Compute Infrastructures at DESY - Outline

- BIRD
  - Quick overview
- GRID, NAF, PAL
- HPC resources in Zeuthen
- HPC cluster
  - Hardware & Layout
  - Setup & Numa
  - Getting started
  - Monitoring & Operational model
  - Storage
  - Examples: Mathematica & Matlab
BIRD - Batch Infrastructure Resource at DESY

- See http://bird.desy.de/info (being updated)
- General purpose batch farm
- Getting access: via registry batch resource → UCO@desy.de
- Problems: UCO@desy.de; Bird.Service@desy.de

<table>
<thead>
<tr>
<th>OS</th>
<th># Hosts</th>
<th>Cores/Host</th>
<th>Total cores</th>
<th>RAM/Core</th>
<th>Total RAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLD6-64</td>
<td>17</td>
<td>4-12</td>
<td>168</td>
<td>8-48GB</td>
<td>0.5TB</td>
</tr>
<tr>
<td>SLD5-64</td>
<td>157</td>
<td>8-48</td>
<td>1520</td>
<td>8-126GB</td>
<td>4.1TB</td>
</tr>
<tr>
<td>Total</td>
<td>174</td>
<td>4-48</td>
<td>1688</td>
<td>8-126GB</td>
<td>4.6TB</td>
</tr>
</tbody>
</table>

- Environments for mpi or GPU jobs
  - High latency 1GE backbone
  - Jobs with heavy inter-process communication will suffer
### Compute Infrastructures at DESY - BIRD

- GPUs are embedded into the BIRD farm

<table>
<thead>
<tr>
<th>Host</th>
<th>OS</th>
<th>Hardware</th>
<th>GPUs</th>
<th>Drv / Cuda</th>
<th>Bird Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>It-gpu01</td>
<td>SL5-64</td>
<td>Intel Xeon 2.5GHz 16 cores 24GB ram</td>
<td>Fermi 2*M2050 448 cores, 3GB</td>
<td>Nvidia 260.24 Cuda 3.2</td>
<td>-l h=it-gpu01 -l gput=nvidiaM2050 Needs upgrade</td>
</tr>
<tr>
<td>It-gpu02</td>
<td>SL5-64</td>
<td>AMD Opteron 870 2GHz 8 cores 16GB ram</td>
<td>Tesla 1*C870</td>
<td>Nvidia 270.41.19 Cuda 4.0</td>
<td>-l h=it-gpu02 -l gput=nvidiaC870 Will retire soon</td>
</tr>
<tr>
<td>It-gpu03</td>
<td>SL5-64</td>
<td>AMD Opteron 6128 2GHz 16 Cores 24GB ram</td>
<td>Fermi 2*C2070 448 cores, 6GB</td>
<td>Nvidia 270.41.19 Cuda 4.0.17</td>
<td>-l h=it-gpu03 -l gput=nvidiaC2070 Accelereys Jacket</td>
</tr>
<tr>
<td>It-gpu04</td>
<td>SL6-64</td>
<td>Intel Xeon 2.4GHz 24 cores 24GB ram</td>
<td>Fermi 2*M2070 448 cores, 6GB</td>
<td>Nvidia 304.51 Cuda 5.0.35</td>
<td>-l h=it-gpu04 -l gput=nvidiaC2070</td>
</tr>
<tr>
<td>It-gpu05</td>
<td>Coming soon</td>
<td></td>
<td>Kepler K20 2496 cores, 5GB</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Interactive login via qrsh
  - ssh -XY pal; ini bird; qrsh -l h=it-gpu03 -l gput=nvidiaC2070 ...
  - Graphical apps like matlab possible (xhost; DISPLAY)
HPC cluster - History

• Zeuthen has a long HPC history
  - APE1 in 1987 → APEnext in 2005 (array processor experiment)
  - QPACE for lattice QCD simulations based on IBM PowerXCell 8i processors
  - GPE - GPU Processing Experiment (HadronPhysics2 project).
  - NIC – John von Neumann Institute for Computing (DESY, GSI, FZJ)

