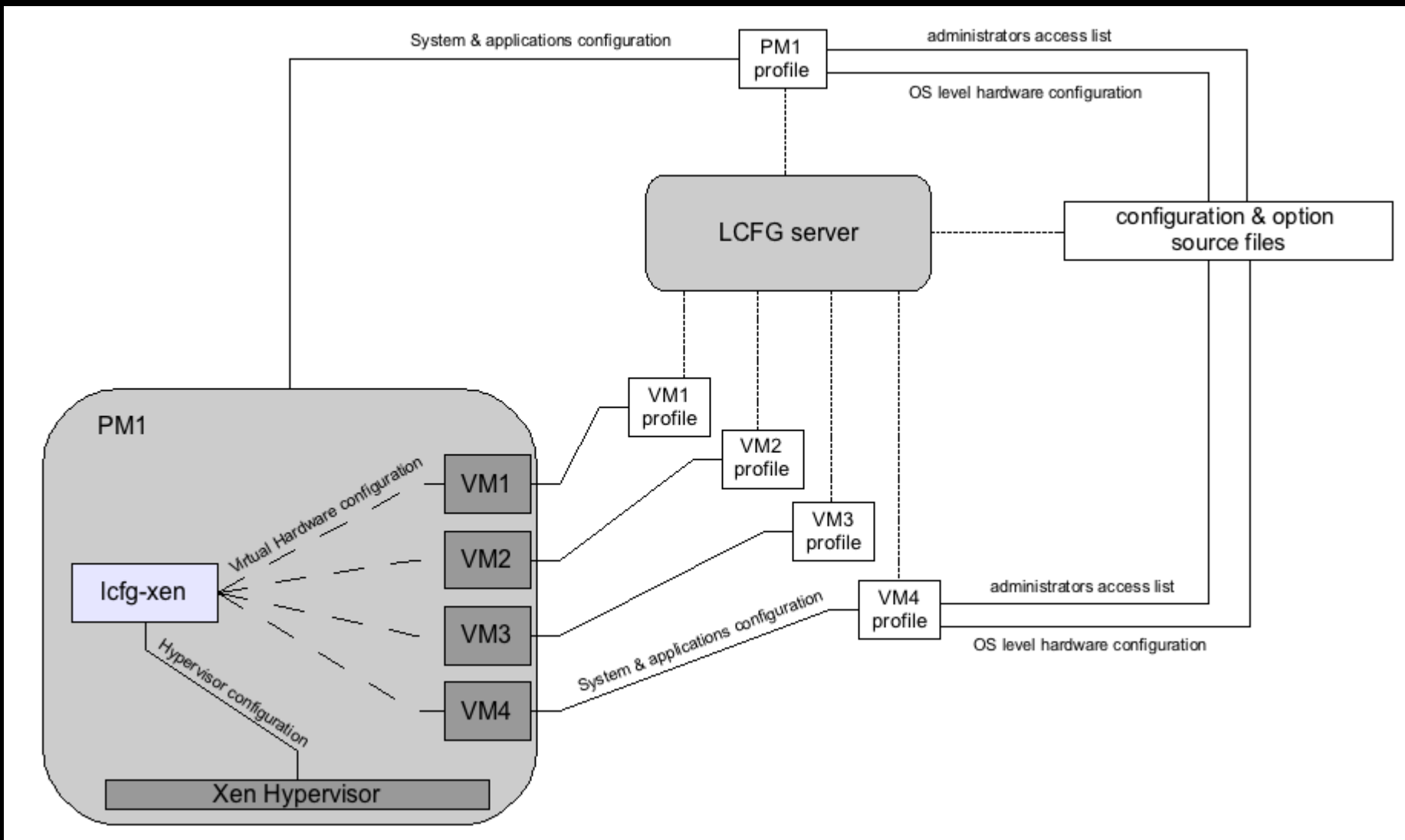


# LCFG meets Virtualisation



Panagiotis Kritikakos  
Computing Support Officer  
School of Physics & Astronomy  
pkritika@epcc.ed.ac.uk

