Transparent I/O Optimization, Hierarchical-Storage Simulation, Earth-System-Data Middleware

Jakob Lüttgau

July 8, 2016

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

DKRZ
DEUTSCHES KLIMARECHENZENTRUM

esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE

Overview

1. Transparent I/O Optimization
2. Hierarchical-Storage Simulation
3. Earth-System-Data Middleware

1. Transparent I/O Optimization

2. Hierarchical-Storage Simulation

3. Earth-System-Data Middleware

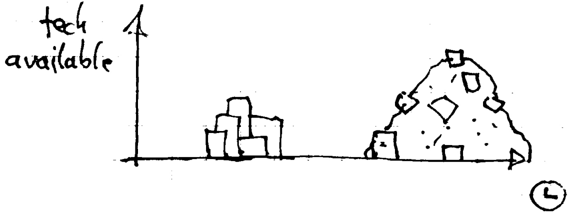
Motivation

The supercomputing langscape.

Mostly cluster systems. Quite complex.

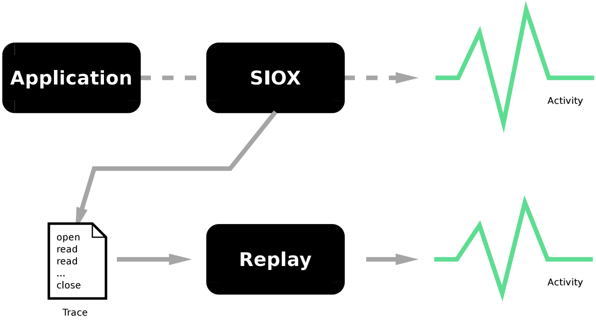
$$Hardware \times Software \times Topologies \times \cdots$$

Combine to suit characteristics of applications! Unfortunately:



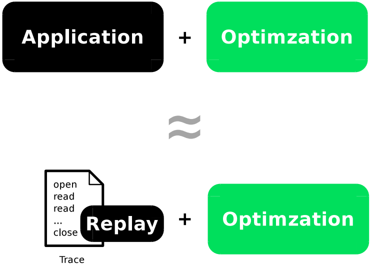
Trace replay to mimic applications

The trace preserves the characteristics.



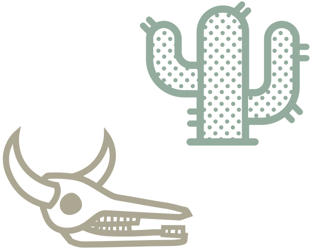
Analogousness

In many cases the following should be true.



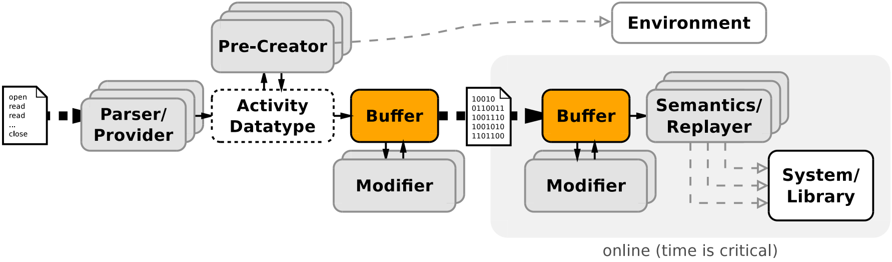
Parallel Trace Replay

Not so many tools available.



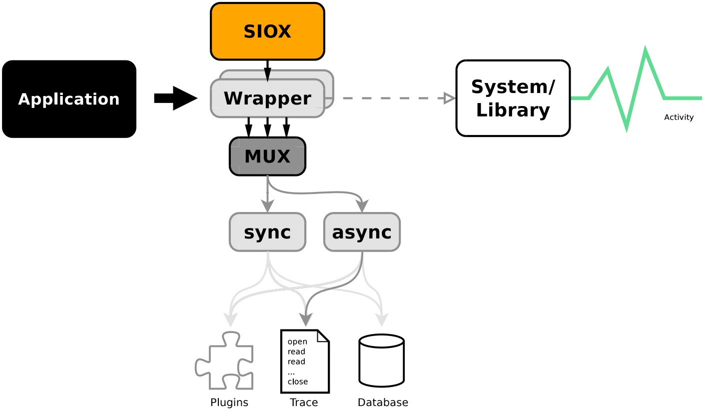
Activity Pipeline

How many hoops does it take to adequately reproduce I/O on different systems?
Prototype to establish good abstractions. Tools are all about convienience.



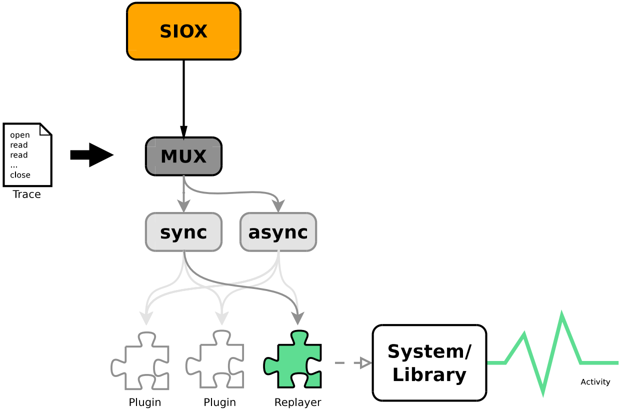
Introducing SIOX

Instrument applications to gain valuable insight. `LD_PRELOAD=..`



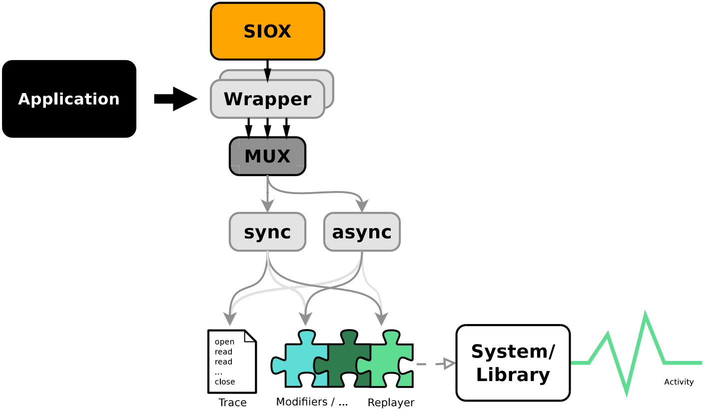
Replay with SIOX

SIOX can also process traces.



