Journal of Physics: Conference Series **331** (2011) 072004 doi:10.1088/1742-6596/331/7/072004

Virtualization for the LHCb Online system

Enrico Bonaccorsi, Loic Brarda, Gary Moine, Niko Neufeld

CERN,

1211 Geneva 23, Switzerland

E-mail: Enrico.Bonaccorsi@cern.ch

Abstract. Virtualization has long been advertised by the IT-industry as a way to cut down cost, optimise resource usage and manage the complexity in large data-centers.
The great number and the huge heterogeneity of hardware, both industrial and custom-made, has up to now led to reluctance in the adoption of virtualization in the IT infrastructure of large experiment installations.
Our experience in the LHCb experiment has shown that virtualization improves the availability and the manageability of the whole system.
We have done an evaluation of available hypervisors / virtualization solutions and find that the Microsoft HV technology provides a high level of maturity and flexibility for our purpose. We present the results of these comparison tests, describing in detail, the architecture of our virtualization infrastructure with a special emphasis on the security for services visible to the outside world. Security is achieved by a sophisticated combination of VLANs, firewalls and virtual routing - the cost and benefits of this solution are analysed.
We have adapted our cluster management tools, notably Quattor, for the needs of virtual machines and this allows us to migrate smoothly services on physical machines to the virtualized infrastructure. The procedures for migration will also be described.
In the final part of the document we describe our recent R&D activities aiming to replacing the SAN-backend for the virtualization by a cheaper iSCSI solution - this will allow to move all servers and related services to the virtualized infrastructure, excepting the ones doing hardware control via non-commodity PCI plugin cards.

1. Introduction

LHCb is a dedicated heavy-flavour physics experiment designed to perform precise measurements of CP violation as well as rare decays of B hadrons Large Hadron Collider (LHC) [1]. The experiment is located at point 8 of the LHC particle accelerator.
As other big experiments in and outside HEP, LHCb depends on a huge and complex IT infrastructure and it tries to solve the common problems of any large IT centre including the server sprawl trend, the scarcity of floor-space, the maintenance costs and server replacement.

The server sprawl trend is based on a decentralized paradigm in which applications and system infrastructure are scaled in a horizontal way: the number of servers implemented within a data center grows at exponential rates as more applications and application environment are deployed.
Natural consequences of the increase in the number of servers are less physical space available in the data center, the number of servers to subsequently replace at the end of the warranty period as well as a greater operating and management efforts.
Taking into account that servers are becoming increasingly powerful, the use of the multi and especially many-core CPUs does nothing but accentuates the sprawl phenomenon.
In order to present a solution to the above problems described the LHCb online team has performed an evaluation of available clustered hypervisors focusing mainly on the free edition of Microsoft core Hyper-V.
The aim of the first phase of study, is the virtualization of the general log-in services (SSH gateways, RDP windows remote desktops, NX Linux remote desktops) as well as the public web services and the essentials infrastructure services (DNS, firewalls and windows domain controllers).
The virtualization of the experiment control PCs, in charge of controlling the detector hardware will be subject of the second phase of study.

2. Virtualization candidates for the LHCb Online System

The LHCb online system, designed to run completely isolated and independent, counts ~1500 servers based on Microsoft windows and scientific Linux, 3 main high density routers and ~100 distributions switches. The only connection to CERN networks and Internet is through the boundary network.
Hosts in the system are grouped as Experiment Control System (ECS) hosts, Data Acquisition (DAQ) hosts and general infrastructure hosts.
LHCb's ECS is in charge of the configuration, control and monitoring of all the components of the online system. This includes all devices in the areas of: data acquisition, detector control, trigger, timing and the interaction with the outside world.
The servers in the ECS network are common data center infrastructure servers (DNS, DHCP, etc) and control PCs that run the standard LHC SCADA system, PVSS, on top of Linux or Windows. While some of them require specific hardware in order to control the experiment such as SPECS cards (special dedicated PCI cards) or USB CANBUS devices, most of them are perfect candidates for migration to virtual platform.

