CERN COMPUTER NEWSLETTER

Volume 42, issue 4 September–October 2007

CNL needs you!

Fill out our survey and you could win a *CNL* sweatshirt (see p2)

Contents

Editor Nicole Crémel, CERN IT Department, 1211 Geneva 23, Switzerland. Email cnl.editor@cern.ch. Fax +41 (22) 766 8500. Web cerncourier.com/cws/latest/cnl.

Advisory board Wolfgang von Rüden (head of IT Department), François Grey (IT Communication team leader), Christine Sutton (*CERN Courier* editor), Tim Smith (group leader, User and Document Services).

Produced for CERN by IOP Publishing Dirac House, Temple Back, Bristol BS1 6BE, UK. Tel +44 (0)117 929 7481. Fax +44 (0)117 920 0733. Web iop.org.

Published by CERN IT Department

IOP Publishing

Grid operations get ready for LHC launch

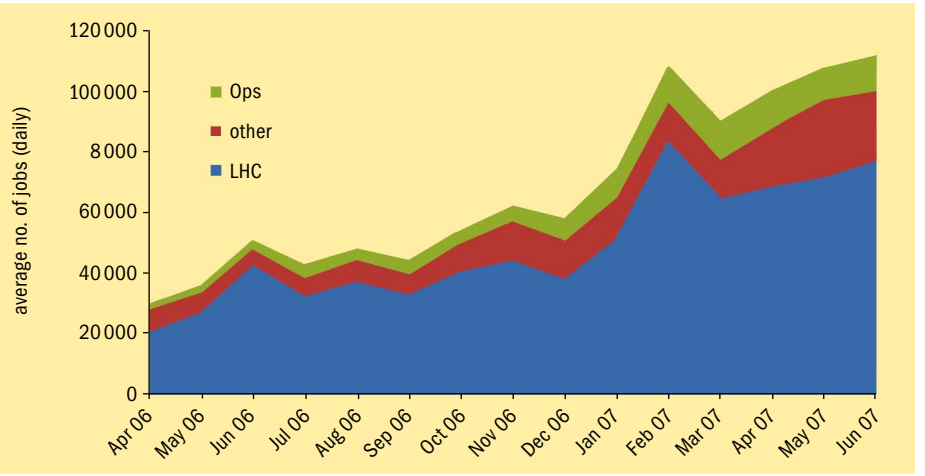


Fig. 1. The rise in the number of jobs run on the EGEE Grid from April 2006 to June 2007.

The Grid Deployment (GD) group in CERN's IT department is in many ways the technical linchpin of CERN's Grid activities. As well as ensuring that the LHC Computing Grid (LCG) is primed for experimental data, it also supports the broader technical goals of the multiscience Grid infrastructure run by the Enabling Grids for E-sciencE (EGEE) project, now in its second phase. *CNL* interviewed Ian Bird, the leader of the GD group, about recent progress in LCG operations, and the road ahead.

How is the GD group structured and funded, and how will this evolve?

There are about 60 people in GD, who are organized in three sections: LCG service coordination; EGEE Grid operations; and middleware certification and data management development. EGEE funds about 20 of those people. Also integrated in the group is ETICS, an EGEE-related project for middleware certification that CERN manages. There are a further five associates who are funded by other related projects in which CERN participates. In total, about half the group is funded by EGEE and its related projects. So the Grid operations for LCG depend strongly on external funding. This is true not just at CERN but also at many other LCG sites.

The Grid operations are a success story of EGEE that is often overlooked. Much fuss is made about the EGEE middleware, but actually the operations form a larger part of the EGEE effort, and today we can safely say that this is now working well. Certainly there is room for improvement. In particular, the operations still require many people. This is partly a reflection of the immaturity of the software, and partly a reflection of the fact that everything is so distributed. One focus at the moment is on automation and improved monitoring of operations, in order to reduce the number of people involved. This is not just to be cheaper but also to provide a more reliable service. This trend to rationalize the operations of the Grid will be emphasized even more strongly during the third phase of the EGEE project, which is expected to start in 2008.

How have Grid operations evolved over the last year?

I think the main achievement is the general increase in scale. In the last year the amount of work being run on the EGEE service has increased by more than a factor of two. This is something that the operators don't notice – you only see it when you look at the accounting after the

fact (figure 1). For me that is an indication that the processes we've got in place are capable of dealing with even more than we are currently running.

One of the important steps in getting more reliable operations is the Service Availability Monitoring (SAM) developed in GD. This is running Grid jobs to a site and probing each site externally on an hourly basis using a whole suite of tests of the services they are running. This has been the basis of the reliability measurements that we do for the Tier-1 centres in LCG. Previously, a criticism from the sites has been that this only gave output centrally. So over the last few months we have put effort into feeding back this information to the local fabric monitors at the sites. This is a prototype now but it should make a big difference, since site administrators will receive alarms – previously only seen centrally – directly in their own fabric monitoring systems.

The philosophy that we are following is: whatever can be done locally is best. The system administrators should be able to address a problem at source rather than us having to open a ticket centrally and then contact the site administrator. As part of this push, we have achieved agreement on a common format for publishing data from monitoring systems, so that this data can be shared between systems. The result is that whatever appears on an experiment dashboard about a particular site can be fed directly back to the site, so the site administrator knows what is going on.

We've also started a collaboration with EDS this year, via CERN openlab, that will focus on better ways of visualizing the complex results being recorded by the various monitors. The idea is to get some high-level visualization for the whole Grid, so you can see what is going on at a glance. I'm quite optimistic that this will provide a more intuitive way to assess the situation.

What are the challenges facing the LCG service?

We've heard a lot about service challenges and data rates over the last couple of years. I don't think that data rates are the main challenge anymore, it is the reliability that needs to be improved. The issue is to understand site problems; things stop working and it is not always clear why. A lot of effort is spent understanding why one experiment obtains a high data rate from a site, and another doesn't. So GD's role is a coordination effort with the experiments, trying to systematize how we handle problems. Today the LCG service focuses on the data transfer to the Tier-1s. For the rest, we rely primarily on either EGEE or its US counterpart, the Open Science Grid (OSG), to manage operations.

Interoperability remains a challenge for the LCG service. There has been good progress on some fronts. We now have a joint operation between EGEE and OSG, with OSG participating in the weekly Grid operations meeting that GD runs. A positive outcome of this is that CMS has been using the interoperability between the two Grids in their production for nearly a year: they submit work through the EGEE Resource Broker middleware, which runs in OSG. We are also making progress with the Naregi Grid project in Japan and NorduGrid in the Nordic countries. The Japanese Tier-2 sites would like to receive their support locally through Naregi. How interoperable the two middleware systems will become remains to be seen, but this scenario is not excluded. There is funded effort for interoperability between NorduGrid's ARC middleware and EGEE's gLite. We're close to being able to run jobs from the EGEE Resource Broker into ARC, but the other way around is further off.

What is the status of middleware certification and data management development?

Getting gLite 3 out of the door last year was a major milestone. This was a merger of the two middleware stacks, LCG and gLite. But at the same time it was a merger of two groups, two build systems and all of the processes behind that. And we did it in just six months.

Today the certification activity is taking middleware from EGEE, from the Virtual Data Toolkit (VDT), and building the gLite distribution. There is a large certification test-bed, both at CERN and with EGEE partners, where we do functional testing, integration testing and stress testing of each release. This is an activity that started in the LCG days and has expanded considerably. One of the problems with the middleware has been the lack of portability. There has been a significant effort in the last few months to untangle some of the dependencies in order to make it more portable.

The data management development team is responsible for the File Transfer Service (FTS) software and the corresponding service at CERN. The team is also responsible for the file catalogue that the experiments are using. There have been some major improvements in that area recently, in particular the development of the Disk Pool Manager (DPM) system, a storage element for sites that do not have enough effort to deploy the dCache system, for example.

What lies ahead for GD as the LHC start-up approaches?

One challenge is whether this Grid can really cope with the level of work we anticipate when the LHC starts up. The EGEE infrastructure is now running about 100 000 jobs a day. The number of jobs is often a better indicator of complexity than sheer CPU time, since jobs are a measure of how busy the system is. CMS and ATLAS say that they will each need to run a quarter of a million jobs a day once the data start flowing. I'm confident that the operation itself will deal with this five-fold increase of activity, because resources at sites will increase. The issue is whether the middleware can really scale in a manageable way to these levels, or whether you have to introduce so many service nodes that it becomes hard for each site to manage them.

Another challenge is that we will reach the end of the third phase of EGEE project funding in 2010. The idea is that we will make the transition to a model of national Grid infrastructures with a European coordination, which is a fundamentally different operations model from what we have now. How we make the transition to that, without disrupting ongoing operations, remains to be seen. We don't know what responsibilities each stakeholder will take. While 2010 may seem a way off, this is something that needs to be worked out within the next year if it is to be implemented in time.

