Software Evolution - HSF

Liz Sexton-Kennedy for the Startup Team
WLCG Collaboration Workshop, Okinawa, 12-April-2015

# The Big Picture for HEP in the 10 Year Timescale

- Ultimately we need to guarantee that we can do desired physics at scale and at a cost that funding agencies can pay.

- What we are driven by:

  - The physics objectives of our experiments and the timeline of data-taking

  - Evolution of technologies we use: hardware, operating systems, Grid/Cloud, compilers, etc.

  - Resource limitations or limitations imposed by resource providers
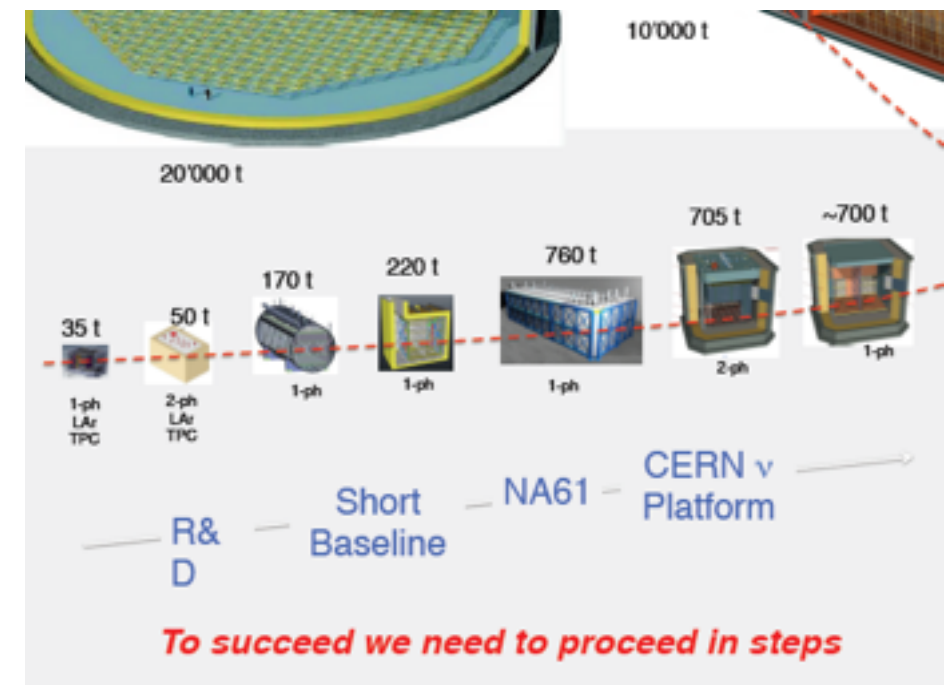
High Granularity Detectors

- The picture on the previous page is an example of what I think will be one of the software and computing challenges 10 years from now.

