

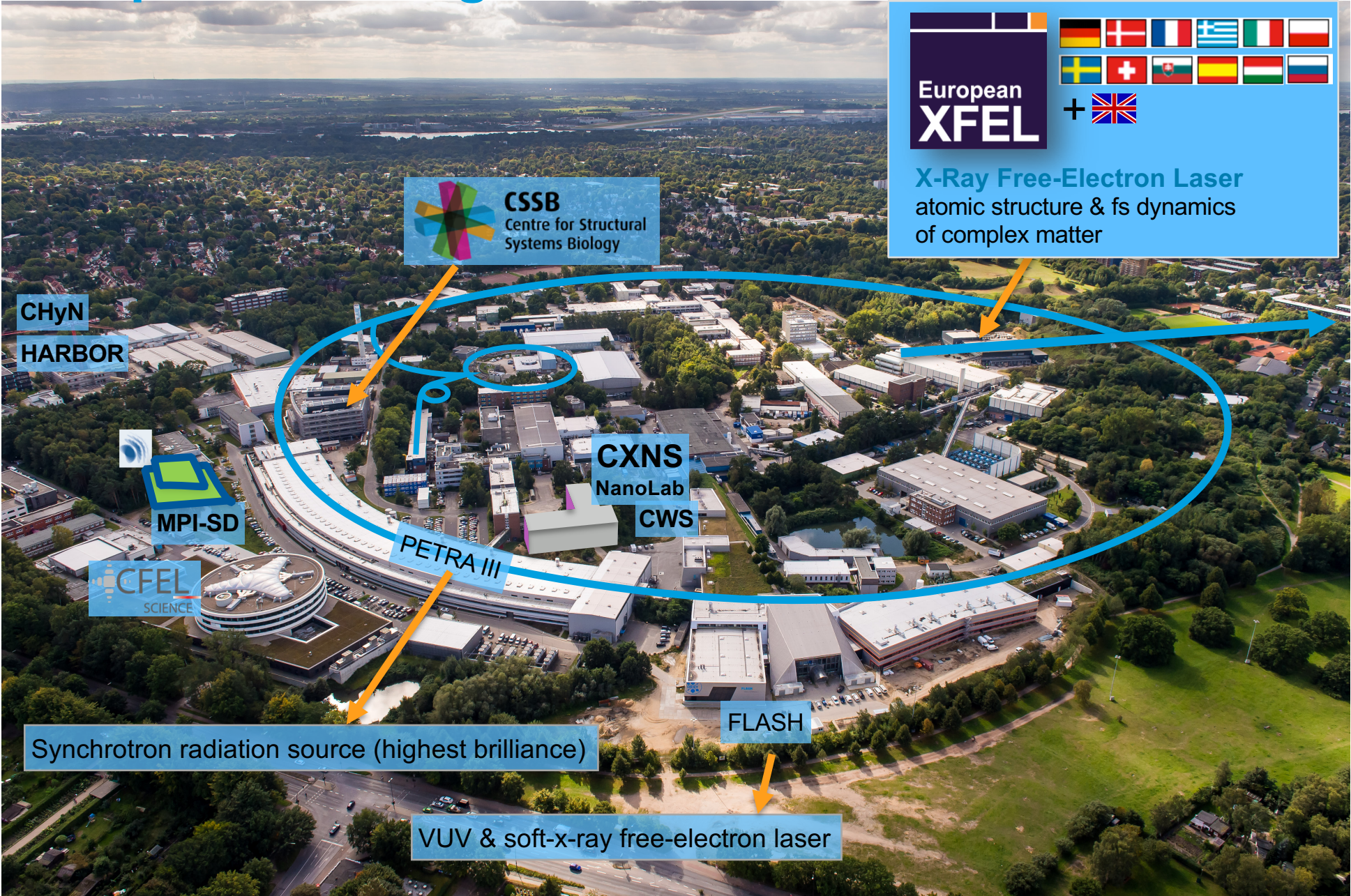
Getting ready for (true) online analysis