Finally, we don't have much experience yet with the experiments doing analysis on the Grid. There are probably about 100 people per experiment who have run a job on the Grid. By next year perhaps half the collaboration will want to look at the data. How the Grid will cope with that is still a big unknown.

Answer CNL survey for chance to win a prize

This issue we say goodbye to Hannelore Hämmerle, who has been co-editor of *CNL* for the past three years. Hannelore has also been a driving force behind the Computing News section that appears in every second issue of *CERN Courier*. Thank you Hannelore!

Partly as a result of the reduction in personnel available to support these activities and also to serve our readership as effectively as possible, we wish to evaluate what CERN users think about *CNL* and elicit suggestions for improvements.

The purpose of *CNL* is to inform CERN users about recent changes and new trends in the computing infrastructure at CERN. *CNL* also includes conference and meeting reports as well as technical briefs and viewpoints that are considered to be of general interest to CERN users.

Your input will help us decide on the future of *CNL*, including the best frequency, format and content for this publication. That is why we are encouraging you to answer the online survey at http://cern.ch/cnl-survey, which should take only a few minutes of your time. Five people will be selected at random from those who submit the survey by 31 October, to receive a limited-edition *CNL* sweatshirt.

The CNL editors

Vista pilot: installation and IT support at CERN

Windows Vista is now available at CERN as part of a pilot (while Windows XP remains the default). Last issue we described Vista's new features, gave details of the hardware required to run the operating system, and talked about application compatibility (see http://cerncourier.com/cws/article/cnl/30298). In this article we report on the installation procedure and the scope of support for Vista at CERN.

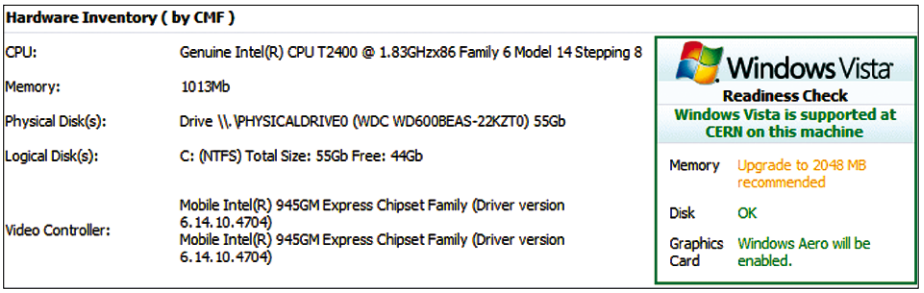


Fig. 1. Users can perform a readiness check to find out if the PC supports Windows Vista.

Hardware requirements

Vista brings many new functionalities and a new user experience, but it also has different hardware requirements. The basic CERN recommendations regarding hardware are given in table 1.

If a user plans to install Vista on a computer bought recently at the CERN stores, it will comply with the requirements. However, a user wanting to install Vista on a computer that now has Windows XP installed should first perform a Vista readiness check on the computer account status page on the WinServices site (https://cern.ch/WinServices/Services/ComputerAccounts/ComputerAccountStatus.aspx). On the right-hand side of the Hardware Inventory section a box gives information about the compliance of the computer with CERN's recommendations (figure 1). The box also tells you whether or not Windows Aero (a sophisticated 3D Vista user interface) is available.

Table 1: Hardware recommendations

CPU	1.0 GHz
Memory	2.0 GB RAM*
Hard disk	60 GB

* The memory requirement may drop to 1.5 GB if your computer has a graphics card with dedicated memory.

Table 2: Supported desktop models

Vista	Windows Aero
HP dc7700	✓
HP dc7600	✓
HP dc5750	✓
Elonex ProSentia	✓
Elonex Xelium	✓
Elonex Elium	✓
Elonex Elix	✓

Table 3: Supported laptop models

Vista	Windows Aero
HP Compaq nc6400	✓
HP Compaq nc6220	
NEC Versa P550	✓
NEC Versa M360	✓

Supported models

The readiness check may indicate that Windows Vista is not supported at CERN on a certain computer. This is because in the first phase of deploying Vista we will support it only on recent models of PCs that have been sold by CERN stores within the last two years. Tables 2 and 3 list seven desktop and four laptop models that comply with NICE Vista. We plan to extend the list in the future.

Installing NICE Vista

To install Windows Vista at CERN the standard procedure for a Diane installation must be followed. Once the computer is turned on, boot the network by pressing F12. Follow the installation Wizard, which will enable you to select an operating system. Windows Vista will only be an available option if the computer is on the list of supported models.

Each Windows Vista computer needs to be activated within 30 days of installing the software. We enforce activation during the Diane installation process; however, the activation has to be done by a direct connection to Microsoft servers over the Internet, so Vista should be installed while the computer is connected to the General Purpose Network and not to the Technical Network – otherwise, after 30 days the Vista computer will be blocked.

The result of the activation can be verified by right-clicking on the Computer item in the Start menu and selecting Properties. "Windows is activated" should be seen in the Windows activation section (figure 2). The IT/IS group is in contact with Microsoft to find a better solution.

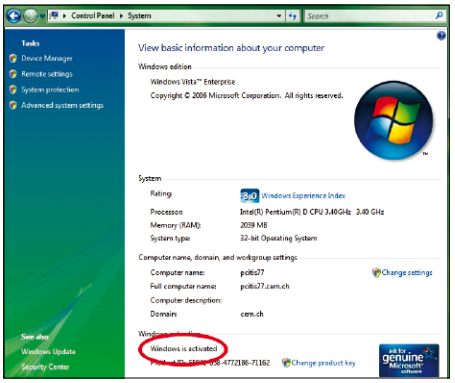


Fig. 2. The Vista activation status result.

Before using Windows Vista for the first time, users are recommended to reorganize their home folder. For details, see the article "IT reorganizes DFS home folder for simplicity" on p13.

Supported applications

The IT/IS group has tested all the standard office applications and ensured that they work with Windows Vista at CERN. This includes Microsoft Office 2007, Adobe Acrobat Reader and Professional, Corel Graphics Suite X3, Hummingbird Exceed 2007, CERN Phonebook etc. Before deciding to reinstall your computer, check with the appropriate support lines that the applications you use are supported on Vista. We encourage other application support groups to test their applications and make them available for Windows Vista.

Documentation and support

Documentation on using Windows Vista at CERN is available on the WinServices site at https://cern.ch/winservices/Help/?kbid=020201. Contact the helpdesk (helpdesk@cern.ch or 78888) if you have any questions.

Rafal Otto, IT/IS

EDH receives a new look and a new workflow engine

CERN's Electronic Document Handling application (EDH) is familiar to most people who work at CERN. It is the most heavily used administrative application developed by the IT department's Administrative Information Services (AIS) group; it has more than 10 500 active users, and at peak times easily exceeds 3 000 accesses a minute. EDH initially supported only a handful of processes, but now there are more than 40 forms that cover almost every aspect of work at CERN.

EDH was originally developed in 1992 as part of a massive project to update and improve CERN's administrative computing. It was developed using the then state-of-the art Client–Server architecture. The server ran SunOS, and the data was held in an Oracle 6 database. The client existed for Windows 3.1, MacOS and SunOS.

Six years later, in 1998, the first web version of EDH was created. Now Web EDH is 10 years old and it's time to perform some "surgery".

Time for a face lift

When Web EDH was first created there was a huge difference between the capabilities of the various web browsers used at CERN. For that reason the developers decided to use only the minimum number of features to create the webpages. Over the past 10 years EDH has been extended and improved, yet many aspects of the user interface have remained frozen since 1998.

During this period there have been many innovations in web technology. With the advent of so-called "rich internet applications", such as Google Maps and Outlook Web Access, CERN users have grown to expect more from web-based applications. To that end a small team of EDH developers has been working hard during 2007 to create a new look for EDH, in time for general release in 2008.

The team has taken account of feedback from EDH users and has developed an extensible framework that takes advantage of contemporary web browser capabilities. At the moment most of the visible changes are mainly cosmetic, although the framework will enable the interface to evolve with more interactive features as time and resources permit.

