Flexibility extension of the Event Picking service for ATLAS experiment

E.I. Alexandrov[1,a], I.N. Alexandrov[1], D. Barberis[2], A.V. Yakovlev[1]

[1]Joint Institute for Nuclear Research (Russia)
[2]University and INFN Genova (Italy)

MMCP Conference, Yerevan, 21–25 October 2024

The ATLAS EventIndex

- The EventIndex is the global catalogue of all ATLAS events

- For each event, each data format and each processing version, it contains:
  - Event identifiers (run and event number)
  - Location (GUID of the file containing it) and provenance
  - Trigger and other useful metadata

- Main use case is event picking for detailed analysis and/or displays
  - Also production checks and overlap counts

EventIndex for Run 3

- The core data storage system was reimplemented during 2021 and deployed in 2022 for the start of LHC Run 3
  - HBase for the dataset and event tables
  - Phoenix interface for SQL queries
  - New client query service CLI also implemented for optimal performance
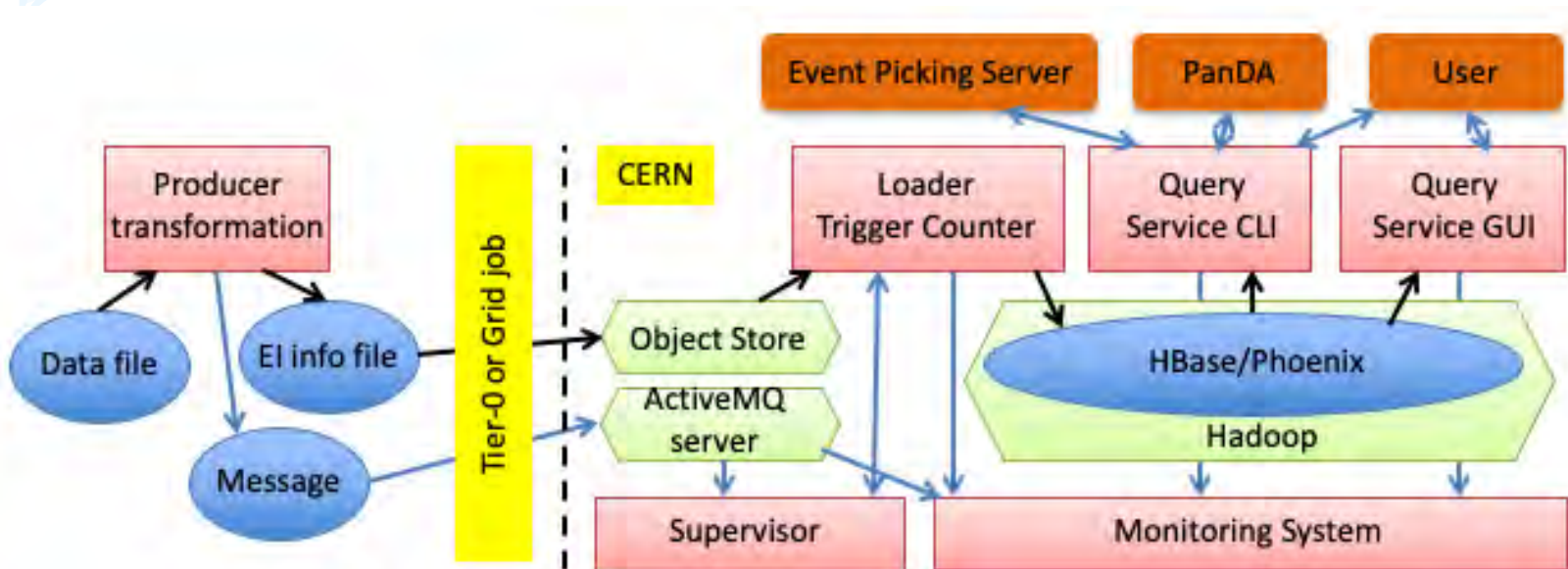
Event Picking Service

- Some physics analyses need to extract many events in order to process them with enhanced algorithms

  - γγ -> WW analysis:
    - The first round 50k events (2019)
    - The second round 136k events (2021)

  - B_c^* -> B_c gamma analysis:
    - 650K events (2023)

  - Z -> TauTau selections:
    - 11K events (2023)

