



НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ ЦЕНТР
«КУРЧАТОВСКИЙ
ИНСТИТУТ»

Data Knowledge Base current status and operation

V. Kotliar NRC “Kurchatov institute” - IHEP, RU-142281, Protvino,
Moscow region, Russia

E-mail: Viktor.Kotliar@ihep.ru

with support from NRC “Kurchatov institute” and from Marina
Golosova, Vasilii Aulov, Mikhail Borodin



The Data Knowledge Base (DKB) project is aimed at knowledge acquisition and metadata integration, providing fast response for a variety of complicated queries, such as summary reports and monitoring tasks (aggregation queries) and multi-system join queries, which are not easy to implement in a timely manner and, obviously, are less efficient than a query to a single system with integrated and pre-processed information would be. In this work the current status of the project as well as its integration with the ATLAS Workflow Management and future perspectives are shown.



Introduction

The Data Knowledge Base (DKB) project is aimed at knowledge acquisition and metadata integration

Started at 2016 v1 main purpose

- Integrate and link pieces of information from independent sources (pdf, indico, wiki pages,...)
- Reconstruct connections between research results and data samples
- Provide fast and flexible access to everything people might want to know about some process or object

2018 v2

- Universal tool for multi-source queries
 - Library pyDKB
 - ATLAS dataflow system
 - ETL flow based on scripts and library
 - System to run and check flow
 - Database to store results
 - REST API to access system
 - Frontend UI for users



Current environment

host	OS	where	Running	Alias	Puppet
aiatlas172	CC7-x86_64	cern-geneva-a	4 years	api.atlas-dkb.cern.ch	master
aiatlas171	CC7-x86_64	cern-geneva-a	4 years	es.atlas-dkb.cern.ch	master
atlas-dkb-dev-1	CC7-x86_64	cern-geneva-c	7 months	-	qa
atlas-dkb-dev-0	CC7-x86_64	cern-geneva-b	7 months	-	qa
dkb-dev-cc7	CC7-x86_64	cern-geneva-b	1 year	-	-

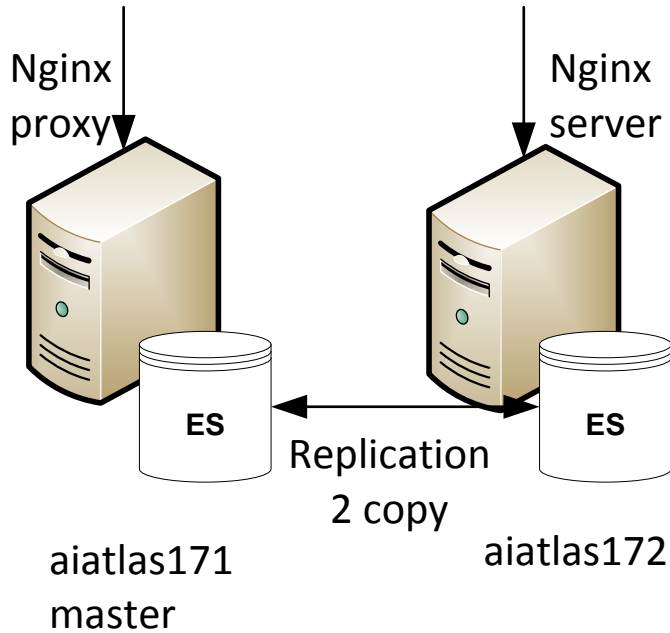
gitlab.cern.ch it-puppet-hostgroup-voatlasdkb
<https://github.com/PanDAWMS/dkb>