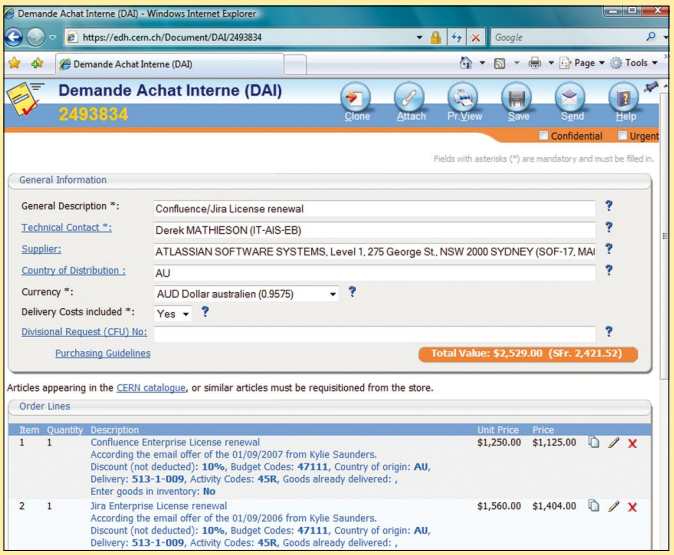
As well as an updated layout (after 10 years of faithful service EDH green is gone), there is a new floating button bar, which is always available – even when the page is scrolled down. The Help system has been updated and it now suggests help automatically if you haven't recently used a form field. But the most significant change is the reduction in the number of "pop-up" windows that EDH uses. Most editing can now take place within the same page, which makes it much easier for users who wish to use the Tabbed Browser capability of most modern web browser software, such as Internet Explorer 7 and Firefox.

Initially both the old look and the new look will be available to everyone, and users can select the one that they prefer. It is planned, however, to phase out the old look in 2008. As of August 2007 the development team is testing the new screens on all the commonly used browsers. To minimize the impact of the change, and as part of this testing phase, members of the IT department will have early access to the new look, most probably this November.

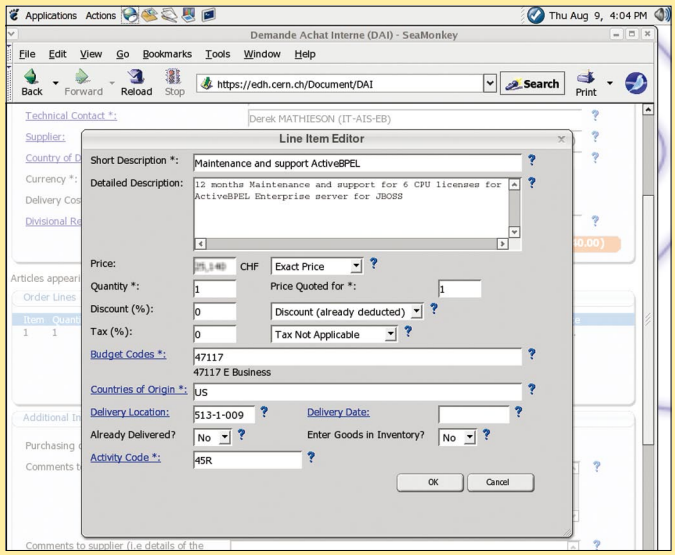
Heart replacement

As well as the more visible changes to the user interface, EDH will also undergo major changes below the surface with the replacement of the routing engine (the component that decides who should approve an EDH document). The current routing engine was introduced in 1999, based on a commercial product called Oracle Workflow. Although the tool has served well over the years, processing almost 1.5 million documents and sending nearly 4 million emails, it is no longer supported by the manufacturer and so it is time to change.

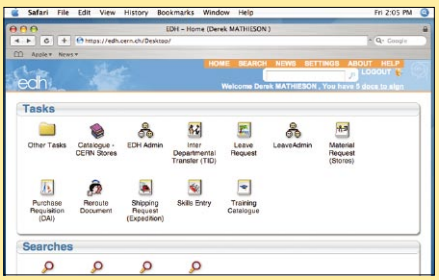
After much research, a more modern replacement engine has been selected from a relatively small American company called Active Endpoints. The product, ActiveBPEL, is a modern process orchestration engine,



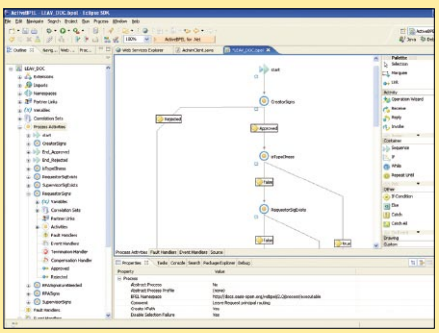
The new-look Internal Purchase Request (on Vista).



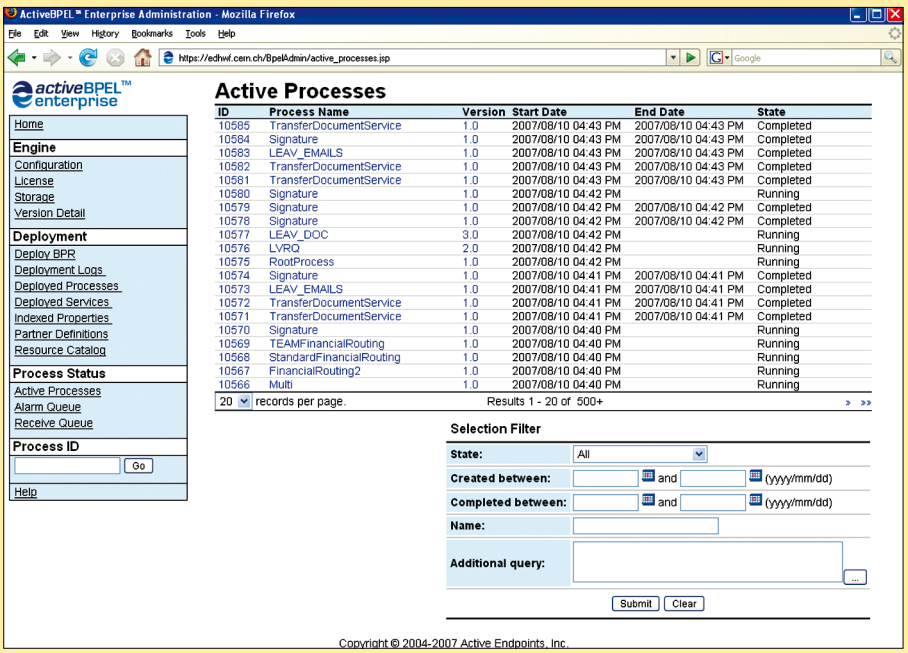
Example of an in-page line editor (on Scientific Linux).



The new EDH desktop with Safari (on Mac).



BPEL process design tool (in Eclipse IDE).



Web-based tools enable support staff to monitor documents as they are routed by EDH.

based on the Business Process Execution Language (BPEL) standard. Although the engine is available as an open-source project (see www.active-endpoints.com/open-source-architecture.htm), EDH has opted for the Enterprise edition to benefit from commercial support.

The selection of ActiveBPEL as the replacement workflow engine for EDH is a vital step: it will pave the way for a more open architecture and enable other CERN applications to use EDH services.

As well as replacing Oracle Workflow, the BPEL engine will enable the support of several new features. Most notably it will support services that operate as a pool of users, where any member of the pool can process a document that arrives at the service. This is especially useful for claims processing (school fees, travel etc) and registration documents.

Throughout 2006 and 2007, EDH developer (and Marie Curie fellow) Tommi Juhani Heikkinen has been rebuilding the workflow engine using a new, open and modular service-oriented architecture. Since January 2007 the BPEL-based engine has been operating in parallel with the existing workflow engine. Automatic jobs compare and highlight any differences between the routing decisions. Once the team is confident that no errors have been introduced during the reimplementation of the processes, the BPEL engine will take over completely and the old system will be switched off. Progress to date indicates that this will occur during the Christmas shutdown this year. It is a pity (at least for Tommi) that, if everything goes well, hardly anyone will notice the switch-over to the new system.

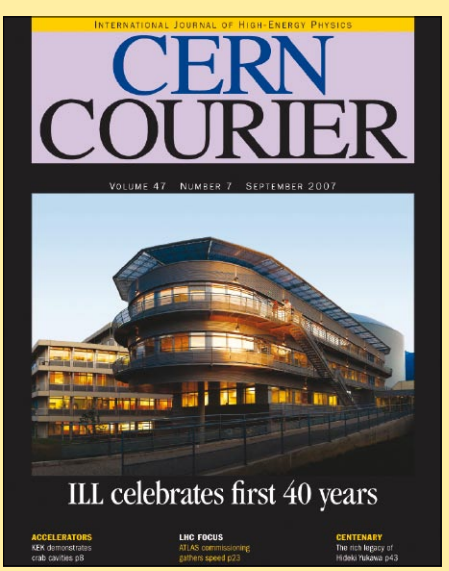
The future

We hope that with all of these changes EDH will continue to help CERN users spend less time trying to understand how CERN's administrative processes work, and instead focus their energies on understanding how the universe works.

Derek Mathieson, IT/AIS

Computing articles featured in this month's CERN Courier

The articles listed below appear in the September issue of *CERN Courier*. Full-text articles and the rest of the issue's contents are available atcerncourier.com.