... and the software to liven the place up

Martin Gasthuber, Sergey Yakubov, Carsten Patzke
San Diego, March 2019



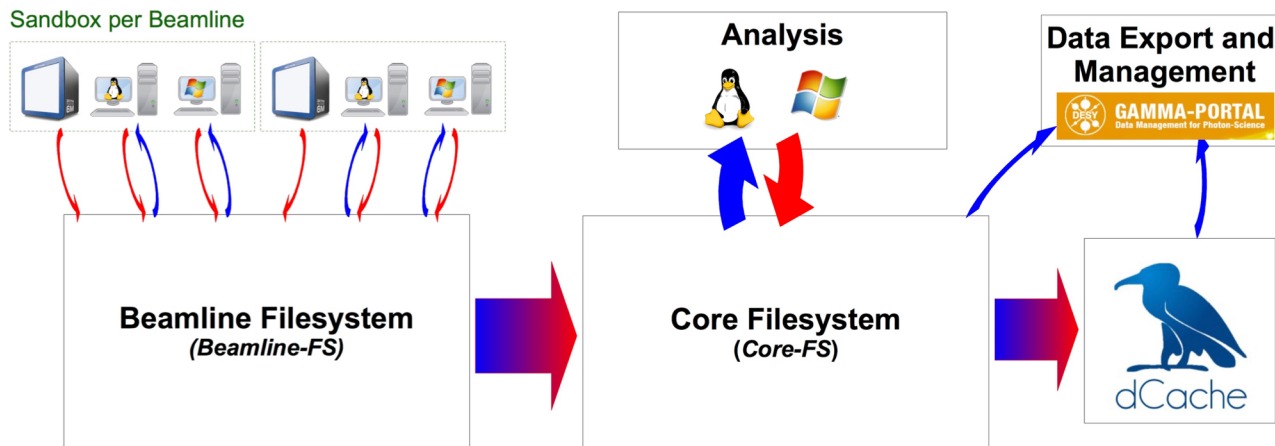
DESY Campus Hamburg – much more communities



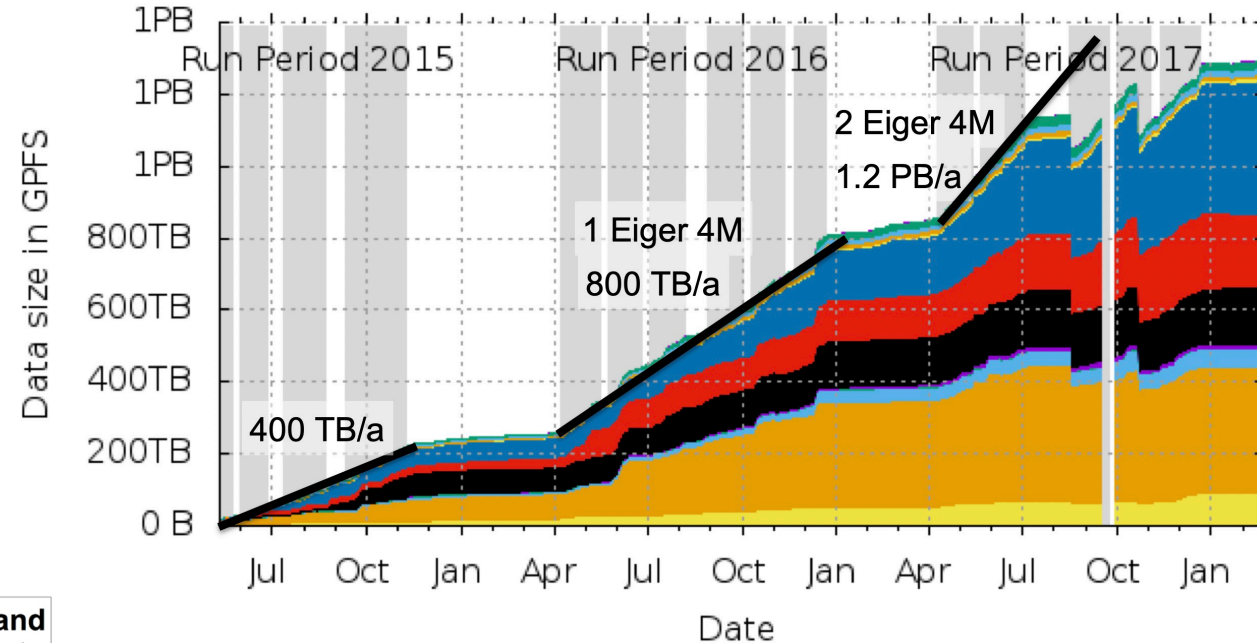
different context – photon science experiments

