

## Multiple archivers in use:

- EPICS channel archiver: (SLS machine and beamline)
- EPICS archiver appliance: (SwissFEL)
- SF-Databuffer BSREAD (beam-sync.): (SwissFEL)
- SF-Databuffer Channel Access: (HIPA, Proscan, Cryo, ESI, TWLHA, ...)

## Issues with current setup:

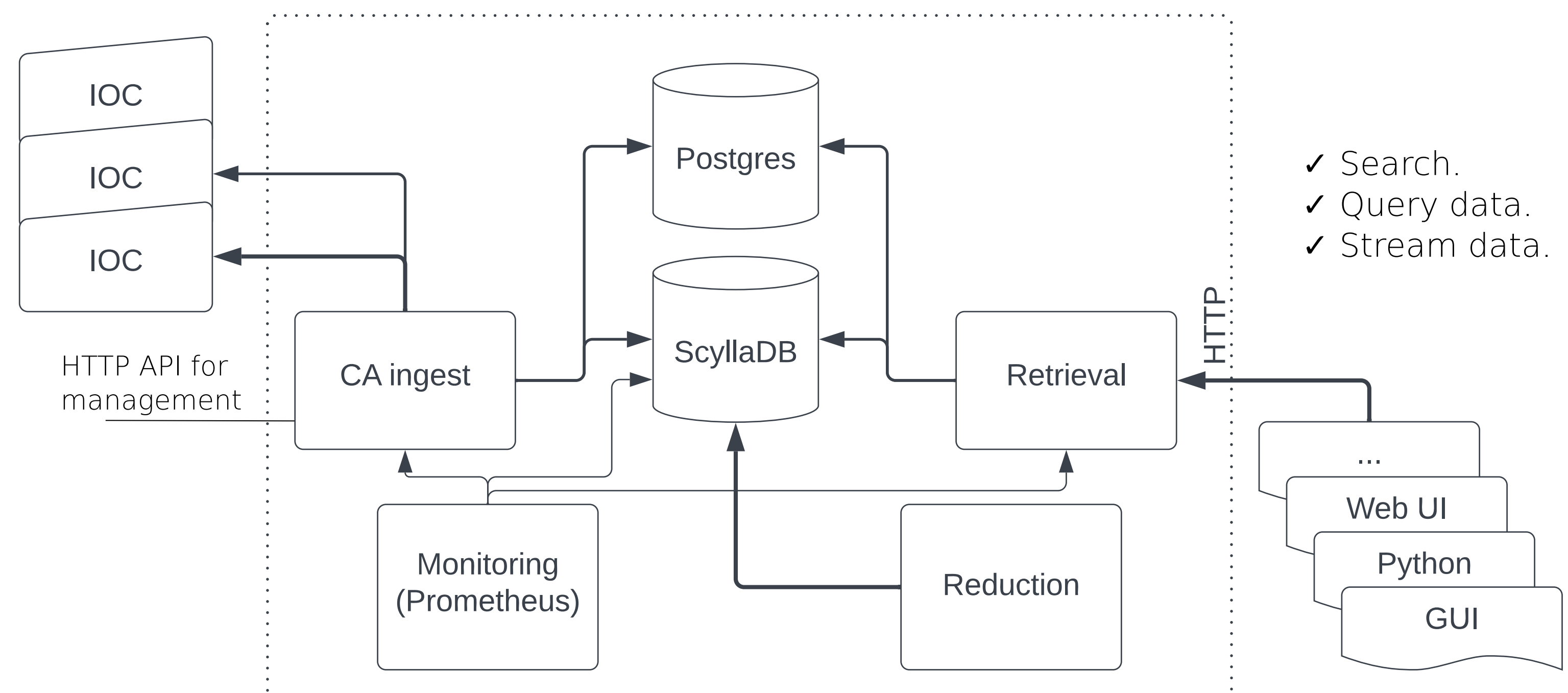
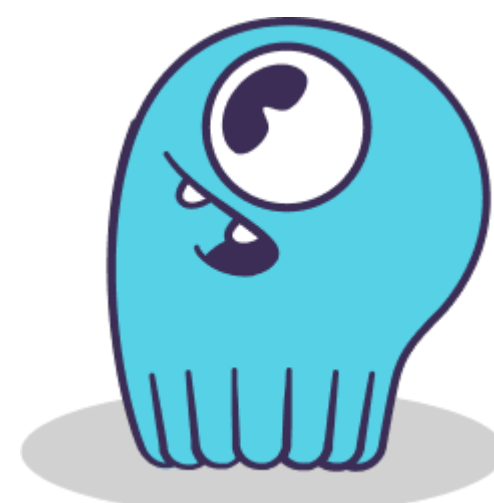
- Overlap in functionality.
- Large maintenance burden.
- Not enough personnel to support all.
- Many underlying custom file formats.
- Missing expertise in sensitive areas (storage engine).
- EPICS channel archiver deprecated.
- Not enough insights into operational issues.