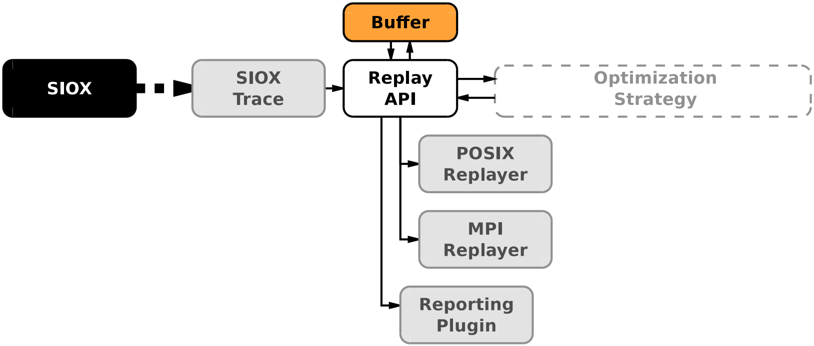
Runtime Optimizations using SIOX

Stack plugins to automatically optimize I/O



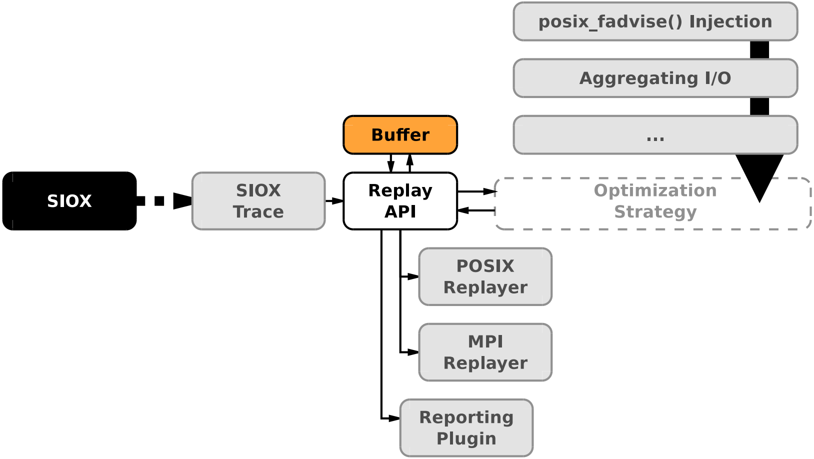
Virtual Lab

Stack plugins in different ways to craft new tools.



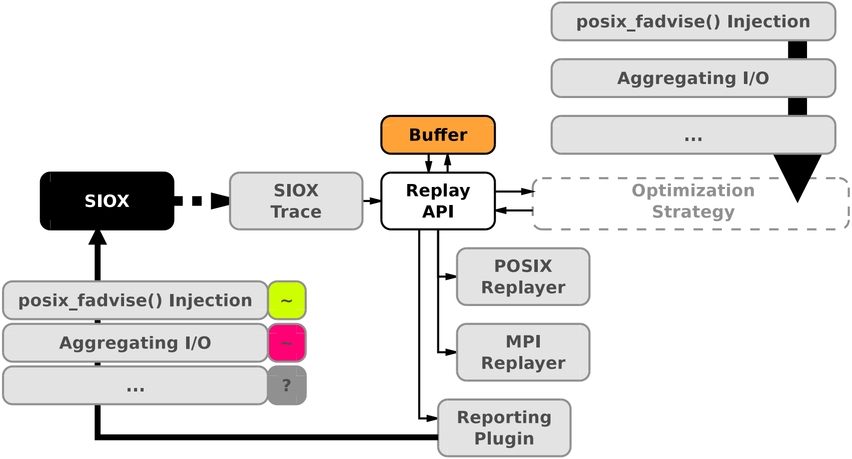
Virtual Lab (2)

Provide plugins that automatically apply optimizations to traces.



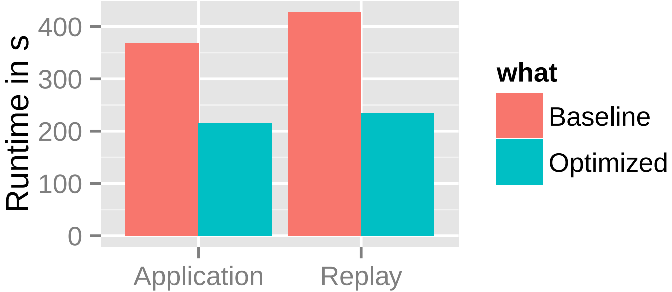
Virtual Lab (3)

Have reporting plugins to propagate results back to optimization engine.



Evaluation: POSIX fadvise injection

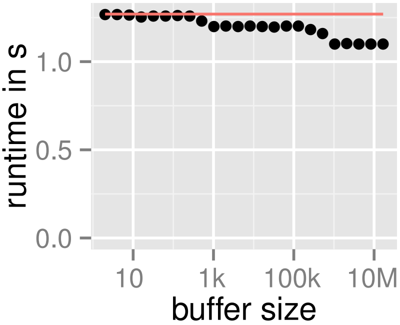
Find successive `lseek()` `read()` patterns and timely inject `fadvise()`.




```
                  ..
                  fadvise(pos,len, WILL_NEED)
..                ..
lseek(pos)        lseek(pos)
read(bytes)       read(bytes)
..                ..
```

Evaluation: Coalescing

Merge adjacent `read()` or `write()` operations. Show that optimzation works by sampling parameter space for optimum.




```
..                  ..
write(10)           write(30)
write(10)           ..
write(10)
..
```

1. Transparent I/O Optimization

2. Hierarchical-Storage Simulation

3. Earth-System-Data Middleware

Motivation

Long-term storage and upcoming challenges for exascale supercomputers.

Why long-term storage?

- Preservation of human knowledge
- Preservation of cultural goods (arts, literature, music, movies, etc.)
- Archival of organizational data (e.g., raw movie footage)
- Preservation of personal documents and photos
- Compliance with legal requirements

Challenges for scientific users (e.g., DKRZ, CERN):

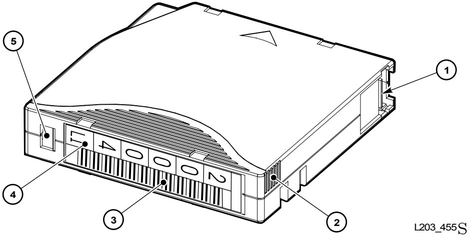
- Supercomputers highly parallel
- Produce data faster than can be stored persistently
- Producing insight was expensive and results should be preserved
- Deep storage hierarchies to balance cost and performance
- Scientific users already approaching exascale storage systems
- Innovation mostly dependent on vendors

History of Magnetic Tape Storage

1890s	Valdemar Poulsen invents **Magnetic Wire Recording**. Only limited use through the 1920s and 1930s, but popular from 1946 to 1954. One hour of audio recording required about 2200m of thin wire (0.10 to 0.15 mm).
1928	Fritz Pfleumer uses ferric oxide (Fe_2O_3) as a recording medium. The approach is improved by AEG and reel-to-reel tape recorder for tapes produced by BASF is released. The method was kept secret during World War II.
1947	John Bardeen, Walter Brattain and William Schockley invent the **Transistor**
1950	Reel-to-Reel recording and playback devices become affordable enabled by transistors.
1951	Data storage UNIVAC I (UNIVersal Automatic Computer I) 128 chars per inch, written on 8 tracks
1952	IBM introduces the first magnetic data storage devices often referred to as *7 Track*.
1962	Phillips invents **Compact Cassete** for audio recordings, though it was also sometimes used for data storage.
(1956)	*Focus on tape from here on, as other media such as floppies and diskettes are beyond the scope of the section.*
1959	Toshiba introduces helical scan as tape draw speed determines the maximum recordable frequency. The signal may not get imprinted which was a problem for video recording. Sony later pushed this technology further forward.
1980s	Introduction of **automated robotic tape libraries** by Sun with the Brand StorageTek. Tape is suddenly accessible within tens of seconds instead of hours or days. The term *nearline storage* gains traction to describe such systems.
1990s	Linear Tape Open (LTO) Consortium is founded. LTO is todays most wide-spread format.

LTO Tape Format

Linear Tape Open - Standards are beneficial for customers and vendors.



