task

- Request could be found by filtering request table page
 - https://prodtask-dev.cern.ch/prodtask/request_table/
- Task could be found by filtering task table page
 - https://prodtask-dev.cern.ch/prodtask/task_table/
- Task in request could be filtered by clicking 'Manage tasks' button

The screenshot displays the Prodsys2 interface. At the top, a slice output table shows the following data:

| Step | Status |
|-------|---------|
| r9309 | done |
| p3095 | failed |
| p3097 | aborted |

Below the table are several action buttons: **Select All**, **Save**, **Approve**, **Clone**, **Fix**, **Reject**, **Hide**, and **Find input**. There is also a section for **Work with selected slices:** with buttons for **Approve only steps before:** (with a dropdown set to 1), **Retry**, **Trains**, **Bulk steps modification**, **Bulk slice modification**, and **Manage tasks**. A blue arrow points to the **Manage tasks** button.

Below the buttons is a table with the following columns: TaskID, Owner, RequestID, Step, (Current) Priority, Total jobs, Done jobs, Failure %, and Status. The table contains 12 entries:

| TaskID | Owner | RequestID | Step | (Current) Priority | Total jobs | Done jobs | Failure % | Status |
|----------|-------|-----------|------|--------------------|------------|-----------|-----------|---------|
| 11060501 | mann | 11607 | Reco | 900 | 40 | 0 | 100 | failed |
| 11060550 | mann | 11607 | Reco | 900 | 45 | 0 | 100 | failed |
| 11060561 | mann | 11607 | Reco | 900 | 40 | 0 | 100 | failed |
| 11060572 | mann | 11607 | Reco | 900 | 45 | 0 | 100 | failed |
| 11060506 | mann | 11607 | Reco | 900 | 0 | 0 | | aborted |
| 11060553 | mann | 11607 | Reco | 900 | 0 | 0 | | aborted |
| 11060564 | mann | 11607 | Reco | 900 | 0 | 0 | | aborted |
| 11060575 | mann | 11607 | Reco | 900 | 0 | 0 | | aborted |
| 11060493 | mann | 11607 | Reco | 900 | 81 | 80 | 0 | done |
| 11060546 | mann | 11607 | Reco | 900 | 83 | 83 | 0 | done |

At the bottom of the interface, there are search filters for various fields (TaskID, Owner, RequestID, Step, Site, Cue, Totals, Done, Failure, Status) and a navigation bar with buttons: **Abort**, **Finish**, **Retry**, **Reassign**, **Parameters**, **Obsolete**, **Kill jobs**, and **Ctrl**. The page shows 1 to 10 of 12 entries, with page 1 of 2 selected.



Hashtags

- “A **hashtag** is a type of label or [metadata tag](#) used on [social network](#) and [microblogging](#) services which makes it easier for users to find messages with a specific theme or content.” © Wiki
- Hashtag provides a way for the cross-request task filtering
 - Better monitoring
 - Simplified management
- Hashtag could be set manually for task and request
- For MC Hashtag fills from joboption keywords

| | |
|------------------------|---|
| Input dataset: | mc15_13TeV:mc15_13TeV.159000.ParticleGenerator_nu_E50.evgen.EVNT.e3711 |
| Output dataset: | mc16_13TeV.159000.ParticleGenerator_nu_E50.simul.log.e3711_s3126_tid10728597_00 mc16_13TeV.159000.ParticleGenerator_nu_E50.simul.HITS.e3711_s3126_tid10728597_00 |
| Physics tag: | |
| Hashtags: | singleParticle ParticleGenerator mc16campaign nu MC16a_CP |



Hashtags views(under active developmnet)

https://prodtask-dev.cern.ch/prodtask/request_hashtags_campaign/ #/ https://prodtask-dev.cern.ch/reqtask/hashtags/&MC16a_CP

Hashtags for mc16 campaign

Selected:

Manage tasks

Show statistic

Filter

| Step Name % | Simul | Merge | Reco | Rec Merge |
|------------------------|-------------------------------|-------------------|-------------------|-------------------|
| Processed/Input events | 40.94% | 100.0% | 100.0% | 100.0% |
| | 1,058,688,310 / 2,585,674,600 | 899,600 / 899,600 | 449,800 / 449,800 | 449,800 / 449,800 |

| Main(AND) | Filter(OR) | Exclude(NOT) | HashTag | N tasks |
|--------------------------|--------------------------|--------------------------|-----------------------|---------|
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | Higgs | 8 |
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | bottom | 1 |
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | 2tau | 49 |
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | MadGraphPythia8EvtGen | 12 |
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | Z | 100 |
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | electron | 21 |
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | jets | 174 |
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | drellYan | 33 |
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | neutrino | 15 |
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NLO | 154 |
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | Sherpa | 164 |
| <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | 2electron | 33 |

Show 10 entries

Search:

Select all

Select Filtered

Deselect all

| TaskID | Owner | RequestID | Step | (Current) Priority | Total jobs | Done jobs | Failure % | Status |
|----------|-------|-----------|------|--------------------|------------|-----------|-----------|---------|
| 11060501 | mann | 11607 | Reco | 900 | 40 | 0 | 100 | failed |
| 11065650 | mann | 11607 | Reco | 900 | 45 | 0 | 100 | failed |
| 11065661 | mann | 11607 | Reco | 900 | 40 | 0 | 100 | failed |
| 11065672 | mann | 11607 | Reco | 900 | 45 | 0 | 100 | failed |
| 11060506 | mann | 11607 | Reco | 900 | 0 | 0 | | aborted |
| 11065653 | mann | 11607 | Reco | 900 | 0 | 0 | | aborted |
| 11065664 | mann | 11607 | Reco | 900 | 0 | 0 | | aborted |
| 11065675 | mann | 11607 | Reco | 900 | 0 | 0 | | aborted |
| 11060493 | mann | 11607 | Reco | 900 | 81 | 80 | 0 | done |
| 11065646 | mann | 11607 | Reco | 900 | 83 | 83 | 0 | done |

Search TaskID

Search Owner

Search RequestID

Search Step

Search (Current) Priority

Search Total jobs

Search Done jobs

Search Failure %

Search Status

Showing 1 to 10 of 12 entries

Previous 1 2 Next

Abort

Finish

Retry

Reassign

Parameters

Obsolete

Kill jobs

Ctrl



Authentication

- System authentication is based on CERN SSO
 - <https://sso-management.web.cern.ch>
 - Familiar for everyone
- Authorisation is based on e-groups
 - PMG approval
 - Tasks actions
 - Step submission
 - ...
- API could use same authorisation mechanism



How to control Production System and JED

...



How to control Production System and JEDI. Event generation

- Finding inputs and parameters using JO on cvmfs
 - /cvmfs/atlas.cern.ch/repo/sw/Generators/\${campaign}JobOptions/latest/
 - JO file: share/DSID\${YYY}xxx/\${capmaign}.\${YYY}000.name.py
 - DEFT finds inputs for specified energy using dictionary file: share/evgeninputfiles.csv

| DSID | Energy | inputeventfile | inputconffile |
|------|--------|----------------|---------------|
| | | | |

- If JO file contains 'evgenConfig.inputconfcheck' then 'inputconfigfile' value should be filled (inputGenConfFile)
- If JO file contains 'evgenConfig.inputfilecheck' then 'inputeventfile' value should be filled (inputGeneratorFile)
- If there are no rows for specified {DSID, energy} or necessary values are not filled then DEFT generates error
- nEventsPerJob: DEFT reads this parameter from JO file ('evgenConfig.minevents=\$')



Event generation (2)

| | nEvents | nEventsPerJob | nEventsPerInputFile | randomSeed | firstEvent | Input offset |
|-------------------------------------|------------------|-----------------------|--------------------------------|---|--|----------------------|
| NO input | Number of events | Number events per job | -- | Number of processed events / events per job | Number of processed events | -- |
| 1 input file inputGenConfFile | -- | Number events per job | Number of events | Number of processed events / events per job | Number of processed events | -- |
| N input files inputGeneratorFile | -- | Number events per job | Number of events in input file | Number of already used files | Number of used files * events per file | Number of used files |

If 'isOTF=yes' is specified in project_mode then the task is submitted without input



