Mitglied der Helmholtz-Gemeinschaft

JÜLICH
FORSCHUNGSZENTRUM

Big data challenges arising from future experiments

D. Pleiter | PASC | 09 June 2016

Outline

- Relevant hardware technology trends
- Requirements analysis for use cases of extreme scale data challenges
  - Focus on observatories/physics experiments
  - Use cases with need for HPC resources
- Implementation examples exploiting new architectures and technologies

Relevant Hardware Technology Trends

Compute acceleration

Typical features

- Relatively low clock frequency
- Extremely high level of parallelism or deep compute pipelines
- Relatively large memory bandwidth
  - Not necessarily large memory bandwidth vs. operation throughput
- Moderately fast connection to processor

Most popular compute accelerator = GPU

- Example:
  NVIDIA Tesla P100

FP64 / cycle	3584
Clock rate [GHz]	1.33 … 1.48
Memory bandwidth [GByte/s]	720
Memory capacity [GByte]	16
Host interface bandwidth [GByte/s]	20 … 80

Compute acceleration (cont.)

Other compute accelerators

- FPGAs
  - Already widely used in physics experiments
- Xeon Phi

New developments

- Stand-alone „accelerator“ devices
- Tighter integration of processor and accelerator
  - Example: POWER8
    - CAPI interface
    - NVLink interface

Hierarchical data stores

Technology driven development

- Capacity optimized technologies: HDD, tape
  - Large capacity for money
- Performance optimized technologies: NVRAM
  - Large capability for money

Need for capacity + performance → hierarchy

- Upper tier: higher performance, lower capacity
- Lower tier: Larger capacity, lower performance

Main challenge: usability of hierarchical storage architecture

Hierarchical data stores (cont.)

EC funded FET project

- 10 partners
- 4 countries

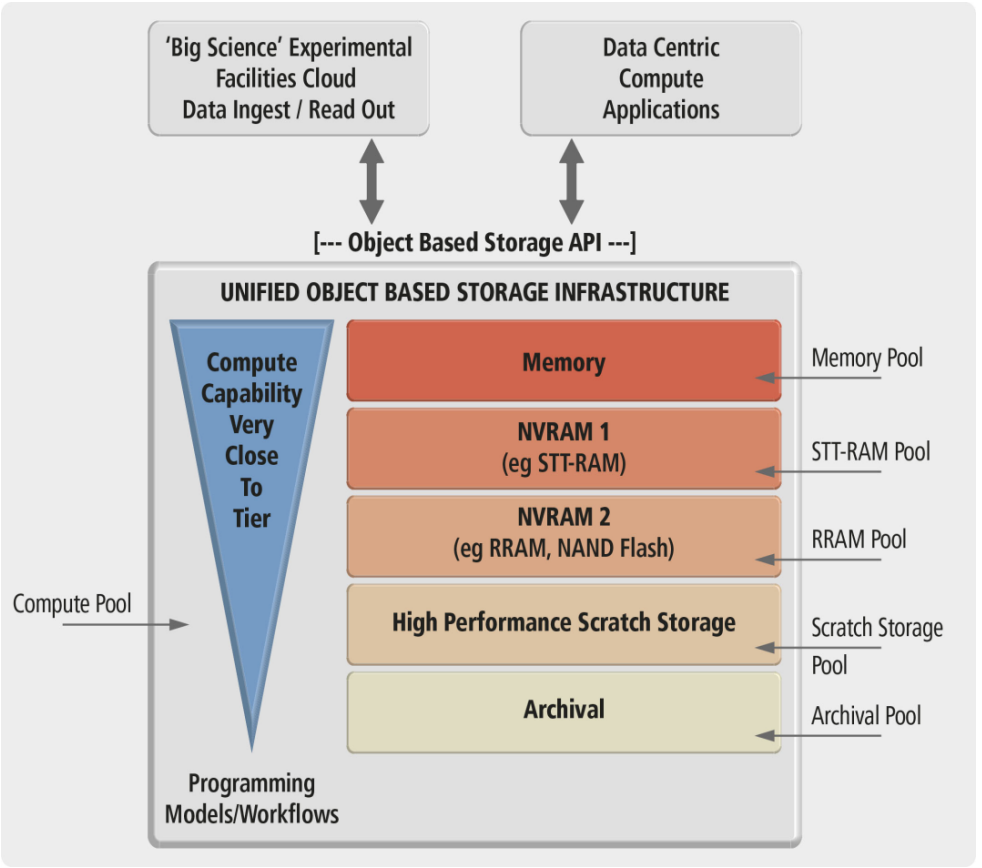
Hierarchy-aware object store

- Aim for data-centric approach

Consider up to 5 tiers

- Memory to archival tier

In-storage processing capabilities



Requirements Analysis

Application requirements analysis methodology

Input/output data volumes or rates

- Data that needs to be moved per time unit
- Space for buffering data may allow to reduce peak requirements

