Self service for software development tools

Michal Husejko, behalf of colleagues in CERN IT/PES

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Self service for software development tools & Scalable HPC at CERN

Michal Husejko, behalf of colleagues in CERN IT/PES

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Agenda

CERN IT Department

- Self service portal for software development tools
- Scalable HPC at CERN
- Next steps

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Services for (computing) projects

- A place to store code and configuration files under revision control
- A place to document development and to provide instructions for users
- A place to track bugs and plan new features
  - Other services such as build and testing frameworks, but this is outside the scope of this talk

Version Control Services at CERN

CERN IT Department

- SVN: Still the main version control system. 2100 repositories, over 50000 commits per month (SVN Statistics)
- GIT: Available as a service since spring, about 700 repositories

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Collaborative Web documentation

CERN IT Department

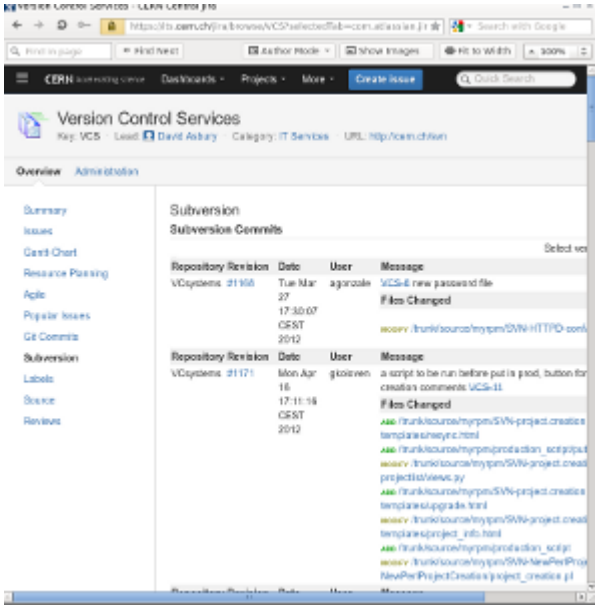
- Typically, TWiki for web documentation
  - In use at CERN since 2003, ~10000 users
  - Documentation divided into project “Webs” and “Topics”
  - Currently running TWiki 5.1
  - Many plugins and active user community
- Could also be on Drupal or another web site for the project
  - A link provides flexibility

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

JIRA issue tracking service

CERN IT Department

- Central JIRA instance
  - CERN : SSO, eGroups, Service-now link
  - Plugins : Git, SVN, Agile, Issue Collector, Gantt charts
  - 165 projects and ~2000 users of central instance, growing (10 projects/week)
  - More users in the other JIRA instances, that have been migrated to the central Issue Tracking infrastructure
  - Savannah migration to JIRA on-going (e.g. ROOT migrated)