## Wishlist:

- Modularity, interfaces, contracts.
- Factor into:
  - Data store engine.
  - IOC / source communication (ingest).
  - Aggregations and transforms.
  - Retention.
  - Retrieval of data.
  - Metrics, monitoring, alerting.
  - Ingest data from Channel Access, BSREAD, etc.

## ScyllaDB as data store:

- Deployed in industry.
- Open source with paid enterprise support.
- Distributed, typed key-value store.
- Scalable, highly available.
- No single point of failure.
- Tunable redundancy.
- Hot scaling and decommissioning.
- Solid management tools.
- Existing product off the shelf.
- Clearly defined access and formats.
- Suits time-series, but also more general workloads.



- ✓ Search.
- ✓ Query data.
- ✓ Stream data.

## Channel Access Ingest:

- Adapter between Channel Access and ScyllaDB.
- Open and monitor channels.
- Detect broken connections.
- Record details about TCP and channel status.
- Support configuration changes at runtime.
- Protect against free running sources.
- Monitor clock offsets between ingest service and source.
- Offer metrics for Prometheus.
- Make detailed status and config available via http api.

### Configure ingest:

```

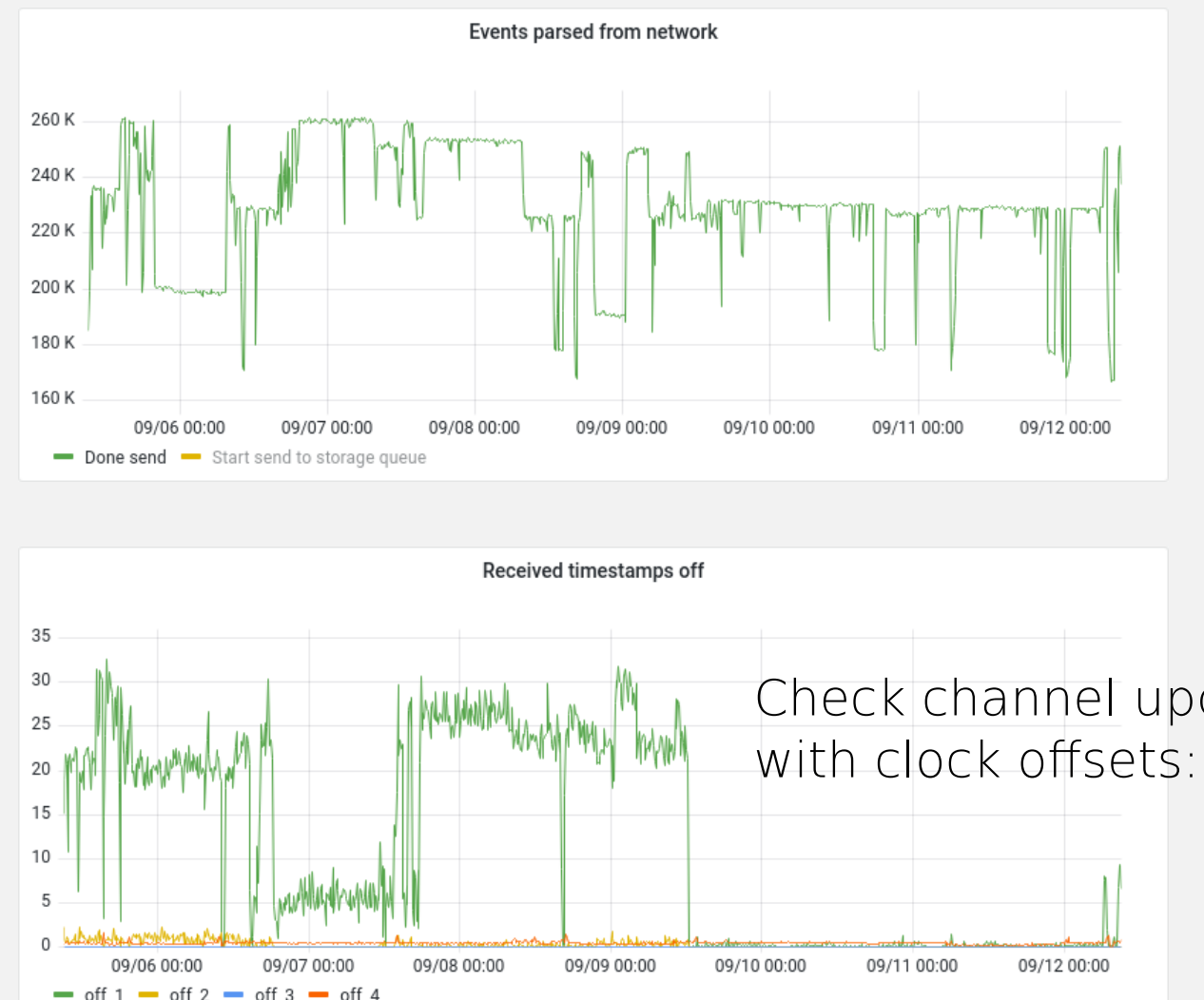
backend: scylla
api_bind: 0.0.0.0:3011
channels:
  - CHANNEL-1:A
  - CHANNEL-1:B
  - CHANNEL-2:A
search:
  - 172.26.0.255
  - 172.26.2.255
postgresql:
  host: host.psi.ch
  port: 5432
  user: USER
  pass: PASS
  name: NAME
scylla:
  hosts:
    - sf-nube-11:19042
    - sf-nube-12:19042
  keyspace: ks1
    
```