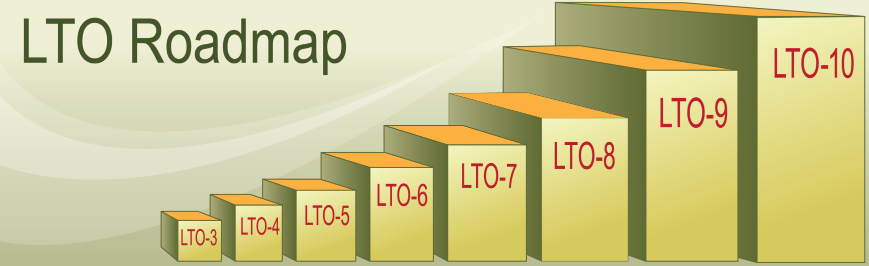
Sun (2006)

Gen	Thickness (μm)	Length (m)	Tracks	Bit Density	EEPROM
1	8.9	609	384	4880	4 kb
2	8.9	609	512	7398	4 kb
3	8.0	680	704	9638	4 kb
4	6.6	820	896	13250	8 kb
5	6.4	846	1280	15142	8 kb
6	6.1	846	2176	15143	16 kb
7	5.6	960	3584	NA	16 kb

- LTO-6: 0.011 USD/GB native, 0.005 USD/GB compressed, (2.5 to 6 TB)
- LTO-7: 0.028 USD/GB native, 0.012 USD/GB compressed, (6 to 15 TB)

Linear Tape Open (2)

LTO release strategy: Backwards-compatibility; New generation every 2-3 years.



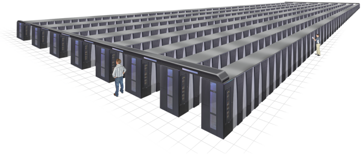
	LTO-3	LTO-4	LTO-5	LTO-6	LTO-7	LTO-8	LTO-9	LTO-10
Shipment Year	2005	2007	2010	2013	2015	TBD	TBD	TBD
Native Capacity	400GB	800GB	1.5TB	2.5TB	6.0TB	Up to 12.8TB	Up to 25TB	Up to 50TB
Compressed Capacity	800GB	1.6TB	3.0TB	6.25TB	15TB	Up to 32TB	Up to 62.5TB	Up to 125TB
Native Transfer Rate	80 MB/s	120 MB/s	140 MB/s	160 MB/s	300 MB/s	Up to 472 MB/s	Up to 708 MB/s	Up to 1100 MB/s
Compressed Transfer Rate	160 MB/s	240 MB/s	280 MB/s	400 MB/s	750 MB/s	Up to 1180 MB/s	Up to 1770 MB/s	Up to 2750 MB/s

(Spectralogic, 2016a)

Automated Tape Libraries

Archives; Data reduction and compression; Encryption; Self-describing tape formats;

IBM TS3500 Library Complex (IBM, 2011b)



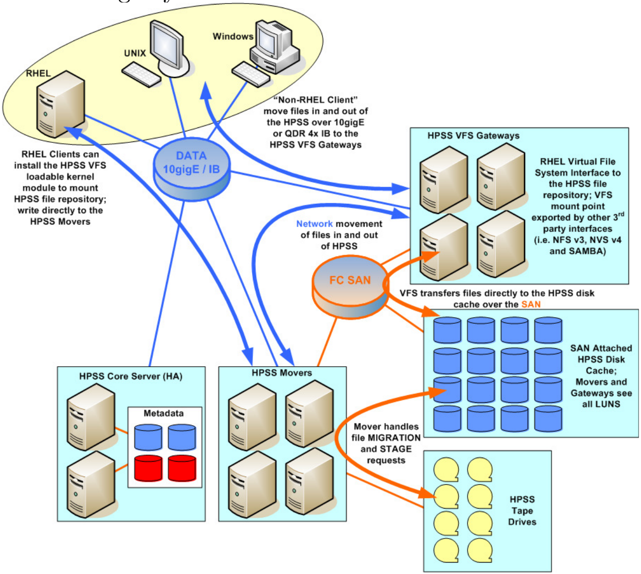
TFinity Library Complex (Spectralogic, 2016b)

StorageTek SL8500 Library Complex (Oracle, 2015)

Scalar i6000 Library Complex (Quantum, 2015)

HPSS

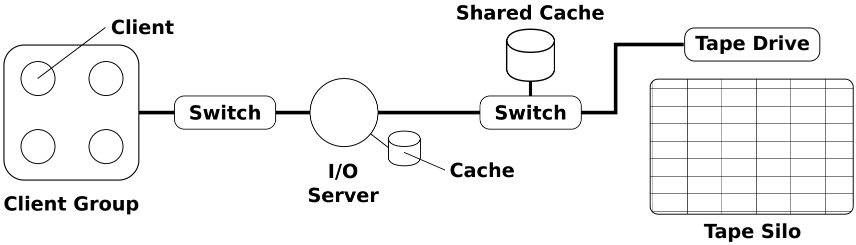
High Performance Storage Systems



(IBM, 2011a)

A simple model to get started

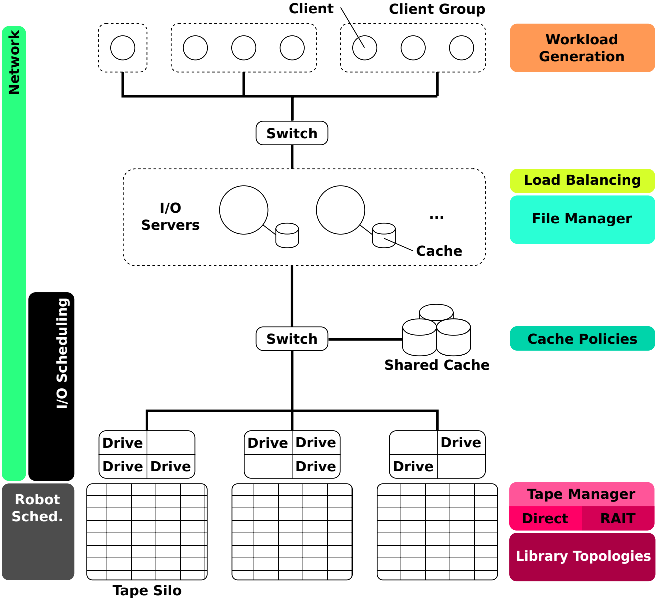
Introduction of the most important components.



1. Multiple *clients* which may issue requests to `read` and `write` data
2. An *I/O Server* to receive and handle the requests
3. Different *cache levels*, to speed up access for recently touched files
4. Automated *tape silos* and *tape drives* to access the archive

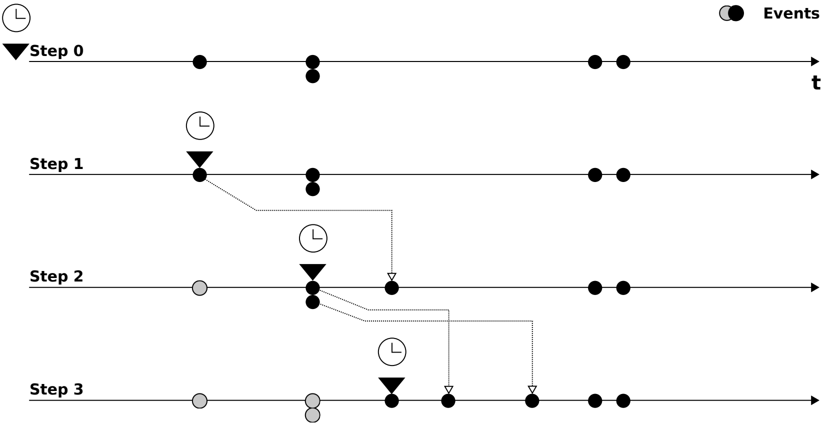
Model Overview

Hardware and software components in a combined overview.