like CERN is for the LHC experiments

- Tier-0 for Petra III and FLASH experiments
- around 30 experimental stations (hutches)
 - mostly parallel operations
- mostly conventional ‘modus operandi’
 - Det -> GPFS -> offl. Ana -> tape
 - lot of small files (300 mill. files, 2.5PB)
 - data cooling down after 3-9 months



Storage consumption in size (per Beamline)

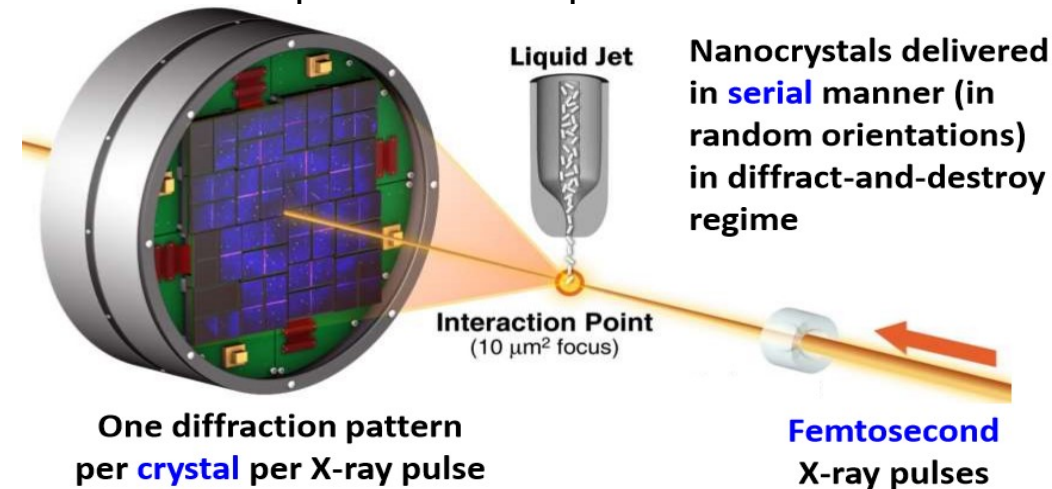


all done ?

not too long

why that mode can't be continued

- detectors growing faster than Moore's law – volume and read-out rate -> more data in less time
 - today: several 10Gb ports – going to several 100Gb ports
 - from a few KHz to MHz
- multi detector experiments – multiple data sources
- no simple experiments anymore – i.e. serial crystallography (lots of snapshot diffraction patterns to be processed individually and then merged)
 - realtime feedback for experiment adjustments – physics in the data ?
- statements from the science perspective
 - *Keeping track of which dataset is where, which is the latest/best etc. Do we have copies of everything that went into each paper?*
 - *The Datapocalypse is coming!!!*
 - *Repent and your data will be saved!*
 - *Don't and your data will be deleted!*
 - *Online monitoring:
At the forefront of data reduction!*



next big thing – Petra IV (2026)

machine upgrade and more experimental stations

> growth of data volume (next 10 years):

100 x - 1000 x increase in brilliance (PETRA IV)