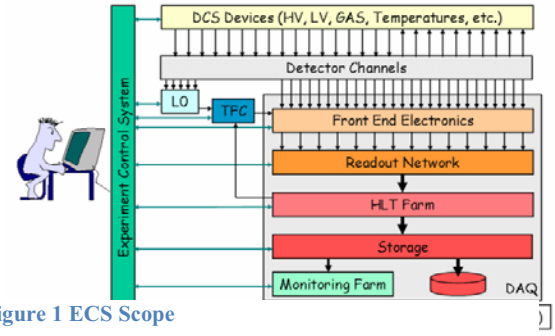

Figure 1 ECS Scope

Category	Candidates
Public availables	Web sites, SSH gateways, Windows and Linux Terminal Services
Common Infrastructure	Firewall, DNS, Domain Controllers
ECS	Control PCs

Table 1 Candidates for virtualization

The role of the DAQ system is to collect the data, zero-suppressed in the front-end electronics, and assemble complete events in CPUs for further data-reduction by the Level-2 and Level-3 triggers [2].
For the time being the high computing resources, necessary to reduce the amount of data acquired from the experiment doesn't make the DAQ servers ideal candidates for a migration to a virtual infrastructure.
Better candidates for virtualization are general infrastructure and public services such as web sites and the Public Login User Service (PLUS). As public available services, they have been virtualized taking care especially of their inherent security aspects: for this
purpose a sophisticated combination of VLANs, firewalls and virtual routing (illustrated in figure 4) has been deployed.

3. Hypervisor

The hypervisor, also called virtual machine monitor, is a virtualization platform that allow multiple operating system to run on a host computer at the same time.
Four hypervisors solutions with clustering and active community supports have been considered: XEN, VMware, KVM and Hyper-V.
While VMware was not considered suitable mainly because of its high license price, XEN has been excluded from the tests because of Red Hat/Scientific Linux choice to do not support it anymore in the next releases in favour of KVM. The free version of Hyper-V and the commercial product System Center Virtual Machine Manager (SCVMM) have been selected for the implementation.

Figure 2. KVM Architecture

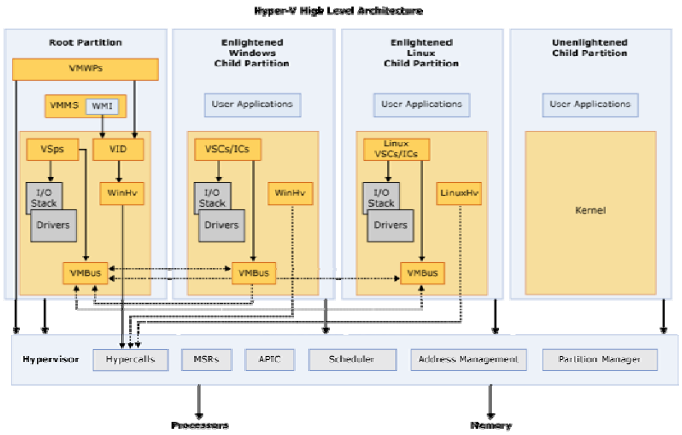

Figure 3. Hyper-V Architecture

4. Hardware & Storage Area Network (SAN)

The first implementation has been deployed on a cluster of ten blade servers Dell Poweredge M610 based on Intel Xeon E5530 processors using Hyper-V failover cluster and SCVMM: each server has eight real cores plus eight virtual CPUs based on hyperthreading. The specifications about the memory and the I/O cards are summarised in table 2.

CPU	2 x E5530 @ 2.4GHz (8 real cores + Hyper Threading)
Memory	3 x 8 GB = 24GB RAM
Network adapters	2 x 10Gb network interfaces (for VLAN sharing, 1 linked to LHCb) 2 X 1Gb network interfaces (1 linked to CERN network, 1 used for cluster communications)
Fiber Channel adapters	2 X 8Gb Fiber channel switches (linked to two isolated fabrics)

Table 2 Hardware Specifications

Each node of the cluster has been connected to the storage area network through a redundant fiber channel connection.
The infrastructure could concurrently support ~260 virtual machines with an average of ~1 GB of memory each one. A logical unit of 10 Tera-Bytes has been allocated in order to store the virtual hard drives. The preferable block size for the LUN in this case would be a multiple of 4 Kilobyte (512 Bytes per 8 disks).
The LUN has been exported to each member of the cluster, setting up the zoning in both fiber channel switches and disk controllers.

5. LHCb virtualized networks

The need to virtualize public services led us to re-design and virtualize the networks as well. In order to protect the online system from potential network attack. According to common security procedures three virtual firewalls based on netfiltes have been put in place in order to isolate virtual networks and demilitirez zones. These are shared between the real machines using VLAN eaated 10Gb/s link.
The two 1Gb/s links are dedicated respectively to cluster management communications and as up-link to CERN network/Internet.
For high-availability reasons LHCb network have been linked trough a 10Gb/s connection per server with a switch uplink the LHCb core router of 20 Gb/s made by two link on two different linecards using the Link Aggregation Control Protocol (LACP).