• DESY Hamburg
  - Started in 2011 with retired hardware. Initiated by Ilya Agapov
  - 64*8 cores and distributed Fraunhofer filesystem (fhgfs)
  - 20GE Infiniband backbone (on an old switch)
  - Proofed to be rather useful, just the old hardware was dying rapidly.
  - Retired the old test-cluster last year and migrated to a new system → it-hpc cluster
HPC Cluster – Hardware

> Compute Nodes:

- **16 Dell PowerEdge R815**
  - 4 CPU AMD Opteron(TM) Processor 6276 2.3GHz
    - Per CPU 16 Bulldozer Cores with 64 GB RAM
  - 256 GB RAM
  - Numa Architecture
  - 10GE Ethernet
  - QDR Infiniband (40GB/s)

- **2 PowerEdge C6145**
  - 4 CPU AMD Opteron(TM) Processor 6276 2.3GHz
    - Per CPU 16 Bulldozer Cores with 48 GB RAM
  - 192 GB RAM
  - Numa Architecture
  - 10GE Ethernet
  - QDR Infiniband (40GB/s)
Storage Nodes:

- **4 Dell PowerEdge R510**
  - 2 CPU Intel(R) Xeon(R) CPU E5503 @ 2.00GHz
    - Per CPU 2 Cores with 6 GB RAM
  - 12 GB RAM
  - QDR Infiniband (40GB/s)
  - 20 TB / 12 Disc / Raid 6

- **1 PowerEdge 2950**
  - 2 CPU Intel(R) Xeon(R) CPU E5345 @ 2.33GHz
    - Per CPU 4 Cores with 4 GB RAM
  - 8 GB RAM
  - Numa Architecture
  - 1 GE Ethernet
  - QDR Infiniband (40GB/s)
InfiniBand is a switched fabric communications link used in high-performance computing and enterprise data centers. Its features include high throughput, low latency, quality of service and failover, and it is designed to be scalable. The InfiniBand architecture specification defines a connection between processor nodes and high performance I/O nodes such as storage devices. (wikipedia)

> Infiniband Infrastructure:

- Mellanox SX6036 InfiniBand/VPI Switch Systems
  - 36 Ports
  - 56 GBIT/s FDR
  - Managed
- Mellanox Connect-X3 QDR Cards
HPC Cluster – Layout

1024 (+128) cores – 4TB (+0.38TB) memory – >100TB temporary storage
56G infiniband backbone – interlagos PCI-E2

10G

40G

IB switch

it-hpc101..116

Compute

Management

Storage

56G

IP switch

2*10G

netapp

netapp4

AFS

dCache

64 cores 256GB

1024 (+128) cores – 4TB (+0.38TB) memory – >100TB temporary storage
56G infiniband backbone – interlagos PCI-E2
HPC Cluster – Setup/Optimization

> Setup
- Scientific Linux 6.3 (always updated)
  - No automatic updates → maintenance
  - No salad/yum/cron jobs
  - Xymon monitoring
- AFS/Kerberos
- IPoIB configuration

> Optimizations
- Bound infiniband card to core
- Switch of timer interrupt
- Unlimited locked-in-memory address space
Non-Uniform Memory Access (NUMA) is a computer memory design used in multiprocessing, where the memory access time depends on the memory location relative to a processor. Under NUMA, a processor can access its own local memory faster than non-local memory (memory local to another processor or memory shared between processors). (wikipedia)

- All compute nodes are numa machines
- To gain the full potential of the nodes you should consider the architecture
- Tools like likwid are helpful (necessary?)
  - Examine Numa topology
  - Profile software on Numaarchitectue
  - Bind processes to cores
- Multicore CPU make it even more complicated
  - Shared caches
  - Shared FPU
  - Different Levels of aggregation
- Commercial software is not always numa aware
HPC Cluster – Numa Hardware