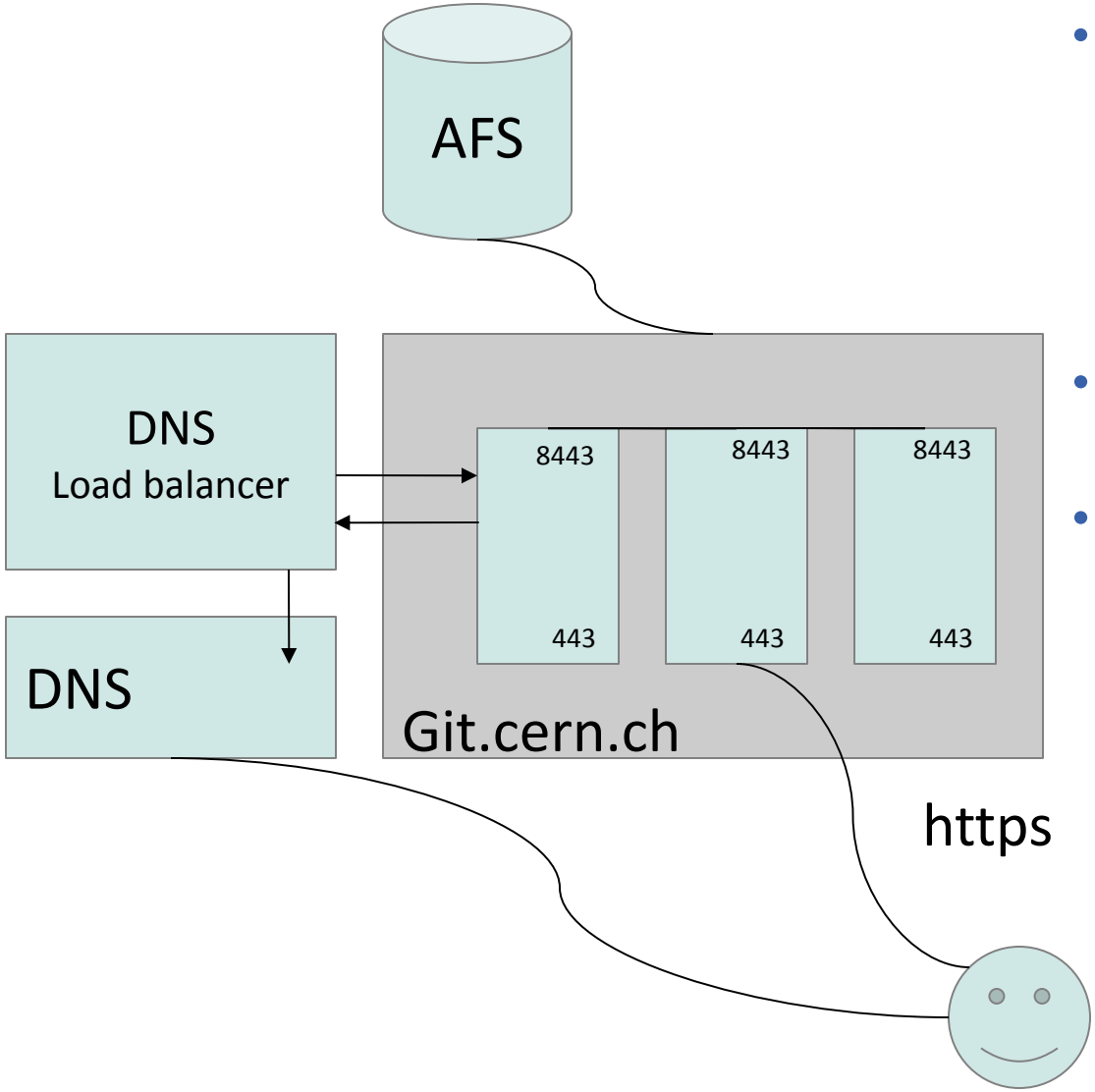


CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

The case for automation

- Creation of a project on a self-service basis is easy for the user and avoids a repetitive task for support teams.
- Administrators of projects would benefit if they can set e.g. authorization rules for multiple tools at the same time
- Certain settings of a Git or SVN projects are beyond the scope of a project administrator, if an intervention requires access to server infrastructure
- JIRA project administrators have restricted rights to change their configuration. Providing more freedom would be desirable.