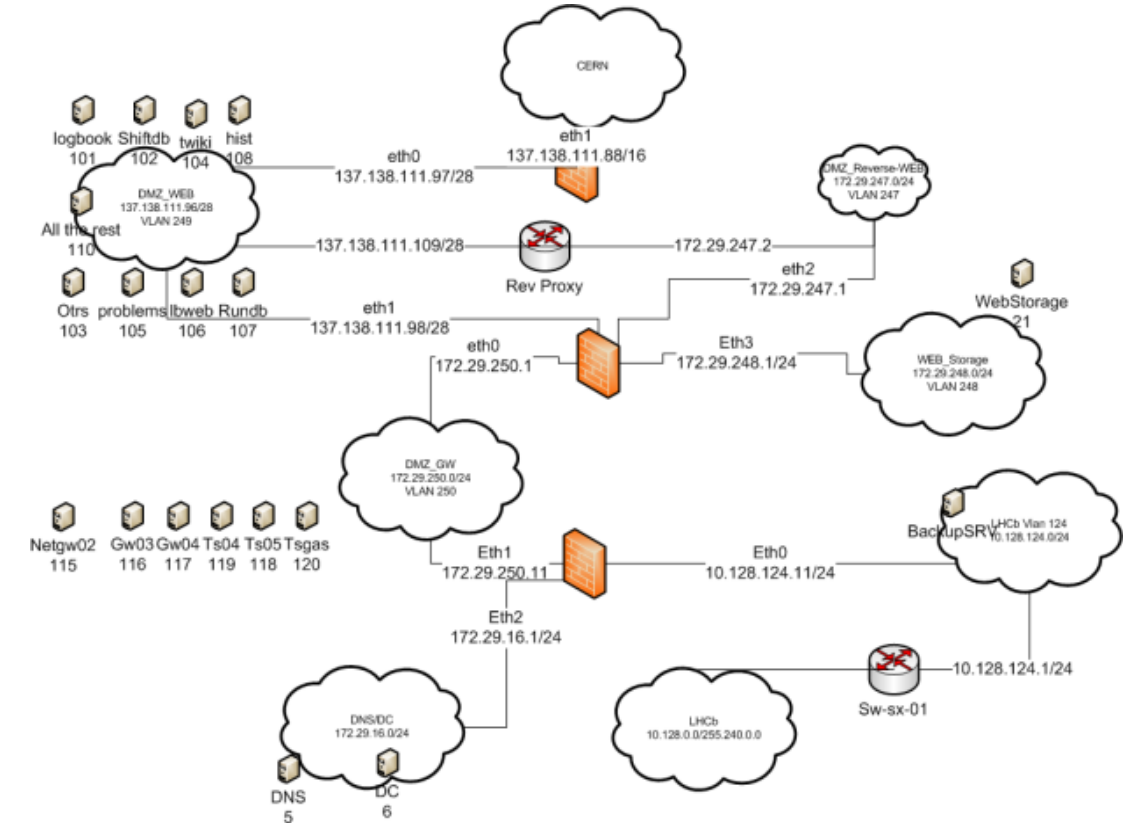

Figure 2 LHCb virtual networks

6. Hyper-V Performances

We measured the network throughput and the network latency from a virtual machine with the paravirtualized drivers installed, to a real server inside the LHCb network linked to the core router. The tests has been done with iperf and ICMP mesuring respectively ~900 Mb/s of throughput and ~0.2 ms of latency.
In the measurement of the disk throughput, since the Microsoft cluster filesystem only supports LUN with 512 Bytes block size, we expect a degradation in the performance also as a results of a not optimized striping.
Taking inot account to drop the cache at each run of the benchmark and disabling it in the storage, we measured a read speed of ~45 MB/s and a write speed of ~35 MB/s far away from the nominal speed of our storage of ~400MB/s.

7. Integration with quattor cluster management tool

Virtualization technologies offer the possibility of cloning machines templates as a quick way to deploy virtual machine. Unfortunately the windows unique identifiers and the static information of the linux /etc configuration directory are cloned as well during the deployment process. The cloned virtual machine in this state is normally not suitable for production before an additional tuning of the all unique data, including static IP addresses. Anyway this process could be automatized with a combination of tools like sysprep and SMS for windows and with a set of SYSV scripts for linux; since a deployment infrastructure for windows and for linux is already in place, a more traditional approach using Preboot Execution Environment (PXE) installation would be preferable. Regrettably the Microsoft hypervisor does not support PXE on the paravirtualized interface; moreover the deployment of a virtual machine by PXE needs about 5 hours because of the slow speed of the fully emulated interface. A solution to reduce the installation time for linux virtual machines has been found mixing the virtual machine cloning method with quattor (system administration toolkit that provides a powerful, portable, and modular set of tools for the automated installation, configuration, and management of linux clusters and farms) [3].