- An automatic system to extract the requested events and deliver them to the requestors is therefore needed: the Event Picking Service

Various input data

- **Data format**
  - Raw (only for real data)
  - AOD
  - Evnt (only for Monte Carlo)
- **Type**
  - Real data
  - Monte Carlo
- **Trigger stream** (only for Real data)
- **AMI tag** AMI tag (software version, not used for RAW data)
- **File containing run and event numbers**

Tasks and problems (1)

- The list and order of tasks may change over time
- Split by Run: required for correct work with minimal time
  - The number of input data can be large
  - Input data may not be in order
  - 1 run 1 file
- Get GUIDs: this information should be added to panda and is required to get the dataset name template
  - Possible error answer from EI

Tasks and problems (2)

- Get Dataset Name: require for panda job
  – Rucio has no information about some GUID
- Start Panda Job: make a real copy of events
  – Long working time
  – The result may be an error (even for valid inputs)
- Validate : should validate the output data
  – Possible duplicate/ skip events
- Set Metadata in RUCIO: panda does not set event count of events
  – Possible big number of output files
- Restart: events that were skipped or have errors
  – Possible big number such events in different runs

Additional requests

Common way

8
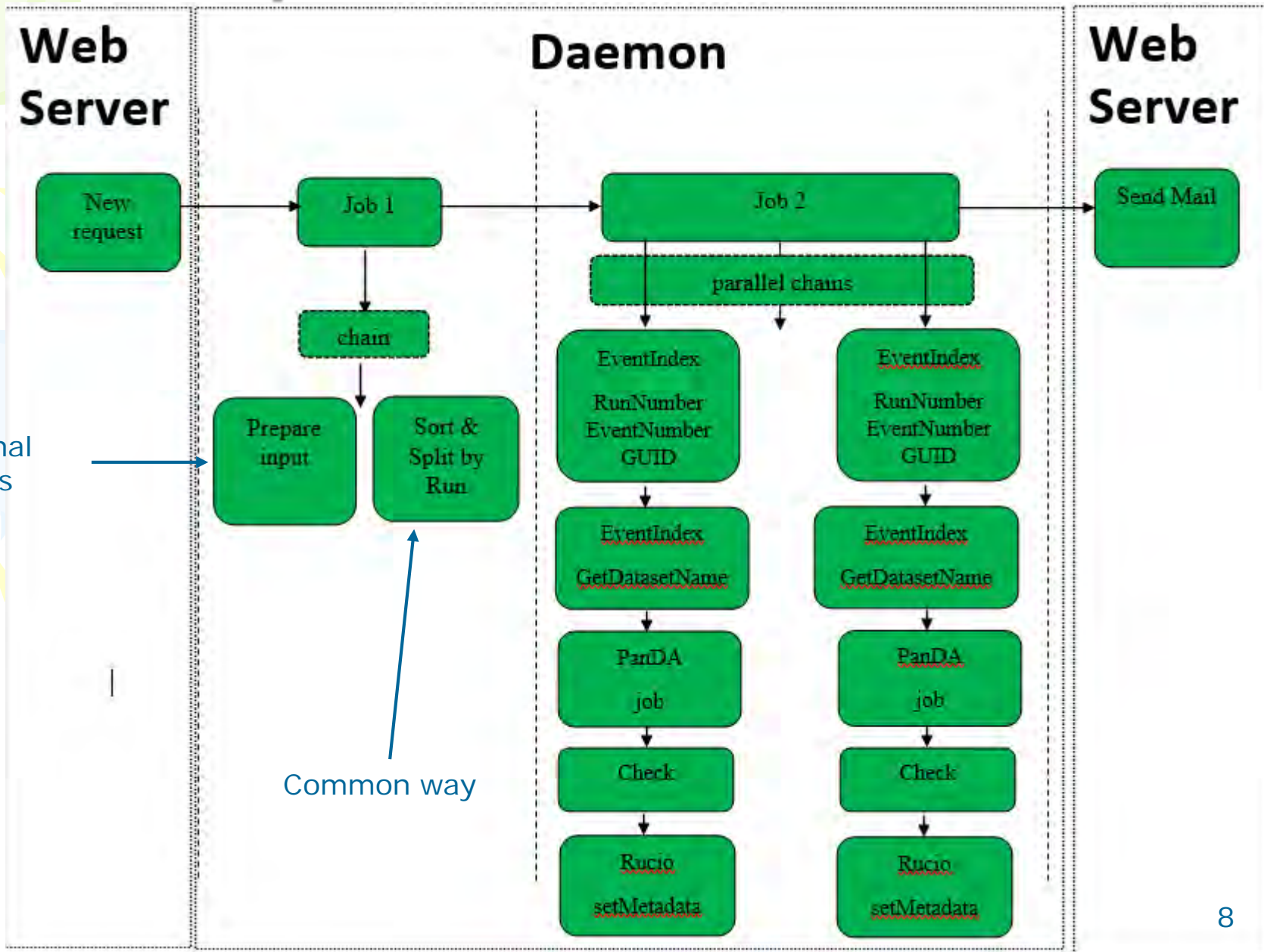
Object model (briefly)
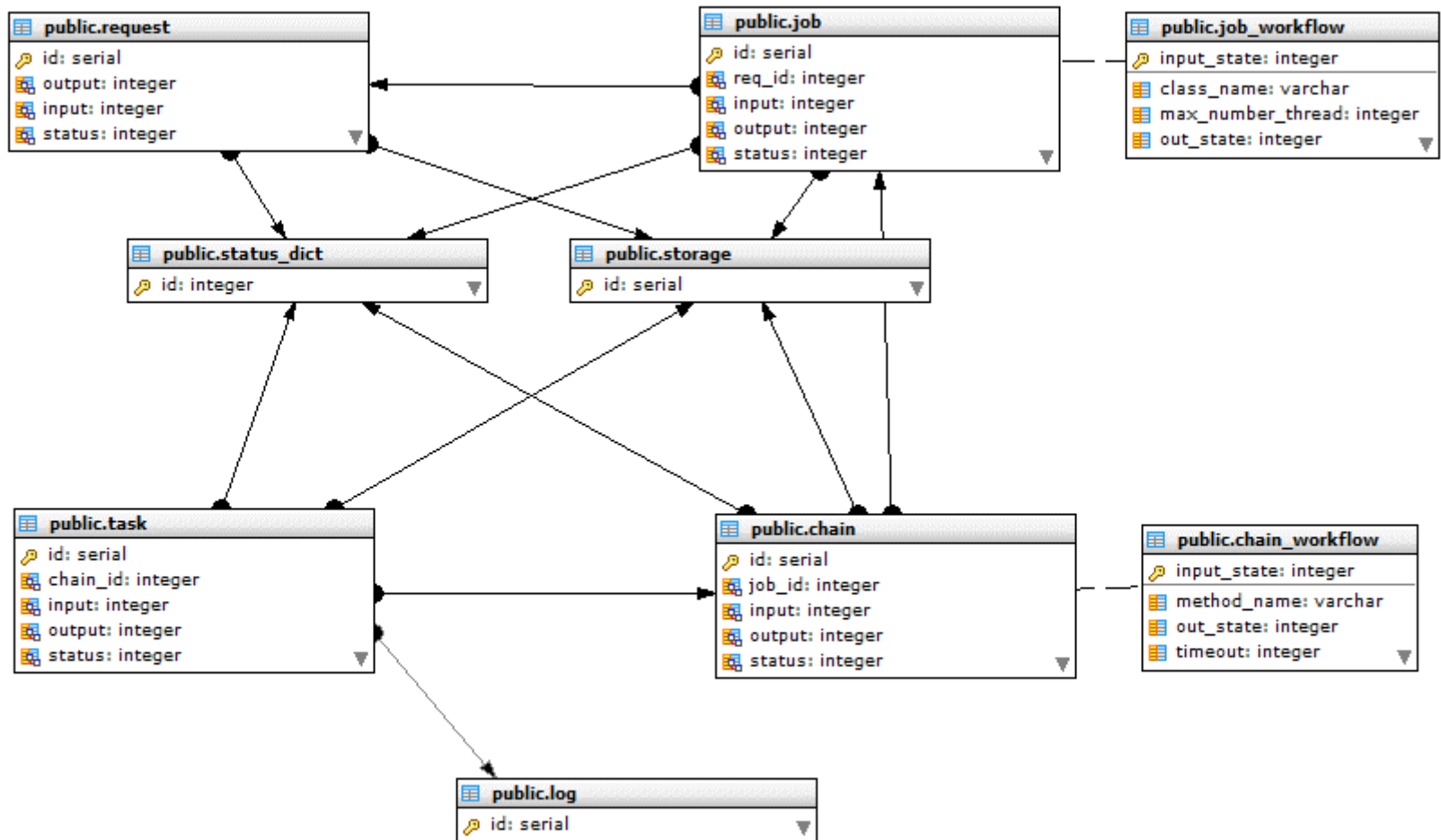
Architecture of the Event Picking service