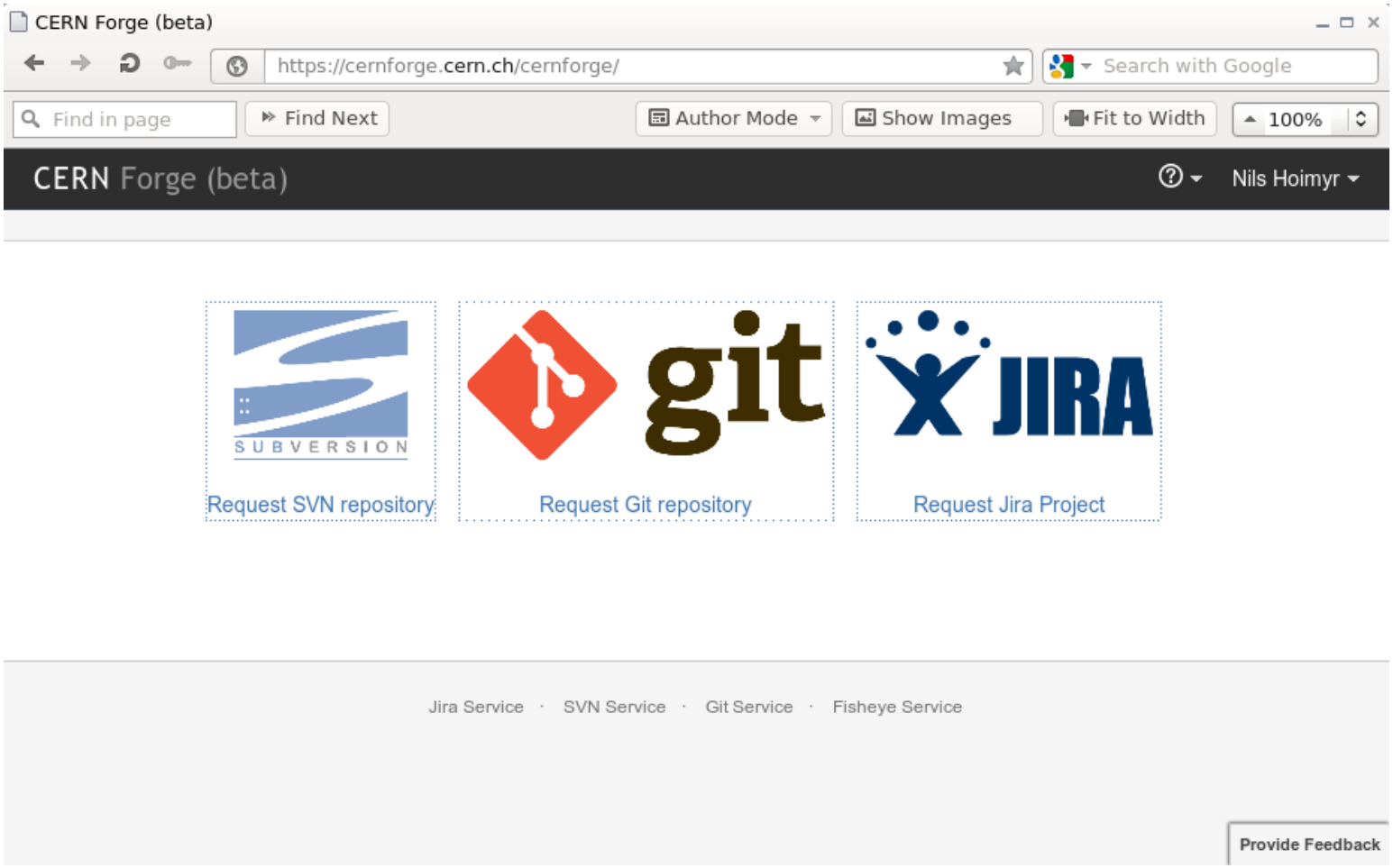
Example: Git service



- Gitolite
  - Permissions, integrated with e-groups
  - World or Cern visibility option
- Gitweb
  - Browser access
- Infrastructure
  - Puppet
  - DNS LB
  - Apache LB
  - HTTP
  - AFS (FS agnostic)

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

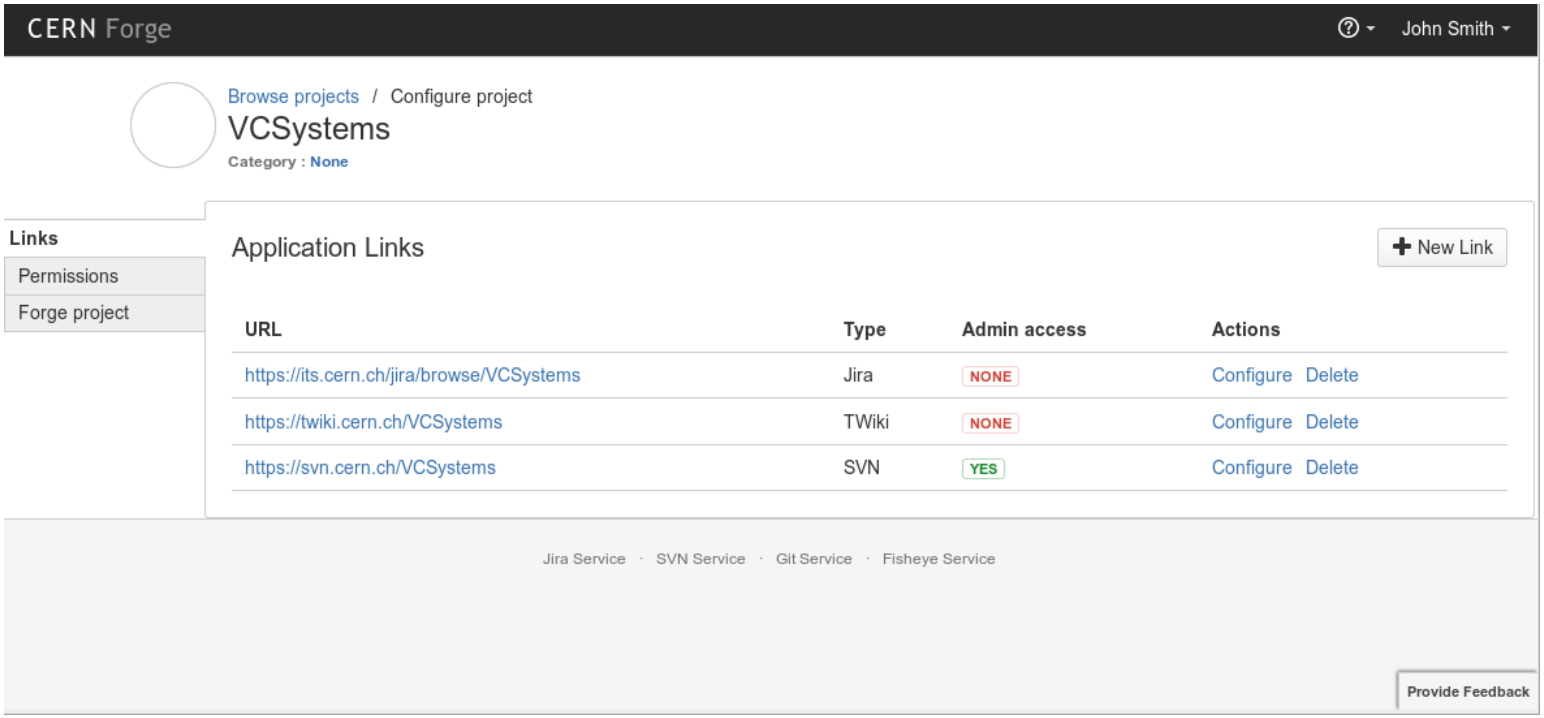
CERN Forge admin portal



- Django Web interface, with backend DB for service metadata
- Use REST API to JIRA and other services for administrative actions

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

CERN Forge admin portal - 2



In the future, users will be able to configure links between VCS and Issue tracking, as well as other settings (public/private etc)

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

More in the next chapter!