5 x more beamlines taking large data sets

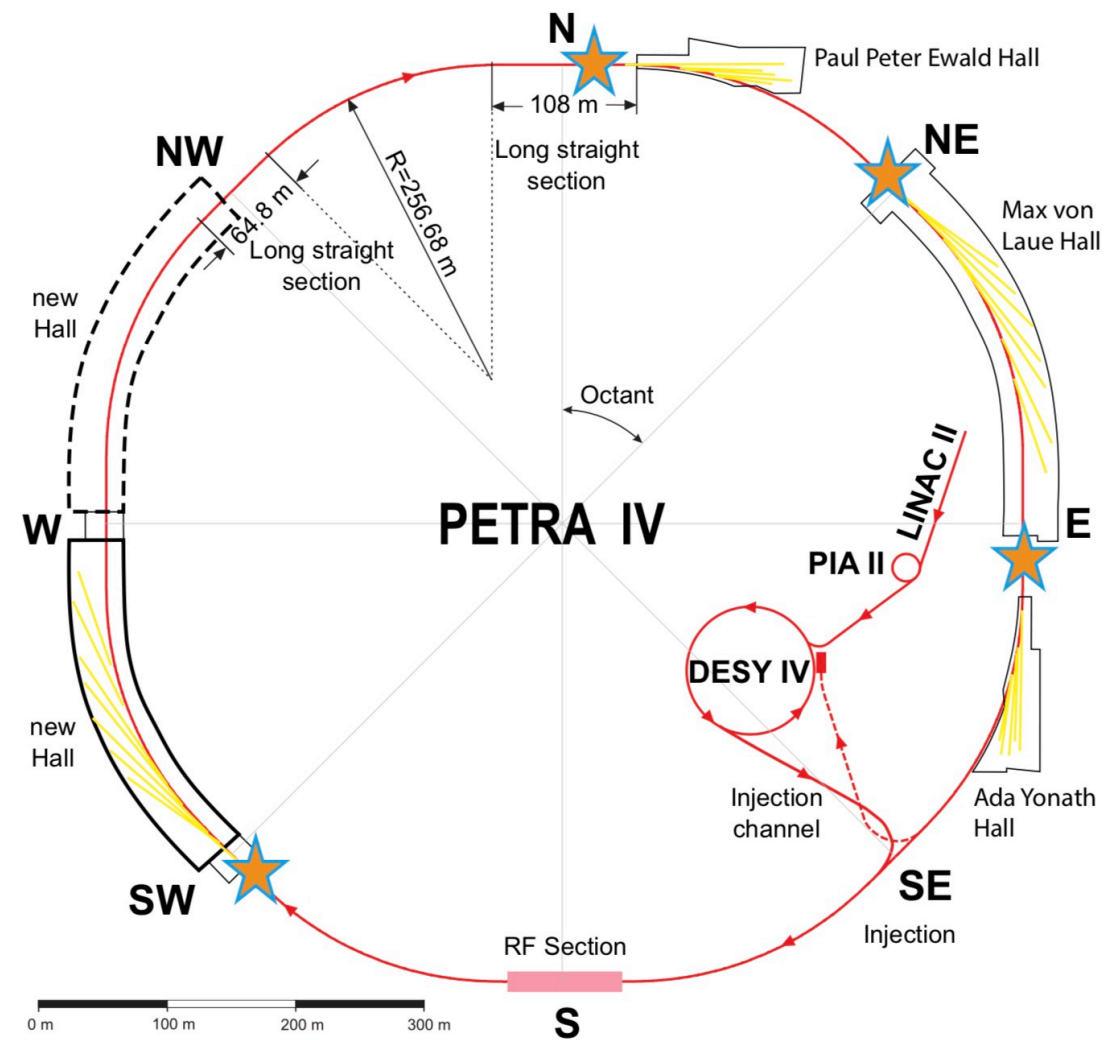
10 x automation of data acquisition

→ $10^4 - 10^5$ x increased data rate

Increasingly complicated experiments:

→ Users do not have resources or might not be experienced enough to process data by themselves