Example: 2 Socket Bulldozer System
HPC Cluster – Numa Hardware

RAM_A

CPU_1

CPU_2

RAM_B
HPC Cluster – Numa Hardware

RAM_A

CPU_1

CPU_2

RAM_B

HT

HT
HPC Cluster – Numa Hardware

User starts process P1
HPC Cluster – Numa Hardware

- **CPU_1**
  - Process P1 runs on CPU_1 and allocates memory on RAM_A

- **CPU_2**

- **RAM_A**
- **RAM_B**
process P1 is suspended and still allocate Memory on RAM_A
process P1 is rescheduled on CPU_2 and allocate Memory on RAM_A
Simple example - binding sockets and IB adapter

host-host ib bandwidth without socket binding

host-host ib bandwidth with socket binding
> Pre-requisites:

- Multi-core capable applications
- Embarrassingly parallel problems → Bird or Grid (if an option)
- 32/64 cores per node with 8/4GB per core → prefer <= 8GB/core
  - Single core performance of AMD interlagos not so great
  - If you need lots of memory/core different architectures might be favorable
  - Could be added to it-hpc if that's a common requirement

> Uncertain your application is suitable?

- Get in touch!
- Run tests on it-hpc-wgs!
  - it-hpc-wgs are freely accessible.
  - Identical configuration like it-hpc(xxx), but slightly less memory (192GB)
Gaining access:

- Access is restricted to „registered“ users
  - No restrictions on it-hpc-wgs!

- Send mail to unix@desy.de or uco@desy.de asking for it-hpc „resource“
  - Let us know what kind of application you intend to run
  - Helps us providing essential dependencies
  - Helps us to plan future expansions
Scheduling:

- Create an account on the it-hpc web-calendar:
  - Use your DESY account name (not the password!)
  - Currently not kerberized.

- Reserve resources
  - Co-operative model, no fair-share
  - Reservation for full nodes – no sharing of hosts
  - Currently fixed 6h time slots (can be adjusted if needed).
  - Don't reserve all nodes for a long time!
  - We do some accounting, if it gets too much out of balance, the model might be adjusted.
HPC Cluster – Monitoring

> Reservations:

- Currently don’t enforce reservations
  - That will change soon!
  - Jobs without reservation have low chance of survival
- Jobs conflicting with a reservation will be terminated
  - Jobs exceeding the reservation will be terminated as well.
- Over-subscription creates an alert
  - Load > 64 will slow everything down
  - No termination of jobs.

5’ update cycle from xymon
HPC Cluster – Running jobs

> Scientific SW & Compiler usually installed in /opt/<app>/<version>
  - idl, intel, maple, mathematica, matlab, pgi
  - AFS-hosted; but no token required
  - Can be changed to local installations if needed
  - Own software: best on /data/netapp or /afs

• OpenMPI in 2 variants
  - openMPI-1.5.x and openMPI-intel-1.5.x
    • +mpich2-1.2.x

> Compiler
  - gnu 4.4.x, pgi-2012/13, intel-2012/13
  - gnu 4.7.x in /data/netapp/it/tools permits interlagos optimization (-march=dbver1)
  - Be careful with intels optimization
    • Floating point model defaults to "value unsafe optimization"
    • Intel recommends -fp-model precise -fp-model source to be on the safe side
    • Other apps (e.g. matlab) have similar "issues"
> Initialize

- `ini` deprecated on SL6, use `module` instead
  - module `avail` to display
  - module `load intel.2011`
  - module `load mathematica`
  - module `load openmpi-x86_64` etc

- For openmpi propagate the prefix
  - `mpirun --prefix /usr/lib64/openmpi ...
  - or: module `load openmpi-x86_64` (in e.g. `.zshrc`)

- Running
  - `/usr/lib64/openmpi/bin/mpirun --prefix /usr/lib64/openmpi -mca orte_base_help_aggregate 0 -mca btl_openib_warn_default_gid_prefix 0 -np 1024 -hostfile mpi.hosts <app>`