Computing News

- **Grid users meet standards at OGF20/ EGEE User Forum**
  The forum was Europe's biggest ever Grid event, with more than 900 registrations.

- **African scientists plug into volunteer computing projects**
  Workshop provides African scientists with know-how to access volunteer computing.

- **ACAT'07 tries to solve computing challenges**
  Amsterdam workshop discusses advanced computing and analysis techniques.

- **CERN makes it into supercomputing TOP500**
  CERN cluster is 115th fastest in the world.

- **European cluster project for supercomputing is completed**
  The FZJ centre's JULI Linux cluster is ready.

Calendar of events

Feature articles

- **50 years of Fortran**
  A review of the use of Fortran at CERN.

- **The greatest challenge: computing the brain**
  Bob Bishop, the Blue Brain Project and CERN.

Tests improve Grid performance

Being a Grid user isn't always straightforward. You might have 10000 CPUs and terabytes of disk space at your fingertips, but can you get your job to work on all of them – or any of them?

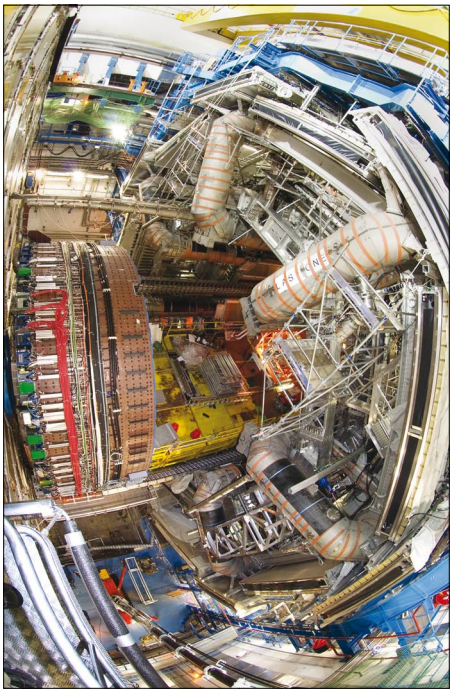
Although there are many more Grid users than there used to be, getting started on a Grid, and getting it to do what you want, is still not for the faint-hearted.

Fortunately there are people trying to make it easier. One of them is Steve Lloyd of Queen Mary, University of London. As part of his work on the ATLAS experiment, Lloyd has been sending jobs to Grids such as GridPP and EGEE for years. But although many of his jobs went off without a hitch Lloyd found some just kept failing, even though they were sent to a working site that had passed all of the Grid's tests.

Six months ago he decided to find out why. And as chair of the GridPP collaboration in the UK he was in a position to get problems fixed.

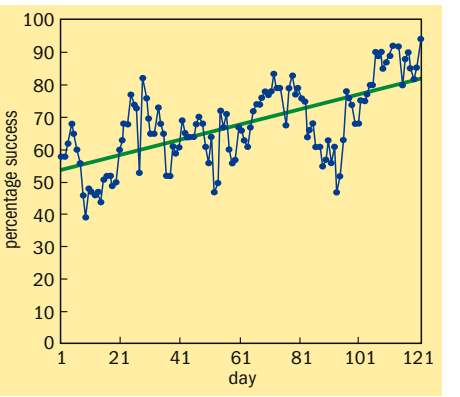
The result of Lloyd's work is a suite of three test jobs that run hourly on sites in the UK particle physics Grid. The complexity of these test jobs range from submitting "hello world" to analysing a file of particle physics data using the latest ATLAS software.

Lloyd explains: "When I first started running these tests their success rate was only around 50%. I'd get a massive range of problems: broken resource brokers, difficulties with the information system,



The inner guts of the ATLAS detector, which will soon be sending streams of data to computers worldwide. (Courtesy CERN.)

proxy certificates timing out, sites that didn't have the latest version of the ATLAS software, and even sites without the required compiler."



The increase in the success rate of test jobs run on GridPP over time. (Courtesy GridPP.)

Using the detailed log files provided by Lloyd's test jobs, and with the aid of the GridPP deployment team, each Grid site got to the bottom of their problems.

Lloyd's test jobs now run at a 90% success rate. This gives him some hope for future Grid users. "I used to wonder how users would ever be able to analyse the ATLAS data on the Grid. Now I'm more hopeful – but we've still got a lot of work to do." Lloyd's experience shows that things don't always go smoothly, even for experienced Grid users. But things are on the up.

Sarah Pearce, GridPP

This article was published online in *iSGTW* on 13 June.

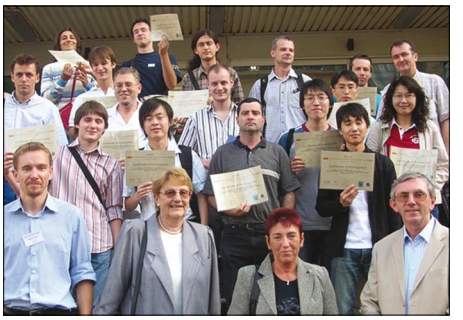
DIY summer school: materials available online

Missed out on attending a Grid summer school this year?

Why not study at home using the teaching material from the Joint Enabling Grids for E-sciencE (EGEE) and South Eastern European Grid-Enabled eInfrastructure Development (SEE-GRID) Summer School on Application Support, held 25–30 June in Budapest, Hungary, and organized and hosted by MTA SZTAKI, the Computer and Automation Research Institute of the Hungarian Academy of Sciences.

The teaching material [at http://indico.cern.ch/conferenceDisplay.py?confId=14387] includes lectures, slides and notes for the hands-on practicals, originally presented by EGEE, SEE-GRID and ICEAGE representatives.

The school aimed to introduce potential users to EGEE and SEE-GRID Grid technologies, and to introduce methods for application development on EGEE Grid networks. Its primary focus was on ways that end-users can apply gLite middleware to operate large-scale distributed applications on top of interorganizational Grids.



Graduates from the Joint EGEE and SEE-GRID Summer School. (Courtesy Gergely Sipos.)

Twenty-two people attended the school, including 17 from European academic institutes and universities, and five from the University of Seoul.

During dedicated sessions, the attendees' own legacy applications were ported onto EGEE. Three applications were successfully ported during the Gridification sessions:

- **Time evolution of spherically symmetric nonlinear fields:** from the Research Institute for Particle and Nuclear Physics, Hungary. The application was ported onto EGEE as a parametric study workflow in P-GRADE that uses storage elements and file catalogues.
- **Simulation of pellet–plasma interaction:** from the Eötvös Lóránd University, Hungary. The application was also ported onto EGEE as a parametric study job with P-GRADE that uses storage elements and file catalogues.
- **HYP3D code, used in quantum mechanics theory:** from the University of Rennes and CNRS, France. The application works with large data files and requires large memory on worker nodes. Two of three jobs were ported during the school with command line tools and JDL. The remaining job will be ported with the help of the EGEE application porting support group.

Introductory and report slides on the Gridified applications are also available.

Gergely Sipos, MTA SZTAKI

This article was published online in *iSGTW* on 8 August.

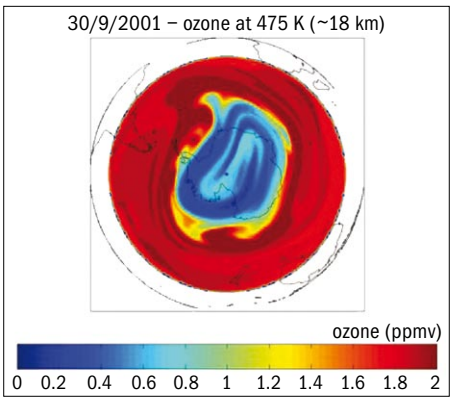
More than just computing power: Earth science on EGEE

You have a good chance of meeting Monique Petitdidier at any event that combines Earth science with computing. Although officially retired, Petitdidier continues in her role as a senior scientist with the French institutes IPSL and CNRS: she coordinates the Earth science contingent of the Enabling Grids for E-sciencE (EGEE) community, and is a driving force behind the DEGREE project (Dissemination and Exploitation of GRids in Earth sciencE). Petitdidier says that scientists in the Earth Science Research virtual organization view Grids as simply a computing means to their research end.

"Most institutes cannot afford intensive computation, such as that required to run millions of jobs or very large data archive exploration," Petitdidier points out. "This means that their science is limited by the computing power they have available."

"We mainly have researchers who need the Grid for specific problems, and will leave once they have obtained their results," she explains. "Sometimes they will come back with a different problem, but most of the time they just move on to other problems where you don't need a Grid. The emphasis really is on the science rather than the technology."

"Some applications need to share large data sets or algorithms, are using a statistical approach, and require rapid solutions. For example, if you have to reprocess a whole database with different parameters, this is a lot more adapted to run on the Grid."