Grafana

Administrator

Atlas Monitoring

Send Mail

Monitoring Data

Web Server

-Input form
-Monitoring

Client

Database

Daemon

External systems
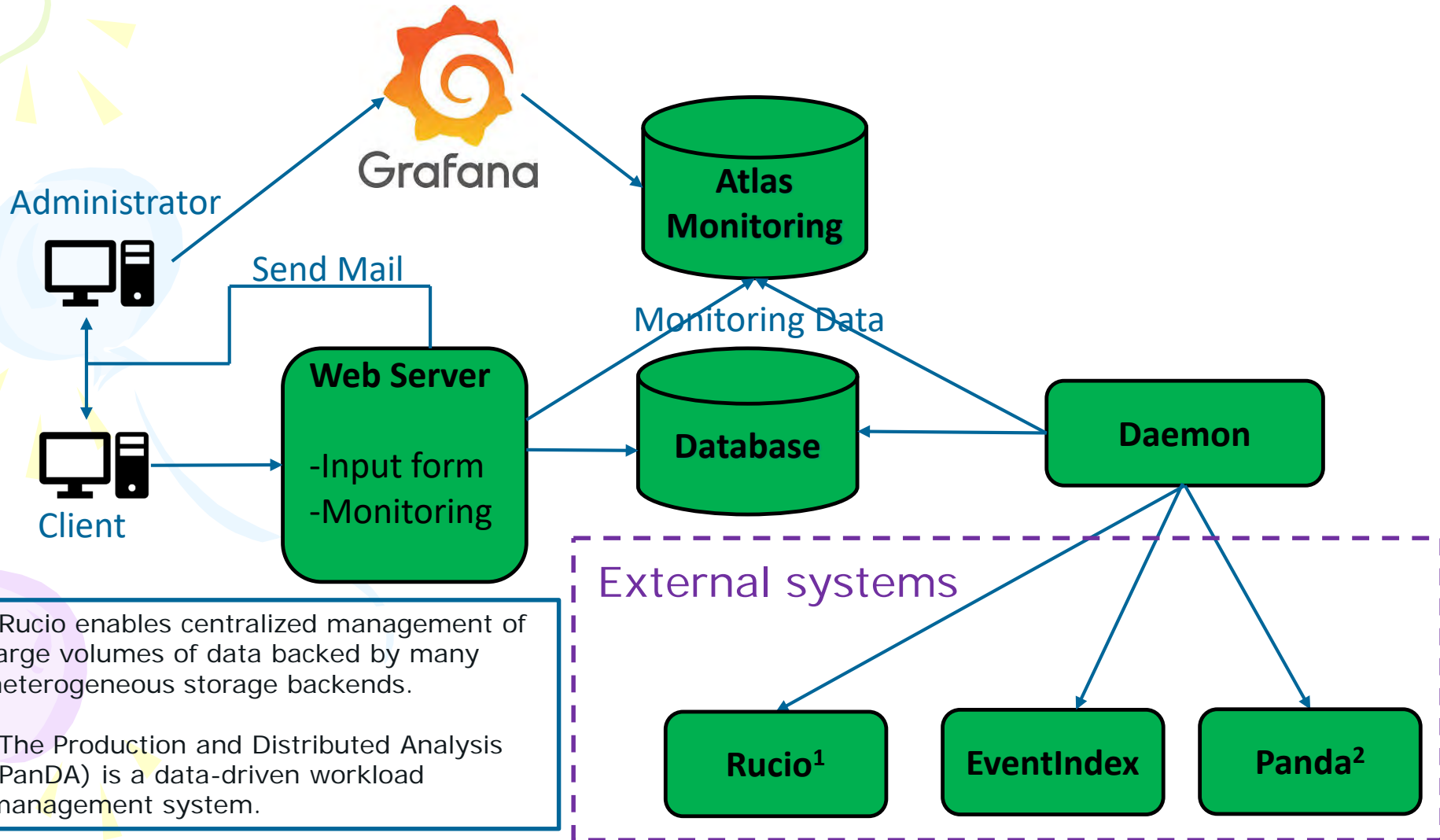
[1]Rucio enables centralized management of large volumes of data backed by many heterogeneous storage backends.