  - ILD

  - Neutrino LAr detectors

  - Maybe CMS…
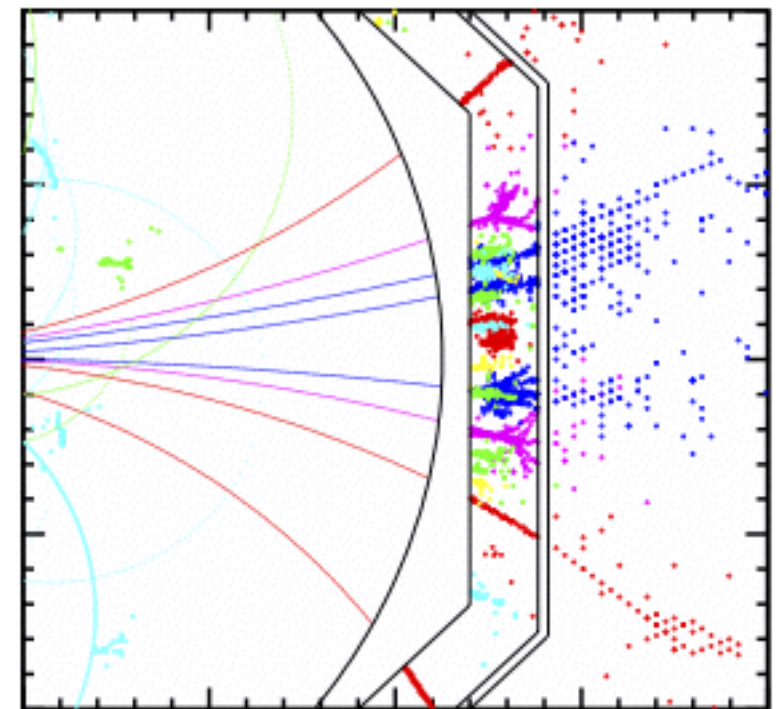
Timelines
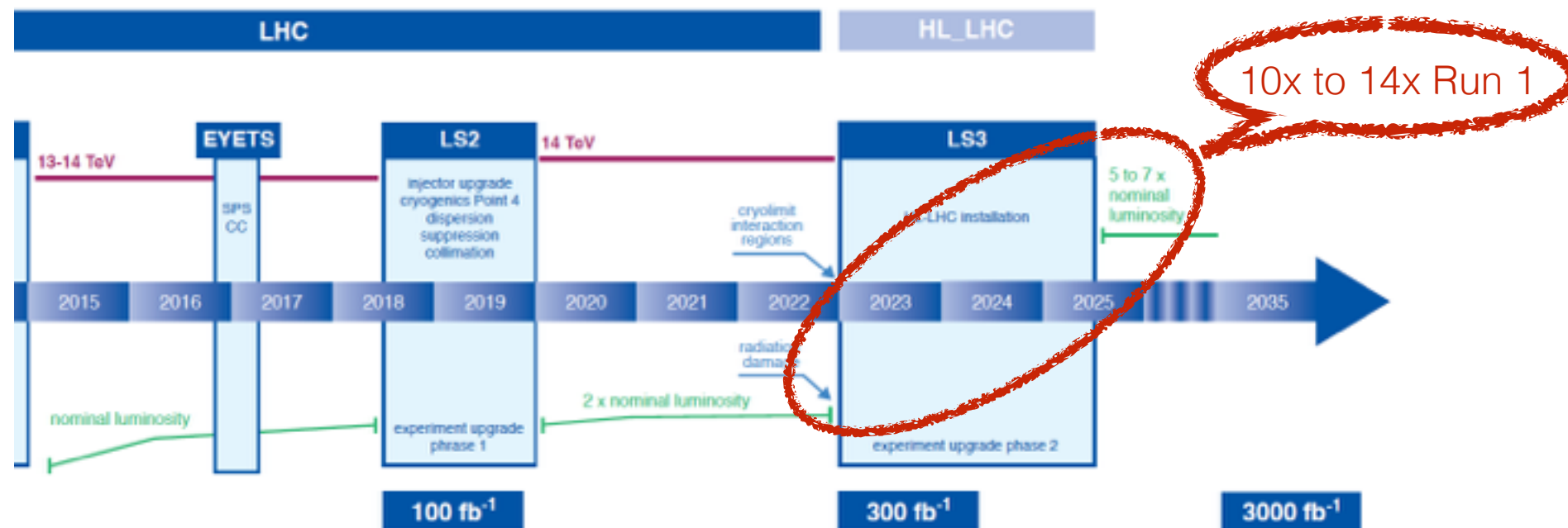
- The above is a graphic of the LHC timeline

- The DUNE timeline builds on the short baseline program and prototypes, 2025?:

- ILC? HIGs factory?

The Later 2020s

- How, where, and by whom will our computing resources be provided?

- We all know that LHC is a global community.

- In 2020s ALL experiments will be global (LHC, DUNE, ILC, etc.).

- A global collaboration is a challenge of it's own.