# lcfg-xen



- Physical host management and configuration
- Xen host management and configuration
  - ✓ Hypervisor (xend, xendomains daemons, hardware resources ... )
  - ✓ Networking (custom script support for bridge, NAT and route configurations)
  - ✓ Support for all of the available Xen configuration parameters
- Xen guest management and configuration (configuration file, disk image)
  - ✓ Can be built from scratch
  - ✓ Can be LCFG managed (enabling transparently all the LCFG functionality)
  - ✓ Can be cloned from a vanilla or LCFG disk image
  - ✓ Can use automate boot
  - ✓ Can be fully virtualised or para-virtualised (not yet applicable for LCFG guests)
  - ✓ On-the-fly changes for para-virtualised guests (VCPUs and memory)
  - ✓ Multiple network interfaces
  - ✓ Multiple disk images

## LCFG meets Virtualisation – Use case / Single virtual machines

```
!xen.virtualmachines      mADD(profiledev)
!xen.name_profiledev     mSET(profiledev)
!xen.uuid_profiledev     mSET(bbdda922-8ee3-4ff8-874e-7c4e712dc4c7)
!xen.type_profiledev     mSET(hvm)
!xen.maxmem_profiledev   mSET(512)
!xen.memory_profiledev   mSET(256)
!xen.vcpus_profiledev    mSET(1)
!xen.disks_profiledev    mADD(root)
!xen.diskname_profiledev mSET(profiledev)
!xen.disktype_profiledev mSET(image)
!xen.disksize_profiledev mSET(12)
!xen.diskpath_profiledev mSET(/guests)
!xen.vifs_profiledev     mADD(vif1)
!xen.mac_profiledev_vif1 mSET(00:15:17:96:40:03)
!xen.template_profiledev mSET(/guests/lcfg_temp.img)
!xen.profileurl_profiledev mSET(http://lcfg.ucs.ed.ac.uk/profiles/epcc.ed.ac.uk/)
!xen.partition_profiledev mSET(5)
```

- Independent virtual machines as stand alone systems
- Use of University's public IPs
- Cloned from template image, either LCFG or standard Linux
- Most common scenario

# LCFG meets Virtualisation – Use case / Multi-core Virtual cluster

```
!xen.xendconfbridgenet      mSET (off)
!xen.xendconfnatnet        mSET (on)

!xen.virtualmachines        mADD (node1 ... node8)

!xen.name_node1            mSET (node1)
!xen.type_node1            mSET (paravm)
!xen.uuid_node1            mSET (26d1876a-85fc-49a4-9a5b-929570e18bb65)
!xen.maxmem_node1          mSET (1930)
!xen.memory_node1          mSET (1930)
!xen.cpus_node1            mCONCATQ ("0,1,2,3")
!xen.vcpus_node1           mSET (4)
!xen.disks_node1           mADD (root)
!xen.diskname_node1_root   mSET (node1)
!xen.disktype_node1_root   mSET (image)
!xen.disksize_node1_root   mSET (20)
!xen.diskpath_node1_root   mSET (/data/guests)
!xen.vifs_node1            mADD (vif1)
!xen.nettype_node1_vif1    mSET (nat)
!xen.mac_node1_vif1        mSET (00:16:2a:24:a3:7c)
!xen.ip_node1_vif1         mSET (10.0.0.1)
!xen.boot_node1            mSET (no)
(...)
!xen.name_node8            mSET (node8)
(...)
!xen.ip_node8_vif1         mSET (10.0.0.8)
```

- Multi-core virtual machines able to form as a virtual cluster
- Private IPs with NAT support. Ability to use more than one network setup.
- Use of multiple cores and pinning VCPUs to specific physical CPUs
- Can be cloned from template