Computational intensity and intrinsic parallelism

- Processing time Δt_{ops} vs. data communication time Δt_{com}
  - Assume perfect overlap → aim for both to be similar
  - Balance condition

$$\frac{I_{ops}}{I_{com}} \approx \frac{B_{ops}}{B_{com}}$$

- Alternative, I/O intensity becomes more relevant, i.e. fraction of time spent in I/O: $\Delta t_{io} / \Delta t$

Analysis methodology (cont.)

Retention time analysis of data objects

- Data objects produced and/or consumed by workflow
- Classification according to retention time
  - Reference time scale =
    typical duration of asingle processing step

Classes

- **Permanent**: Data objects outliving the system on which it is processed
- **Short-term**: Data objects with lifetime exceeding reference time scale without being permanent
- **Transient**: Data discarded after processing step

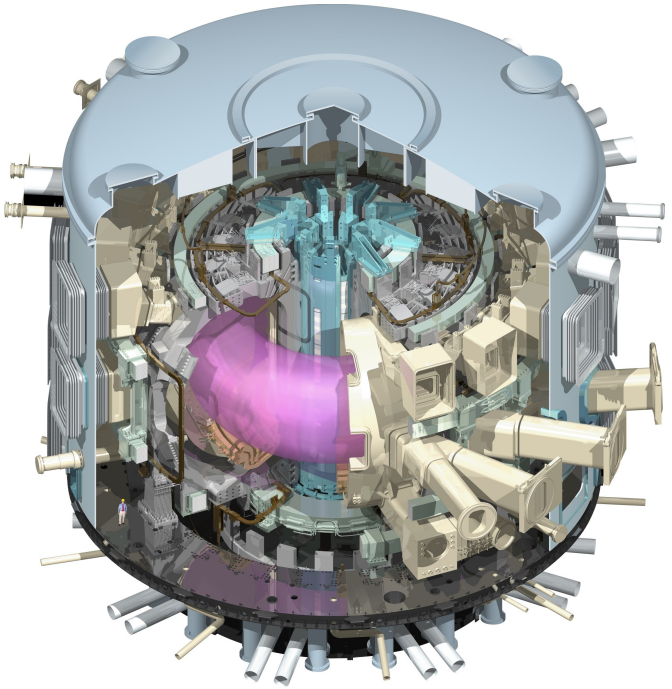
Natural mapping onto hierarchical storage architecture

- Capacity optimized tier for permanent data objects
- Performance optimized tier for short-term/transient data objects

Fusion experiments

Various experiments running or in preparation, e.g.

- Joint European Torus (JET) at CCFE
- International Thermonuclear Experimental Reactor (ITER) at Cadarache



[Shaun de Witt, 2016]