→ Data analysis becomes integral part of experiment



cited from PETRA IV scientific project coordinator

edge conditions

why that mode can't be continued

- non technical
 - multi day experiments – days instead of years – very short setup time – schedule ~10 experiments per day
 - #of groups >300, #of group members 1-30, #of computing experts ~0 (exceptions are rare ;-)
 - the system outlined - will not be configured 5min before beam arrives, not run by side glances, not further developed as a weekend task
 - experts required in the experiment group - from day one (planning) until paper published
 - building and keeping trust – equivalent responsibility to make good physics
 - accept computing nerds as full members of the experiment team
 - expect - nobody will pay for dedicated HW/SW per beamline/experiment
 - leading to
 - **1. data reduction (before storing)**
 - **2. do not store raw data at all (eventually ;-)**
 - **3. more services around workflow, automation, reproducibility, bookkeeping, ... to make scaling happened**
-  **be ready to fully support (true) online data analysis**

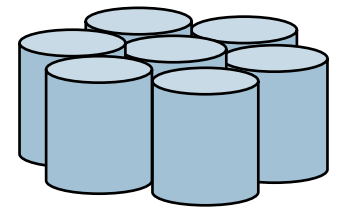
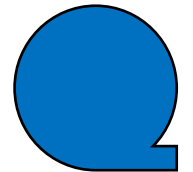
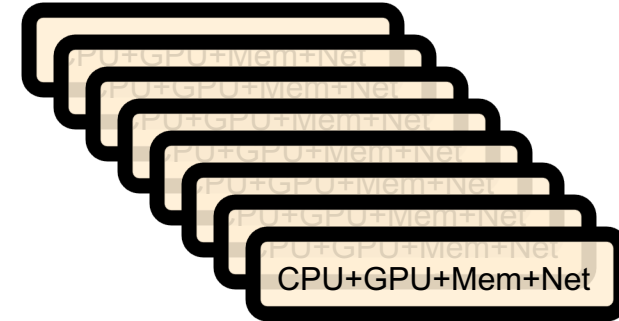
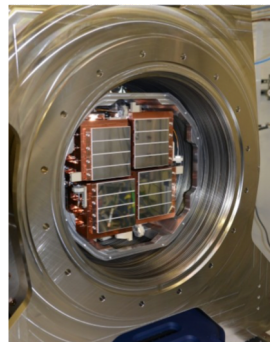
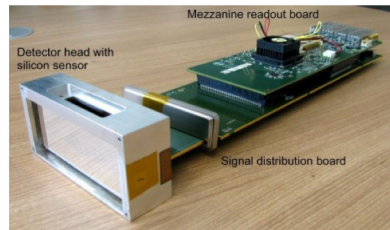
conditions – initial numbers we will address

