EXASCALING AND FEDERATION: USING BRAIN RESEARCH AS SCIENCE DRIVER

Dirk Pleiter | SOS23, Nashville | 28.03.2019
Overview

- Science domain: Human Brain Project
- Annotated use case analysis
- Science cases and use case scenarios
- Reflection on session chair’s questions
Human Brain Project and Fenix

Human Brain Project

- Overall research challenge
  - Create an understanding of brain at different spatial and temporal scales
  - Help to address dysfunctions of the brain causing mental diseases including Alzheimer

- Specific research topics
  - Create high-resolution atlases of the human brain
  - Create realistic models of the human brain
  - Analysis of patient data

Fenix and the ICEI project

- Consortium of BSC, CEA, CINECA, CSCS, JSC
  - Aim for harmonising and federation of services

- Services provided through ICEI
  - Computing services
    - Interactive Computing Services
    - Scalable Computing Services
    - VM Services
  - Data services
    - Active Data Repositories
    - Federated Archival Data Repositories
    - Data Mover, Location and Transport Services
  - Federation level services
    - Authentication and Authorisation Services
    - User and Resource Management Services (FURMS)
Science Cases and Use Case Scenarios

Brain modelling
- Data-driven modelling of the brain

Brain atlas development
- Histological data analysis, image registration

Sharing of data assets
- Allow researchers to access data assets from various data sources
  - Simulation data
  - Experimental data
Annotated Use Case Diagrams

Abstract model components
- Data ingest
- Data repository
- Processing station
- Data transport

Use case/workload specific annotation of components
- Data transport
  - Maximum/average required bandwidth
  - Interface requirements
- Data repository
  - Maximum capacity requirements
- Processing station
  - Data processing hardware architecture requirements
  - Required software stacks
Storage Hardware Architectures in Modularised Setup at JSC

Multiple compute modules
- Different capabilities
- Different user communities

Diversification of storage tiers
- High Performance Storage Tier
- Large Capacity Storage Tier
- eXtended Capacity Storage Tier
- Tape archive

---

**HPST**
**LCST (scratch)**
**LCST (data)**
**XCST**
**Archive**

**capacity**
**bandwidth**
**retention time**

---

**JURECA**
**Interactive Computing Nodes/Cloud (2019)**
**JUWELS**
**High Performance Storage Tier**
**Active data repository (2019)**

**Internet PRACE**
**100 GE Ethernet backbone**

**JUST5: LCST**
**Active data repository**

**JUST5: XCST**
**Archival data repository**

**Data Mover Cluster (TBD)**
**OpenStack Cluster**
Data Sharing and Federated Data Stores: Requirements

Integration in AAI + consistent access control
- Exceeding local control domains → challenge of agreeing on common policies
- Different storage technologies do not provide compatible access control mechanisms

Storage accessible from outside the data be centre
- Need to move away from silo approach

Web-based clients
- No proprietary clients, easy to deploy by any user

Persistent references
- Keep data findable
Approach in Fenix

Active Data Repositories
• Data repository localized close to computational or visualization resources optimised for performance
• Used for storing temporary slave replica of large data objects
• Typical implementation: PFS with POSIX API

Archival Data Repositories
• Data store optimised for capacity, reliability and availability
• Used for storing large data products permanently that cannot be easily regenerated
• Implementation: Object store with SWIFT interface

Data Mover Service
• Asynchronous data transfer between active and archival data repositories
• Optionally controlled by resource manager
Simulation and Post-processing: Use Case Scenario

Use case scenario: Interactive supercomputing

• Scenarios
  - Online monitoring
  - Interrupt simulation, analyse data, resume simulation

• Challenges
  - Provide interactive compute services
  - Generic data transport layer

Use case scenario: Sharing of simulation data

• Access to simulation data to external users through web-based visualisation services

• Challenge: Make data accessible externally
Limitations of today’s architectures

• Lacking data and memory (storage) awareness
• Few abstractions exist that capture data semantics of applications, so reasoning about data movement and memory in software is impossible

Maestro approach

• Develop new data-aware abstractions
• Design a middleware and libraries that allow for modelling of memory/storage hierarchy, reason about data transport, manage data object movement etc.

Key abstractions

• Core Data Objects + Core Data Object Pool
• Producer-consumer API: declare, give, take, dispose
Data Analytics/ML: Science Case

Research goal
• Accurate, highly detailed computer model of the human brain based on histological input

Approach
• Create high-resolution 2-dimensional brain section images (1μm resolution)
• Reconstruct 3-dimensional models from these images
• Reconstruct nerve fibers in histological brain sections
• Automatised analysis of high-resolution images
Data Analytics/ML: Requirements Analysis

Data production
• Data acquisition using 8-9 microscopes at ~15 MByte/s → ~10 TByte/day
• Archival data increase 1-2 PByte per year

Data access speed requirements
• Overlapping learning and I/O → \( \Delta t_{IO} \leq \Delta t_{learn} \)
• Assume \( \Delta t_{IO} = \frac{I_{IO}}{B_{IO}} \) → \( B_{IO} \geq \frac{I_{IO}}{\Delta t_{learn}} \)
• For single NVIDIA P100 we determined a minimal required bandwidth \( B_{IO} \geq 1.1 \text{ GByte/s} \)

Case for I/O optimisation and caching
• Saturating single node bandwidth accessing PFS challenging
• Options for relaxing PFS bandwidth requirements by exploiting data-locality through a software managed data cache (in HPST or node local memory/storage)

[Oden et al., PDP 2019]
Reflection on session chair’s questions

How will changing requirements for HPC platforms influence design?
• Yes: Hardware architectures, system software, mode of operation

Will we see a dramatic shift in the types of applications and use cases for HPC?
• To some extend: New application domains, but mainly extended workflows (HPC in the loop)

What types of data-management services do you expect to be common?
• Data federation services, data transport services

How should we share/arbitrate use of shared, in-system storage among the different parts of the software stack?
• Usage models not yet fully clear, but observe some common patterns
• Need for co-design
  - Provide new capabilities to be explored by pioneering users
  - Improve analysis of science cases