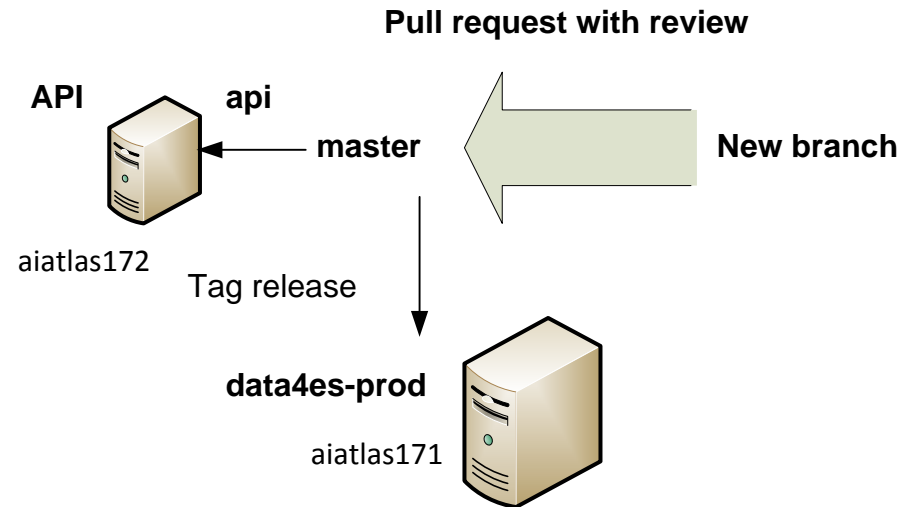
Current environment

es.atlas-dkb.cern.ch

api.atlas-dkb.cern.ch



<https://github.com/PanDAWMS/dkb>



API – api (0.3.3)

DKB flow - data4es-prod (tags v0.2-0)



Current environment: numbers

```
[root@aiatlas171 ~]# curl -sS -X GET 'http://127.0.0.1:9200/_cat/indices?v' | grep -e docs.count -e 0_1_1
health status index          uuid                                pri rep docs.count docs.deleted store.size pri.store.size
green  open   nested_production_tasks_v0_1_1  PxedXrvvTtuDz8oHcx8Tug          4   1    7158575      187500      13.5gb          6.7gb
green  open   nested_analysis_tasks_v0_1_1    mmbaOhMASO-Jry5fz5QALQ          4   1    57789339     196118      51gb           25.5gb
green  open   nested_analysis_progress_v0_1_1  Ab6deEeyR3iRfj5rViNxNg          4   1     2622630         5969      425.2mb         212.2mb
green  open   nested_production_progress_v0_1_1 KR-1DyB3SAischb2ZfqEvA          4   1     1000876         19979     286.4mb         142.6mb
```

DKB workflow runs every hour by cron

- ATLAS production 980 sec (654 tasks 590 datasets)
- ATLAS analysis 316 sec (815tasks 4263datasets)

Consistency check every day

- Check if some records in DB missed in DKB

Access from aipanda[xx].cern.ch

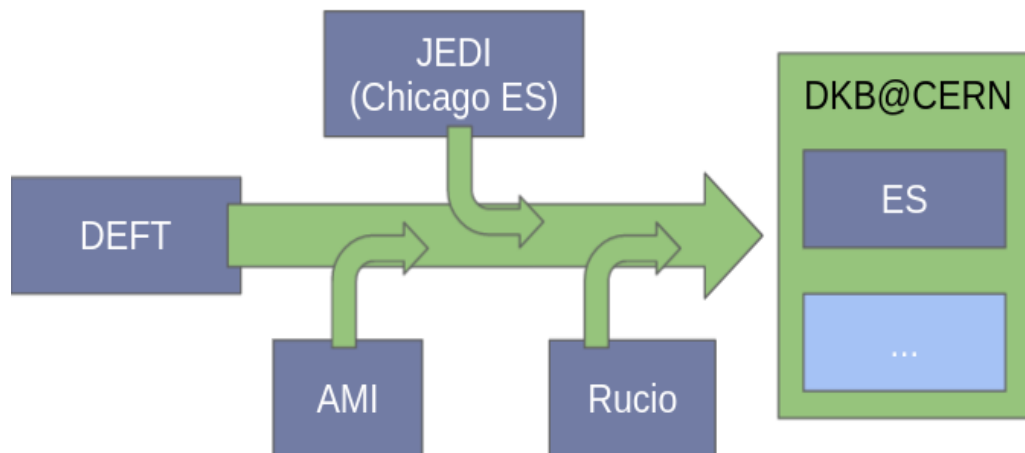
```
cat /var/log/nginx/es.access.log | awk '{print $1;}' | sort | uniq -c
    11 188.184.80.118
    30 188.185.66.16
   2268 188.185.70.23
```