per active experiment

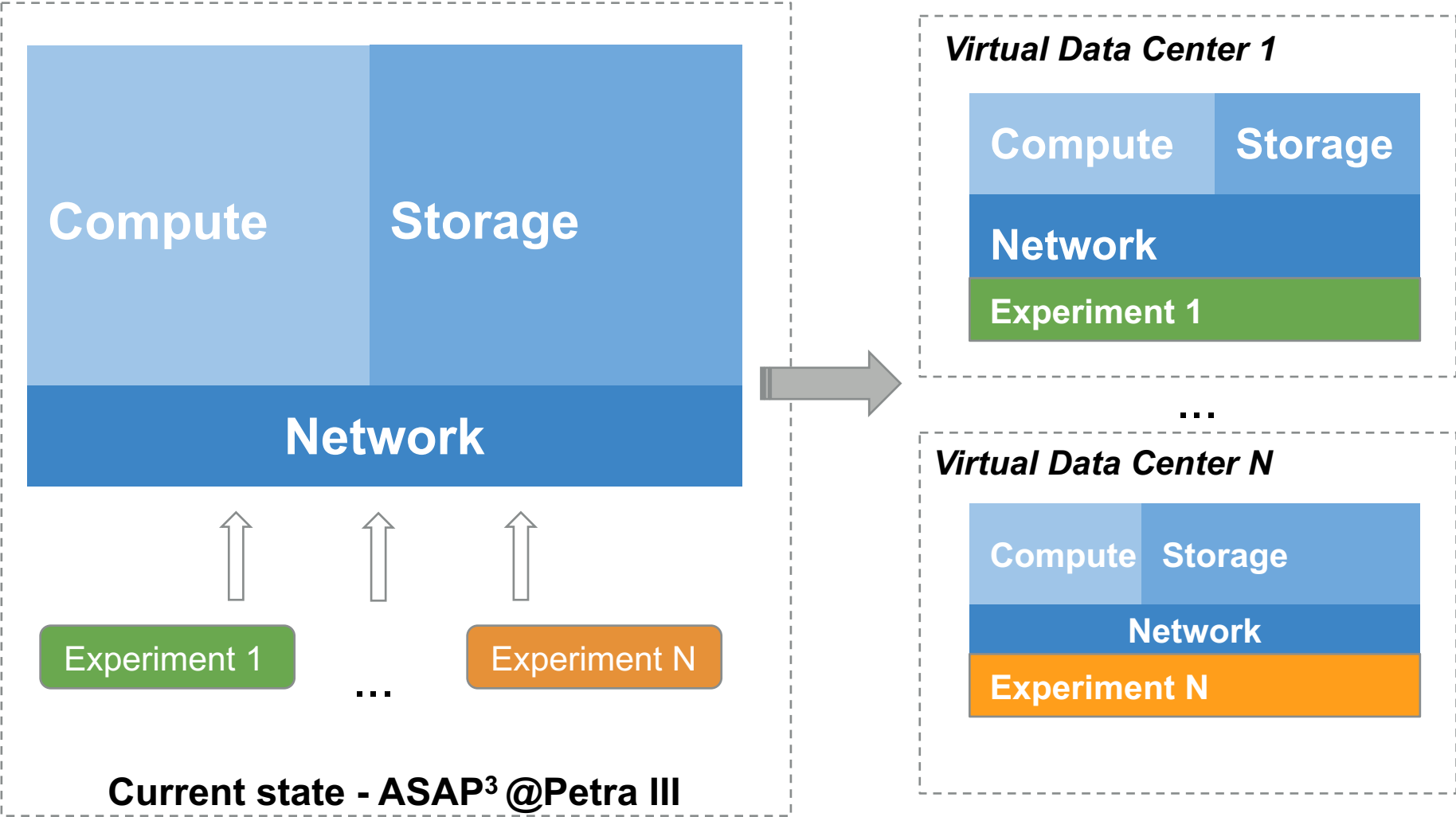
- ~50GB/sec aggregates ingest rate of all sources (10 - 30min average) – daily averages to ~5 GB/sec
- ~1PB of 'hot' data to work on (eventually store much less ;-)
- network - ~100GB/sec with deterministic latencies <1 μ s
- 1k cores (CPU+GPU) with 300 k cpu-hours of computing work
- archive data for 10 yrs – with 1 week delay after creation (second copy asap)

- others - obvious
 - allow 'commissioning' ahead of beam hitting the target
 - keep data in flight mode as long as possible - defer storage access as long as possible