### Check ingest state of specific channel:

```

curl 'http://localhost:8023/daqingest/channel/state?name=S205'
[
  {
    "172.26.16.116:5064",
    {
      "name": "S205Y02-MKAC020:WATER-TEMP03",
      "addr": "172.26.16.116:5064",
      "channel_connected_info": "Connected",
      "scalar_type": "f64",
      "shape": [],
      "ts_created": "2022-09-02T10:14:30.214Z",
      "ts_event_last": "2022-09-09T19:55:16.667Z",
      "item_recv_ivl_ema": 3.736205,
      "interest_score": 0.26765126
    }
  }
]
    
```

### Monitor using Prometheus + Grafana:



## Data Retrieval:

- Access recorded data via http api.
- Search through list of recorded channels.
- Fetch channel updates in a time range.
- Query aggregated events for easier plotting.
- Time-weighted and unweighted binning.
- Aggregation accounts for channel connection status.

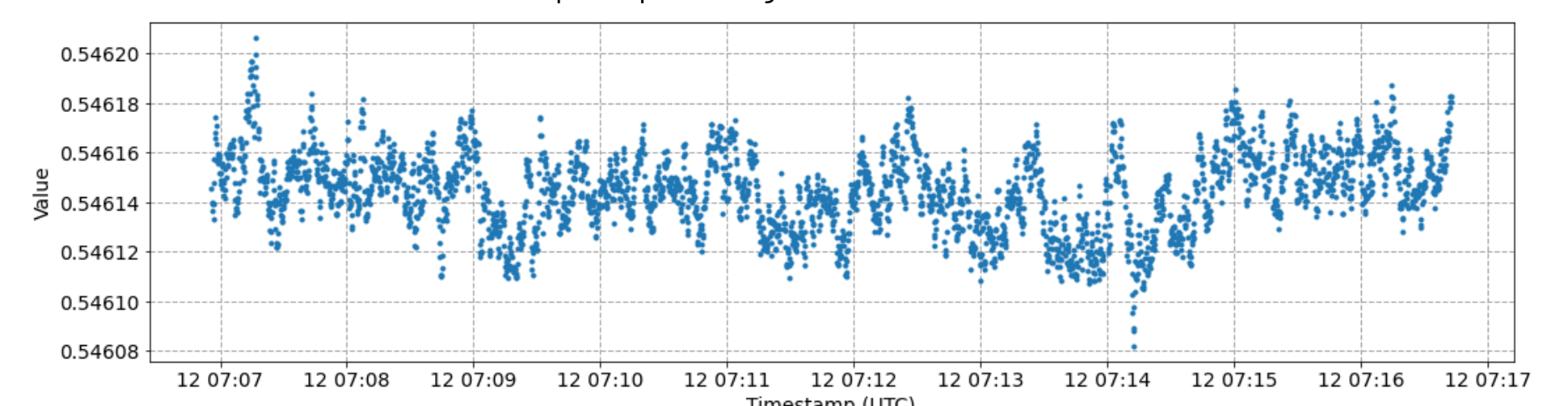
### Fetch channel events from retrieval:

```

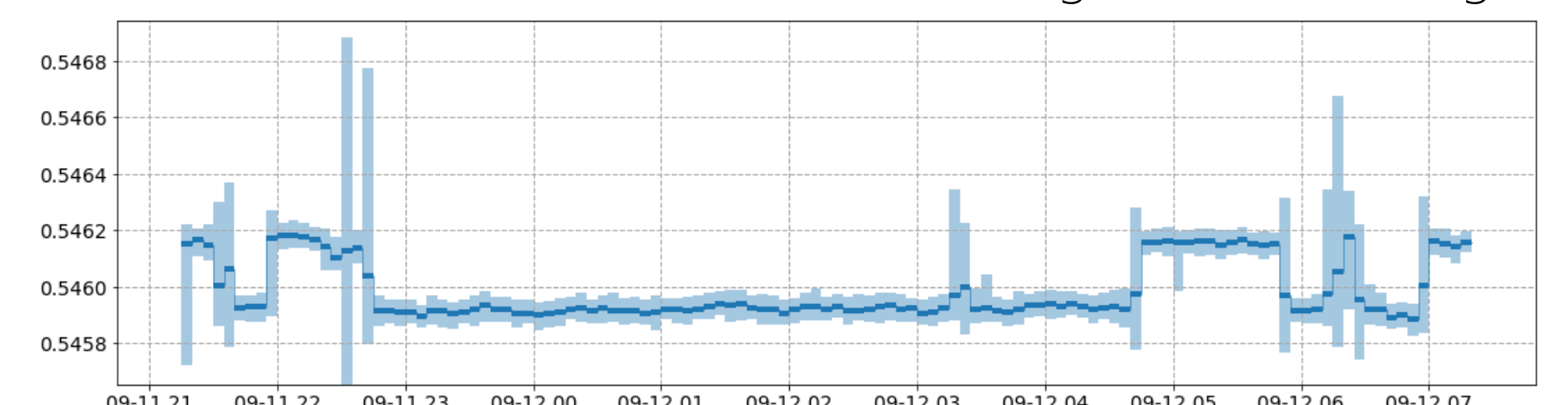
{
  "tsAnchor": 1662733800,
  "tsMs": [
    125,
    324,
    525
  ],
  "tsNs": [
    174507,
    997264,
    119155
  ],
  "values": [
    0.01536537054926157,
    0.01577814482152462,
    0.01731603406369686
  ]
}
    
```