Current DKB metadata integration

ATLAS case:

- information update is based on “task timestamp” from ProdSys;
- main information comes from DEFT (Database Engine for Tasks) and is extended with additional metadata from other systems:
 - AMI – ATLAS Metadata Interface
 - JEDI - Job Execution and Definition Interface
 - Rucio - scientific data management system





Current DKB metadata integration

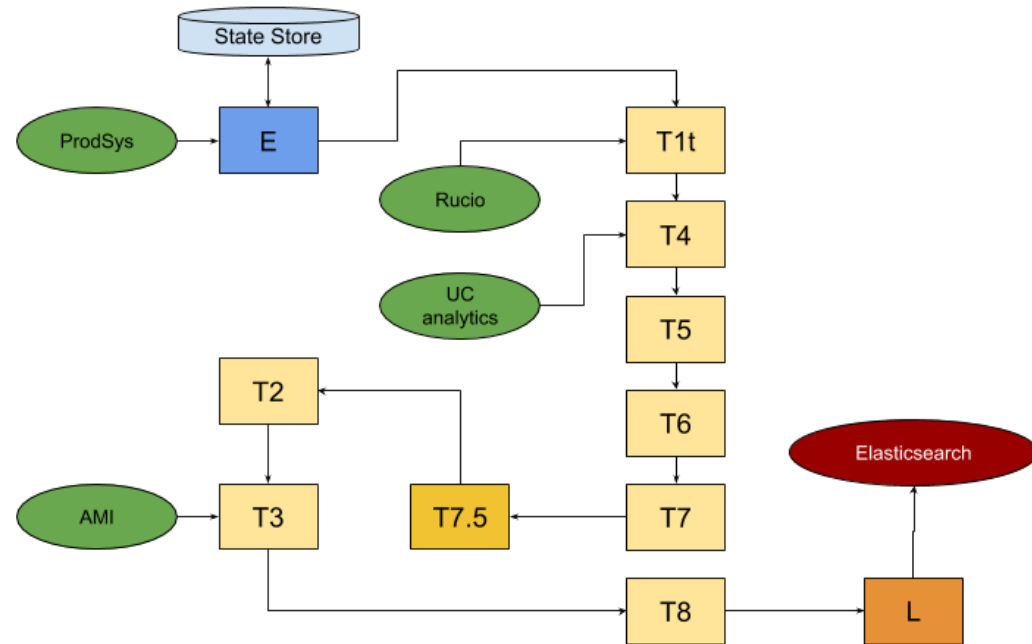
ATLAS flow:

- ETL (Extract, Transform, Load) process
- Single bash script for data flow
 - Stages flow parts
 - DKB python library to build stages

Utils/Dataflow/data4es-nested/run/data4es-start

```
....
source_chain() {
  run_stage '09' | tee $b_process
}
process_chain() {
  cat $b_process | run_stage '91' | eop_filter \
    | run_stage '25' | eop_filter \
    | run_stage '16' | eop_filter \
    | run_stage '17' | eop_filter \
    | run_stage '40' | eop_filter \
    | run_stage '93' | eop_filter \
    | run_stage '95' | eop_filter
}
# Sink chain
sink_chain() {
  [-n "$DEBUG" ] \
  && cmd_69="tee"
  tr $'\n' $'\x1e' | run_stage '19' "$1" | tr -d $'\x1e' | eop_filter \
    | mediator "$1" | run_stage '69' "$1" > /dev/null
}
process_chain | tee $out | sink_chain from95
05.09.2021
```