- Remember to assemble the hostfile corresponding to your reservation (addon to booking?)
- Start the jobs preferably on it-hpc-wgs
- Where to store data?
HPC Cluster – Using storage

> AFS

- Secure, low volume storage.
- TSM Backup possible
- Global File System. Accessible from everywhere
- Good for final results.
- ACL for fine grade right management
- Keep in mind: no token extension, difficult in long running jobs (> 24h)
- Read and Write speed problems
HPC Cluster – Using storage

> NFS (ver. 3/4)

- /data/netapp and /data/netapp4
- 20 TB at the moment without quota
- Secure storage
- Good for long term storage
  - larger volumes of final results
  - Store group specific software
  - Fast user/group home directories
- Per default not accessible from outside it-hpc
- Backup possible
- Cooperative storage
HPC Cluster – Using storage

> dCache

- High performance distributed storage
- Possible connection to tertiary Storage Systems (Tape robot)
- Currently not mounted, but could be mounted via NFS4.1
- Let us know if that's a requirement
HPC Cluster – Using storage

> FHGFS

- /data/fhgfs
- 72TB without quota
- Low latency, fast read and write
- Data transfer via Infiniband
- Good for short term temporary storage.
  - Interprocess communication
  - Staging data files
- No export outside the it-hpc cluster
- No backups, no protection
- Cooperative storage (otherwise we would delete chronology)
HPC Cluster – Using storage

> FHGFS Key benefits

- **Distributed File Contents and Metadata**
  - strict avoidance of architectural bottle necks.
  - distribution of file system metadata (e.g. directory information)

- **HPC Technologies**
  - scalable multithreaded core components
  - native Infiniband support

- **Easy to use**
  - FhGFS requires no kernel patches
  - graphical cluster installation and management tools

- **Increased Coherency** (compared to simple remote file systems like NFS)
  - changes to a file are always immediately visible to all clients.
HPC Cluster – Running mathematica

> Licenses available
  - 440 SubKernel licenses occupying a single main-license
  - Usually a substantial number of licenses available

> Mathematica lightweight-grid
  - Requires startup of tomcat server
  - Currently not an option on the cluster

> Parallel kernel configuration
  - Easily done from within the application
  - Unmanaged
  - Requires some care selecting the number of kernels per host.
Mathematica benchmark with 128 cores ...
> Mathematica benchmarks

- Scales well up to 32-bit cores, gets worse beyond that
  - But true for other architectures as well

- Mathematica has troubles acquiring much more than 50 cores
- Using up to 32-48 cores per node should be fine.
HPC Cluster – Running matlab

> New: Matlab Distributed Computing Server

- 128 kernel licenses → 128 core job with a single matlab / distributed computing toolbox

> Currently need to start the matlab scheduler as root

- No automatic startup yet. If frequently used, we could add that to the web-calendar.
> Once the scheduler is running, job submission from a desktop is easy:

```matlab
>> myCluster = parcluster; matlabpool(myCluster);
>> myrun = matlabpool('open','HPC','64','AttachedFiles',{nb1.m,nb2.m});
```
HPC Cluster – Running matlab

Matlab benchmark with 2*32 cores

Speedup based on job wait time compared to one task
GPU computing - workshops

> PNI-HDRI workshop on (parallel) Computing for Photons and Neutrons
  ▪ 04-05.03.2013
  ▪ See: https://indico.desy.de/event/7333

> Terascale Alliance Workshop on GPU computing
  ▪ 15-16.04.2013
  ▪ See: https://indico.desy.de/event/7143

> Matlab Advanced Tutorial
  ▪ In preparation

> Anything else (mpi, c++)?
  ▪ it-training@desy.de
HPC Cluster – Information

> Web:
  - http://it.desy.de/services/computing_infrastructure
  - http://it-hpc-web.desy.de (only DESY-internal)

> Contact:
  - uco@desy.de
  - unix@desy.de

> Announcements
  - Subscribe to it-hpc-user@desy.de

Questions? Comments?