- How will our software development model work? Which pieces of software can we take from elsewhere? What has to be experiment specific? What can be common?
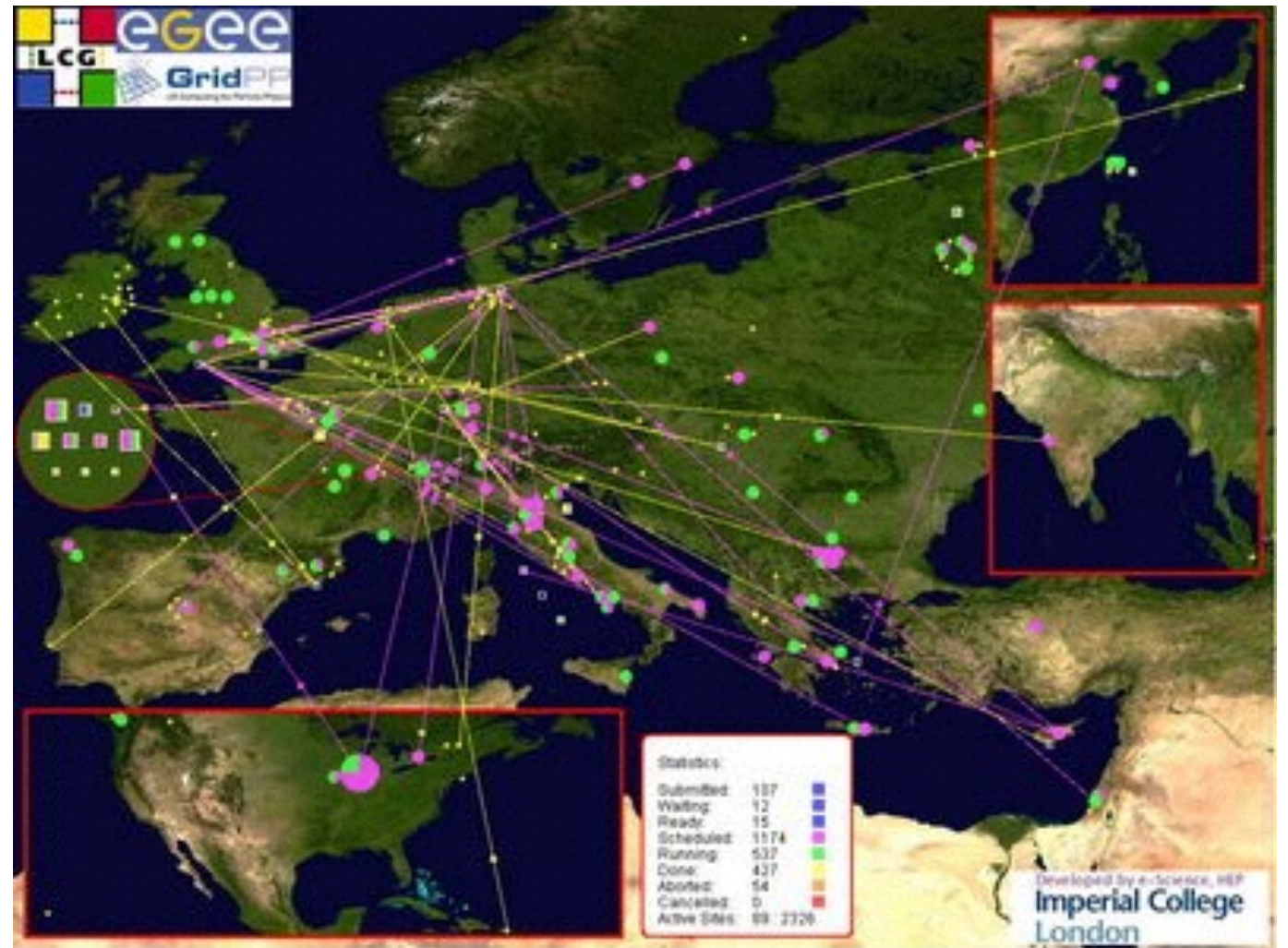
Computing Technology
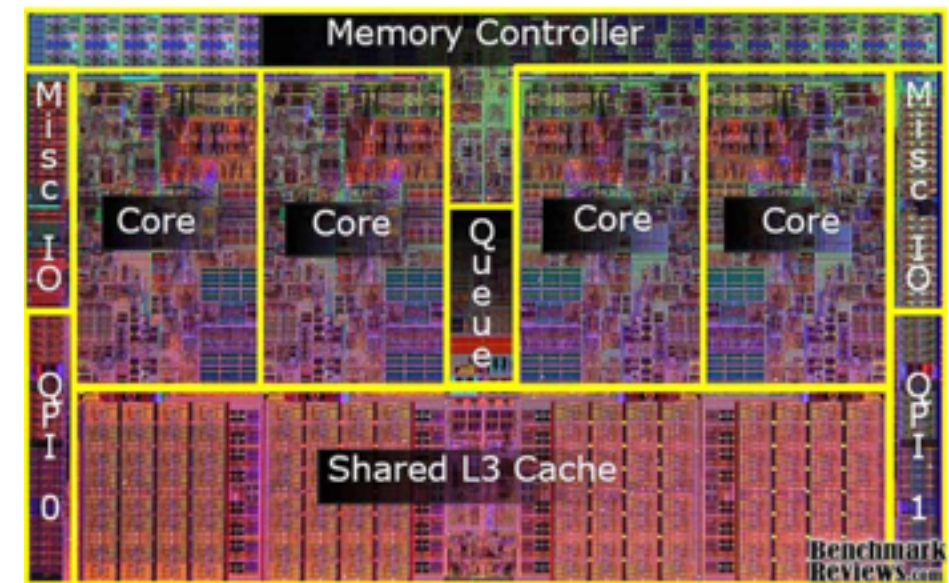
- Will we be able to use the technologies available at the time we need them?