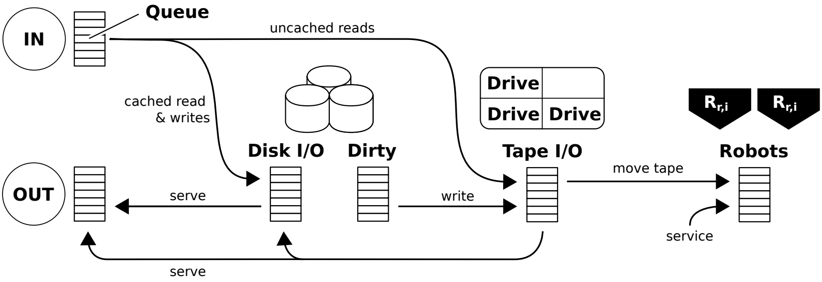
Discrete Event Simulation

Only require calculations when the state of the system changes.



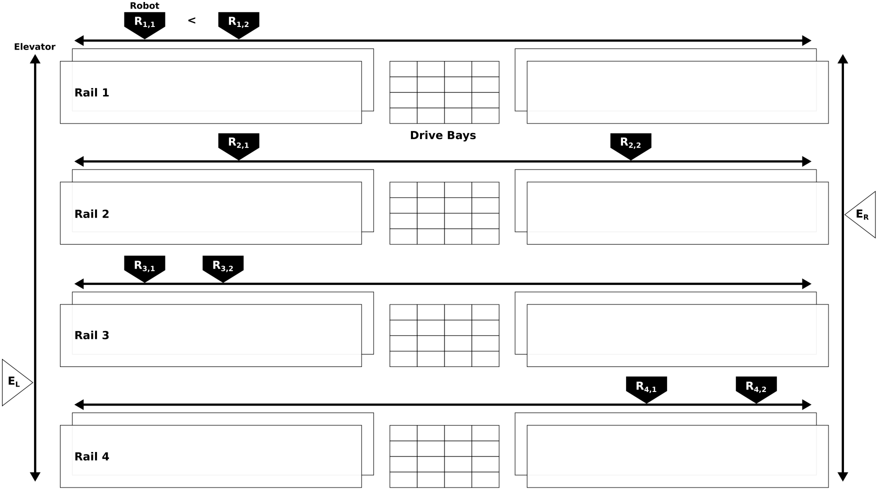
Scheduling and Request Queues

Chaining specialized request queues makes resource allocation manageable.



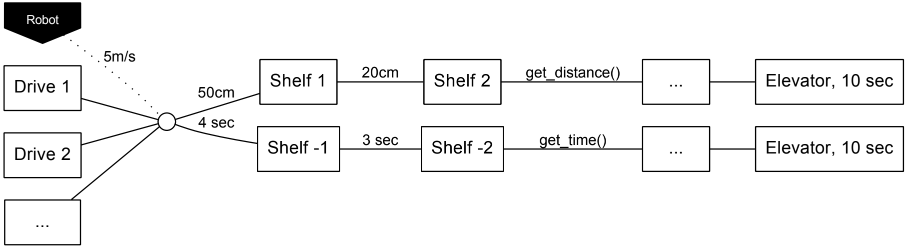
Robot Scheduling

Example: How a single SL8500 library maybe seen by a scheduling component.



Graph-Based Topology Model

Component connecticity graphs with distance or time panalties.

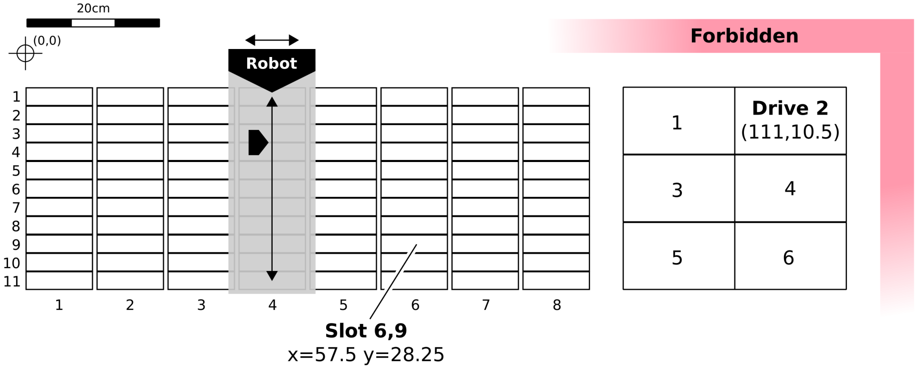


$$\text{get\_time}(e_{v_i,v_j} \text{ or } v) := \begin{cases} t & \text{if } e_{v_i,v_j} \text{ or } v \text{ have time } t \text{ set} \\ \frac{\text{get\_distance}(v_i,v_j)}{v_{robot}} & \text{if } e \text{ but no time is set} \\ 0 & \text{otherwise} \end{cases}$$

$$T_G(v_i, v_j) = \sum_{v_i,v_j}^{\text{shortest\_path}(v_0,v_1)} \text{get\_time}(v_i) + \text{get\_time}(e_{v_i,v_j})$$

2D Topology Model

Flat library projections and tape receive times. Optional with easing.

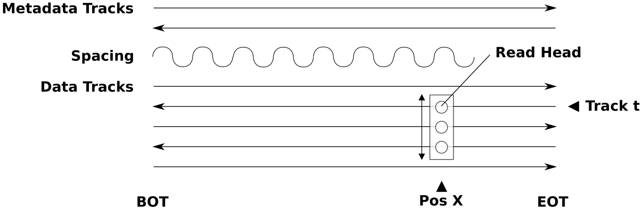


$$T_{2D}(p_j, p_i) = max\left(\frac{|p_{ix} - p_{jx}|}{v_x}, \frac{|p_{iy} - p_{jy}|}{v_y}\right)$$

$$T_{2D}(path) = \sum_{p_i, p_j}^{path} T_{2D}(p_i, p_j) + T_{wait/work}$$

Serpentine Tape Model

Estimating spool and seek times for tape access.

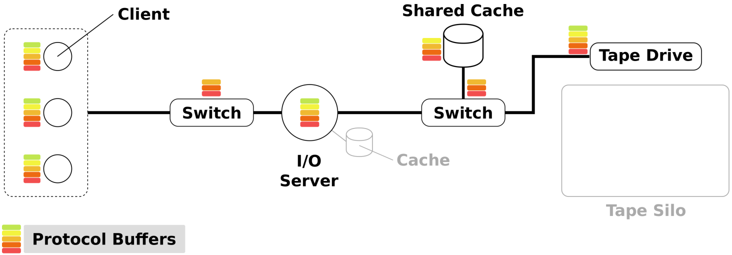


$$T_{seek}(pos_j, pos_i) = max\left(\frac{|pos_{ix} - pos_{jx}|}{v_{spool}}, \frac{|pos_{it} - pos_{jt}|}{v_{head}}\right)$$

$$T_{read/write}(bytes) = \frac{bytes}{v_{read/write}}$$

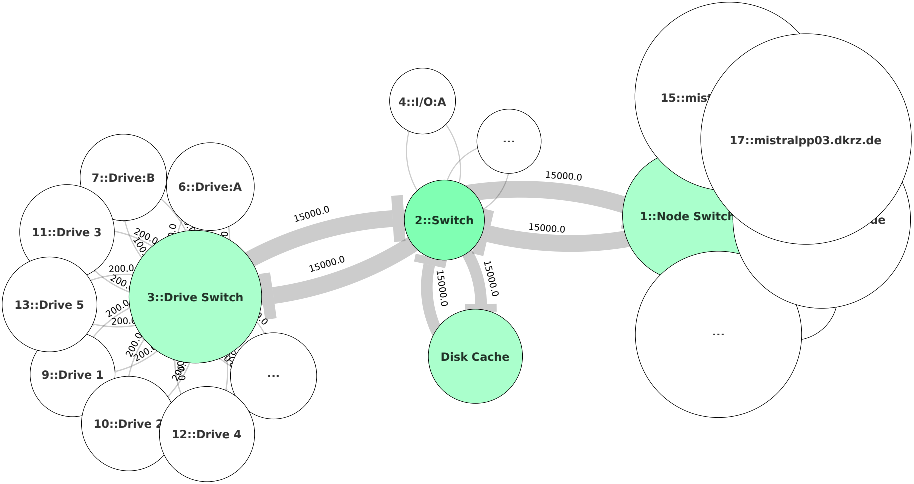
$$T_{busy} = T_{mount} + \left(\sum_{pos_i, pos_{i+1}}^{BOT,\dots,BOT} T_{seek}(pos_i, pos_j) + T_{read/write}(bytes_i)\right) + T_{unmount}$$