## LCFG meets Virtualisation – Use case / Virtual cluster group configuration

```
!xen.virtualgroup          mADD(cluster1)
!xen.members_cluster1     mADD(node1 node2 node3)
!xen.groupmemory_cluster1 mSET(1024)
!xen.groupmaxmem_cluster1 mSET(2048)
!xen.groupcpus_cluster1   mSET(3)
!xen.groupvcpus_cluster1  mSET(4)
!xen.groupbridge_cluster1 mSET(xenbr0)

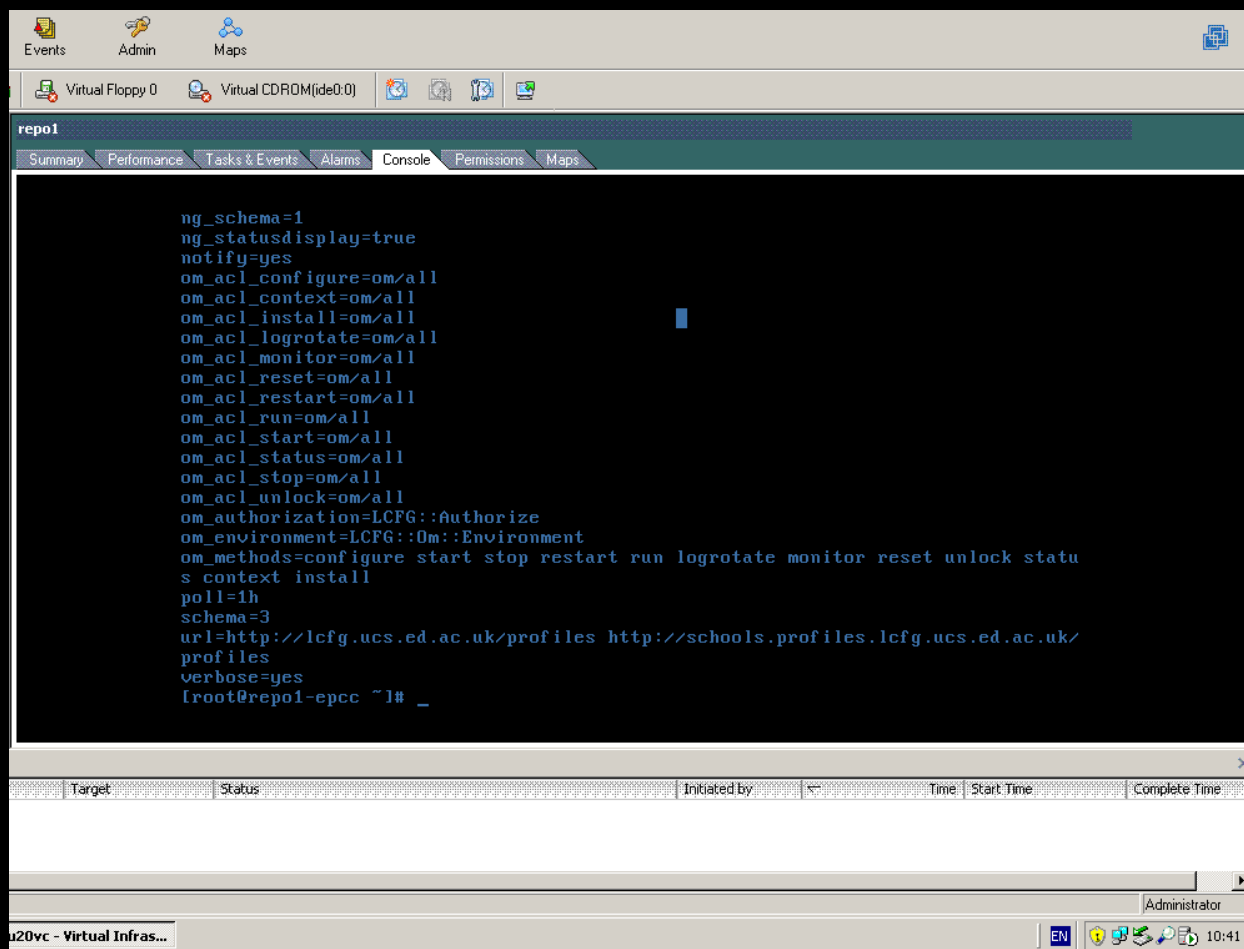
!xen.virtualgroup          mADD(cluster2)
!xen.members_cluster2     mADD(node4 node5)
!xen.groupvcpus_cluster2  mSET(2)
!xen.groupcpus_cluster2   mSET(2)
!xen.groupbridge_cluster2 mSET(xenbr1)

!xen.virtualgroup          mADD(cluster3)
!xen.members_cluster3     mADD(node6 node7 node8)
!xen.groupcpus_cluster3   mADD(2)
!xen.groupnettype_cluster3 mSET(nat)
!xen.groupscript_cluster3 mSET(vif-nat)
```