The ozone-mixing ratio in the southern hemisphere on 30 September 2001, measured using an application that will be run completely on EGEE. (Courtesy S Godin-Beekmann IPSL/Service d'Aéronomie.)

But the benefits of Grids go beyond raw computing power, says Petitdidier.

"Grids are a great way to get people together and thus eliminate doubling of effort. We often see that people from different fields or regions have similar problems. For example, people looking at the flooding of rivers in France or Ukraine have been put in contact with scientists who have developed a similar application for the Danube river."

This doubling of effort can also be eliminated when developing software for geophysical sciences. Such software is often designed for a particular machine with specific characteristics, which makes it difficult to port to other architectures.

However, software developed for a Grid is ubiquitous, eliminating any problems with changing between machines, and leaving more time for pure research.

The Earth Science Research virtual organization in EGEE brings together some 50 scientists at any one time, but Petitdidier says she'd like to see more people get involved.

"We have many users but have not yet achieved a critical mass of institutes and organizations involved, so the challenge is to get new people on board," she says.

Hannelore Hämmerle, EGEE

This article was published online in *iSGTW* on 11 July.

Students contribute to Cosmic Ray e-Lab

As of 3 July 2007, teachers and students in QuarkNet's Cosmic Ray e-Lab had uploaded more than 11 400 data files to their central database, with one file representing roughly one day of data collection.

These files are part of a new education and outreach programme encouraging students to think more about cosmic rays. The data came from schools in 23 states across the US, as well as Canada, Germany, India, Japan and Taiwan.

The Cosmic Ray e-Lab is a pilot for Interactions in Understanding the Universe. It seeks to establish an "educational virtual organization" that supports a portfolio of labs in which Grid tools and techniques facilitate several levels of student activity.

So far 305 teachers and 599 student research teams in 301 schools have e-Lab accounts; QuarkNet has already deployed over 232 cosmic ray detectors, with more on the way.

The success of this pilot e-Lab demonstrates what can be accomplished by connecting formal education, research science and information technology. The project aims to harness Grid computing power to provide a learning environment for encapsulating tools and data sets, provides a mechanism for students to perform data analysis, and helps establish online communities for collaborative learning.

Cosmic rays shower the Earth. This screen shot is part of an introductory animation designed to fire students' imaginations.

In this way teachers can offer their students guided, experiential science investigations. By learning to work as scientists do, students enhance the way they think and their approach to solving problems. They become responsible for their own learning, defining their own research objectives and plans, and evaluating their success.

This integrated approach provides a powerful two-way interaction between experiments and education. By creating a unified tool set that can be used by the education programmes of many experiments, a "critical mass" can be developed, supporting a common approach to teaching the data-analysis skills students require to better understand their universe.

Also under development are e-Labs associated with the ATLAS, CMS, LIGO and STAR experiments, as well as a cosmic ray i-lab for informal and museum settings.

Marge Bardeen, Fermilab

This article was published online in *iSGTW* on 25 July.

Suggestions for designing and writing secure software

In this article we give some general tips for creating more secure software, which software engineers should follow at the various phases of the software development cycle. The list doesn't contain all possible advice but it does include the most important – the suggestions that are worth keeping in mind when designing and developing almost any kind of software with any programming language.

Security should be seen as part of the system from the very beginning, and not added as a layer at the end. The latter approach produces insecure code, may limit functionality, and will cost much more (in both time and money).

Architecture

- Modularity: divide the program into semi-independent parts (small, well defined interfaces to each module/function).
- Isolation: each part should work correctly even if others fail (return wrong results, send requests with invalid arguments).
- Defence in depth: build multiple layers of defence instead of trusting just one protection mechanism. For example, validate user input data at entry point, and check again all values that are passed to sensitive parts of the code (like file handling etc).
- Simplicity: complex solutions are much more likely to be insecure.
- Redundancy: if possible avoid having a single point of failure.

Design

- Make security-sensitive parts of your code small.
- Least privilege principle: don't require more privileges than you need. For example, run your code as the least privileged user necessary (don't run it as root, nor with SUID flag). Make sure that the account on which you will run your code has only the file access and execute privileges that your code really needs. Don't connect to a database with administrative privileges from your software.
- Choose safe defaults: for example, a random password that users are likely to change rather than a standard default password that many won't bother to change.
- Deny by default: for example, when validating user input accept only characters that you expect rather than trying to block known "bad" characters.
- Limit resource consumption, to limit the likelihood of a "denial of service" attack.
- Fail securely: for example, if there is a runtime error when checking user's access rights, assume the user has none.
- In distributed or web applications don't trust the client: don't expect it to validate user input, perform security checks or authenticate users – it all has to be done (again) on the server side. Remember that HTTP response header fields (cookies, user-agent, referrer etc) and HTTP query string values (from hidden fields or explicit links) may be forged or manipulated.
- Cryptography: use trusted, public algorithms, protocols and products. Do not invent your own cryptographic algorithms or protocols, nor implement existing ones.

Implementation

- Read and follow guidelines for your programming language and software type.
- Think of the security implications of what your code does.
- Reuse trusted code (libraries, modules).
- Write good-quality, readable, maintainable code (bad code won't ever be secure).

Coding

- Don't trust input data – data coming from potentially malicious users is the single most common reason for security-related incidents (buffer overflow, SQL injection, Cross Site Scripting (XSS), code inside data etc). Input data includes command-line arguments, configuration files (if accessible by not-trusted users), environment variables, cookies and POST/GET arguments etc.
- Validate (sanitize) all input data: consider all input dangerous until proven valid, deny by default if not sure, validate at different levels; for example, at input data entry point and before really using that data.
- Don't make any assumptions about the environment: make sure your code doesn't break with modified/malicious PATH, CLASSPATH and other environment variables, current directory, @INC Perl variable, umask, signals, open file descriptors etc.
- Beware of the race condition: can your code run parallel? What if someone executes two instances of your program at the same time, or changes environment in the middle of its execution?
- Deal with errors and exceptions: don't assume that everything will work (especially file operations, system and network calls), catch exceptions, check result codes. Don't display internal error messages, failing SQL query, stack trace etc.
- Fail gracefully: if there is an unexpected error that you can't recover from, then log details, alert the administrator, clean the system (delete temporary files, clear memory) and inform the user.
- Protect passwords and secret information: don't hard-code passwords (it's hard to change them and easy to disclose), use external files instead (possibly encrypted) or already existing credentials (like certificates or Kerberos tickets), or simply ask the user for the password.
- Be careful when handling files: if you want to create it, report an error if it already exists; when you create it, set file permissions; if you open a file to read data, don't ask for write access; check if the file you open is not a link with the lstat() function (before and after opening the file); use absolute pathnames (for both commands and files); be extra careful when the filename (or part of it) comes from a user.
- Temporary files: don't fall for the symbolic link attack (someone guesses the name of your temporary file and creates a link from it to another file e.g. "/bin/bash", that your program overwrites). Temporary files must have unique names that are hard to guess (use tmpfile() for C/C++, mktemp shell command etc).
- Be careful with shell calls, eval functions etc: such functions evaluate the string argument as code and interpret it, or run it on the shell. If an attacker managed to inject malicious input to that argument, you're executing his code.

After implementation

- Review your code and let others review it.
- When a (security) bug is found, search for similar ones.
- Use tools specific to your programming language: bounds checkers, memory testers, bug finders etc.
- Turn on compiler/interpreter warnings and read them ("perl –w", "gcc –Wall").
- Disable debugging information ("strip" command, "javac -g:none" etc).

Further information

Find useful links and language-specific tools at http://cern.ch/info-secure-software/SecurityChecklistForSoftwareDevelopers.pdf. See also http://cern.ch/SecureSoftware.

Sebastian Lopienski, IT/FIO

J2EE Public Service achieves new levels of functionality

Today at CERN you can host a Java web application (servlet/jsp) in a centrally managed deployment platform maintained by the IT department. The aim of the J2EE Public Service is to provide you with the necessary infrastructure and tools.

Since it went into production at the end of 2005, the J2EE Public Service has been used extensively by the CERN Java community. Today the service hosts approximately 120 Java web applications (like CERN project websites, third-party software, Java web services etc). The J2EE Public Service is widely used by physicists. For example, it hosts ATLAS web applications (to monitor and manage documentation related to the construction of the detector), various LHC web applications, and many other sites (like CERN's Technology Transfer website, shown in figure 1).

Fig. 1. CERN's Technology Transfer website.

What we provide:

- A servlet and JSP container that is compatible with Servlet 2.4 and JavaServer Pages 2.0 specifications (currently Apache Tomcat 5.5). If you own a Java web application you will be assigned your own Tomcat instance.
- A series of management, configuration and monitoring tools that will enable you to manage your web application. You can deploy the application, monitor it, have access to log files and many other features via an administration web interface.
- Oracle JDBC drivers are preinstalled (thin), and NICE authentication is configured in case you need to implement access control to the web application.