PDU/Package-based Network Model

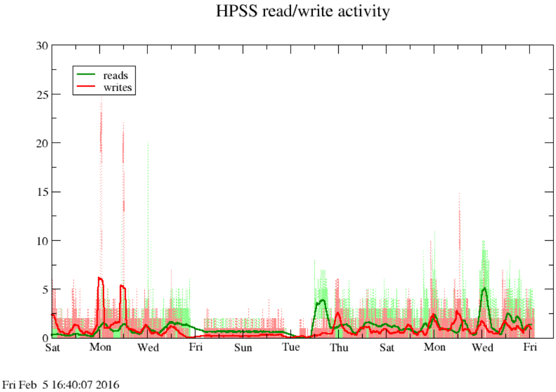


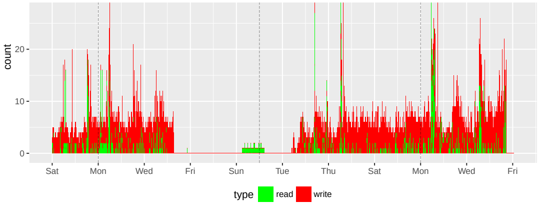
	Layer	Protocol data unit	Function / Example
Host layers	**7 Application**	**Data**	CSS, HTML, Javascript FTP, NFS, HTTP, HTTPS, RPC, SMTP
	6 Presentation		
	5 Session		
	4 Transport	**Segment (TCP) / Datagram (UDP)**	TCP, UDP, SSH, NetBIOS
Media layers	**3 Network**	**Packet**	IPv4, IPv6, ICMP
	2 Data link	**Frame**	IEEE 802.2, L2TP, MAC, PPP
	1 Physical	**Bit**	Ethernet, SCSI, USB, ISDN, DSL

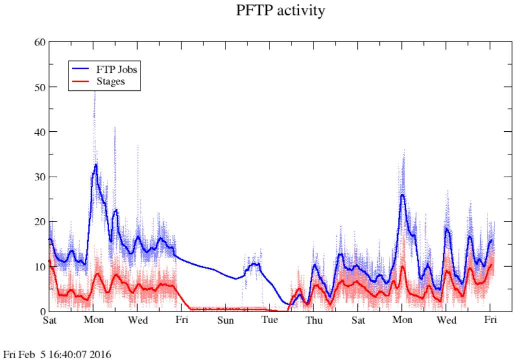
Network Topology used for Evaluation

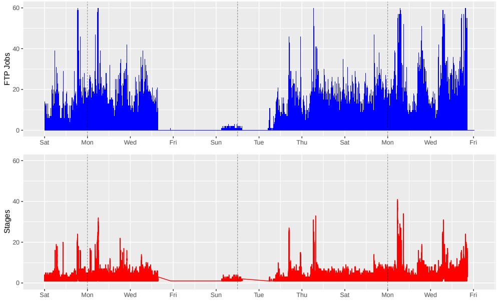


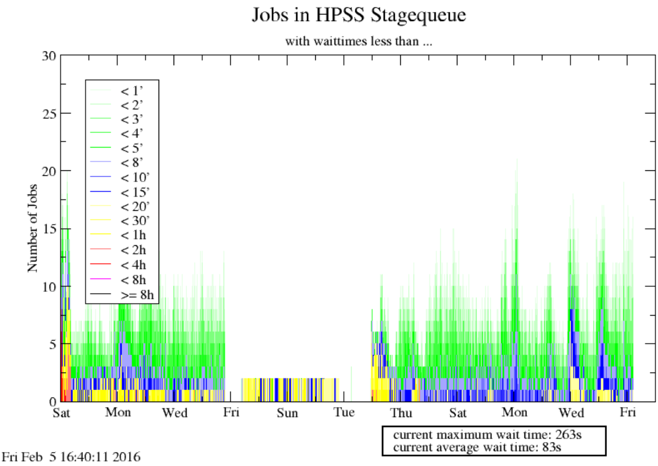
A familier face: Workload Trace for Verification

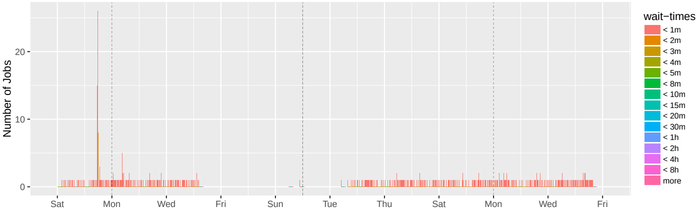





PFTP activity
FTP Jobs
Stages
0
10
20
30
40
50
60
Sat
Mon
Wed
Fri
Sun
Tue
Thu
Sat
Mon
Wed
Fri
Fri Feb 5 16:40:07 2016


FTP Jobs
Stages
0
20
40
60
Sat
Mon
Wed
Fri
Sun
Tue
Thu
Sat
Mon
Wed
Fri


Jobs in HPSS Stagequeue
with waittimes less than ...
Number of Jobs
0
5
10
15
20
25
30
< 1'
< 2'
< 3'
< 4'
< 5'
< 8'
< 10'
< 15'
< 20'
< 30'
< 1h
< 2h
< 4h
< 8h
>= 8h
Sat
Mon
Wed
Fri
Sun
Tue
Thu
Sat
Mon
Wed
Fri
current maximum wait time: 263s
current average wait time: 83s
Fri Feb 5 16:40:11 2016


Number of Jobs
0
10
20
Sat
Mon
Wed
Fri
Sun
Tue
Thu
Sat
Mon
Wed
Fri
wait−times
< 1m
< 2m
< 3m
< 4m
< 5m
< 8m
< 10m
< 15m
< 20m
< 30m
< 1h
< 2h
< 4h
< 8h
more