Task splitting

- DEFT uses container as input in the following cases
 - Evgen
 - Simul
 - project_mode contains mergeCont=yes
- Task splitting is activated in other cases. The following actions are automated:
 - Lists all datasets in a container
 - Gets number of already processed events in other tasks
 - Calculates offset for each dataset or skips dataset if fully processed
 - Submits one task per dataset
- mc15_13TeV.410503.PowhegPythia8EvtGen_A14_ttbar_hdamp258p75_dil.merge.AOD.e5475_s2726_r7772_r7676/
 - mc15_13TeV.410503.PowhegPythia8EvtGen_A14_ttbar_hdamp258p75_dil.merge.AOD.e5475_s2726_r7772_r7676_tid09886344_00 → task 11071455
 - mc15_13TeV.410503.PowhegPythia8EvtGen_A14_ttbar_hdamp258p75_dil.merge.AOD.e5475_s2726_r7772_r7676_tid10872901_00 → task 11071459



Task configuration (1)

- Task can be configured using
 - Web I/F: AMI tag, Events per Input file, Events per job, Total events, Output formats (“XXX.YYY.ZZZ”), etc.
 - Special parameter project_mode

| | | | | |
|--|--|-----------------------|--------------------|---------------------------|
| AMI tag p3075 | Step is skipped <input type="checkbox"/> | Events per Input file | Events per job | Total events -1 |
| Files per job | nGB per job 25 | | | max failure attempt 15 |
| Output formats DAOD_HIGG2D2 | | | Input format | |
| project mode merging=p3077:corecount=8;cloud=WORLD:outputRatio=16 | | | Priority 560 | |
| Destination | | | | |
| JEDI internal merging: | | | | |
| merging tag | nFilesPerMergeJob 1 | nGBPerMergeJob | nEventsPerMergeJob | nMaxFilesPerMergeJob |
| Previous tasks | | | | |

- “param1_name=param1_value;param2_name=param2_value;...;paramN_name=paramN_value”



Task configuration (2)

- Most project_mode parameters are defined in TWiki (https://twiki.cern.ch/twiki/bin/viewauth/AtlasComputing/ProdSys#PROJECT_MODE)
- cmtconfig
 - DEFT checks that cmtconfig is exist in specified cache using AGIS API
 - There are default cmtconfig values
 - If default cmtconfig is not exist in cache then DEFT uses one of existed
- site: site=XXX,YYY,ZZZ;
 - DEFT checks each site provided using AGIS API
 - Site string is transferred to JEDI if all sites are valid

| Cache release | cmtconfig |
|---------------|-----------------------|
| <=13.X.X | i686-slc3-gcc323-opt |
| <15.6.3 | i686-slc4-gcc34-opt |
| <19.0.3 | i686-slc5-gcc43-opt |
| <20.1.0 | x86_64-slc6-gcc47-opt |
| >= 20.1.0 | x86_64-slc6-gcc48-opt |



Offsets (except Evgen)

- Task parameters with offset controlled by DEfT:
 - randomSeed
 - jobNumber, digiSeedOffset1, digiSeedOffset2
 - primary input offset
- Number of events per file (nEventsPerInputFile) is used from step configuration or from Rucio
- Getting the number of used input files (NF) from the previous tasks with the same input name and configuration (project, tag, input data name) – extensions
- randomSeed = NF, primary input offset = NF
- Generating error if all events are processed in the previous tasks with the same configuration



Task handling

- The following task actions can be performed using Web I/F through DEFT API
 - abort_task, finish_task, reassign_task, change_task_priority, retry_task, obsolete_task, change_task_*, etc.
- All actions are executed asynchronous
- DEFT API performs suitable method of JEDI CLI as 'prodsys' user
- Status of each action request can be checked using DEFT API
 - https://aipanda015.cern.ch/api/v1/request/?username={user}&api_key={key}
 - It supports for searching statuses using name patterns, ids, filters
- All actions are logged in JIRA
 - in the ticket associated with corresponding task
 - <https://its.cern.ch/jira/browse/ATLPSTASKS-XXX>
 - [2017-03-31 17:20:51.808664+00:00] action = "abort_task", owner = "xxxxx", result = "success", parameters: task_id = "11043353"

```
{  
  "action": "change_task_priority",  
  "body": "{\"priority\": \"420\", \"task_id\": 11012061}",  
  "created": "2017-03-31T20:19:32.191041",  
  "id": 237580,  
  "owner": "gingrich",  
  "resource_uri": "/api/v1/request/237580/",  
  "status": "{\"jedi_info\": {\"status_code\": 0, \"return_code\": 1,  
  \"timestamp\": \"2017-03-31T20:19:32.944983\"  
  },
```



