

DKB/DCC STATUS REPORT

Grigorieva M., Golosova M., Borodin M.

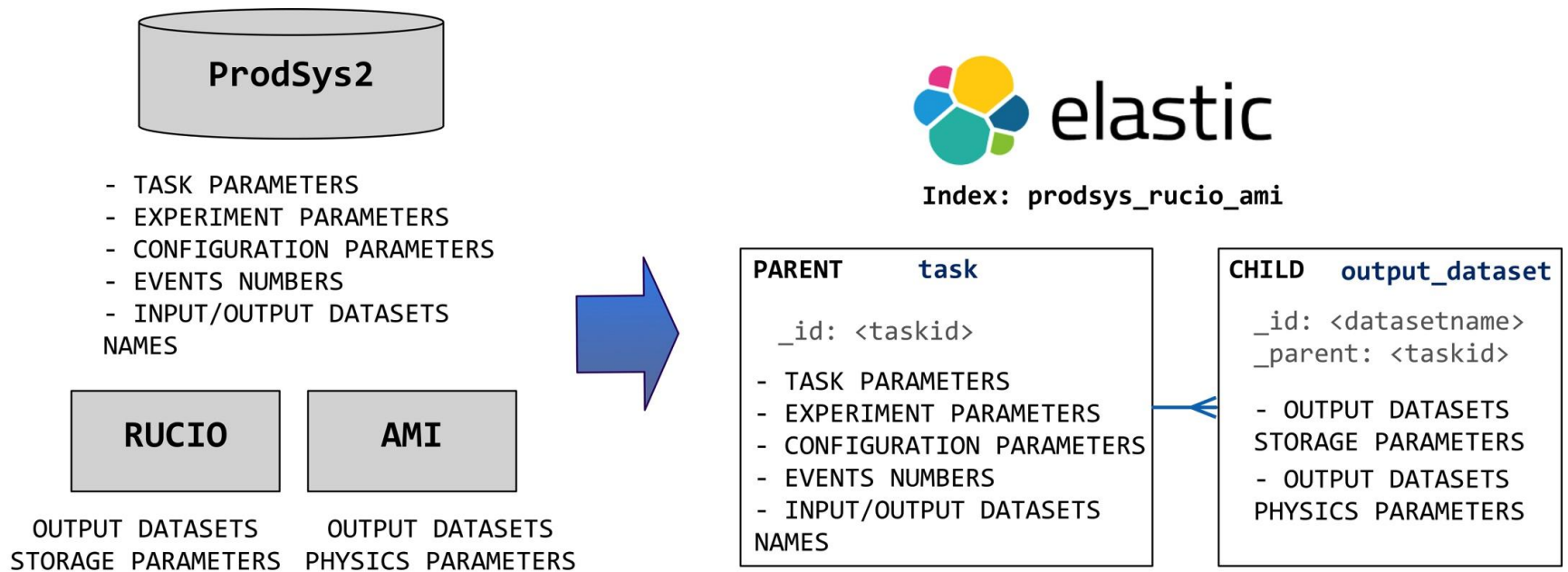
DKB/DCC Outline

2

- Developed the following data processing scripts:
 - Extract metadata from the initial data sources:
 - ProdSys2 (bulk (timestamp-based) extraction)
 - AMI (by dataset name)
 - Rucio (by dataset name)
 - Preparation of the metadata to import into the ElasticSearch
 - adding the indexing meta information to the records
 - Bulk ES import procedure
 - Supervising script that chains the scripts, extracting metadata from ProdSys2, AMI and Rucio together
- Metadata from ProdSys2, Rucio, AMI was integrated and imported to the ElasticSearch Storage
- Data update is possible only basing on timestamp parameter from ProdSys2