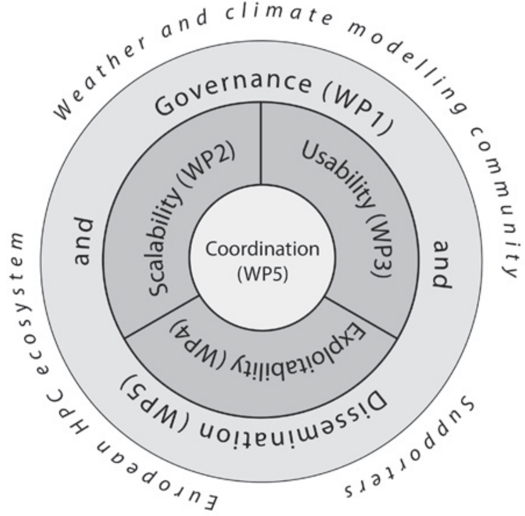
1. Transparent I/O Optimization

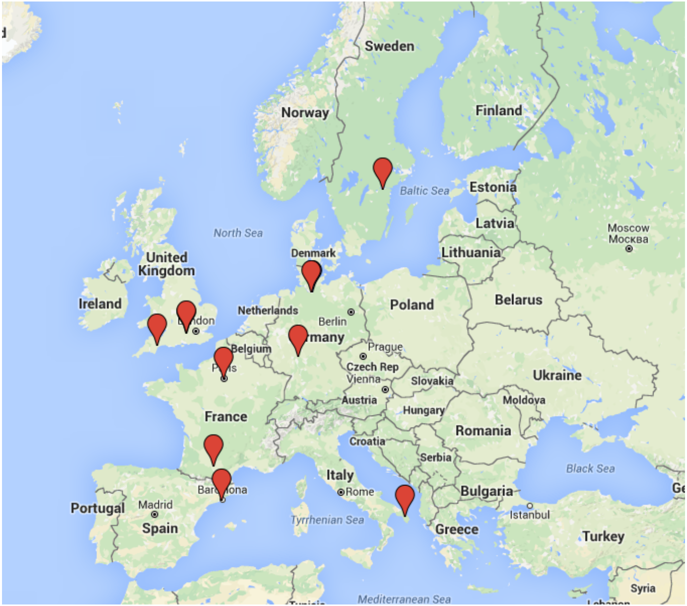
2. Hierarchical-Storage Simulation

3. Earth-System-Data Middleware

esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE


Site Map | Contact | Log in
IN SHORT
RESULTS
EVENTS
SERVICES
CONTEXT
esiwace
CENTRE OF EXCELLENCE IN SIMULATION OF WEATHER AND CLIMATE IN EUROPE
This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 675191
Search Site
OK
ESiWACE
ESiWACE stands for Centre of Excellence in Simulation of Weather and Climate in Europe
We are a new initiative of the HPC ecosystem in Europe and we leverage two established European networks: the European Network for Earth System modelling (ENES) http://enes.org representing the European climate modelling community and the world leading European Centre for Medium-Range Weather Forecasts (ECMWF) http://www.ecmwf.int
Our goal is to substantially improve efficiency and productivity of numerical weather and climate simulation on high-performance computing platforms by supporting the end-to-end workflow of global
UPCOMING EVENTS
Joint final workshop on IS-ENES2 Workflow Solutions in Earth System Modelling and on Meta-Data Generation during Experiments
Sep 26, 2016 - Sep 28, 2016 — Lisbon (Portugal)
Previous events...
Upcoming events...
TWITTER @ESIWACE
Steven_VdB80 : RT @Eudat_eu: Keep in mind the guiding principles for data management: make data F.A.I.R. #DMP #einfrastructures @MarjanGrootveld https://t...
Jul 07, 2016 04:19 PM
IFB_Bioinfo : RT @PhnmlH2020: We still need feedback in our #metabolomics data survey,


Weather and climate modelling community
and
Europe HPC ecosystem
and
suppliers
Governance (WP1)
Scalability (WP2)
Usability (WP3)
Exploitability (WP4)
Coordination
(WP5)
Dissemination (WP5)



STFC, DKRZ, ECMWF, CMCC, SEAGATE

WP4 Exploitability

Model Systems and Use-Cases

- Better understand how to handle high volumes earth system data and how to exploit storage hardware
- Convergence of weather and climate HPC Workflows

Earth-System-Data Middleware

- Disk Storage Layout for ESD, overcome limitations of current formats and APIs
- New interface for data-concious storage (beyond bytestreams)
- Account for more heterogenous disk storage environments

Tape access strategies

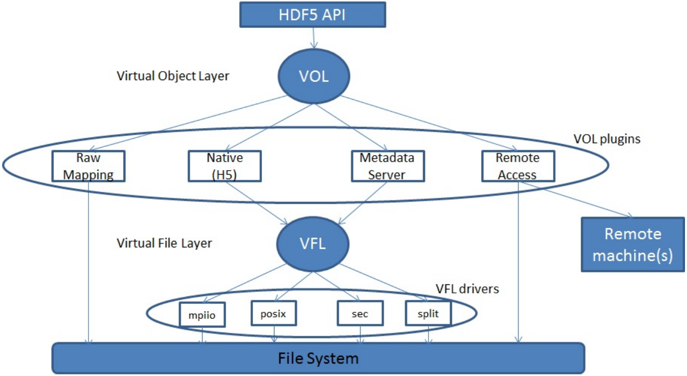
- New tape access strategies
- Improve bandwidth and redundancy (e.g. RAIT)

Semantic Mapping between climate and weather formats

- GRIB (primalary used in weather, e.g. satellite images)
- netCDF (popular within climate community)

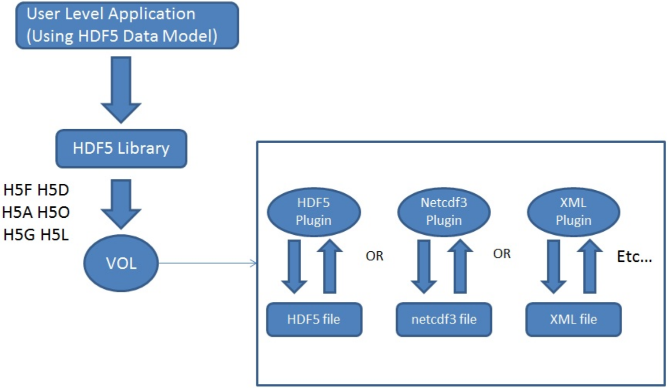
New in HDF5: VOL

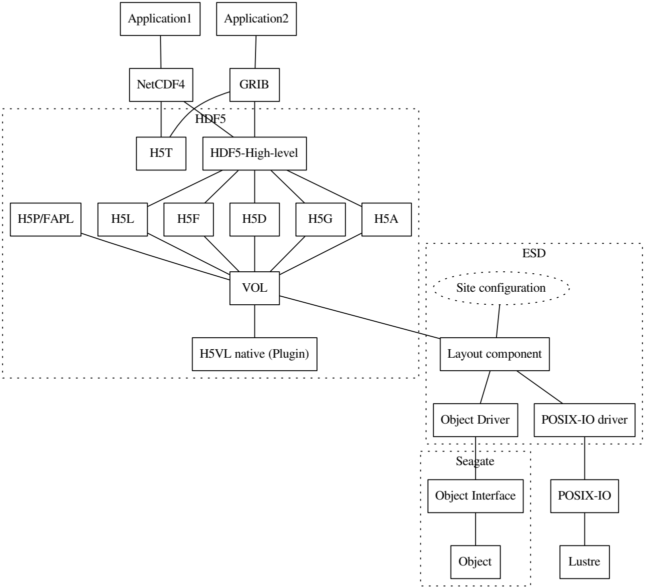
Virtual Object Layer



New in HDF5: VOL

Virtual Object Layer




Application1
Application2
NetCDF4
GRIB
HDF5
H5T
HDF5-High-level
H5P/FAPL
H5L
H5F
H5D
H5G
H5A
VOL
H5VL native (Plugin)
ESD
Site configuration
Layout component
Object Driver
POSIX-IO driver
Seagate
Object Interface
POSIX-IO
Object
Lustre

Questions?

Appendix

4. Library Management

5. Concurrency

6. Runtime and Memory Requirements

7. Misc

Bibliography I

Fontana, R. E., Decad, G. M., and Hetzler, S. R. (2013). The Impact of Areal Density and Millions of Square Inches (MSI) of Produced Memory on Petabyte Shipments of TAPE , NAND Flash , and HDD Storage Class Memories. *Proceedings of the 29th IEEE Symposium on Massive Storage Systems and Technologies*.

