iDDS integration

Wen Guan, Tadashi Maeno, Gancho Dimitrov
Brian Bockelman, Torre Wenaus
Fernando Barreiro, Fahui Lin, Rui Zhang, Misha Borodin, Paul Nilsson

May 28, 2020
WFMS
iDDS Status

❖ Main architecture (production)
  ➢ iDDS database, core, REST API
  ➢ Plugins
  ➢ Agents
  ➢ Watchdogs

❖ Documents & monitors
  ➢ Home page: https://idds.cern.ch
  ➢ Codes: https://github.com/HSF/iDDS
  ➢ Documents: https://idds.readthedocs.io (dev)
  ➢ ATLAS Monitor: https://bigpanda.cern.ch/idds/

❖ Use cases
  ➢ Fine-grained data carousel -- ready
  ➢ Hyperparameter tuning -- integrating
  ➢ Decision making for active learning -- developing
  ➢ Other usecases in 2020
iDDS Data Carousel
Integration Test Status with data15 reprocessing

- 99% staged in 4 days
  - Many files were handled shortly after the request was submitted
  - Few files were stuck on tape, finished some days later.
- 63 datasets
- 426,331 files
- 850TB
- iDDS was not fully occupied, will test with more tasks

<table>
<thead>
<tr>
<th>COUNT(*)</th>
<th>SUM(IN_TOTAL_FILES)</th>
<th>SUM(IN_BYTES)</th>
<th>SUM(OUT_PROCESSED_FILES)</th>
</tr>
</thead>
<tbody>
<tr>
<td>63</td>
<td>426331</td>
<td>932290670456064</td>
<td>426331</td>
</tr>
</tbody>
</table>

![Request Table](image)
iDDS data carousel with all datasets

- **Data Carousel Integration with all datasets (mainly data15 and zerobias)**
  - 667 datasets
  - 768, 115 files
  - 1.2 PB
  - iDDS was not fully occupied, will test with more tasks, finished zerobias datasets in few days.

<table>
<thead>
<tr>
<th>request_id</th>
<th>scope</th>
<th>name</th>
<th>status</th>
<th>transform_status</th>
<th>in_status</th>
<th>in_total_files</th>
<th>in_processed_files</th>
<th>out_status</th>
<th>out_total_files</th>
<th>out_processed_files</th>
</tr>
</thead>
<tbody>
<tr>
<td>724</td>
<td>21423307</td>
<td>21423307</td>
<td>Finished</td>
<td>Finished</td>
<td>Closed</td>
<td>1</td>
<td>1</td>
<td>Closed</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>723</td>
<td>21422500</td>
<td>21422500</td>
<td>Transforming</td>
<td>Transforming</td>
<td>Closed</td>
<td>1</td>
<td>1</td>
<td>Processing</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>722</td>
<td>21420035</td>
<td>21420035</td>
<td>Transforming</td>
<td>Transforming</td>
<td>Closed</td>
<td>1</td>
<td>1</td>
<td>Processing</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>721</td>
<td>21407107</td>
<td>21407107</td>
<td>Finished</td>
<td>Finished</td>
<td>Closed</td>
<td>1</td>
<td>1</td>
<td>Closed</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>720</td>
<td>21405829</td>
<td>21405829</td>
<td>Transforming</td>
<td>Transforming</td>
<td>Closed</td>
<td>1</td>
<td>1</td>
<td>Processing</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>719</td>
<td>21404562</td>
<td>21404562</td>
<td>Transforming</td>
<td>Transforming</td>
<td>Closed</td>
<td>1</td>
<td>1</td>
<td>Processing</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>718</td>
<td>21404113</td>
<td>21404113</td>
<td>Transforming</td>
<td>Transforming</td>
<td>Closed</td>
<td>1</td>
<td>1</td>
<td>Processing</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>717</td>
<td>21391145</td>
<td>21391145</td>
<td>Finished</td>
<td>Finished</td>
<td>Closed</td>
<td>1</td>
<td>1</td>
<td>Closed</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>716</td>
<td>21391132</td>
<td>21391132</td>
<td>Transforming</td>
<td>Transforming</td>
<td>Closed</td>
<td>1</td>
<td>1</td>
<td>Processing</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>715</td>
<td>21390625</td>
<td>21390625</td>
<td>Transforming</td>
<td>Transforming</td>
<td>Closed</td>
<td>1</td>
<td>1</td>
<td>Processing</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