A set of templates, which differ only in hardware allocations, has been created with a minimal installation and specific firstboot scripts; these last ones configure the quattor client software according to the reverse resolution of the virtual machine's IP address. In this way the paravirtualized driver can be used and once the quattor client is configured, the virtual machine starts to configure automatically itself as specified in the quattor server. This method lets us deploy a fully configured linux based virtual machine in less then 10 minutes.

8. Issues

During the tests we encountered many issues, some of them common to virtualization technologies, specific to the hypervisor in use, regarding software licensing linked to the changes of hardware, or related to the hardware in use. One of the main common problem related to virtualization technology is the management of the time: while for the windows guest the problem does not exist (since the microsoft paravirtualized drivers includes a full support to time synchronization with the host operating system), the same maturity level for the linux version of the driver has not been reached yet. The linux virtual machine time is not reliable unless settle for an accuracy around the minute.

Microsoft Hyper-V linux driver does not support multicast and jumbo frames as well as the possibility for the guest to use PCI cards or USB ports while other hypervisors such as KVM do.

We experienced only on the linux virtual machines a degradation after a SCSI timeout or a SCSI Status BUSY that after few retries brings the virtual file systems to be considered as read-only, preventing the Linux guest of most of its operations. This problem does not appear on windows virtual machines where the driver and the filesystem works as expected without stopping the I/O and retrying.

An issue that partially prevents us to virtualize the Control PCs is linked to the software license method of PVSS and the way how Hyper-V abstract the hardware: if the guest machine that runs the PVSS at the moment of the license's check is not running on the host where the license has been generated, PVSS does not recognise the valid license and stops working after 30 minutes. Anyway it could be seen as a minor problem which could be workaround-ed in two ways: in a first way the virtual machine can be started on the right host and then migrate it to another host taking advantage of load balancing; as a second solution we could change the startup script of PVSS to download the correct license file from a central server. In any case on other virtualization platform such as VirtualBox the PVSS licence system works without any hacks.

A rare incompatibility causing a series of blue screen of death of the host operating system has been found in the combination of the Intel 5500 CPU series, ACPI and windows 2008 R2. The problem occurs because incorrect interrupts are generated on the computer that uses Intel processors that are code-named Nehalem as the Intel E5530 cpu[4], microsoft released an hotfix[5] for all 2008 R2 versions except the core editions.

As a result disabling ACPI is the only method to avoid random blue screen.

9. Conclusions

Virtualization can provide a solution to the server sprawl phenomenon with the consolidation of several operating systems on a single server as well as reducing the number of servers to be managed in the data-center and consequentially the hardware maintenance costs.
Virtualization of windows based operating system works without any dramatic problems, the many issues have been tackled and solved, the infrastructure is high-reliable in almost all of the aspects and reasonable secure from the network point of view.
On the opposite the lack of maturity of the microsoft linux integration components - for instance the read-only file system problem that is still not solved yet - has prevented us to start a production phase for the linux based virtual machines.
In parallel we are starting the second phase of the tests in which we will focus on other hypervisor like KVM clustered by pacemaker/corosync, KVM clustered by CMAN/Red Hat Cluster Suite and KVM managed by Red Hat Enterprise Virtualization for Servers. In order to virtualize almost every Control PCs we are looking forward in the encapsulation of the USB protocol over IP in both standalone hardware devices and software solutions.
For the backend we already started tests and benchmark on iSCSI technology as replacement for the expensive fiber channel solution.
In an security-oriented perspective a dedicated resource will design and deploy an intrusion prevention system for the public availabe services in Q2 2001.

References

[1] LHCb Reoptimised Detector Design and Performance TDR, R. Antunes Nobrega et al., LHCb TDR 9, CERN/LHCC/2003-30, 2003.
[2] The LHCb Trigger and Data Acquisition System J.-P. Dufey, M. Frank, F. Harris, J. Harvey,B. Jost, P.Mato, H. Mueller
[3] http://www.quattor.org
[4] Intel Xeon Processor 5500 Specification Update – Reference Number 321324-013
[5] http://support.microsoft.com/kb/975530/en