Considered application: Monitoring of magnetics signal data at JET

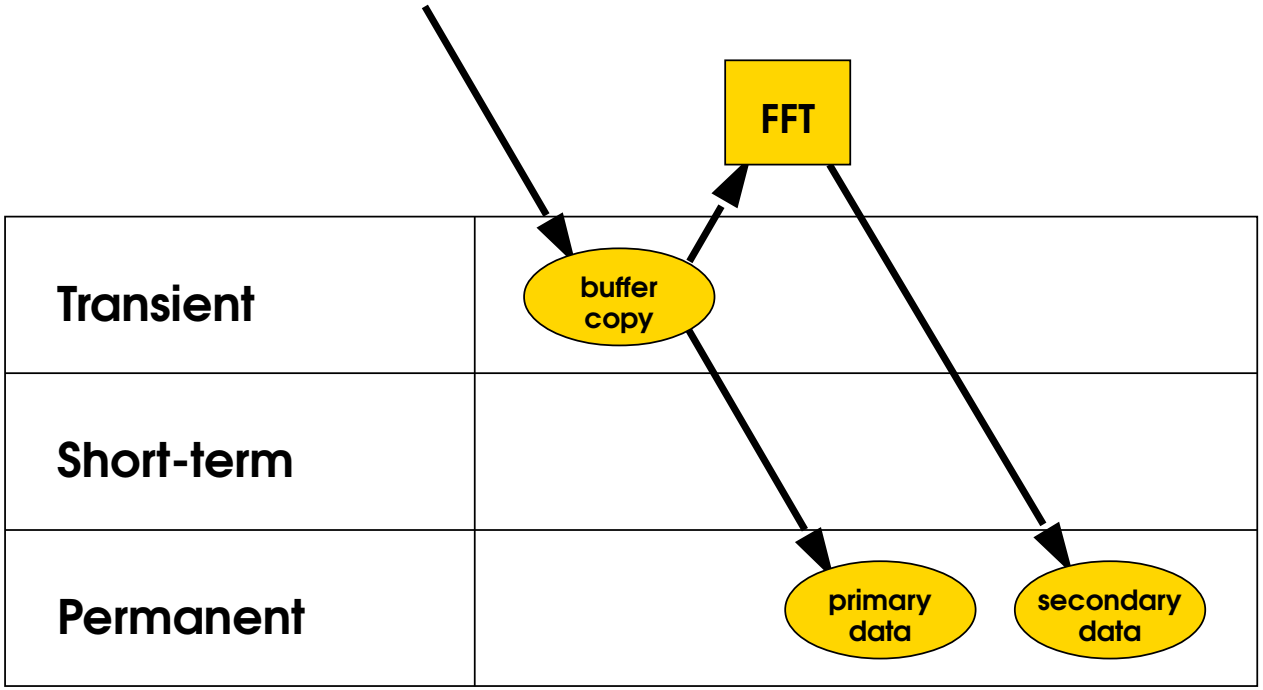
- Near-real time monitoring required

Performance characteristics

- Non-continuous data taking
- Input data rate today about N_{src} * 40 MByte/s (N_{src} = 200)
  - Sources can be processed independently
- Mainly FFT, i.e. moderate computational intensity I_{fp}/I_{com}

Fusion experiments (cont.)

Retention time analysis



Satellite data processing

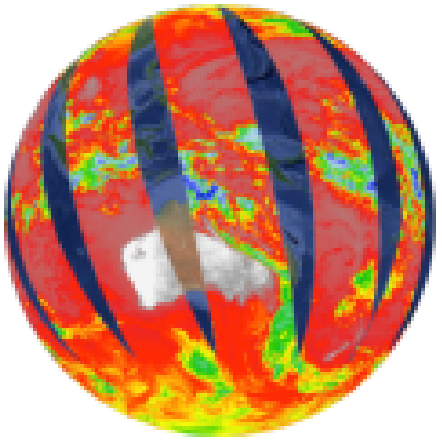
Jülich Rapid Spectral Simulation Code (JURASSIC)

- Fast radiative transfer model
- Use for retrieval of atmospheric data
  - Atmospheric InfraRed Sounder (AIRS) measurements