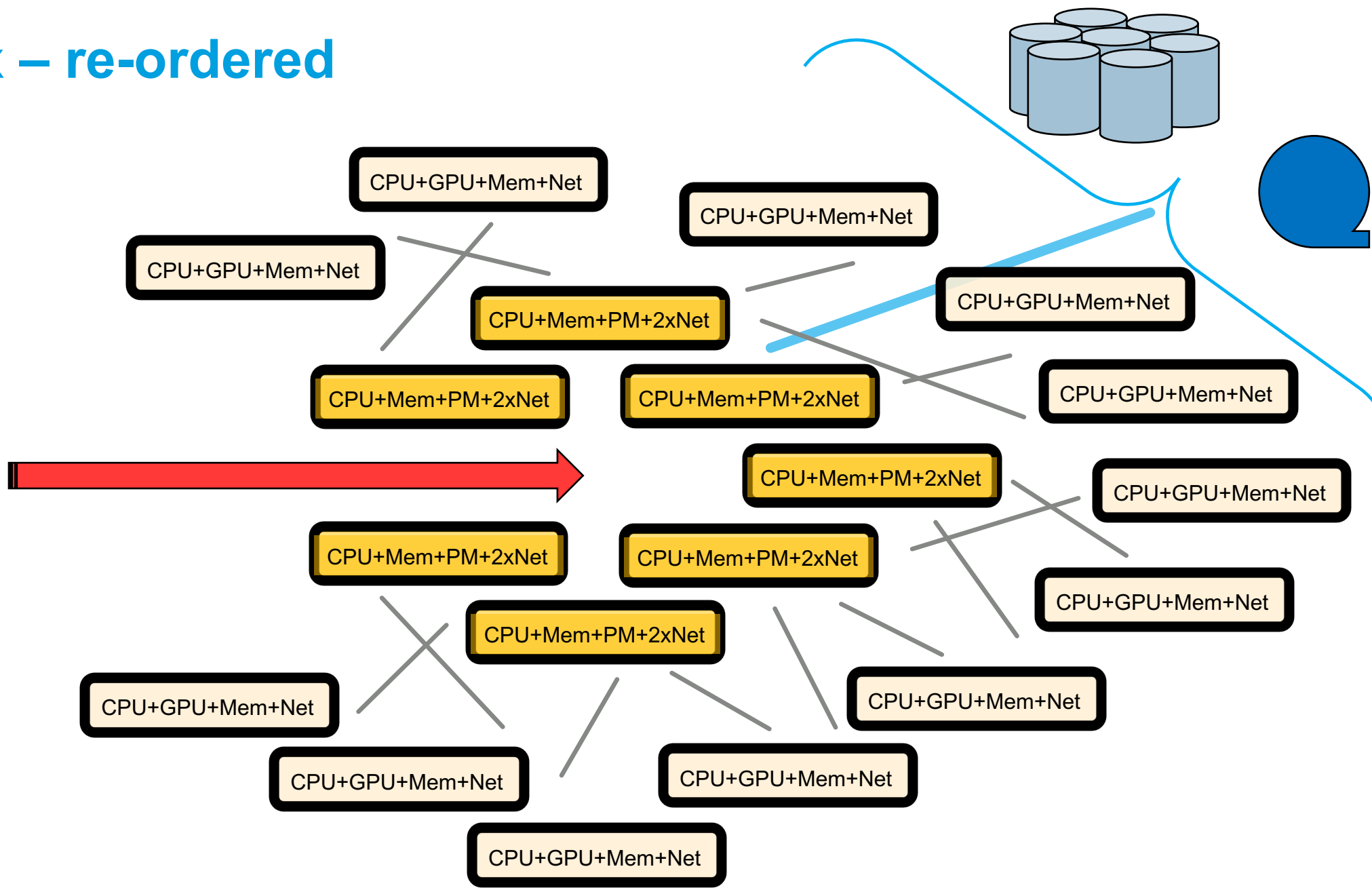
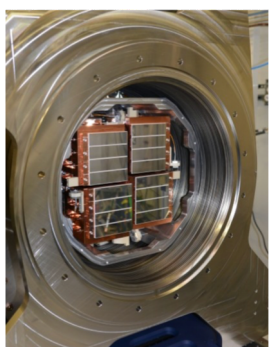
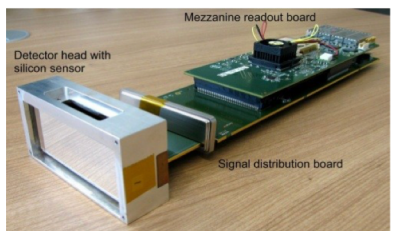
in the toolbox... all pooled, beside detector