IBM (2011a). High Performance Storage System. Technical report.

IBM (2011b). IBM System Storage TS3500 Tape Library Connector and TS1140 Tape Drive support for the IBM TS3500 Tape Library. pages 1–15.

Oracle (2015). StorageTek SL8500 Modular Library System User's Guide.

Quantum (2015). Quantum Scalar i6000 Datasheet.

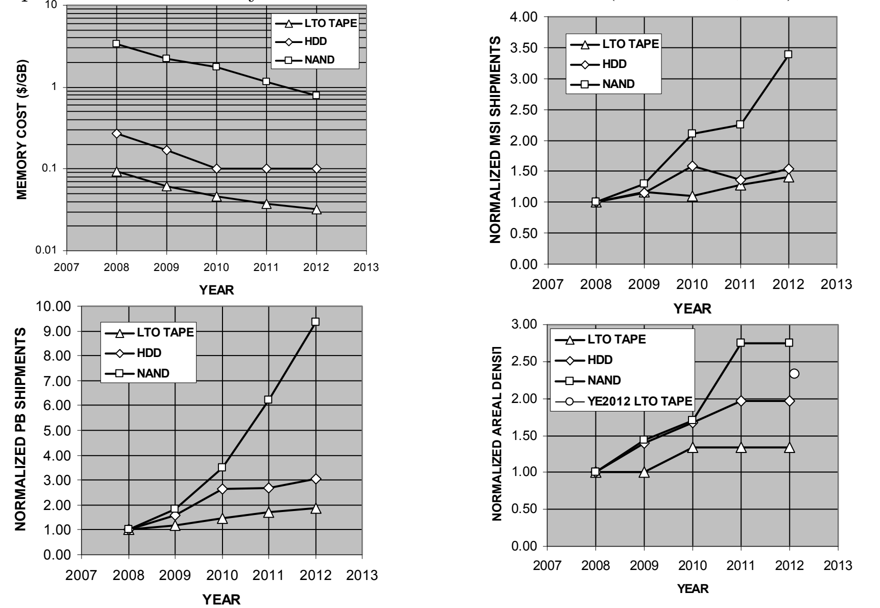
Spectralogic (2016a). LTO Roadmap. `https://www.spectralogic.com/features/lto-7/`. [Online; accessed 2016-01-24].

Spectralogic (2016b). Spectralogic TFinity - Enterprise Performance. `https://www.spectralogic.com/products/spectra-tfinity/tfinity-features-enterprise-performance/`. [Online; accessed 2016-02-12].

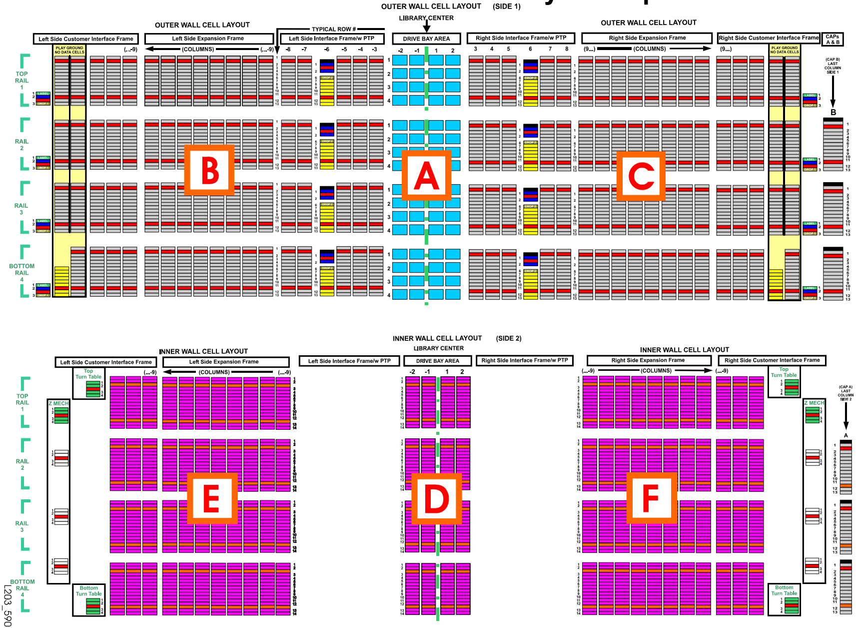
Sun (2006). StorageTek StreamLine SL8500 - User Guide. (96154).

Future of Tape

Is tape obsolete? Probably not for another decade or two. (Fontana et al., 2013)



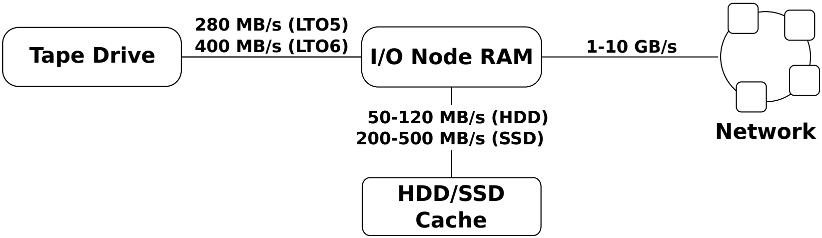
Outer and Inner Wall Cell Layout Map



(Sun, 2006)

Network Model Granularity

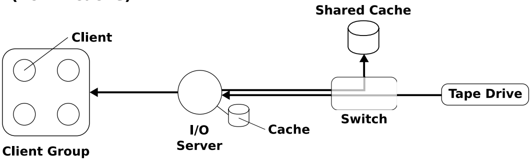
Tape Drive and HDD/SSD throughput are limiting factors.



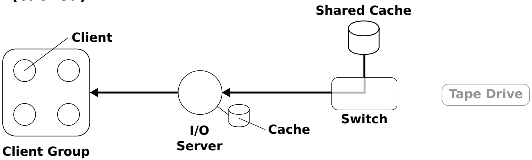
Handling READ Requests

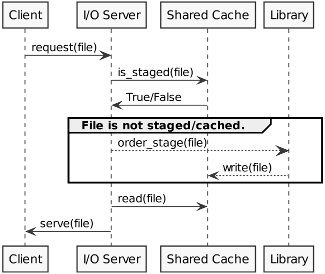
Staging of recently accessed files for reads.