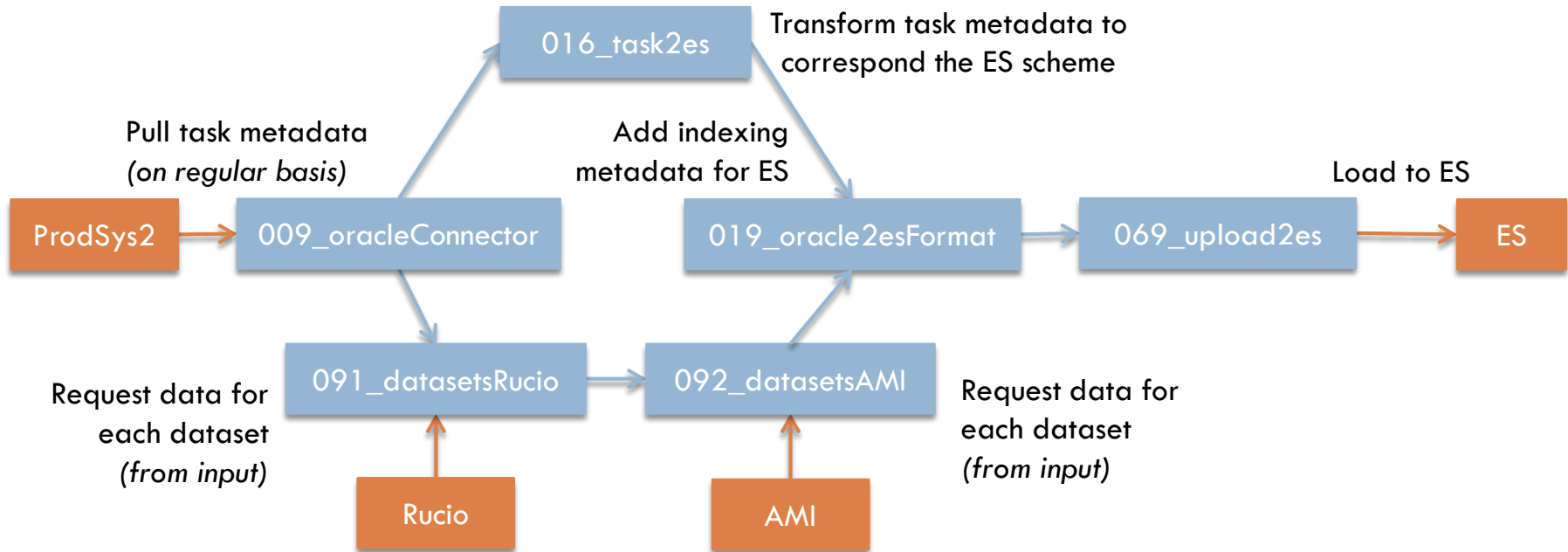
DKB/DCC Data Model

3



Data Processing Pipeline

4



Supervising script that chains, extracting data from ProdSys2, AMI and Rucio together, organizing a continuous pipelines to ES:

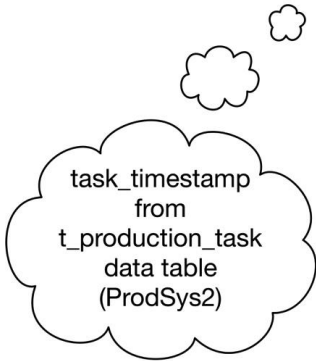
- ProdSys2 records with task metadata are appended with categorization (physics category) information and go to ES;
- in parallel, from task metadata are taken output dataset names to query data from Rucio and AMI;
- information from AMI is appended to the data from Rucio;
- records with data from both AMI and Rucio go to ES.

Data Synchronization

Initial data period

01/01/2016 -
01/01/2017

_id 8268906
_id 8268912
_id 8345439
_id 8178689
_id 8268904
_id 8268910
_id 8308266
_id 8301471
_id 8275546
_id 8308262
_id 8217090
_id 8176528
_id 8176783
_id 8192209
_id 8178729
... ..
... ..
... ..
_id 8177813
_id 8187288



Next Day

01/01/2017 -
02/01/2017

_id 9384509
_id 9383786
_id 9247635
_id 9174823
_id 8301471
_id 8308262

Updated data period

01/01/2016 -
02/01/2017

_id 8268906
_id 8268912
_id 8345439
_id 8178689
_id 8268904
_id 8268910
_id 8308266
_id 8301471
_id 8275546
_id 8308262
_id 8217090
_id 8176528
_id 8176783
_id 8192209
_id 8178729
... ..
... ..
... ..
_id 8177813
_id 8187288
_id 9384509
_id 9383786
_id 9247635
_id 9174823

Rewriting records

Add new records

DKB/DCC Conclusions

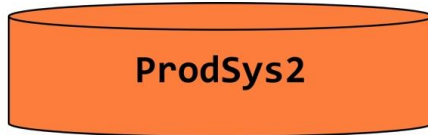
6

- Metadata since 01/01/2016 for tasks and datasets are loaded into the ElasticSearch of DKB instance at CERN
 - ▣ aiatlas171.cern.ch:9200
- Synchronization:
 - ▣ Data update process for tasks and related datasets is initiated when the task timestamp in ProdSys2 is updated to the "current".
 - ▣ How do we get updates from AMI and Rucio?
 - The easiest way would be to have a possibility to ask for data "changed since moment X" on a regular basis.
 - We also thought about receiving notifications about changes from the external systems in more or less real-time mode (e.g. via some REST API).
- Future Plans:
 - ▣ Automatic deployment via Puppet
 - ▣ Kafka-driven data processing flow

7

Addition slides

Metadata in ProdSys2, Rucio and AMI



TASK PARAMETERS

taskid
taskname
status
task_timestamp
start_time
end_time
request_id
ticket_id
username
description
step_name
task_type
output_formats
ctag

EXPERIMENT PARAMETERS

energy_gev
campaign
subcampaign
project
phys_group
phys_category
hashtag_list
run_number

CONFIGURATION

geometry_version
conditions_tags
core_count
architecture
trans_home
trans_path
trans_uses
vo
trigger_config
job_config
evgen_job_opts
cloud
site