- Can we coordinate better between the Software and Computing sides of our community?
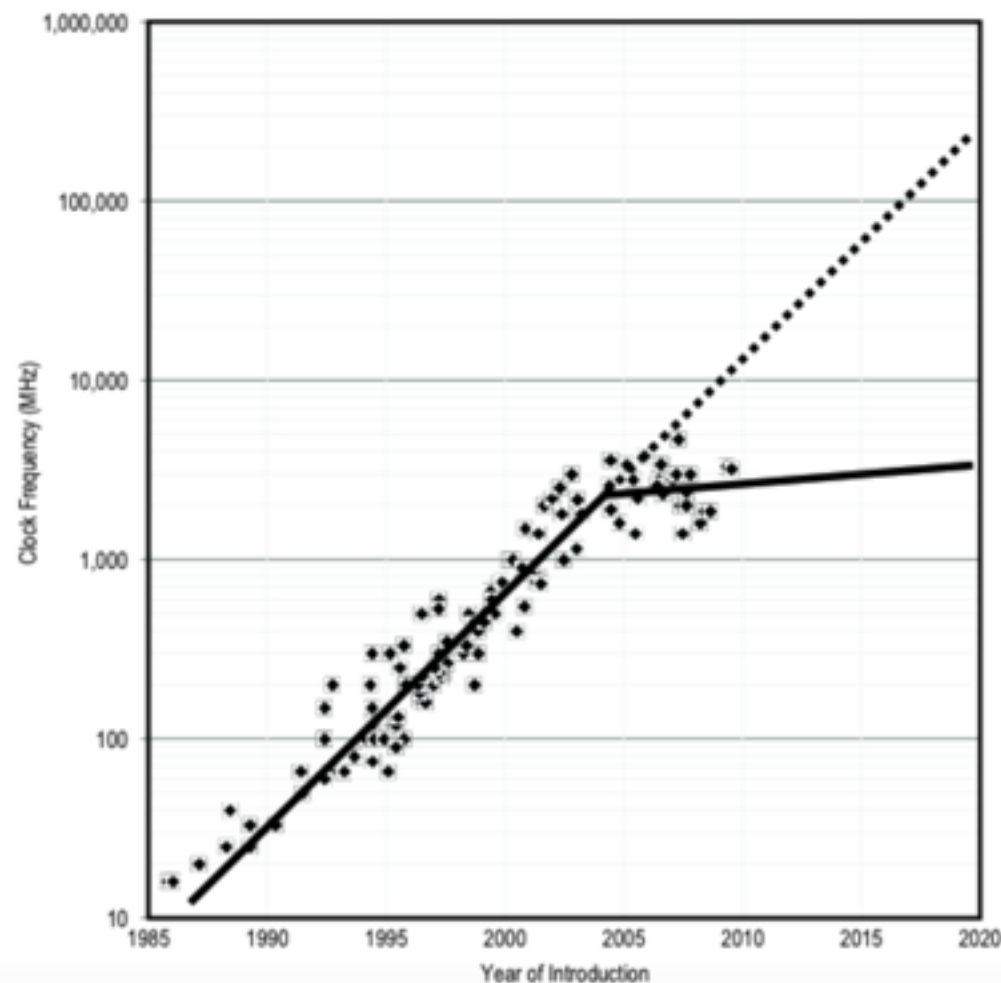
Multicore was the initial solution, but it just buys time. The same power limits still exist and are driving the introduction of newer architectures.