READ (not in cache)



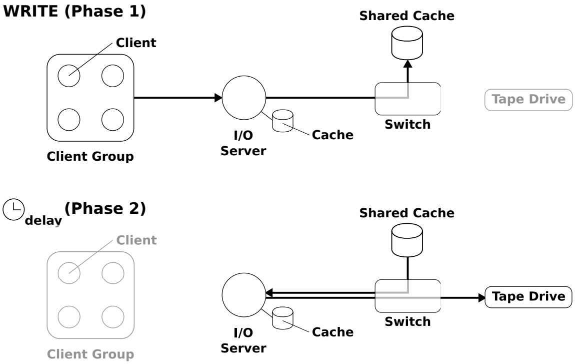
READ (cached)

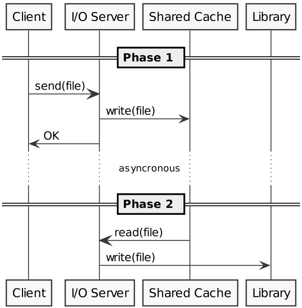




Handling WRITE Requests

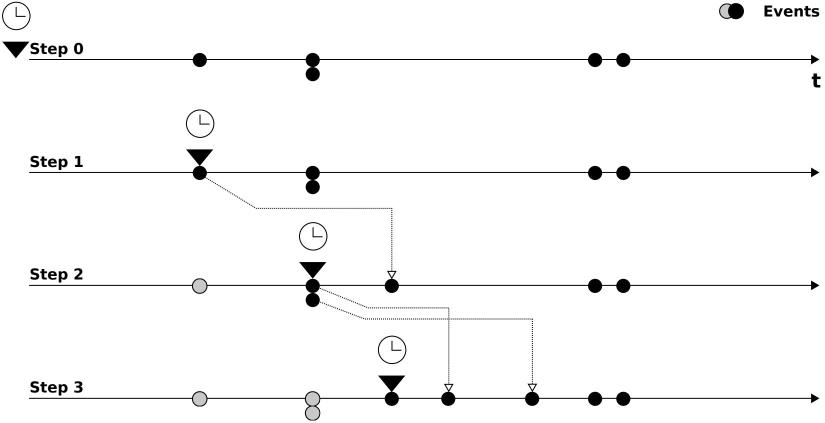
Two-Phase write with delayed persistence on tape.





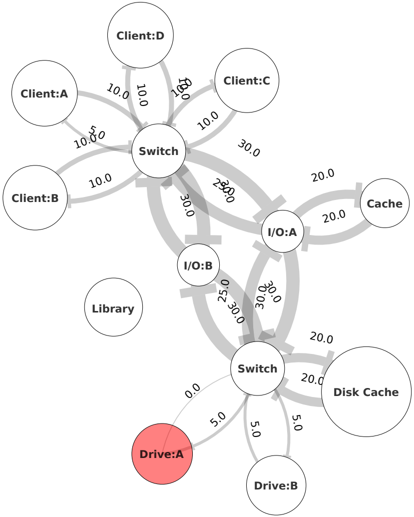
Discrete Event Simulation

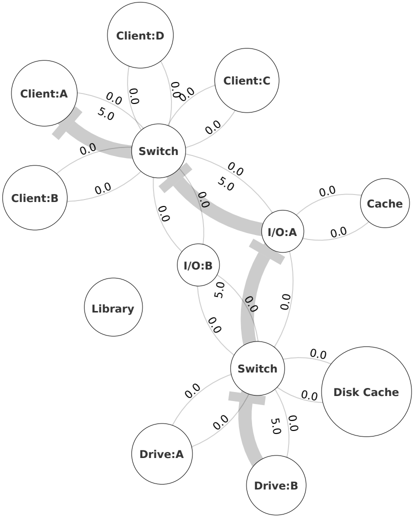
Only require calculations when the state of the system changes.



Network Model (Implementation)

Example: The network with one busy drive. Max-flow used to estimate throughput.



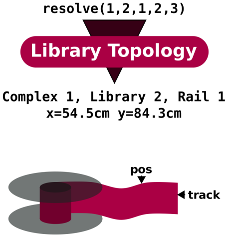


Library Organisation and Management

File and tape management.

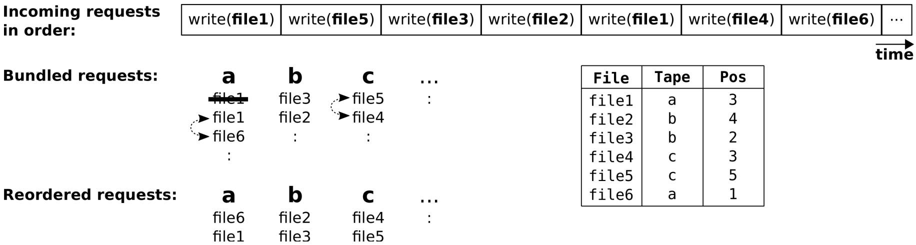
File	Size	Position	Tape
file1	3134	pos,track	012345L1
file2	6483	45,447	LTO834L5
file3	39485	1623,187	274344L4
file4	38474	2245,184	274344L4
file5	345	3749, 47	LTO834L5

Tape	Slot
012345L1	1,1,1,18, 9
264653L4	3,1,3, 7, 5
274344L4	1,4,2,-6,12
267753L4	2,2,4, 3, 5
LTO834L5	1,3,3, 7, 1
CLN004CU	2,3,1,-7, 8
CLN031CU	1,2,1, 2, 3



Concurrency

I/O scheduling and strong vs. weak ordering semantics

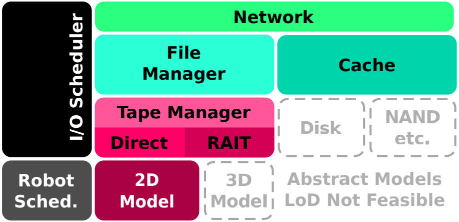


File	Tape	Pos
file1	a	3
file2	b	4
file3	b	2
file4	c	3
file5	c	5
file6	a	1

1. $O_i = read(D)$, $O_j = read(D)$. Maybe handled concurrently.
2. $O_i = read(D)$, $O_j = write(D)$. Can not be handled concurrently.
3. $O_i = write(D)$, $O_j = read(D)$. Can not be handled concurrently.
4. $O_i = write(D)$, $O_j = write(D)$. Can not be handled concurrently.

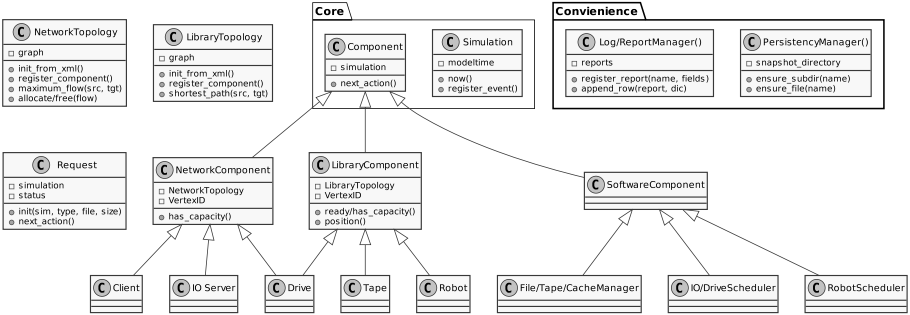
Tape System Software Stack

A similar stack should also allow to run a real tape system.



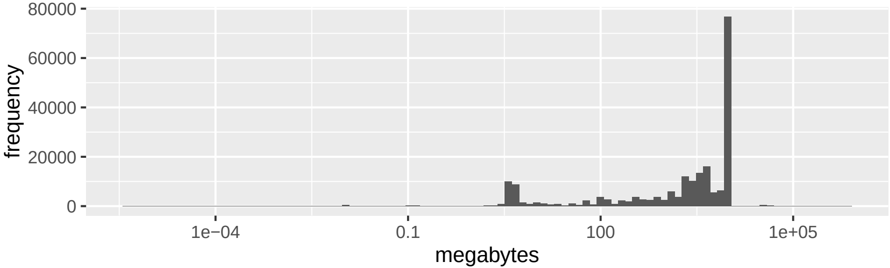
Components and Classes

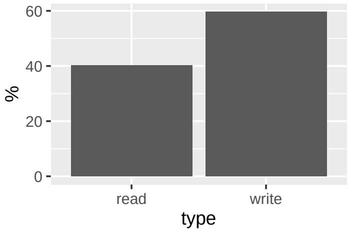
UML Class Diagram



Workload Trace (2)

Request size and request type distributions





Runtime and Memory Consumption

Only request data is immediately written to disk. Some other data accumulates.