Communication with AMI

- DEFT uses the last available version of pyAMI – 5.0.6
- AMI supports configurable failover
 - `client = pyAMI.client.Client(['atlas-replica', 'atlas'])`
 - **atlas-replica** (@CERN) is selected as higher priority data source in DEFT
 - To ensure better stability of AMI during task definition
- Extracting task parameters from AMI tag
 - Support for all types of AMI tags: PS1, oldStructure, newStructure
- Getting the list of parameters of TRF to define necessary task parameters
- Syncing information about projects (every 1h), data types (every 1h) and physics containers (1 time per day) in DEFT DB and AMI



Communication with Rucio

- Getting information about datasets, files and containers from Rucio during task definition
 - To find suitable input for each task
 - To perform auto task splitting
 - To define special task parameters (Overlay, etc.)
- Reading metadata of datasets to fill necessary task parameters (depending on nEventsPerInputFile) and to provide protection against duplication
- DEFT performs data placement (registers containers) and data deletion (task and chain obsoleting) during post production



Error handling in JIRA

- CERN SSO is used to access to JIRA API
- DEFT creates JIRA tickets associated with requests and tasks automatically
- All errors during task definition appear in the associated JIRA ticket (for request) automatically
- If some tasks are not submitted then the 'Red box' with link to JIRA ticket appears on the request page
https://prodtask-dev.cern.ch/prodtask/inputlist_with_request/XXXX/

Error: Some task can't be created by DEFT: [ATLPSTASKS-1009601](#)

- All user actions on task are logged in JIRA ticket associated with task



Common task definition errors

- “The task is rejected because of inconsistency. XXX”
 - nEventsPerJob of parent is not equal to specified nEventsPerInputFile
 - To fix: nEventsPerInputFile should be changed
 - nEventsPerJob is not divisible by nEventsPerInputFile without remainder AND nEventsPerInputFile is not divisible by nEventsPerJob without remainder
 - To fix: nEventsPerInputFile or nEventsPerJob should be changed
- “Input data list is empty”
 - No inputs or JO are provided
 - To fix: check step/request parameters
- “Invalid request parameter: DSID”, “Invalid request parameter: Energy”, “Suitable XXX candidate not found in evgeninputfiles.csv”
 - Wrong energy provided for Evgen step of there is no necessary line in the file ‘share/evgeninputfiles.csv’ on cvmfs
 - To fix: check request parameter or update/fix ‘evgeninputfiles.csv’ file



Common task definition errors

- “Output data are missing”, “These requested outputs are not defined properly: ZZZ”
 - The task cannot be defined without output but TRF does not support for some of specified output formats (“XXX.YYY.ZZZ”)
 - To fix: check step parameters (output formats), check TRF (asetup ..., *_tf.py -dumpgargs)
- “Number of events to be processed is mandatory when task has no input”
 - The task input is not properly defined. Missing dataset/container or JO
 - To fix: check input parameters of step (“Dataset”)
- “[Check duplicates] The task is rejected”, “No more input files”
 - All available events are already processed with given configuration
 - To fix: change step configuration (project, tags, formats, etc.)



Task and job brokerage

- Task
 - Assigned to a nucleus based on
 - Storage and site status
 - Input data locality
 - Accumulation of workload
 - Data transfer backlog
 -
- Jobs
 - Generated from a task
 - Assigned to the nucleus, where the task has been assigned, and satellites based on
 - Storage and site status
 - Requirements on software, memory, walltime, IO intensity, and core count
 - Data transfer backlog
 - Restriction on types of jobs which can run on the site
 - Site activity
 - Satellites are dynamically associated to nuclei based on static configuration and dynamic measurements of network connectivity between nuclei and satellites



Acknowledgements

Thanks to many ADC colleagues for materials used in this presentation