Stay tuned for updates on
http://cernforge.cern.ch

Over to another challenge where we are working towards on demand flexibility:

Scalable High Performance Computing

CERN IT Department
CH-1211 Genève 23
Switzerland
www.cern.ch/it

Scalable HPC (1/2)

- Some 95% of our applications are served well with bread-and-butter machines
- We (CERN IT) have invested heavily in AI including layered approach to responsibilities, virtualization, private cloud.
- There are certain applications, traditionally called HPC applications, which have different requirements
- Even though these applications sail under common HPC name, they are different and have different requirements
- These applications need detailed requirements analysis

Scalable HPC (2/2)

- We contacted our user community and started to gather continuously user requirements
- We have started detailed system analysis of our HPC applications to gain knowledge of their behavior.
- In this talk I would like to present the progress and the next steps
- At a later stage, we will look how the HPC requirements can fit into the IT infrastructure

CERN HPC applications

- Engineering applications:
  - Used at CERN in different departments to model and design parts of the LHC machine.
  - Tools used for: structural analysis, fluid dynamics, electromagnetics, and recently multiphysics
  - Major commercial tools: Ansys, Fluent, HFSS, Comsol, CST
    - but also open source: OpenFOAM (fluid dynamics)
- Physics simulation applications
  - HEP community developed simulation applications for theory and accelerator physics

One of many eng. cases …

- Ansys Mechanical
  - LINAC4 beam dump system, single cycle simulation
- Time to obtain a single cycle solution:
  - 8 cores -> 63 hours to finish simulation
  - 64 cores -> 17 hours to finish simulation
- User interested in 50 cycles: would need 130 days on 8 cores, or 31 days on 64 cores

- It is impossible to obtain simulation results for this case in a reasonable time on a standard user engineering workstation

Challenges

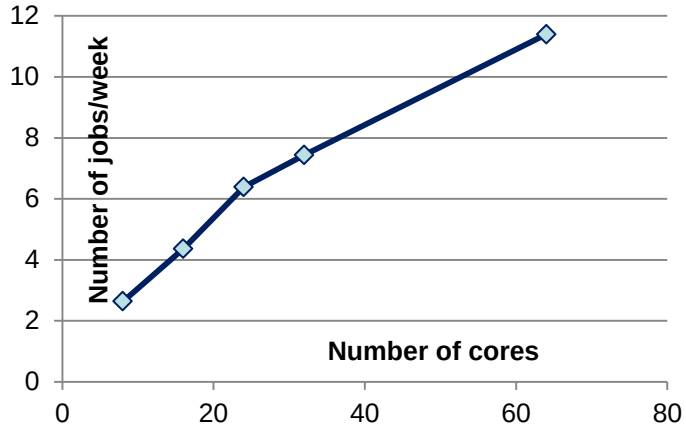
- Why do we care ?
- Challenges
  - Problem size and its complexity are challenging our users' workstations
  - This can be extrapolated to other Engineering HPC applications
- How to solve the problem ?
  - Can we use current infrastructure ?
  - … or do we need something completely new ?
  - … and if something new, how this could fit into our IT infrastructure
  - We are running a **Data Center** and not a Super Computing Center

Current infrastructure

- CERN lxbatch – Optimized for HEP data processing
  - Definitely not designed with MPI applications in mind
- Example lxbatch node:
  - Network: mostly Ethernet 1 Gb/s, rarely 10 Gb/s (very limited number with low latency)
  - 8-48 cores, 3-4 GB RAM per core
  - Local disk, AFS or EOS/CASTOR HEP storage
- Can current lxbatch service provide “good-enough” HPC capability?
  - How interconnect affects performance (MPI based distributed computing)
  - How much RAM per core
  - Type of temporary storage
  - Can we utilize multicore CPUs to decrease time to solution (increase jobs/week) ?