Please keep in mind that the service provider will help you to deploy your web application but not to develop it.

The scope of the service

The J2EE Public Service is designed to host non-critical, medium-sized J2EE server-side applications. Mission-critical applications are not in the scope of the service: we provide full support during CERN working hours, but outside working hours support is provided on a best-effort basis (http://cern.ch/j2ee-public-service/sla.html).

Advantages of using the service

The J2EE Public Service is convenient for potential CERN J2EE users because it releases them from tedious maintenance tasks. We take care of the Java platform server installation, back-ups, monitoring, software upgrades and security.

Improving monitoring

Since January the service has offered a Java Management Extensions (JMX) monitoring interface to consult runtime information about web applications (e.g. memory use, CPU process time and compilation time).

This is possible thanks to the integration of JMX Monitoring Component (JMX MC) within the service. JMX MC collects this data from the Tomcat instances and feeds it into the LHC Era MONitoring (Lemon) monitoring system. This enables users to consult the data over the lifetime of their web applications, helping them to maintain and recognize any emerging problems (figure 2).

Solution to permanent storage

As many potential users of the service ask for permanent file storage, we provide, on demand, a library that uses WebDAV to connect to the Windows Distributed File System (DFS). As this solution requires modifications in the source code of the web application, we are now in the process of testing the Andrew File System (AFS) as an alternative solution.

How to start using the service

The J2EE Public Service is integrated with CERN's central web services. Creating a Java website is simple, just follow the steps explained at http://cern.ch/j2ee-public-service/stepbystep.html (figure 3).

Visit the service webpage at http://cern.ch/j2ee-public-service and join up if it fits your needs. We welcome feedback, and if you have any questions please ask us (j2ee-public-service.support@cern.ch).

Lucia Moreno, IT/DES

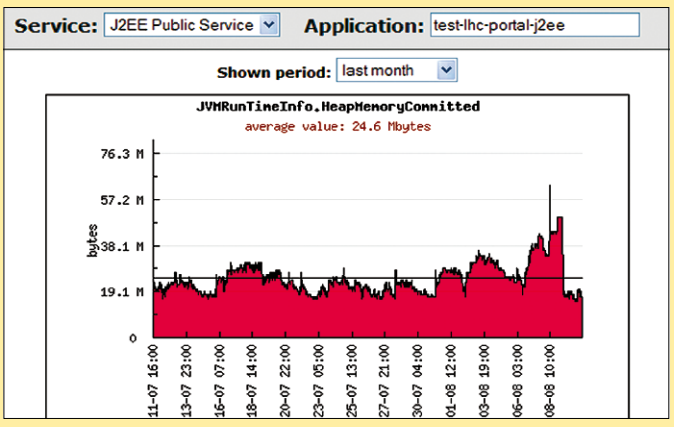


Fig. 2. JMX enables users to monitor runtime information.

Fig. 3. Create a Java website from the central web services page.

Software freedom defender Stallman addresses CERN

On 18 June Richard Stallman gave a talk at CERN on the ethics and practice of free software. Earlier in the day he had given the keynote speech at the AIM 2007 (Association Information and Management) conference in Lausanne.

It was a great pleasure to receive Stallman at the CERN Council Chamber, which, unsurprisingly, was not big enough for the large audience that came to listen to him. This article attempts to give a flavour of his talk, but it goes without saying that we cannot reproduce the show that was given by this well known speaker.

Stallman's talk on the ethics and practice of free software was extremely well attended.

About Stallman

As most people know, Stallman is a software freedom activist, hacker (in the positive, original sense) and software developer (he wrote the first version of the Emacs editor in 1975). In September 1983 he launched the GNU Project to create a free Unix-like operating system (OS), and he has been the project's lead architect and organizer. The GNU Project is regarded as the start of the free-software movement. In October 1985 Stallman set up the Free Software Foundation to support the movement. In 1989 he co-founded the League for Programming Freedom to fight against software patents and the extension of the scope of copyright. Stallman pioneered the concept of copyleft and is the main author of several copyleft licences, including the GNU General Public Licence, the most widely used free software licence.

Freedom in computing

Stallman started his talk in an unusual way, by asking for all computers and video-projectors to be switched off; in fact he only used a microphone and did not show any slides or transparencies. When asked by the chairman how long his talk would be, given that the average is an hour plus 30 minutes for questions, he seemed to think it was not enough, and indeed he talked for more than an hour.

The first part of the talk was related to the concept of "freedom", and it was a personal attack on all the people in the computing world who break this freedom for computer users by imposing on them what they must do and what they need. Microsoft is obviously Stallman's biggest enemy, with all its licensed products, imposed updates and upgrades, and intrusions in your computer.

We have to teach the rest of our community about freedom in computing, says Stallman.

For Stallman "freedom" is the key concept that users should strive for in computing: the freedom to take what you need, to modify the code according to your needs, to distribute the code with or without these modifications, to make updates when you like and only if you need them, and so on. When using Microsoft products or other widely used commercial software, even if available at no cost, users lose this freedom and are forced later on to accept what is decided by the providers. To access audio and video media, Stallman insisted that users should resist using RealPlayer and instead employ the free formats provided by Helix, for example.

Stallman prefers the term "Free" software to "open source" software, which is used today by commercial companies that distribute free software that is often mixed with other components that are not free. "Free" means that the software can be used, studied and modified without restriction, and it can be copied and redistributed in modified or unmodified form, either without restriction or with certain restrictions to ensure that end-users have the same freedoms as the original authors. "Most computer customers have never heard this idea of FREEDOM!" said Stallman.

The Free Software Foundation and GNU

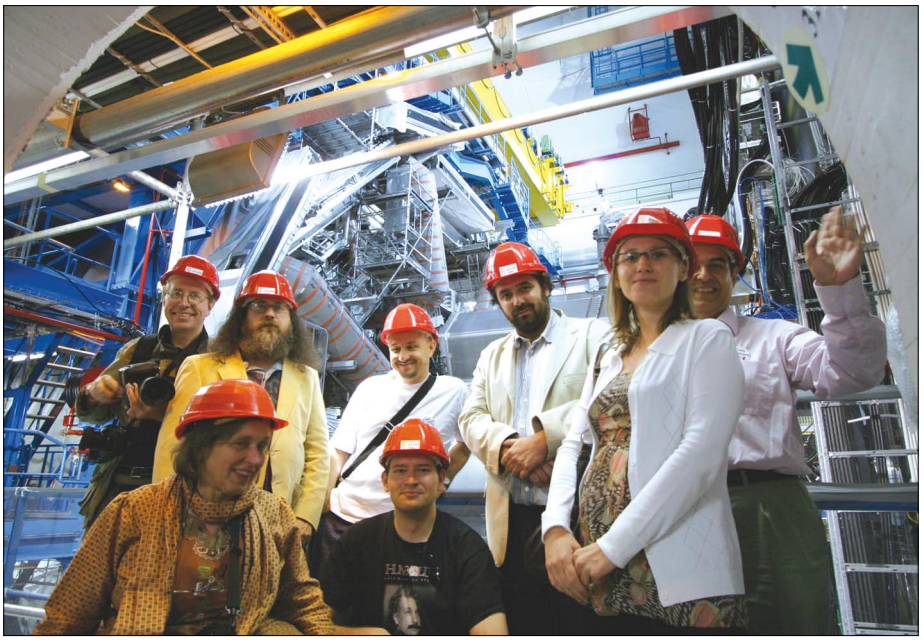
Stallman continued his talk by describing the Free Software Foundation (FSF), the non-profit organization that he founded

to support the free-software movement and the GNU Project.

"We need your help! If you know devices that do not work with free software, then please tell us!" said Stallman.

The original purpose of the FSF was to promote the idea of free software ("free" as in "freedom"). The organization developed the GNU OS as a first example of this concept. The GNU General Public Licence (GPL) is the most widely used licence for free-software projects. The most current version (GPL v.3) has just been released after a long period of deliberation; it is meant to address various new issues (such as new distribution technologies) and threats (e.g. patents, digital rights management, locked hardware and the DMCA). The FSF has also published the GNU Lesser General Public Licence (LGPL) and the GNU Free Documentation Licence (GFDL).

The GPL is the most popular and well known example of the type of strong copyleft licence that requires derived works to be made available under the same conditions. Under this philosophy, the GPL is said to grant the recipients of a computer program the rights of the "free software" definition, and it uses "copyleft" to ensure the freedoms are preserved, even when the work is changed or added to. This contradicts the "permissive free software licences" that include a set of "exclusive rights", often of limited duration, which regulate the use of some parts of this software.



During his visit at CERN, Stallman found time to go on a tour of the ATLAS experiment.