Performance characteristics

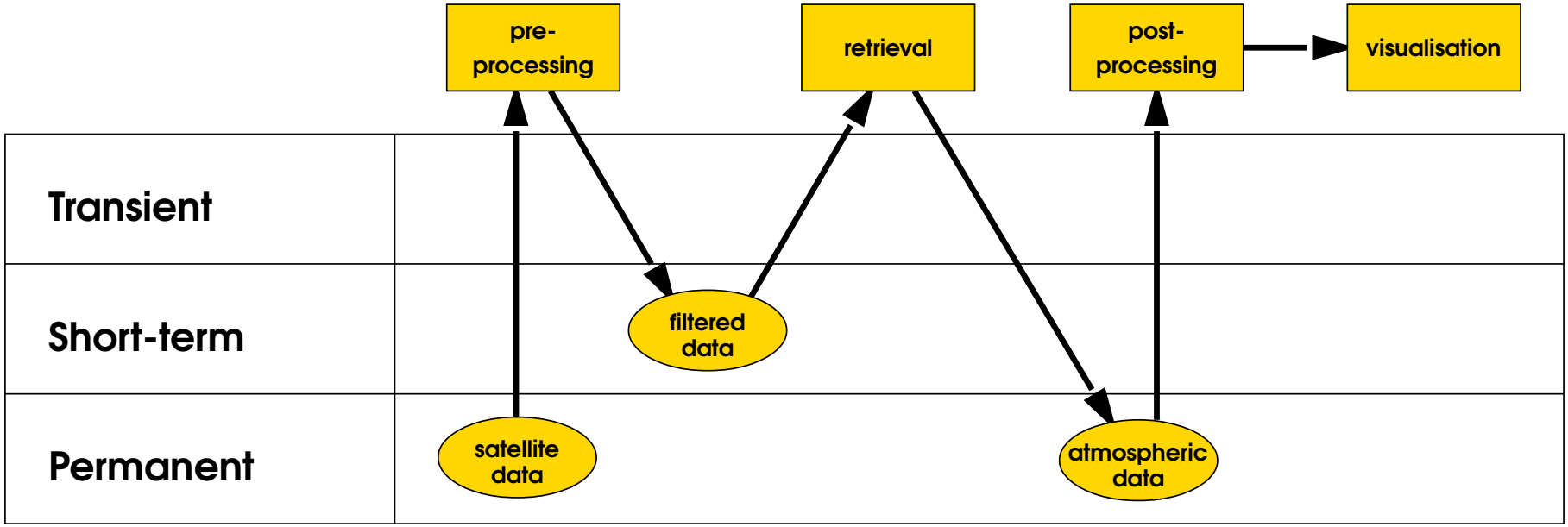
- Forward modelling approach
  - Loop over different realisations of temperature functions
  - Loop over different rays

  → massive parallelism
- Accumulated data-sets about 100 TByte
  - Data-sets comprising many objects that are processed independently of each other



Satellite data processing (cont.)

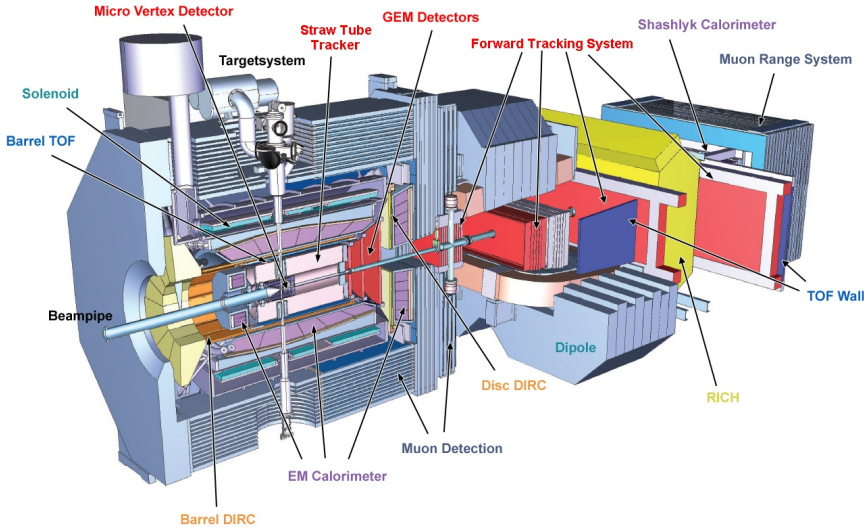
Retention time analysis



High-energy physics

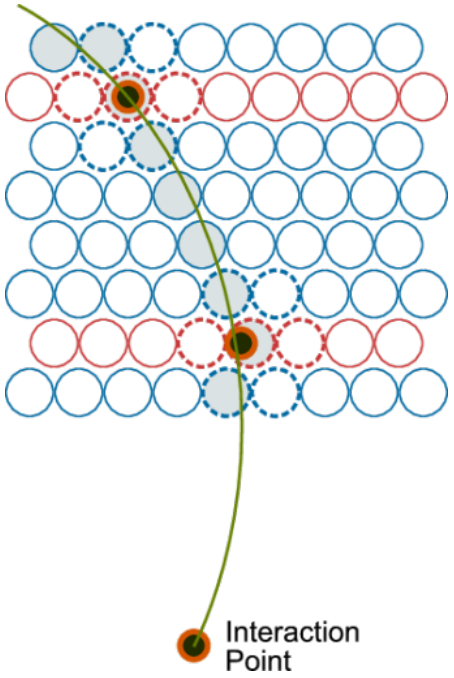
PANDA detector

- Planned as part of FAIR facility at GSI (Germany)
- Aim for software-based triggerless online reconstruction