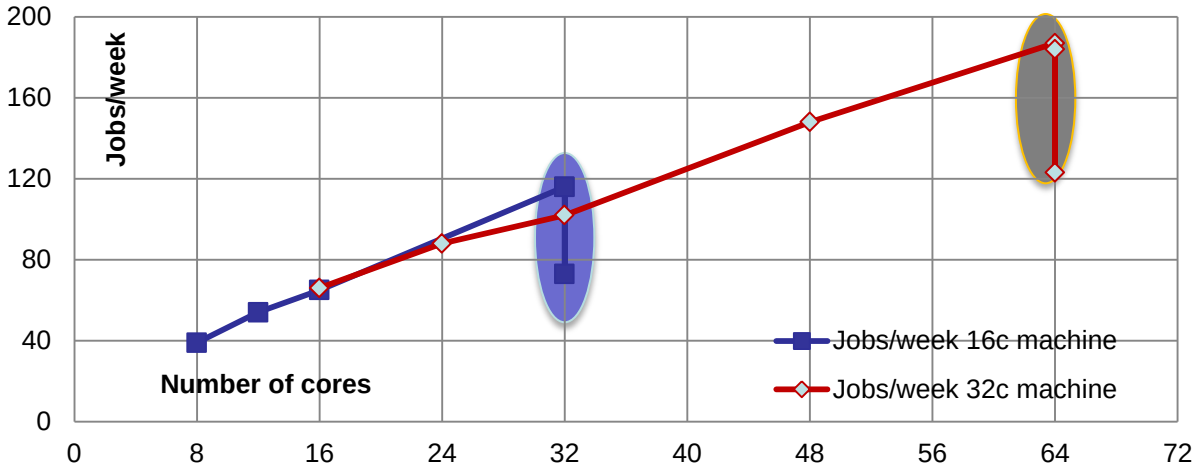
Multi-core scalability

- We know that some of our tools have a good scalability (example Ansys Fluent)
- How about other, heavily used at CERN (example Ansys Mechanical)?
- One of many test cases: LINAC4 beam dump system, single cycle simulation – results:
  - Scales well beyond single multi-core box.
  - Greatly improved number of jobs/week, or simulation cycles/week
- Conclusion
  - Multi-core distrbuted platforms needed to finish simulation in reasonable time



Interconnect latency impact

- Ansys Fluent – Commercial CFD application
- Speedup beyond single node can be diminished because of high latency interconnect.
  - The graph shows good scalability for 10 Gb low latency beyond single box, and dips in performance when switched to 1 Gb MPI
- Conclusion:
  - Currently 99 % of nodes in CERN batch system are equipped with 1 Gb/s NIC
  - Low latency interconnect solution needed.

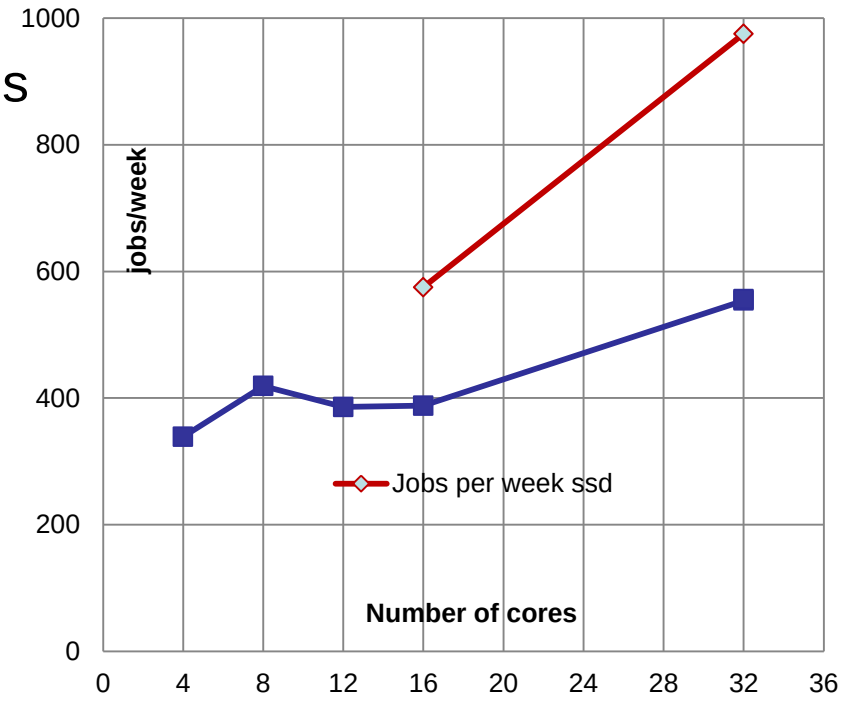


Memory requirements

- In-core/out-core simulations (**avoiding costly file I/O**)
  - In-core = most of temporary data is stored in the RAM (still can write to disk during simulation)
  - Out-of-core = uses files on file system to store temporary data.
  - Preferable mode is in-core to avoid costly disk I/O accesses, but this requires increased RAM memory and its bandwidth
- Ansys Mechanical (and some other engineering applications) **has limited scalability**
  - Depends heavily on solver and user problem
  - Limits possibility of problem splitting among multiple nodes
- All commercial engineering application use some **licensing scheme**, which can put skew on choice of a platform

Impact of HDD IOPS on performance

- Temporary storage ?
- Test case: Ansys Mechanical, BE CLIC test system
- Disk I/O impact on speedup. Two configurations compared.
  - Using SSD (better IOPS) instead of HDD increases jobs/week almost by 100 %
- Conclusion:
  - We need to investigate more cases to see if this is a marginal case or something more common



Analysis done so far for engineering applications

- Conclusions
  - More RAM needed for in-core mode, this seems to solve potential problem of disk I/O access times.
  - Multicore machines needed to decrease time to solution
  - Low latency interconnect needed for scalability beyond single node, which by itself is needed to decrease simulation times even further.
- Next steps:
  - Perform scalability analysis on many-node clusters
  - Compare low latency 10 Gb/s Ethernet with Infiniband Cluster for all CERN HPC Engineering applications

Last but not least

- If low latency interconnect then … Ethernet or Infiniband ?