GNU Linux

Linux is a typical example of these concepts and of the difficulty of identifying real "free software". Many users are not fully aware of the distinction between the kernel, which is Linux, and the whole system, which they also call "Linux". These users often think that Linus Torvalds developed the whole OS in 1991. However, according to Stallman's definition of the OS, Linux is only the kernel: i.e. the program in the system that allocates the machine's resources to the other programs that you run. The kernel is an essential part of the OS but is useless by itself; it can only function in the context of a complete OS. Linux is normally used in combination with the GNU OS: the whole system is basically GNU with Linux functioning as its kernel. This is why Stallman invited us to be careful and to speak about "GNU/Linux" for the fully free OS. "Please attach GNU to Linux, and use either GNU+Linux or GNU/Linux!" said Stallman.

You will find more details on this subject on the GNU website (www.gnu.org/gnu), including several articles written by Stallman himself: for instance "GNU's not Unix! What's in a name?" (at www.gnu.org/gnu/why-gnu-linux.html) or "Linux and the GNU Project" (at www.gnu.org/gnu/linux-and-gnu.html). There you will read: "Our community's strength rests on commitment to freedom and co-operation. Using the name GNU/Linux is a way for people to remind themselves and inform others of these goals…"

A great challenge to the future of free software comes from the tendency of the Linux distribution companies to add non-free software to GNU/Linux in the name of convenience and power. All the major commercial distribution developers do this; none produces a distribution that is entirely free. Most of them do not clearly identify the non-free packages in their distributions. Many even develop non-free software and add it to the system. Some outrageously advertise Linux systems that are "licensed per seat", which gives the user as much freedom as Microsoft Windows.

In conclusion, programmers should be be very careful when choosing software components. According to Stallman, programmers should not develop free software that depends on non-free packages or libraries; this problem has been demonstrated with software like Motif and Qt, and now also with Java since Sun has added non-free Java implementations to the package.

"We have to teach the rest of our community about freedom," said Stallman.

Nicole Crémel, IT/UDS (CNL editor)

The deadline for submissions to the next issue of the CNL is

12 October

Please e-mail your contributions to cnl.editor@cern.ch

If you would like to be informed by e-mail when a new issue of CNL is available, subscribe to the mailing list cern-cnl-info. You can do this from the CERN CNL website at http://cern.ch/cnl

openlab/Intel workshop offers programmers multicore training

Most readers have probably heard of the multicore revolution and the increasing impact that multicore processors are having on the way software needs to be written.

The possibility of gaining additional processing power through frequency scaling has diminished greatly over the last few years, while at the same time Moore's law continues to increase the number of transistors that can be put on a chip. Therefore processor manufacturers have found an alternative way to use spare transistors, and that is by including additional processing cores in a single processor package.

This relatively new feature of modern computers implies a different model for writing and maintaining software than the one we have been using up until now. Today we can usually get away with adding a new process to the system and throwing in more memory as the number of cores grows – at CERN this is typically 2 GB per core. Very soon, when the number of cores in a single package reaches 16, 32 or more, adding more memory will become very costly, and the need to change the approach to programming will become instantly and painfully apparent.

While many of the challenges ahead are well known from basic university parallelism-related classes, some of them are quite tricky and difficult to meet, even for experienced programmers. With non-expert CERN and high-energy physics (HEP) programmers in mind, CERN openlab and Intel have teamed up to bring you a workshop on multithreading and parallelism, concentrating on the key software development differences between the linear and parallel worlds.

Spring workshop

The first workshop took place on 31 May and 1 June at CERN. We had the pleasure of receiving two experienced teachers from Intel, Herbert Cornelius and Hans-Joachim Plum, who will probably also join us for the second workshop in October. At the spring workshop, the first day comprised a series of lectures and the second a hands-on lab.

Day one began with Cornelius explaining the reasons for holding the workshop: the push towards multicore and Intel's plans for the next few years, which include several upcoming generations of ×86 CPUs. The participants asked a range of questions, especially about the classified Intel 80-core processor prototype described by Cornelius. Some common parallelism and scalability

Workshop participants were able to work on several of CERN's multicore machines.

issues were also presented. In the afternoon Plum spoke about OpenMP, POSIX threads and Intel's Threading Building Blocks in more detail, going as far as describing the peculiarities of individual API calls.

On day two the participants were divided into pairs and given access to several of CERN's multicore machines to work on. Cornelius and Plum demonstrated the powerful abilities of Intel's multithreading debugging software, Thread Checker and Thread Profiler.

Another useful Intel product, a performance analyser called VTune, was also described but not demonstrated. It is worth mentioning that openlab is actively involved in a collaboration with Hewlett-Packard to develop pfmon, an open-source tool for low-level performance monitoring, and this has some of the same functionality as the Intel tool. During several exercises pfmon was used as a replacement for VTune, to show certain non-obvious side effects of incorrect thread programming.

Because the attendees on day two were familiar with POSIX threads, the instructors concentrated mostly on OpenMP-related exercises. The participants were asked to write multithreaded code possessing certain characteristics and then debug the code as needed.

We were pleased to hear the teachers say later that they considered the audience at CERN to be very quick at grasping the concepts covered. Feedback received after the first workshop will heavily influence the shape of the next event.

The second workshop

It was a surprise that so many people subscribed to the first workshop. Given the huge interest and the lack of available seats, openlab has decided to turn the workshop into a semi-regular event that will gradually gain more and more CERN-specific content with time.

The next workshop will take place on 4–5 October at the CERN Meyrin site. Fifty places will be available for the first day, and 40 places for the second day. The programme will be similar to the last workshop but will be enhanced with more CERN-specific topics. On day one, the instructors from Intel will speak about the future challenges of multicore architectures and processing, as well as general parallelism principles. Some multithreading technologies, such as OpenMP and POSIX threads, will also be covered again. In addition it is hoped that several CERN scientists will give an insight into the most prominent multithreading technologies that are deployed at CERN, and will highlight the relationship between the multicore revolution and popular CERN software.

On day two, participants will be able to use the knowledge gained during the first day and get acquainted with several Intel tools that are used for developing and debugging multithreaded applications. Moreover, several lab assignments concerning OpenMP and POSIX threads will be given out in the afternoon, with the instructors at hand to guide the participants through the exercises. While some of the exercises will be quite advanced, rest assured that day two is being prepared with non-expert users in mind. As usual, the best places to look for announcements are the openlab webpage (http://cern.ch/openlab) and the internal CERN webpage.

Conclusion

Multicore technologies began to dominate everyday computing some time ago. It is our joint responsibility to harness this free processing power and use it to the limit. Let us hope that during this time of change in the computing landscape, CERN and HEP will harvest the fruit of this new technology to its fullest extent.

Andrzej Nowak, IT/DI, openlab

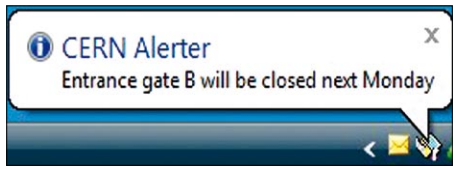
CERN Alerter is now based on RSS

In July a new version of the CERN Alerter was installed on all centrally managed NICE computers. Since then it has been used by central IT services to send urgent messages.

This latest version is based on Internet Explorer 7.0. All alerts are now displayed on the screen in the form of the Internet Explorer window (figure 1). To print the message it is necessary to right-click on the background of the alert window and select the Print option from the menu.

When a message is not urgent the alert may not pop up on the screen immediately. In such cases, before an alert window is shown (which normally takes place at 6 a.m. the next morning or at the next log-on) users are notified by the balloon shown in figure 2. When the balloon is clicked, the alert will pop up immediately.

Users can access the history of a message at any time. Simply right-click on the CMF icon in the Windows system tray, select the CERN Alerter option and click on History (figure 3). This will again open the Internet Explorer window, with a list of all the alerts that have been sent to the user in the past.



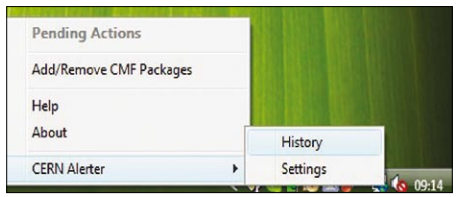


Fig. 1 (top). CERN Alerts now appear in an Internet Explorer window. Fig. 2 (middle). A balloon may notify non-urgent messages. Fig. 3 (bottom). Users can have easy access to the history of messages at any time.

Non-Windows operating systems

The new system is based on RSS (Really Simple Syndication) as a transport protocol that makes the system accessible from other platforms, not only NICE computers. Any RSS reader can be installed (all recent web browsers can act as RSS readers, including Firefox and Safari). Users can then subscribe to the RSS feed at http://cernalerts.web.cern.ch/cernalerts/alerts.aspx to see all of the messages ever sent by the central services. A user name and a computer name can be added to the URL to filter messages and see only those that concern a specified user and computer: http://cernalerts.web.cern.ch/cernalerts/alerts.aspx?login=MYLOGIN&computername=MYCOMPUTER.