Splitted timestamps because javascript always uses f64 which leads to precision issues.

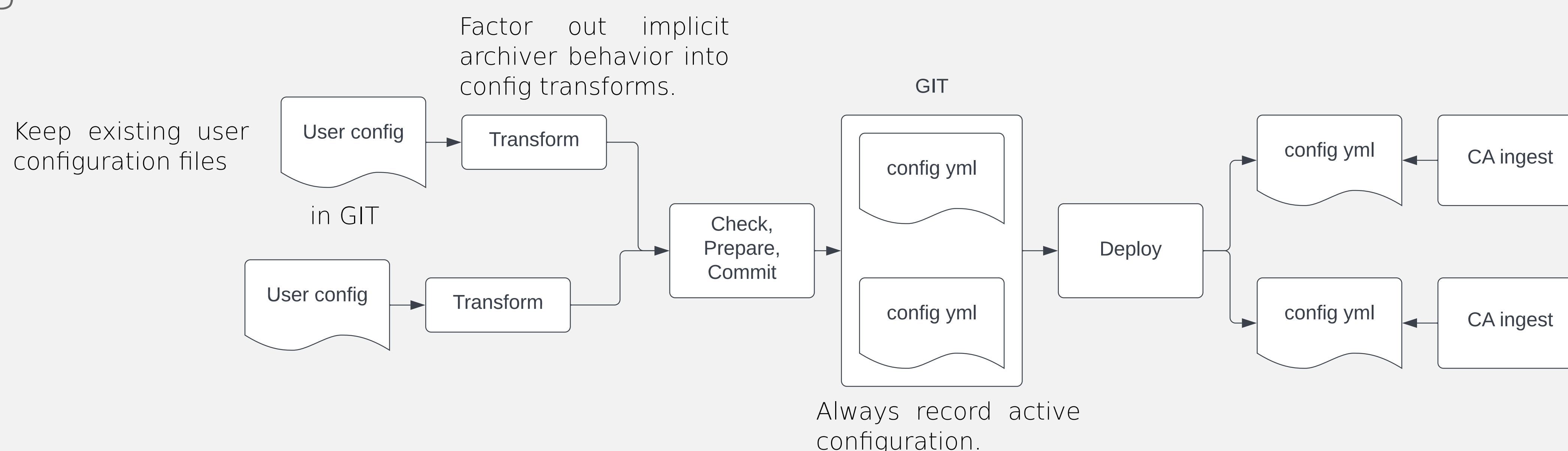
### Plain events via http api in json format from the retrieval:



### Binned data from retrieval: here e.g. Min, Max, Avg.



## Configuration



## BSREAD Ingest

- Similar to CA ingest for beam-synchronous data.
- Early testing phase.

## Outlook

- Continue testing of Channel Access ingest at SwissFEL.
- Plan for having Channel Access ingest for user testing in 2022.
- Commission production hardware.
- Finish support for BSREAD.
- Add caching for aggregated (e.g. binned) data.