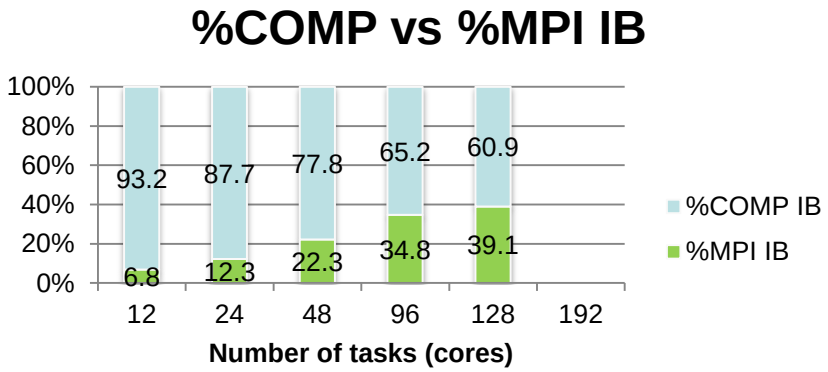
Lattice QCD

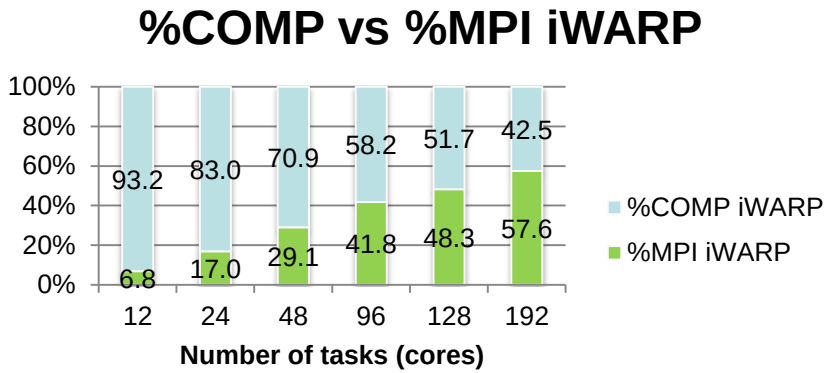
- Lattice QCD:
  - Highly parallelized MPI application
- Main objective is to investigate:
  - Impact of interconnection network on system level performance (comparison of 10 Gb Ethernet iWARP and Infiniband QDR)
  - Scalability of clusters with different interconnect
  - Is Ethernet (iWARP) “good enough” for MPI heavy applications ?