different mixture of ingredients

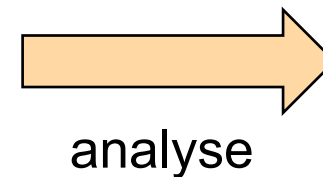
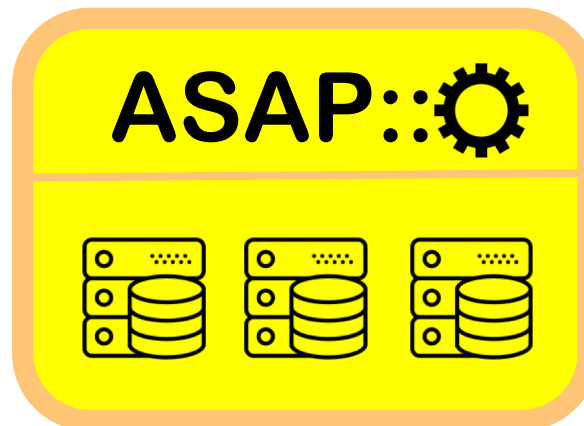
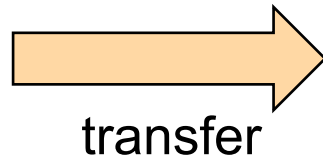


the toolbox – re-ordered



ASAP::

- A middleware for high-performance next-generation detector data analysis
 - Takes care of the “first mile” between the experimental hall and the compute center (high-performance data transfer)
 - Provides API for data analysis synchronous (online) and asynchronous (offline) to data taking
- Basic characteristics
 - Scalable (N detectors, K network links, L service nodes, M analysis nodes)
 - Highly available (services in Docker containers managed by Nomad/Consul)
 - Efficient (C++, multi-threading, RDMA, ...)
 - Provides user friendly API interfaces (C/C++, Python, REST API)
 - Runs on Linux/Windows/...



ASAP::O – A Crystallography Experiment with PILATUS 6M Detector

Detector generates a new image – saves a file to RAM disk

ASAP::O monitors RAM disk and detects the new file

ASAP::O sends the file to the data center

```
In [2]: %matplotlib inline

In [3]: import asap_worker
import numpy as np
import matplotlib.pyplot as plt
import tempfile
import cbf

broker, err = asap_worker.create_server_broker("asapo-server:8400",

In [4]: data, meta, err = broker.get_last(meta_only=False)
plt.imshow(data, cmap='gnuplot', vmax=500)
plt.show()
```

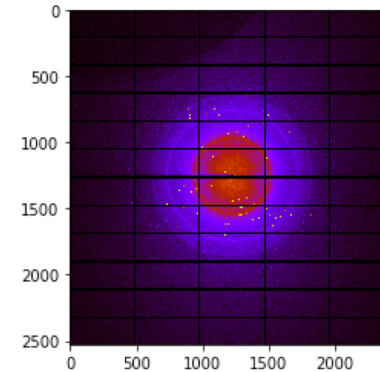


Image appears in a Jupyter Notebook



PETRA III Experiment Hall

A web browser somewhere in the world

ASAP::O - Current State and Future Plans

- Done
 - Basic functionality (data transfer, online/offline analysis, parallel processing)
 - Scalability/high availability/logging/performance monitoring/CI/CD
 - C++/Python API – basic functions
 - Detector plugin - file monitoring
- In Progress
 - RDMA support
 - Virtual detectors
- ToDo
 - Other detectors plugins (http, zeroMQ, ...)
 - Work with metadata – inject, use it during analysis (tagging data at source, filter data, custom queries, ...)
 - multi-stage processing/pipelining (input processed data to ASAP::O, connectors to Apache Spark/Storm,...)

further To-Do's

- scientific payload slipped in container – single/multithreaded, MPI based
 - templated (by us) container to start developing process (tuned for network/storage/co-processor)
 - served as contract – scientists does its 'wild-west' software R&D, we run it efficiently on 10k machines
- carving out a DAQ (sub)network – guaranteed performance, latency and jitter
- same for storage (some techniques already known and tested)
 - control/manage the 'noisy neighbor' – network & storage
- further services – similar importance
 - automated workflows - covering the complete life-cycle (from the cradle to the grave ;-)
 - automated bookkeeping – derived data (when, who, code, based on)