Utils/Dataflow/data4es-nested/
009_oracleConnector
025_chicagoES
091_datasetsRucio
016_task2es
040_progress
093_datasetsFormat
017_adjustMetadata
069_upload2es
095_datasetInfoAMI
019_esFormat
071_esConsistency





ATLAS DKB UI + REST API

<https://prodtask.cern.ch/dkb/>

Added for ProdSys by Mikhail Borodin

- Google-like search for tasks and related datasets
- High-level aggregated view
- Generated on demand (up to date information)
- Takes only few seconds to load the page
- Based on #tag("universal" may be used for different use-cases)
 - `dkb/#/output_stat`
 - `dkb/#/steps_stat`
 - `dkb/#/deriv_ratio`
 - `dkb/#/task_keywords`

<http://api.atlas-dkb.cern.ch:5080/>

- `/server_info` - Server info
- `[/path/to/category]/info` - Method/category info
- `/task/kwsearch` - Tasks keyword search
- `/task/chain` - Task chain reconstruction
- `/task/hist` - Tasks distribution over time (by steps)
- `/task/deriv` - Derivation efficiency
- `/campaign/stat` - Campaign statistics
- `/step/stat` - Steps statistics

The screenshot shows the ATLAS DKB web interface. At the top, there is a navigation bar with tabs for ATLAS PanDA, Requests, Tasks, Jobs, Train, Datasets, Meta, PanDA Config, Help, and Login. A search bar contains the keyword 'p3553'. Below the search bar, it displays 'Keywords: "p3553"' and 'Tasks found: 3802 Tasks displayed: 2000'. A 'Manage tasks' button is visible. Below this, there is a table with columns for TASK, EXPERIMENT, and CONFIG. The table contains detailed information for a specific task, including taskID, Request, status, Description, User, timestamp, start-end time, HS06 per event, and Outputs. At the bottom of the table, it shows 'EVENTS Requested / Processed' as '25468023 / 25468023'.

TASK		EXPERIMENT		CONFIG
taskID	14119975 BigPanda	AMI tag	p3553	Step Name
Request	17194	Campaign	data18_13TeV	ticket_id
status	done	Subcampaign	openended18	Architecture
Description	open-ended derivation production on data18 main stream with 21.2.30.0 - skim			Core Number
User	egramsta	Energy	13000	ATLAS Geometry
timestamp	17-05-2018 20:17:55	Physics Group	PHYS	Conditions Tags
start - end time	15-05-2018 17:16:51 17-05-2018 17:41:50	Physics Category	["Uncategorized"]	trigger_config
HS06 per event	10	Hashtags		transPath
Outputs	["DAOD_STDM3","DAOD_EXOT19","DAOD_TOPQ5","DAOD_TOPQ2","DAOD_JETM4","DAOD_JETM3"]			transUses
EVENTS Requested / Processed				run number
25468023 / 25468023				



Resent changes (ES store)

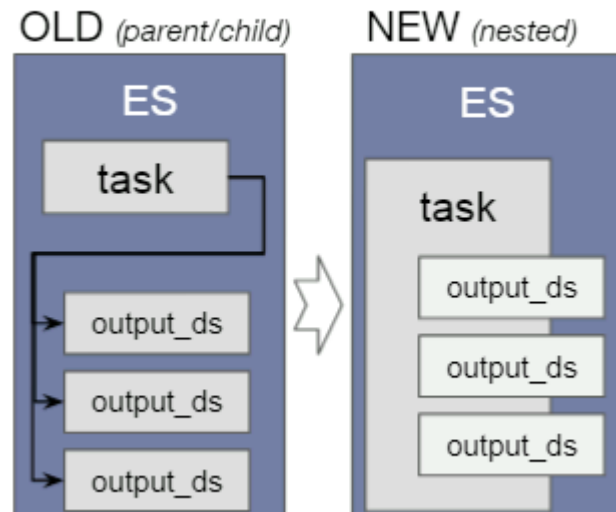
Task (parent)

- task metadata
 - User
 - creation time
 - computing site...
- resource usage
 - CPU
 - Walltime...
- derived values
 - reconstructed task chain root...