From: "The Future of Computing Performance: Game Over or Next Level?"

Resource Limitations

CHF/HS06

Price/performance evolution of installed CPU servers

improvement/year
- 15%
- 30%
- 20%
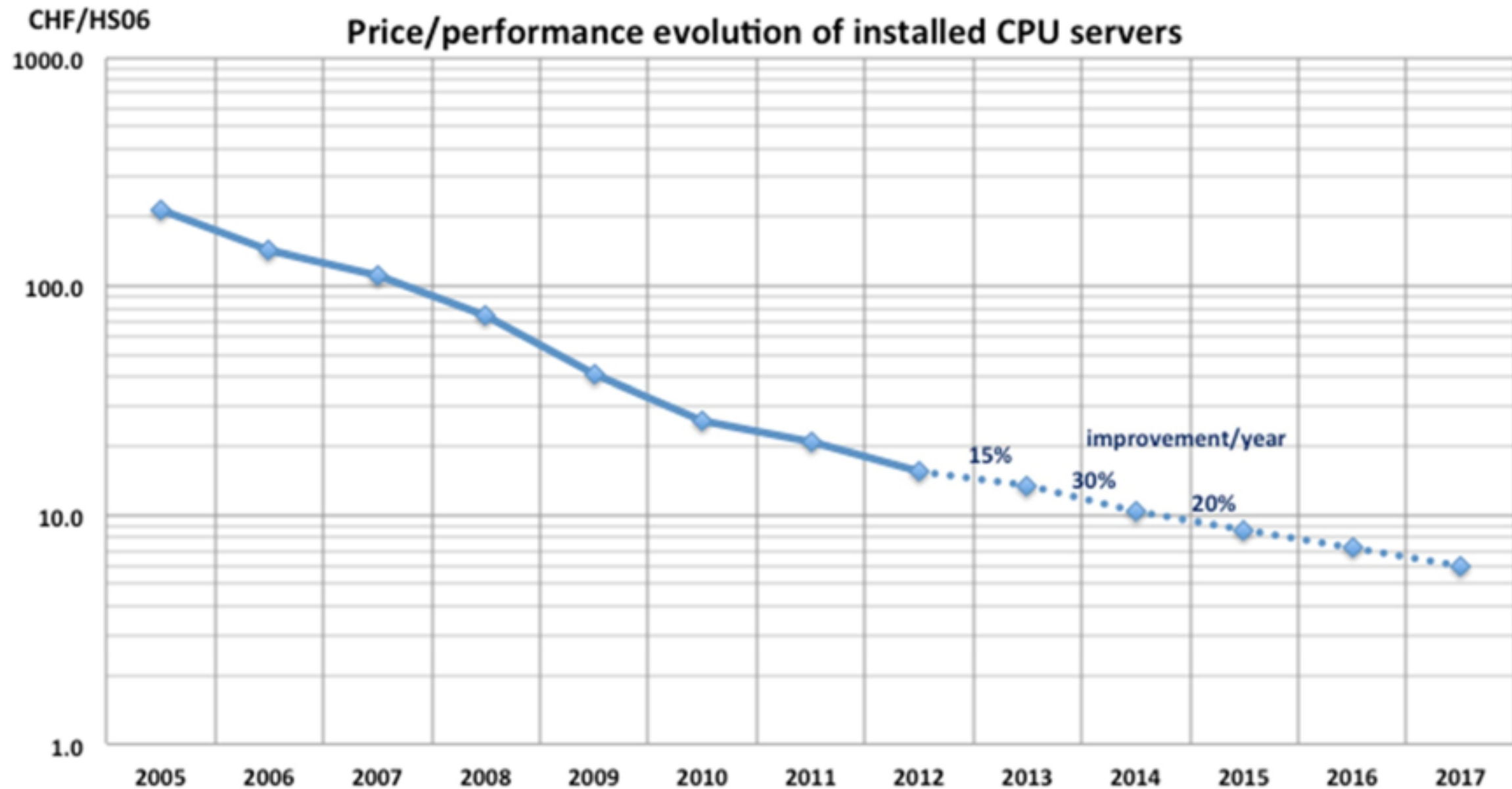
Bernd-Panzer Steindel (CERN IT market trend 2012)

- Will we be able to buy ourselves out of this problem? probably not…

7

Why HSF?