Posting messages

The only authority allowed to transmit messages is the IT manager on duty (MOD). If another authority at CERN wishes to send a message using CERN Alerter, he/she should send an email to mod@cern.ch. The IT MOD can then decide whether or not to send the message.

User documentation for the CERN Alerter is available at https://cern.ch/winservices/Help/?kbid=060810. Documentation on how to read RSS feeds from an SLC computer is available at http://cern.ch/linux/documentation/rss.shtml.

Rafal Otto, IT/IS

DFS home folder is reorganized for simplicity

Each user at CERN has three folders created for him/her on central servers: a DFS (Distributed File System) home folder, a Favorites folder and a Desktop folder. At the moment these folders are stored in different physical places on the network. Your home folder, where normally all documents should be stored, is accessible from DFS. The Favorites and Desktop folders are more or less hidden and not easily accessible without browsing "desktop" during your Windows session or "favorites" in Internet Explorer. Users sometimes find this structure confusing, so the IT department's Internet Services group (IT/IS) proposed reorganizing these folders in a more straightforward way.

Let's take as an example an account called "rotto". Its folders are going to be reorganized in the following way (figure 1):

- the content of Desktop will be moved to \\cern.ch\dfs\Users\r\rotto\Desktop;
- the content of Favorites will be moved to \\cern.ch\dfs\Users\r\rotto\Favorites;
- the content of \\cern.ch\dfs\Users\r\rotto will be moved to the Documents subfolder with two exceptions:

– the public folder will stay at the root (for better security on Documents);
– the www folder will stay at the root;

- two Vista-specific folders, Links and Contacts, will be added.

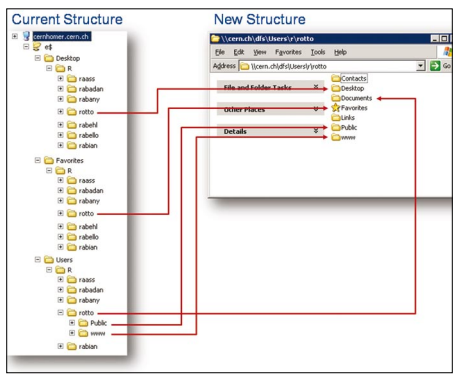


Fig. 1. The new structure of the DFS home folder will be much easier to understand.

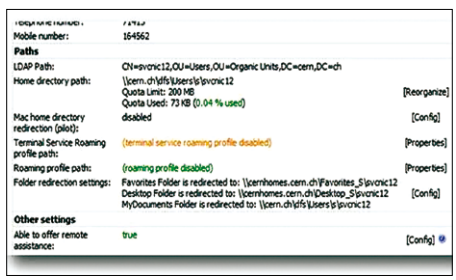


Fig. 2. Users can reorganize their home folder via the Check Account Status page.

Schedule

At the end of July the IT/IS group prepared a pilot where users could ask for their folders to be reorganized by visiting a dedicated webpage. No issues were identified during that phase so all new accounts are now created with the new folder structure. The next step will be to do an overnight reorganization of all remaining accounts. This should take place by the end of summer.

How to proceed?

If you would like to proactively reorganize your home folder before it is done globally, visit the Check Account Status page at https://cern.ch/WinServices/Services/UserAccounts/AccountStatus.aspx. Click the [Reorganize] link, which is located in the Paths section of the page (figure 2), and follow the instructions. The procedure should take about five minutes and requires that you close all applications. Afterwards it is necessary to log off from your computer and log on again.

Rafal Otto, IT/IS

SLC3 support to end on 31 October

End of the road for SLC3

Scientific Linux CERN 3 (SLC3), the old CERN Linux distribution, will be laid to rest this autumn. SLC3 was certified as the default CERN Linux version back in November 2004. Since SLC4 has been the CERN Linux production version for more than a year (and the default on LXPLUS and LXBATCH for more than six months), general support for SLC3 will end on 31 October. This date was agreed with all the major CERN Linux user communities in February 2006 because it was in line with the end of life of the "upstream" Scientific Linux 3.

This means that from 1 November:

- the Helpdesk will provide no more support for SLC3;
- no more SLC3 installations will be possible via the central install servers;
- and, most importantly, no SLC3 updates (security or otherwise) will be distributed from our servers.

The CERN computing rules (OC5 § II.12, http://cern.ch/computingrules) require that all computers are protected proactively against attacks. Since CERN Linux support will no longer provide security updates for SLC3, it will be up to the machine owners (registered responsibles) to migrate their machines to our supported SLC4 (this is now the preferred Linux distribution, and users will receive automatic updates). Otherwise machine owners will be required to secure their machines, which will involve investing a lot of effort to make sure their machine does not create a security risk for the site.

The registered responsibles for SLC3 machines at CERN were sent several emails in July and will receive several more warnings. We may also have to communicate more directly with the users of affected machines, since experience shows that our warning mails sometimes go unheeded (for example, during the 7.3 phase-out). These warnings are intended to make sure that no SLC3 machines are unintentionally left insecure.

More information about this phase-out campaign and answers to frequently asked questions can be found at http://cern.ch/linux/slc4/docs/migration-campaign.shtml.

SLC4++, or what's next?

SLC4 is the current production version and will be so at the start-up of the LHC. It seems clear that none of the major user groups plans to switch the majority of their machines to a completely new Linux distribution until well after the start-up.

At the same time we are seeing more and more recent hardware that isn't supported by the now two-year-old SLC4; this affects laptops in particular, which are rather short-lived in terms of production lifetime.

Not all desktop models work well with SLC4 either (machines in the computer centre tend to demonstrate fewer problems, since fewer "features" are required to work on these). Even updating our distribution to the "upstream" Scientific Linux 5 will not solve all these issues.

This means that for now we see no urgent need to provide a supported SLC5 distribution, and, depending on our users' timelines, we might decide to skip SLC5 altogether in favour of a future SLC6.

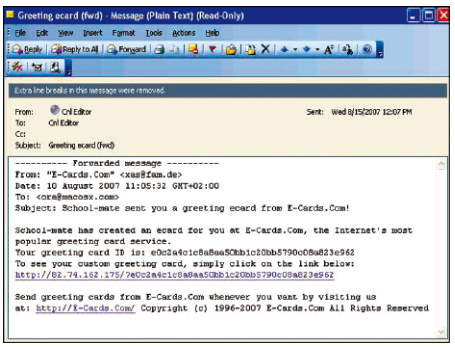
At the time of writing, no decision has been taken and alternative solutions (such as just updating the Linux kernel) are still being discussed.

The CERN Linux team

E-card spam attack installs Trojan horse

We would like to remind computer users that spam emails regularly circulate asking users to click on a link. A recent example was a link to a greetings card that arrived during the summer holiday period. Clicking on the link installed a Trojan horse program that gave the attacker control of the user's computer. Several Windows PCs at CERN were affected and needed to be re-installed from scratch.

Users need to be vigilant about such tricks and remain cautious about clicking on links or opening attachments in any emails they are not expecting. Antivirus systems may not be able to detect all of the latest viruses, which are now highly sophisticated.



Screenshot of the spam greetings card link that breached CERN computer security.

The User Support team

Grid publication iSGTW celebrates 30th issue

The weekly online e-newsletter *International Science Grid This Week*, better known as *iSGTW*, published its 30th issue in August. *iSGTW* is emailed free to more than 3000 subscribers and covers news about Grid and distributed computing, as well as Grid-empowered science and scientific discoveries.

iSGTW is a collaboration between the Open Science Grid and Enabling Grids for E-sciencE, and was launched last November.

You can receive all the latest Grid-related news and announcements by subscribing free at www.isgtw.org.

Calendar

September

17–21 **Cluster 2007 and Grid 2007**
Austin, TX, US
http://cluster2007.org
www.Grid2007.org

October

1–5 **Enabling Grids for E-sciencE, EGEE'07**
Budapest, Hungary
www.eu-egee.org/egee07

15–19 **International Conference on Accelerator and Large Experimental Physics Control Systems, ICALEPCS'2007**
Knoxville, TN, US
www.sns.gov/conf/icalepcs07

15–19 **The 21st Open Grid Forum, OGF21**
Seattle, WA, US
www.ogf.org/OGF21/events_ogf21.php

29–31 **GridNets 2007**
Lyon, France
www.Gridnets.org

November

5–9 **HEPiX Fall 07**
St Louis, MO, US
www.hepix.org

10–16 **International Conference for High Performance Computing, Networking, Storage and Analysis, SC'07**
Reno, NV, US
http://sc07.supercomputing.org

26–30 **MGC 2007**
Newport Beach, CA, US
http://mgc2007.lncc.br