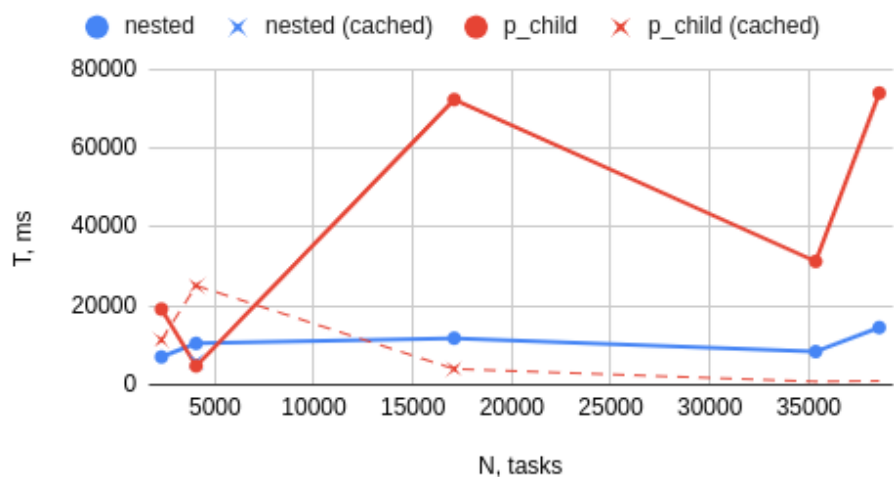
Dataset (child output for task)