- The challenges are large, but why do we think the HSF can help?

- It's a mechanism to facilitate coordination and common efforts in HEP software and computing.

- We need to exploit all the expertise available in our community, and outside it, to meet these challenges.

- The affordable way to do it, is collaboratively.

2015 Status

- The current HEP software generation is > 20 years old

- Due to the previously illustrated technology trends we need:

  - Paradigm shift due to many-core from serial to parallel execution

  - New resources such as HPC, (which already require the above).

  - commercial clouds, etc waiting to be exploited.

- New software needs to interoperate with other software - inside and outside our community

- We need to attract new people with the right skills. The Concurrency forum was a nice example how this might work

What's Happened

- In April of 2014 a brain storming workshop was organized at CERN; Outcome:

  - Strong negative reaction to top down organization and governance.

  - However lots of agreements in principle that this was a necessary and useful initiative.

  - Groups were asked to self-identify and write down their thoughts.

  - 10 white papers were submitted with ideas for scope, goals, formation process, governing models, etc. .

What's Happened 2

- An all volunteer "Interim Foundation Board" was formed. It's like a general assembly, receives startup team digests on a monthly basis.

- Torre Wenaus and Pere Mato agreed to organize a startup team of more volunteers.

  - People agreeing to spend part of their time to bootstrap the HSF

  - Fall 2014: synthesis of white papers + proposal of a HSF startup plan

  - http://hepsoftwarefoundation.org/sites/default/files/HSFwhitepaperanalysisandstartupplanV1.1.pdf

  - Made contact with HNEP-Astro communities not at the spring workshop

  - Preparation of a HSF "kick-off" workshop at SLAC, January 2015

Agreed HSF Goals

- Share expertise

- Raise awareness of existing software and solutions

- Catalyze new common projects, create an incubator

- Promote commonality and collaboration in new developments to make the most of limited resources

- Aid developers and users in discovering, using and sustaining common software

- Support training career development for software and computing specialists

- Provide a framework for attracting effort and support to S&C common projects

- Provide a structure for the community to set priorities and goals for the work

- Facilitate wider connections; while the HSF is a HEP community effort, it should be open enough to form the basis for collaboration with other sciences

Goals at SLAC

- Hear from small and large projects what HSF could bring to them and what they can bring to it

- Hear from other similar initiatives in different contexts like Apache SW Foundation

- Hear from experiments, science communities and individuals Hear views from funding agencies

- Discuss related initiatives that could benefit from HSF or could be launched under the HSF umbrella

- Set up the relative priorities between the possible activities/ services for HSF and ensure we have the required resources committed by participants and funding agencies

SLAC 2

- As you can see this was a success:

  - http://indico.cern.ch/event/357737/

  - <Add workshop summary link here>

- 10 sessions, 44 presentations 80 local and ~20 external participants

- Good non-EU participation: mainly US but also Asian

- Many non-HEP and IF experiments were represented: Dayabay, LSST, Photon science.

- Decided to adopt the Apache model and start doing.

Learning from Others

- Very interesting and useful session

- Apache Software Foundation

  - Goal similar to ours: umbrella for related projects, no central management of projects, they remain autonomous

  - Difference: ASF started before most projects, invented the model when developing

  - Do-ocracy: no long-term planning, active people have their say

  - Darwinian approach: ASF provides an infrastructure for projects, users decide the projects that will survive by their adoption

  - ASF focuses on providing an incubator for new projects and on ensuring the project sustainability. Avoids projects bound to 1 individual (hit-by-the-bus problem!)

  - Transparency is essential: a pillar of ASF culture

Learning from Others 2

- Dan Katz on Building Scientific Software communities: a very nice summary on lessons learnt from successful and failed projects

  - Avoid too much planning, try-and-fail is the most productive approach

  - Governance: flat layer of peers generally better than benevolent dictator to create a community: forces to work together

  - Make it easy for people to contribute, with little time and effort

  - Give credit for every work done, motivate people

  - Get people involved rather than having them reinventing the wheel

- Software Sustainability Institute (UK) - Neil Chue Hong

  - Helps SW projects to address sustainability, great focus on training

  - Same message as D. Katz, insistence on not designing the perfect HSF

  - Lobbying/communication about career path for Research SW Engineers

Community and Project Views

- Every community and project mentioned that HSF could help in some ways, lots of repeated requests.

