PROCEEDINGS OF SCIENCE

Computing in High Energy Physics at the time of the Grid

René Brun (eds.)
CERN
1211 Geneva 23, Switzerland
E-mail: Rene.Brun@cern.ch

Federico Carminati (eds.)
CERN
1211 Geneva 23, Switzerland
E-mail: Federico.Carminati@cern.ch

Giuliana Galli Carminati (eds.)
Hôpitaux Universitaire de Genève, Unité de la Psychiatrie du Développement Mental
2 ch du Petit Bel Air, 1225 Chêne-Bourg, Switzerland
E-mail: Giuliana.GalliCarminati@hcuge.ch

During the past forty years High Energy Physics (HEP) computing has faced the challenge of software development by distributed communities, and of exploiting geographically distributed computing resources. HEP computing has been very successful in fulfilling its mandate.

For many aspects, HEP computing has been today where computing was going to be tomorrow. This book, due to appear in the second half of 2009, describes the evolution of HEP computing, and in particular those aspects that have been most innovative.

This book contains contributions from R. Brun, P. Buncic, F. Carminati, F. Furano, G. Galli Carminati, P. Méndez Lorenzo, L. Pinsky, F. Rademakers, L. Robertson, B. Segal and J. Shiers.

XII Advanced Computing and Analysis Techniques in Physics Research
Erice, Italy
3-7 November, 2008

 http://pos.sissa.it

PoS(ACAT08)127

1. Computing in HEP at the time of the Grid

1.1 Introduction

HEP Computing has developed very large program libraries that have been used by generations of physicists over many years on several combinations of hardware platforms and operating systems. New programming languages and paradigms have been introduced, following the evolution and the offer of the IT industry.

The impressive changes in the HEP software in the past 10 years have required a big effort of adaptation from thousands of users. The trend towards larger and larger applications will continue to grow. This will require more discipline to organize the application in a set of libraries that can be dynamically configurable such that the running code in memory is a small fraction of the total code.

A large fraction of the software for the next decade is already in place or shaping up. Core Software requires Open Source and international cooperation to guarantee stability and smooth evolution. Parallelism will become a key parameter. More effort must be invested in software quality, training and education.

One of the major achievements of HEP in terms of IT was the development of the Web [1], a paradigm-shift invention not only for HEP or IT, but for our whole society.

In fact one man, Tim Berners-Lee, invented the Web, not "HEP". So our question should really be re-phrased as: "What was the influence of HEP in leading to the Web's invention?"

The Web's invention, like many other such leaps, was in fact "coincidental". It was certainly not ordered, planned or anticipated in any way by "HEP", by CERN [2], or by TB-L's programme of work there. But for that particular leap to occur, a certain number of pre-conditions were essential and all these existed at that time at CERN.

Code development has always been a central issue in HEP Computing, as geographically distributed communities of programmers with little or no hierarchical relationships have developed the large programs used by HEP to analyse the data coming from the detectors. This activity has been indeed rather successful, as software has hindered no experiment in reaching its physics goals.

In spite of its rather honourable record, HEP computing has never claimed to use, nor has it used on a large scale, traditional Software Engineering methods.

However, some intuitive Software Engineering was applied without ever being formalised. Some of it was quite innovative, and indeed ahead of its time. The failure to recognise this led to a complicated and not very efficient relationship with Software Engineering and software engineers, that still continues today. It is important to analyse what happened in this field to promote a more constructive interaction between software engineering and HEP computing.

1.2 HEP Computing at a glance

Although born in the late 1990s Grid computing burst onto the HEP computing scene in the year 2000. In many ways a continuation and evolution of previous work, HEP Grid

PoS(ACAT08)127

computing succeeded in catalysing not only significant funding but also mobilising a large number of sites to pool resources in a manner that had never been achieved before.

After many years of research and development followed by production deployment and usage by many VOs, worldwide Grids that satisfy the criteria in Ian Foster's "grid checklist" [3] are now an everyday reality for HEP users.

One of the major trends in today's IT is the virtualisation of computing resources. This technique seems to offer a good solution specifically to the problems raised by distributed computing, which aims at integrating heterogeneous resources in a seamless system.

In this context, the virtualisation technology also opens a possibility for user communities and even individual users to create ad-hoc, overlay Grids capable of efficiently running specific applications by deploying a pool of virtual machines on top of physical hardware resources. In doing so, they would create a virtual Grid or virtual cluster that would have a much smaller scale of the physical Grid.

This could give Virtual Organisations and users a complete freedom of choice when it comes to selecting the middleware that is best suited for their applications. At the same time this could enhance security on the resource owner's side by completely isolating user application while running on the site resources.

From the point of view of the computing resources, computing in HEP has always required more computing power than could be provided by a single machine. In this context, the idea of using a group of CPUs or computers in parallel to increase the combined CPU, memory and disk available to a program, is only natural.

A problem that can be easily parallelized in large chunks is called embarrassingly parallel and most HEP processing falls in that category. In theory running an embarrassingly parallel program on multiple machines looks fairly trivial, but the practice is much more complicated.

In some sense, HEP computing has been one of the pioneers of the free distribution of software, which is one of the basis of the Open Source [4] revolution. It is interesting to analyse how the concept of Intellectual Property applies to the new world of software and Internet, in the era of free software.

Let's begin with the concept of property. Normally when we think of property, we think of tangible things like your laptop, or maybe the land your home is on. How does all of this work when the property in question is intangible, like an idea? How does one own and idea? Intellectual Property is intangible, but nonetheless can possess almost the same bundle of rights that tangible property can.

1.3 Outlook

This book describes, via different contributions, the evolution of HEP computing over more than twenty years. There is no doubt that the era described above, and in particular the more recent years, was at times turbulent - both the move to distributed computing and from "FORTRAN to OO" resulted in heated debates and often diametrically opposed opinions.

We have not yet gained sufficient experience in this environment for a fully objective analysis - this must wait another few years, including the onslaught of full LHC [5] data taking and analysis. In spite of this, we can already see how the evolution of computing, in which HEP

Computing has had an important role, is changing our view not only of computing or science, not only of society, but also of ourselves, and leading us to ask questions such as:

- Can we describe our brain as a "thinking machine"?
- Can a theory of the brain as a thinking machine explain, or instruct us about, the evolution of global connection?
- Are thoughts physical objects? Are thoughts matter?
- Are the elements of the Web and the Grid and their interconnections evolving toward a planetary brain?

References

[1] http://en.wikipedia.org/wiki/World_Wide_Web
[2] http://www.cern.ch
[3] I. Foster, *What is the Grid? A Three Point Checklist*, Argonne National Laboratory & University of Chicago, July 20, 2002, http://www.mcs.anl.gov/~itf/Articles/WhatIsTheGrid.pdf
[4] See for instance http://www.opensource.org/ and references therein
[5] http://public.web.cern.ch/public/en/LHC/LHC-en.html