- Storage information
- Data characteristics

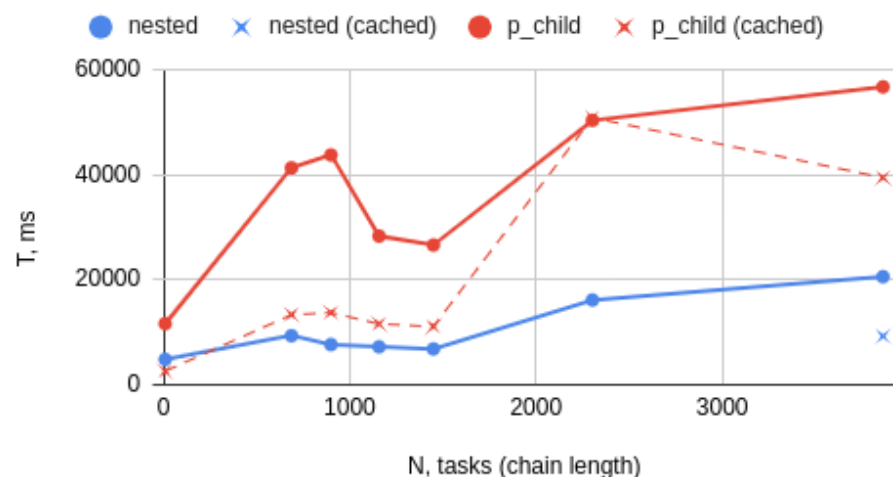
Indexing scheme



Derivation statistics request execution time



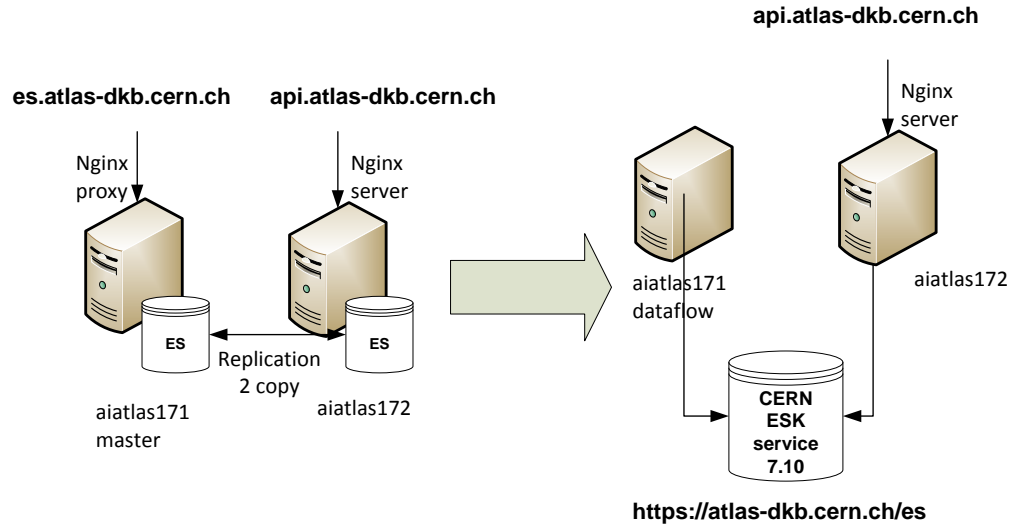
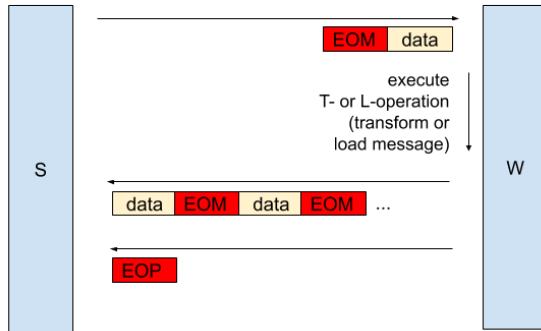
Chain reconstruction request execution time



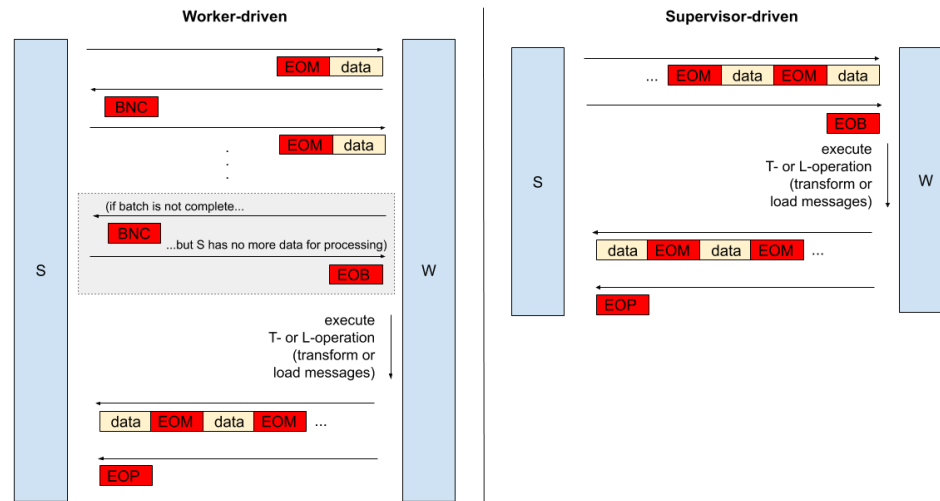


Works in progress

Supervisor-worker interaction: T-, L-stage (single message)



Supervisor-worker interaction: T-, L-stage (batch)



- Batch processing
- ES migration

<https://github.com/PanDAWMS/dkb/wiki/Internal-communication-protocol>



Thank you!

Any questions?