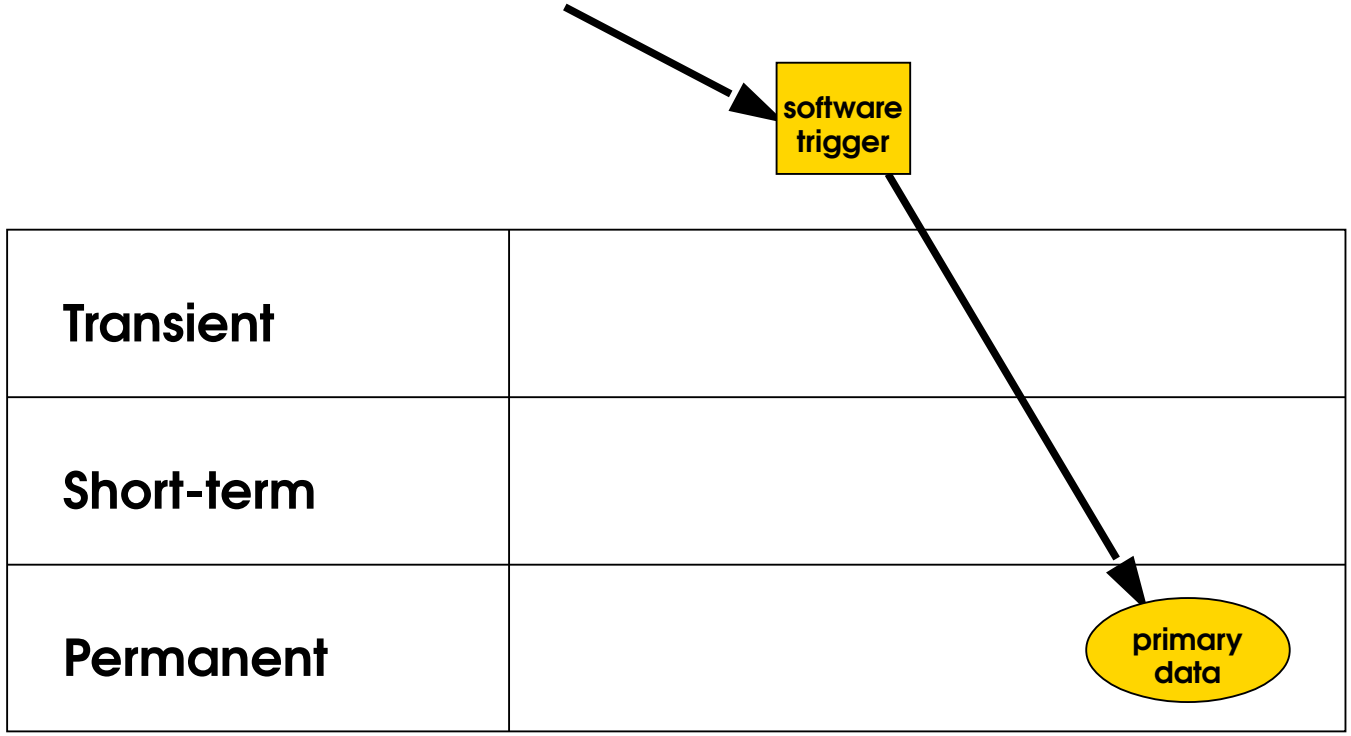
Performance characteristics

- Input data rate 200 GByte/s
- Need for real-time data processing
- Number of operations depends on algorithm that is used, e.g.
  - Hough transformation
  - Triplet finder
- Events sufficiently separated in time can be processed indepedently



High-energy physics (cont.)

Retention time analysis



Astrophysics

Square Kilometre Array (SKA)

- Next-generation radio-telescope that will be built in South Africa and Western Australia
- Multi-stage processing pipeline
  - Central Signal Processing (CSP)
  - Science Data Processor (SDP)
  - Regional Science Centres