- Configure hardware and network resource on per group basis
- Dynamic reconfiguration of virtual group members
- Nodes can use generic hardware configuration and individual network configuration or vice versa
- Nodes may not be part of an actual virtual cluster but share common hardware/network specifications

# LCFG meets Virtualisation – Use case / LCFG managed hosts on VMware ESX

- Managing only virtual hosts
- Manual procedure of initialise virtual machine configuration
- Manual procedure of initial virtual machine installation (PIE, PXE, ...)
- Use of VMware Workstation headers for virtual host hardware configuration



The screenshot shows a VMware Workstation console window for a virtual machine named 'repo1'. The console displays the following configuration text:

```
ng_schema=1
ng_statusdisplay=true
notify=yes
om_acl_configure=om/all
om_acl_context=om/all
om_acl_install=om/all
om_acl_logrotate=om/all
om_acl_monitor=om/all
om_acl_reset=om/all
om_acl_restart=om/all
om_acl_run=om/all
om_acl_start=om/all
om_acl_status=om/all
om_acl_stop=om/all
om_acl_unlock=om/all
om_authorization=LCFG::Authorize
om_environment=LCFG::Om::Environment
om_methods=configure start stop restart run logrotate monitor reset unlock statu
s context install
poll=1h
schema=3
url=http://lcfg.ucs.ed.ac.uk/profiles http://schools.profiles.lcfg.ucs.ed.ac.uk/
profiles
verbose=yes
lroot@repo1-epcc ~]# _
```

Below the console output, there is a table with columns: Target, Status, Initiated by, Time, Start Time, Complete Time. The table is currently empty.

The VMware Workstation interface shows the 'Virtual Floppy 0' and 'Virtual CDROM(ide0:0)' devices. The bottom status bar indicates the user is 'Administrator' and the time is 10:41.

## LCFG meets Virtualisation – Use case / Multitple network configuration

```
!xen.xendconfbridgenet          mSET (on)
!xen.xendconfnatnet             mSET (on)

!xen.xendconfcustombridge       mSET (my-network-bridge)
!xen.bridgeinterfaces           mADD (eth0 eth1 eth2)
!xen.bridgenetdev_eth0         mSET (eth0)
!xen.bridgeif_eth0             mSET (xenbr0)
!xen.bridgevifnum_eth0         mSET (0)

!xen.bridgenetdev_eth1         mSET (eth1)
!xen.bridgeif_eth1             mSET (xenbr1)
!xen.bridgevifnum_eth1         mSET (1)

!xen.bridgenetdev_eth2         mSET (eth2)
!xen.bridgeif_eth2             mSET (xenbr2)
!xen.bridgevifnum_eth2         mSET (2)
```

- Ability to support more than one network setup per physical host
- Ability to support different network types – bridge, NAT, routed
- Attaching virtual machines to different networks as per project basis



## LCFG meets Virtualisation – Future development

- Development of a libVirt component – *lcfg-libvirt*
- Support for Xen (transferring functionality from *lcfg-xen*)
- Support for Qemu / KVM (RHEL 6 / SL6 will come with KVM by default)
- Ability to extend the component and add support for
  - OpenVZ (container system)
  - UML – User Mode Linux (paravirtualisation support)
  - Virtual Box (hypervisor)
  - LXC – Linux Container (container system)

*“One Component to rule them all”*



## LCFG meets Virtualisation – Future development

- Do we get to a setup of a private “Cloud” ?
  - Virtual machines can be requested
  - However, it’s not the user who creates it, but the sysadmin
  - No fancy tools and use of web services (still XML and HTTP are core elements in LCFG infrastructure)
- The “Cloud” is nothing more than the “Internet” – A network of networks
  - There was always the ability to have a webapp running on a remote system that belongs to company X, served by DNS Y. The webapp was in the “Cloud”. But there was no “Cloud” as we mean it today.
  - The shape of a cloud was always representing the Internet
  - Intercommunicative multiple applications forming a system that communicates and is accessible via Web Services
  - In the aspect that Web 2.0 introduced user generated content, Cloud-Computing introduces user generated services
- Open Source Cloud Computing Infrastructure – Could LCFG manage these ?
  - Eucalyptus
  - OpenNebula
  - Globus Nimbus



THANK YOU



**Presentation**

<http://www2.epcc.ed.ac.uk/~pkritika/lcfg/lcfgmv.pdf>