- No real conflicting view but each with a different focus.

- Experiments: SW knowledge base to increase SW reuse, consultancy for new projects, SWAT teams, value of discussion fora and exchange, continuous integration, better training, licensing

- Projects: technical forums, help in organizing technical discussions with other projects, help in organizing meetings with users, build/test infrastructure for smaller projects, licensing

- Common SW or expertise: avoid to reinvent the wheel (example with HPC), help with convergence and sustainability (pyroot/rootpy)

Time to "do"

Working Group	Objectives	Forum - Mailing list
Training	Organization of training and education, learning from similar initiatives	hep-sf-training-wg
Software Packaging	Package building and deployment, runtime and virtual environments	hep-sf-packaging-wg
Software Licensing	Recommendation for HSF licence(s)	hep-sf-tech-forum
Software Projects	Define incubator and other project membership or association levels. Developing templates	hep-sf-tech-forum
Development tools and services	Access to build, test, integration services and development tools	hep-sf-tech-forum
Communication and information exchange	Address communication issues and building the knowledge base Technical notes	hep-sf-tech-forum

Areas considered high priority coming out of the workshop.
The next set of slides will expand on each of these.

Training WG

- Consensus that it should be one of the initial HSF focus

  - Several volunteers... and several existing initiatives being leveraged in the HSF context

  - Should cover different kind of training from "traditional schools" to virtual seminars or webinars

- Learn from the Software Carpentry's very successful experience

  - Launched by Sustainable Software Institute. A very valuable input from the workshop

- A working group created and active, led by G. Stewart

  - Volunteers welcome to join

  - hep-sf-training-wg+subscribe@googlegroups.com

Software Packaging WG

- Package Building and Deployment

  - Runtime Environment / Virtual environments

  - Role of new technologies like Docker

  - Define a build 'protocol' to orchestrate the combination of various open software projects into combined builds

  - Allow adoption by well established projects, like Geant, and users, like experiments.

- Created an HSF organization at GitHub with an active "issue" discussion. https://github.com/HEP-SF/packaging/issues/1

- I've agreed to lead this one…

Software Licensing WG

- An open-source license is mandatory to participate to HSF.

- Many SW projects in the community without an explicit license. Apache people recommend use of one existing open-source license. It defines the foundation.

- Need to start with what exists…

  - Building on CERN's investigations: CERN study available on the web site

- Aim for first conclusions/recommendations by the summer

Software Projects WG

- Experiment with the incubator idea with a few guinea pig projects.

  - What services are needed to support projects? (what they would benefit from HSF)

  - What they can bring to the community?

  - Which are the criteria to move out of the incubator: avoid too high thresholds, remain pragmatic

- Experiment with the inter-project relationships under HSF umbrella

- What kind of membership or affiliation should there be for projects?

- Develop templates for new projects

- A few projects declared their interest: rootpy, Gaudi, Vac/Vcycle,

Development Tools WG

- Will not do anything already provided by others. For example, project hosting: use GitHub

- Several labs with hardware and software resources that could be shared and benefit other projects

  - E.g. CERN TechLab, FNAL…

  - Currently discussing how to make it possible for HSF projects to get access to them

- Benedikt is leading this one.

Communication & Information Exchange WG

- SW Knowledge Base - Think of it as a Yelp! for HEP software

- Started at http://hepsoftwarefoundation.org

- May migrate to a new platform in the future but data will be migrated

- About all the software developed and used in the community

  - Contains a project description with standard items

  - Cross-reference the usage by experiments

- Everybody encouraged to register its favorite software

  - Request an account on the web site

- Everybody is encouraged to write a review… buyers beware model

More Information

- HSF Website

- http:// hepsoftwarefoundation.org

- Main contact point for HSF

- Contains work done so far

- Mailing lists, meeting notes, document repository

Conclusions

- There is a field wide, large challenge ahead of us on the timescale of 10+ years.

- Limitations of power, cooling, and funding will not allow us to keep our current software paradigm, nor our current mode of computing resource provisioning.

- Collaboration on these challenges is key

- The work of the HSF collaboration has begun, come join us!