Extreme increase in data rates

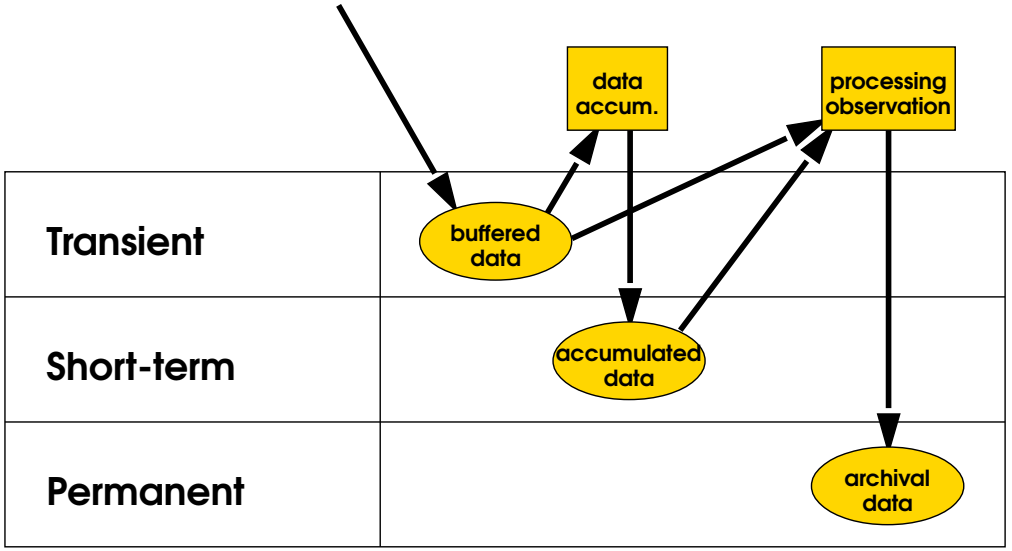
- Selected tentative SKA1 SDP characteristics

	SKA1 mid/SA	SKA1 low/AU
SDP input [Tbit/s]	5.2	4.7
Compute capability [PFlop/s]	24	5.7

- Comparison LOFAR (before correlator!): 3 Gbit/s per station, ~64 stations
- Compute intensive processing of many independent data streams

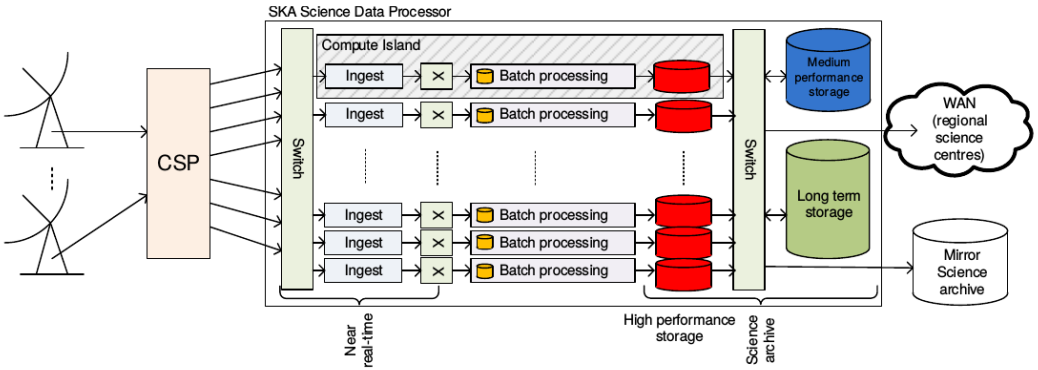
Astrophysics (cont.)

Retention time analysis



Compute island concept

- Includes „fast buffers“



[C. Broekema et al., 2015]

Implementation Examples

Satellite data processing: JURASSIC

Idea: Allow storage to manage data object staging

- Cache = Storage holding temporary data object copies
  - Data object may be evicted if necessary
- Semi-persistent = Cache content may survive job boundaries → short-term retention time
- Active = Support pre- and post-processing in cache

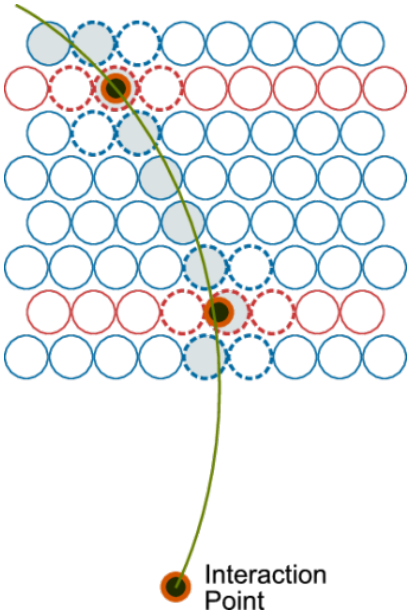
Work-flow

- Cache stages data objects depending on availability
- Compute nodes are informed about staged objects
- Optional mechanism for result file migration