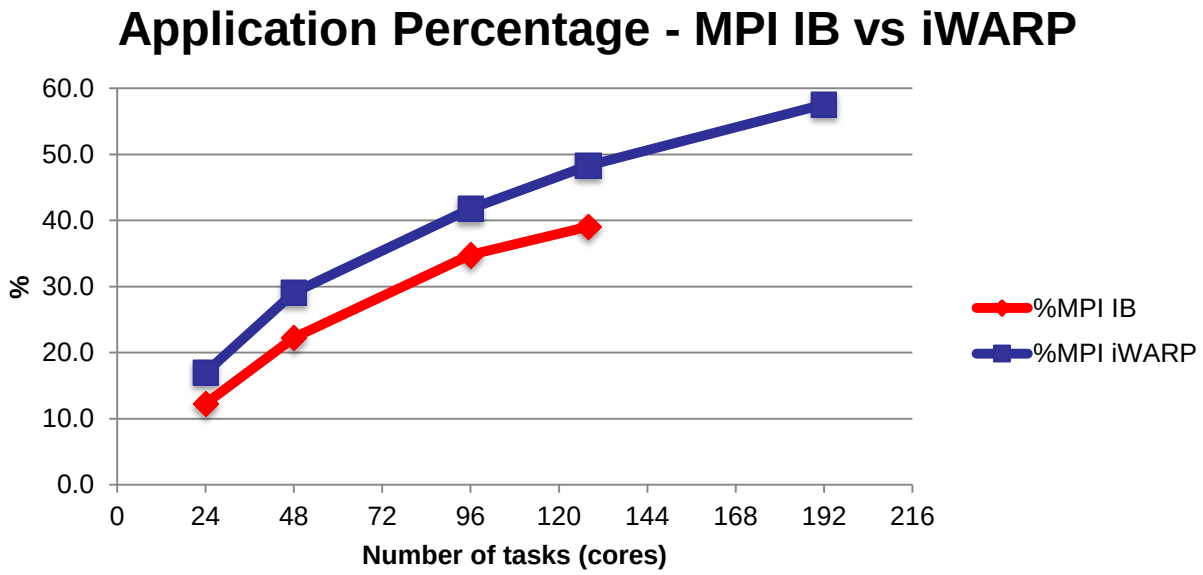
IB QDR vs. iWARP 10 Gb/s

- Two clusters, same compute nodes, same BIOS settings.
  - Dual Socket, Xeon SandyBridge E5-2630L, 12 cores total per compute node
  - 64 GB RAM @ 1333 MT/s per compute node
- Different Interconnect networks
  - Qlogic (Intel) Infiniband QDR (40 Gb/s)
    - 12 compute nodes + 1 frontend node (max. 128 cores)
  - NetEffect (Intel) iWARP Ethernet (10 Gb/s) + HP 10Gb low latency switch
    - 16 compute nodes + 1 frontend node (max. 192 cores)

IB QDR .vs iWARP 10 Gb/s (1/3)

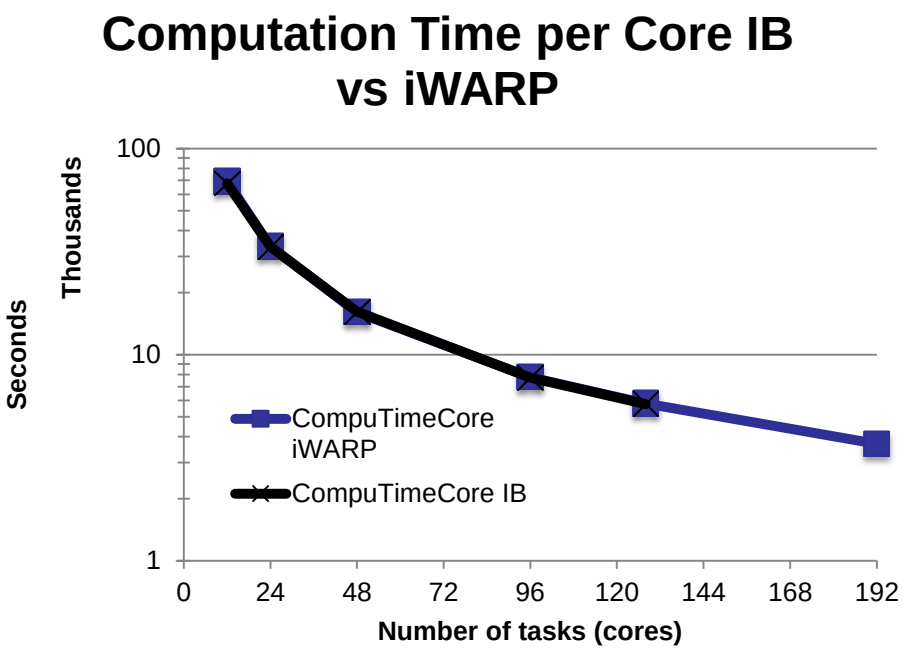


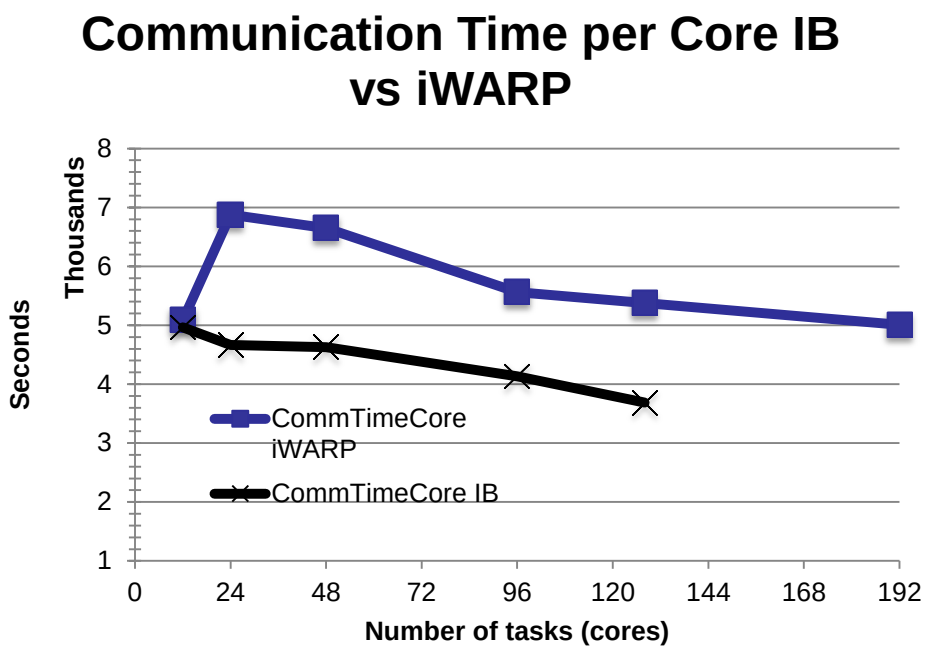




IB QDR .vs iWARP 10 Gb/s (2/3)

