

Table of Contents

PanDA share distribution and monitoring	1
What is the idea behind?.....	1
Shares at batch level.....	1
Optimising resources.....	1
Design.....	2
Open questions/things to discuss.....	2

PanDA share distribution and monitoring

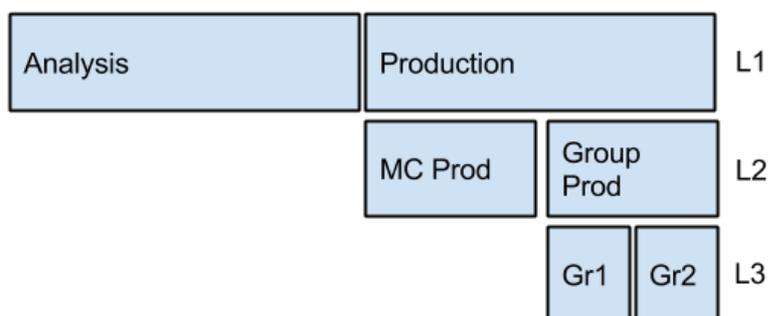
🔧 WORK IN PROGRESS

What is the idea behind?

Currently PanDA job assignment is done according to priorities: higher priority jobs are first in line to be sent to a site. At occasions, the current implementation is not sufficient to let the operations team control the jobs traffic and prioritize urgent campaigns without manually reshuffling priorities.

Therefore the request from ADC Ops is to have a multi-level/nested share system that allows changing the fraction of resources assigned to the different activities. One example:

- 50% of resources go to Analysis and 50% to Production.
- Out of the Production share, the operator wants to define that 50% goes to MC Production and 50 % to Group Production, and so on.
- If MC Production is not consuming their 50% share, then Group Production can overflow and fill it up - but not Analysis... Analysis can only overflow if Production is not filling the Production share.



Shares at batch level

Sites keep shares at batch level to control the ratio between production and analysis jobs, based on APF sending pilots with two different roles -production and pilot role- to the sites. Because of security reasons (e.g. an analysis user accidentally deleting data), this situation will likely not change in the short-term future. Therefore, the PanDA distributor system should only keep track of production shares.

Optimising resources

Sites usually have sets of different machines available (e.g. 8 core,16 GB | 16 core, 32 GB | 16 core, 48 GB) and ATLAS needs to match the different job types:

- MSCORE: 8 core, 16 GB
- MSCORE_HM: 8 core, 24 GB
- SCORE: 1 core, 2 GB
- SCORE_HM: 1 core, 4 GB

Depending on the combinations of jobs on a machine, we can be using the resources in a suboptimal manner. Obviously it is outside ATLAS'/PanDA's control which machine a job is going to end up on. However PanDA can for example try to keep the flow of MSCORE jobs to a site constant, to avoid draining machines to allocate MSCORE jobs.

Design

The proposal for implementation is an independent and asynchronous service that will communicate with the JEDI and PanDA server. - The PanDA Distributor:

1. Figures out how many slots are available to ATLAS (e.g. 150.000)
2. Figures out the capabilities of the slots
3. Splits the slots according to the shares and capabilities

- PanDA server brokerage:

1. Queries the Distributor service
2. Brokers
3. Updates the Distributor service

There will be a Web interface to monitor and manage shares.

Open questions/things to discuss

- How will the batch shares and the Distributor live together? Focus only on production?
- Interactions between JEDI and Distributor.

This topic: PanDA > PanDADistributor

Topic revision: r3 - 2015-04-15 - FernandoHaraldBarreiroMegino



Copyright &© 2008-2020 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? Send feedback