Challenges

- Pre-fetch strategy
- Load balancing

PANDA triggerless tracking

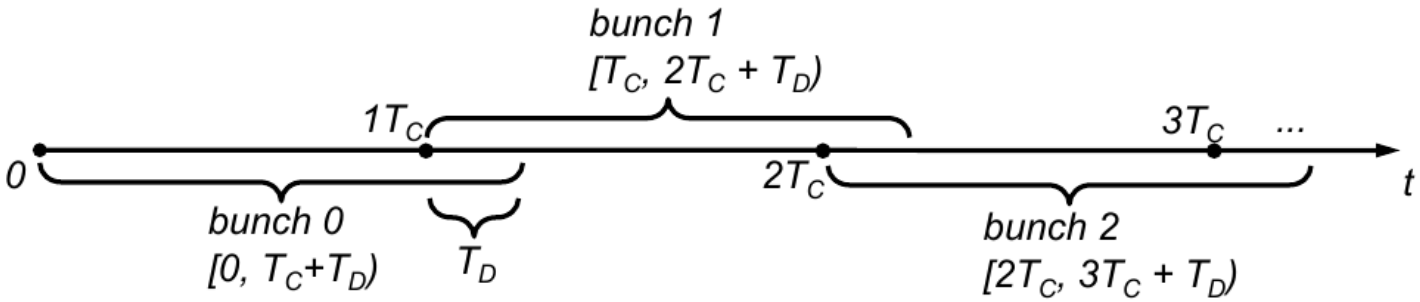


Triplet Finder Algorithm

- Exploits features of Straw Tube Tracker
- Within bunch of events it detects hits in adjacent straws and construct triplet
- Compute particle trajectory from triplets

Complexity challenge

- Algorithm scales $O(N^2)$ with N … number of processed hits
  - For GPU to be efficient, N should be large
- Solution: split data set in smaller bunches
  - T_D … maximum drift speed



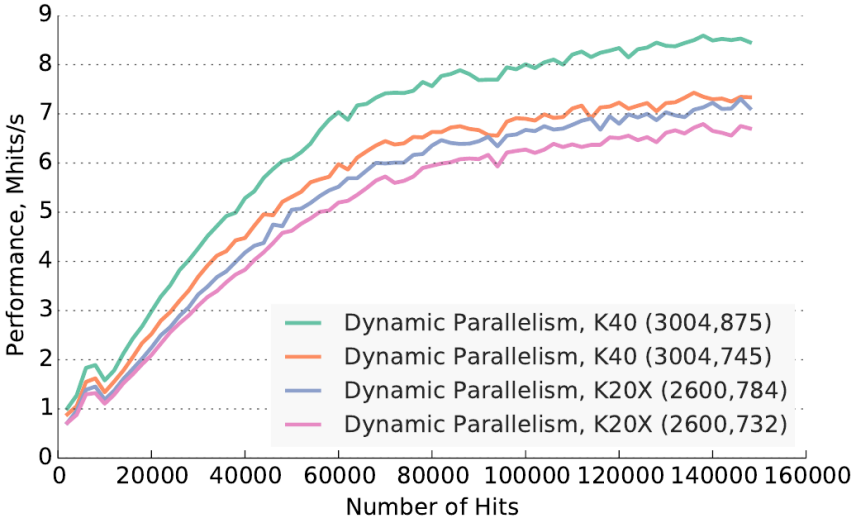
PANDA triggerless tracking (cont.)

Different implementations for processing multiple bunches

- Exploit host streams
- Use dynamic parallelism ← best approach

Final performance results

- On K20x it is able to sustain an event processing rate of 6.19 MHit/s
- PANDA is expected to run at most at 1600 MHit/s
- Using 260 K20x would allow to sustain PANDA data rate



Summary & Conclusions

Summary and conclusions

Various large-scale experiments will challenge the ability to process extreme scale data volumes using HPC technologies

- Satellite data processing
- Future fusion and high-energy physics experiments
- Future radio telescopes

Leverage future hierarchical storage architectures

- Retention analysis provides guidance on how to map data objects to storage tier

Opportunities to exploit compute accelerators

- In particular interesting when data streams are too large for being stored