EVENTS

requested_events
processed_events

DATASETS

primary_input
output

OUTPUT DATASETS STORAGE PARAMETERS

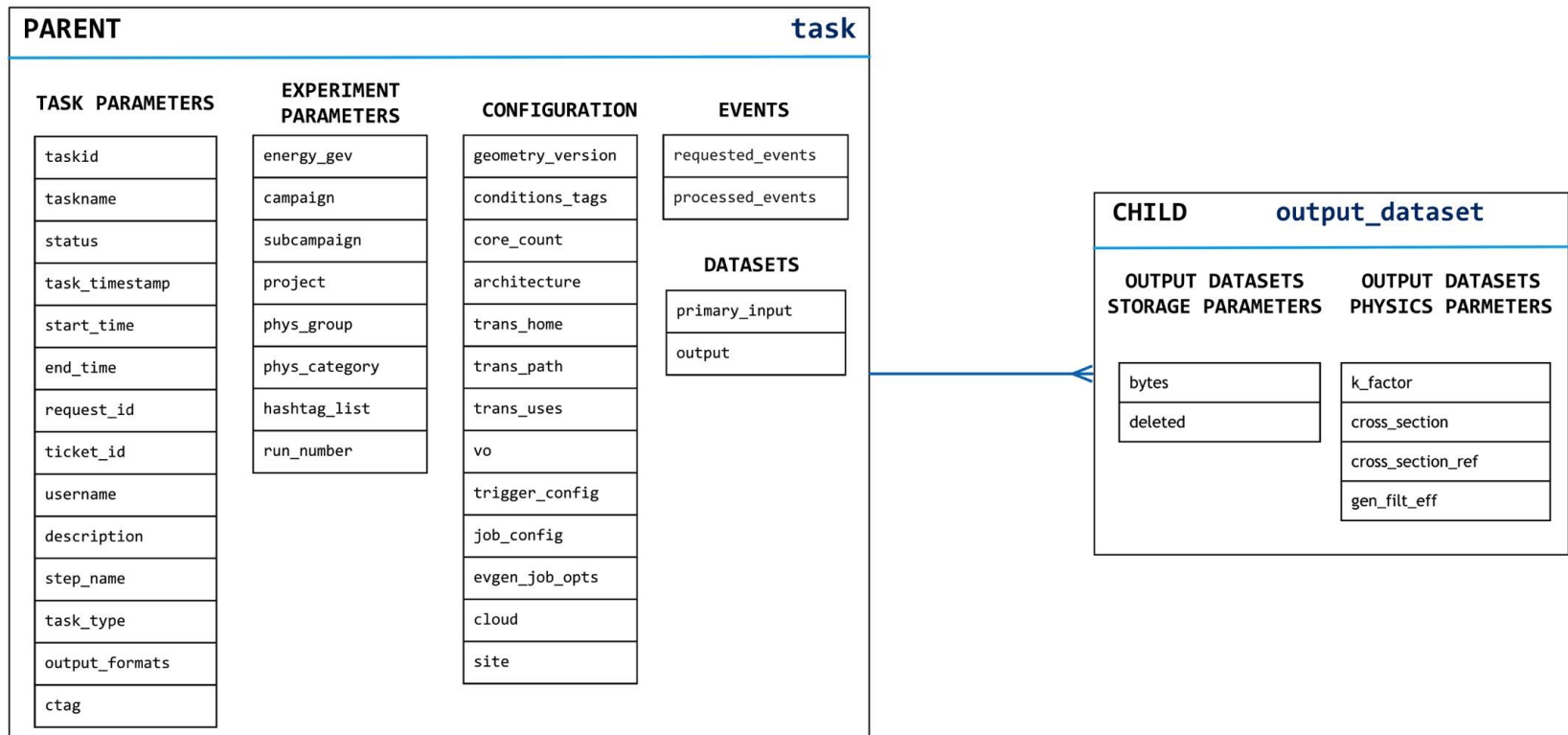
bytes
deleted

OUTPUT DATASETS PHYSICS PARAMETERS

k_factor
cross_section
cross_section_ref
gen_filt_eff

Data model in Elasticsearch

9



Parent-Child Relationship in Elasticsearch

10

- Allows to associate one entity to another in one-to-many relationship, and all entities live within separate documents.
 - ▣ The parent document can be updated without reindexing the children.
 - ▣ Child documents can be added, changed or deleted without affecting either the parent or other children.
 - ▣ Child documents can be returned as the result of a search request.

- Elasticsearch Index Mapping General Structure:

```
{
  "template": "prodsys_rucio_ami»,
  "settings": {
    "number_of_shards": 4
  },
  "mappings" : {
    "task" : {
      "properties" : {...}
    },
    {
      "output_dataset": {
        "_parent": {
          "type": «task»
        },
        "properties": {...}
      }
    }
  }
}
```