[2]The Production and Distributed Analysis (PanDA) is a data-driven workload management system.

Rucio[1]

EventIndex

Panda[2]

Time results

Request	Number of events	Version	Time
$\gamma\gamma$ -> WW	50k	1.0.0	2 weeks
		manual	3 months
$\gamma\gamma$ -> WW	136k	Beta version	3 months
B_c^* -> B_c	16K	1.2.37	84h
Z -> TauTau	11K	1.2.37	40h

Production server

- Production version is available since middle of 2022.
- Web service is available outside of CERN (need CERN SSO authorization)
- Exploitation of the Event Picking Server revealed deficiencies that were solved by creating new Workflows

Critical error for most of chains

Critical error for a few of chains

Error for a check tasks

343 warning (error but result file exists)

Result files (+)

Request progress

Jobs:	Job Name	Initial state	Current state	Created	Changed	Job Status
└ SplitJob	Sort and split (state: SORT_SPLIT)	Panda job (state: PANDA_PART)	03.05.2024 15:34	03.05.2024 15:34	Finalized. Total=1, errors=0, warnings=0 (Done without error and warning)
└ PandaJob	Panda job (state: PANDA_PART)	Request finished (state: DONE_REQ)	03.05.2024 15:34	20.05.2024 23:57	Finalized. Total=343, errors=0, warnings=343 (Done with check errors)

Chains:	Chain number / Run number	Current state	Created	Changed	Chain status
└	1 363664	Chain finished (state: DONE_CHAIN)	03.05.2024 15:34:17	05.05.2024 11:51:18	Finalized (Done with check errors)

Tasks:	Task Name	Initial state	Current state	Created	Changed	Task status
└	getIndex	Panda job (state: PANDA_PART)	Get dataset name for first guid (state: GET_DATASET_NAME)	03.05.2024 15:34:17	03.05.2024 15:34:47	Finalized (Done without error and warning)
└	getDatasetName	Get dataset name for first guid (state: GET_DATASET_NAME)	Chain workflow. Get GUIDs from EventIndex (state: INDEXED_TASK)	03.05.2024 15:34:47	03.05.2024 15:35:06	Finalized (Done without error and warning)
└	panda	Chain workflow. Get GUIDs from EventIndex (state: INDEXED_TASK)	Chain workflow. Start panda task (state: START_PANDA_TASK) PanDA job	03.05.2024 15:35:06	05.05.2024 11:17:07	Finalized (Done without error and warning)
└	check	Chain workflow. Start panda task (state: START_PANDA_TASK)	Chain workflow. Start panda task (state: START_PANDA_TASK)	05.05.2024 11:17:07	05.05.2024 11:21:53	Wait restart (Error state of task)
└	check	Chain workflow. Start panda task (state: START_PANDA_TASK)	Chain workflow. Start panda task (state: START_PANDA_TASK)	05.05.2024 11:26:53	05.05.2024 11:39:04	Wait restart (Error state of task)
└	check	Chain workflow. Start panda task (state: START_PANDA_TASK)	Chain workflow. Start panda task (state: START_PANDA_TASK)	05.05.2024 11:44:05	05.05.2024 11:51:18	Max number of restart (Error state of task)

The Panda task ran for about 2 days.
Other tasks usually ran for minutes.

Available workflows

Main

– Common workflow for get selected events.

Full restart

– Restart request with the same parameters.

Restart errors and warnings job

– Restart only error and warning job.

Restart check task

– Restart job with check error. Start job since check task.

Finalize request (special system type)

– Use in case of critical error of request. Set error for all not finished job.

16

Conclusion

- The new implementation of the Event Index is working without problems.

- The Event Picking Service is running on the production server.

- Error handling and automatic fixes have been improved, which is why the speed of the new version has been increased.

- The number of users of the service is growing.

- The Event Picking Service is flexible and can be used in other experiments once the required workflow is implemented.