- Less is better

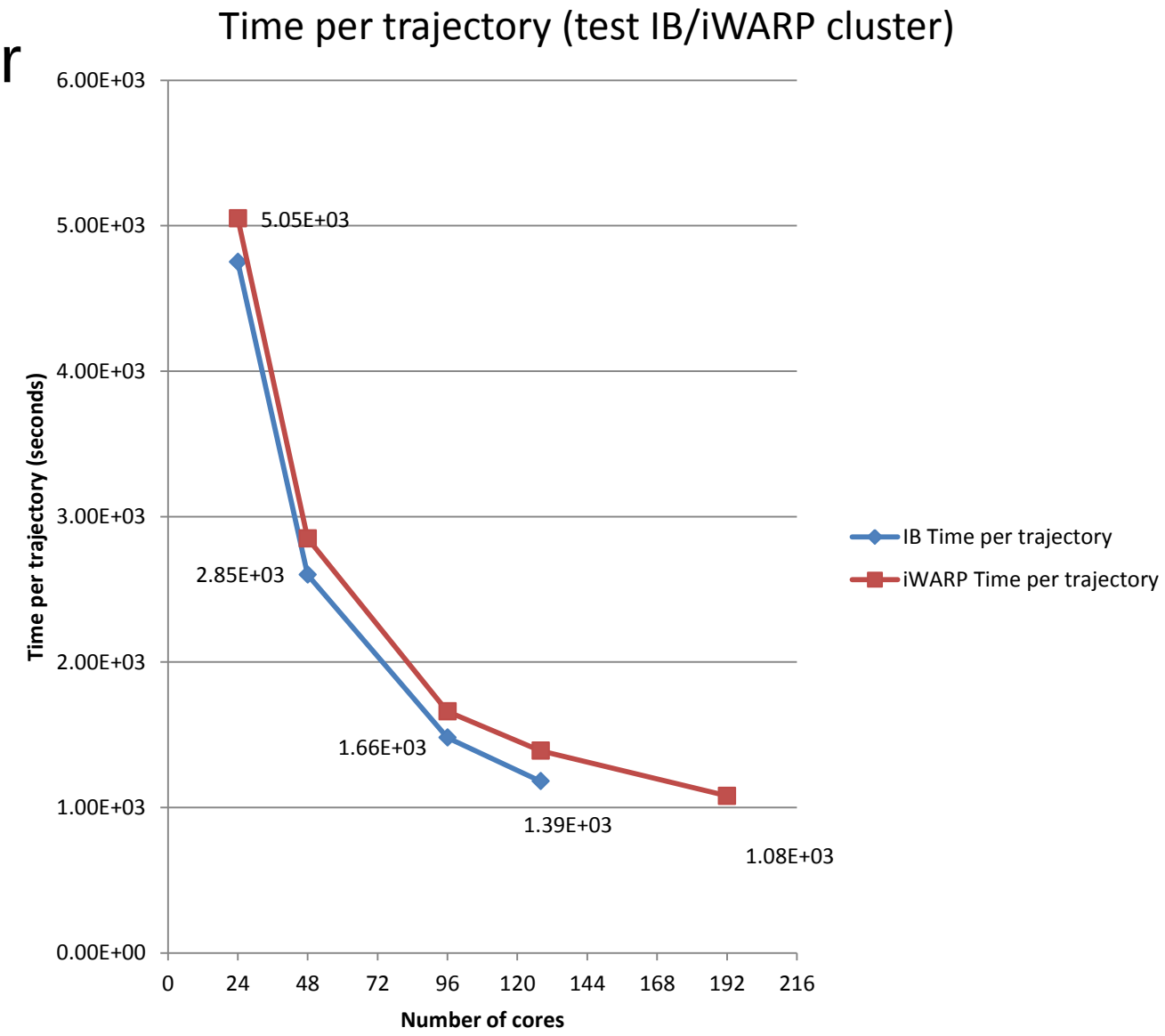




IB QDR .vs iWARP 10 Gb/s (3/3)

CERN IT Department

- Less is better



Conclusions and outlooks

- Activity started to better understand requirements of CERN HPC applications
- Performed first steps to measure performance of CERN HPC applications on Ethernet (iWARP) based cluster – results are encouraging
- Next steps are:
  - Refine our approach and our scripts to work at higher scale (next target is 40-60 nodes) and with real-time monitoring
  - Compare results between Sandy Bridge 2 socket system with SB 4 socket system – both iWARP
  - Gain more knowledge about impact of Ethernet interconnect network and tuning parameters on MPI jobs
  - Investigate impact of virtualization (KVM, Hyper-V) on latency and bandwidth for low latency iWARP NIC.

- Q&A