A lot of tasks have finished.
The monitor has increased from 7 pages to 70 pages
iDDS
HyperParameterOptimization (HPO)
iDDS HPO (Hyper Parameter Optimization)

❖ Purpose
➢ Using iDDS to generate hyperparameters and trigger production system to automatically process training with the new hyperparameters.
➢ Status: Integrating it with JEDI (successful task https://bigpanda.cern.ch/task/21423307/)

❖ workflow:

![Diagram showing the workflow of iDDS HPO](attachment:image.png)
iDDS HPO (Hyper Parameter Optimization)

❖ iDDS Messages
➢ New content: When a new output content(file) is created but not processed.
➢ Available content: When an output content(file) is available (successfully processed or evaluated)
➢ Available Collection: When all contents(files) in a collection(dataset) are available.

❖ iDDS processing:
➢ iDDS runs a hyperparameter generator in a local condor cluster again and again, until the task finishes.
➢ Every time the hyperparameter generator will read all evaluated hyperparameters with registered loss, based on them, new hyperparameters are generated for a new loop.
➢ iDDS currently developed two predefined hyperparameter generators: bayesian and nevergrad.
➢ iDDS also supports docker containers and developed an example for users to define their own generators.
➢ Documents are in https://idds.readthedocs.io.
iDDS HPO (Hyper Parameter Optimization)

❖ iDDS RESTful client for HPO
  ➢ get_hyperparameter: to get hyperparameters
  ➢ update_hyperparameter: to register loss results.

❖ iDDS HPO integration with JEDI, Pilot:
  ➢ JEDI HPO consumes ‘new content’ messages:
    ■ New content: for new hyperparameters: to generate event ranges within the ES framework.
  ➢ New transform runHPO-00-00-01 running in Pilot:
    ■ Get event range from panda.
    ■ Get hyperparameter from iDDS with event range id(hyper parameter id) through iDDS REST.
    ■ Run ML training
    ■ Register loss results to iDDS through iDDS REST.
    ■ Update event range status
  ➢ Finish task
    ■ When all hyperparameters are evaluated, iDDS publish ‘Collection available’ messages.
    ■ JEDI consumes this message and finishes the task.

One successful task: https://bigpanda.cern.ch/task/21423307/
iDDS Active Learning (AL)

To integrate it with Prodsys2
Active Learning

- Running tasks on top of results of old tasks
- Decision making to generate new tasks from old results
  - Light job, good to execute it immediately and then trigger next step. iDDS can get rid of some latency.

Workflow with grid entities

- Production system processes the normal task
- iDDS runs the Decision Making parts (with/without merge parts)

- Each job runs evgen and derivation sequentially
- Each task generates many multi-step jobs
- Once the first task is finished output are merged and some values are calculated
- Parameters for the next task are decided using the values
- Better to get rid of latency in the intermediate steps (merge+physics and decision making)
iDDS Active Learning(AL)

❖ Decision making job will run in iDDS
❖ iDDS messaging and iDDS processing will be similar with HPO.
❖ iDDS AL integration with prodsys2 (to discuss)

➢ Option 1: similar workflow as HPO.
  ■ Prodsys2 sends one AL call to iDDS Rest at the beginning of the AL task.
  ■ iDDS generating parameters with initial parameters.
  ■ Prodsys2 consumes the ‘new content’ message to create an AL task.
  ■ Prodsys2 needs to send information to iDDS when a task finishes, for example, the output dataset. Then iDDS can evaluate the new dataset and generates new parameters.
  ■ Here only one call with multiple updates. However, prodsys2 needs to register new output dataset information when it finishes, just like to register HPO loss in HPO workflow.
iDDS Active Learning (AL)

❖ iDDS AL integration with prodsys2 (to discuss)

➢ Option 2:
  ■ Prodsys2 can send another call to iDDS.
  ■ iDDS runs the Decision making job and returns the outputs.
  ■ Prodsys2 decides whether to stop or generate another task, for example: stop when the returned outputs is empty.
  ■ One call per AL task. Totally there will be multiple calls.

![Diagram of iDDS AL process]

User ➔ Submit a AL request ➔ Production system ➔ Submit one AL call ➔ iDDS

AL task ➔ Generating AL task ➔ The task finishes ➔ Prodsys2 decides whether to create a new task ➔ Generating new parameters, trough new content messages (ActiveMQ) ➔ Prodsys2 decides whether to create a new task