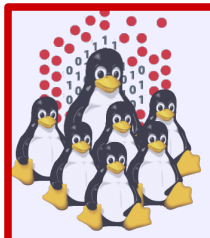
Moreno Baricevic

CNR-IOM DEMOCRITOS
Trieste, ITALY

eLab
Joint DEMOCRITOS/SISSA
Laboratory for e-Science

ePICO

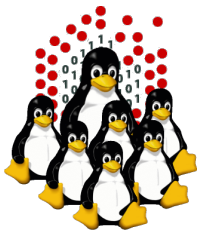
e-Lab Procedure for Installation and COnfiguration



Agenda

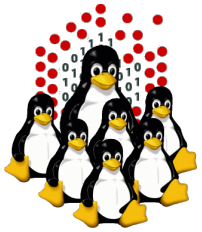
- What is EPICO?
- Why should I use it?
- How to handle complexity
- Profiles, Subprofiles, Pools
- Download, 3rd party, installation and deployment
- Utils and customization
- TODO

- Hands-on Laboratory Session



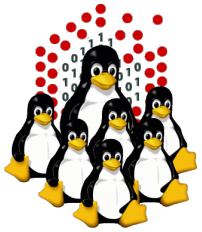
What is EPICO?

- Framework for unattended and distributed deployments of LINUX, focused on the post-{installation,configuration} for HPC
- Collection of procedures, scripts and tricks, built brick-by-brick, whenever new hardware was introduced, in >10 years of on-the-field experience
- Fruit of the experience and requirements on extremely heterogeneous clusters (>250 nodes and ~20 HW/SW profiles @eLab)
- Based on open standards, well-known protocols, open/free tools and standard procedures
- Flexible and customizable, as well as complex
- Open and Free (as in free beer and as in freedom)
- Based and tested on RPM-based distros using Anaconda/Kickstart, even though scripts and procedures should work with other distributed installers too (e.g. FAI)
- Aimed at experienced LINUX system administrators, or users with some knowledge of the services involved (PXE, DHCP, DNS, NFS, RPM-based package repositories, queue systems) and scripting experience



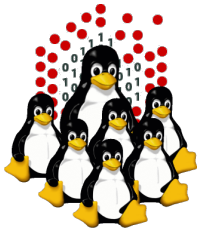
Why should I use it?

- extremely customizable, whole package as plain text scripts
- unattended deployment of a single ad-hoc machine as well as large heterogeneous clusters
- used to install:
  - >250 nodes with ~20 HW/SW profiles @SISSA
  - >100 nodes with 6 HW/SW profiles @TEMPLE
  - >50 nodes with 5 HW/SW profiles @MERCURIO
  - 2 HPC clusters @AAU
  - 1 HPC cluster @SPIN
  - 2 GRID clusters @eLab



How to handle complexity

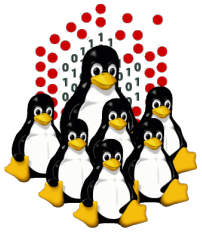
- logical profiles
  - masternode
  - I/O server
  - computing node
  - management/monitoring node, workstation, ...
- hardware/software profiles, pools and subsets
  - ata/sata/sas/scsi hard disks, w/ or w/o raid, attached to NAS, SAN, or just diskless
  - infiniband/myrinet/gigabit network, bonding
  - amd/intel, cpu/gpu
  - grid sw
  - ...



PROFILES

- subset of machines identified and divided by typology, purpose or major differences:
  - master
  - iosrv
  - nodes
  - diskless
  - wks
- handled as:

  /distro/epico/include/profiles.d/<PROFILE>/



POOLS

- subset of machines with:
  - common characteristics
  - similar/identical hardware or special purpose
- identified by hostname / IP / subnet, DNS TXT entry:
  - iosrv-nas, iosrv-san, storage01 – storage04
  - node01 – node20, amd01 – amd20, gpu01 – gpu06
  - “p001 IN TXT planck” (as in /var/named/data/nfs.db):

```
# host -t TXT p001.nfs
p001.nfs descriptive text "planck"
```

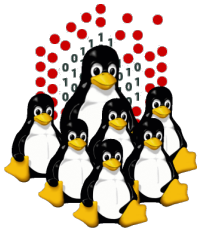
- handled as:

  .../<PROFILE>/hosts.d/<HOSTNAME>/

  .../<PROFILE>/hosts.d/<POOL>/

  .../<PROFILE>/hosts.d/<HOST1> –> <POOL>

  .../<PROFILE>/hosts.d/<HOST2> –> <POOL>



SUBPROFILES

- ad-hoc installations of singular machines w/ minor differences related to network settings, partitioning, ...
- handled by kickstart %include files extension as defined on kernel cmdline (if the file exists, fallback to default otherwise):

  EPICO_KSEXT=*ictp2011* (kernel cmdline)

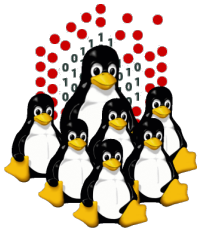
  -> %include ksinclude.partition.*ictp2011*

  -> %include ksinclude.network.*ictp2011*

  -> ...

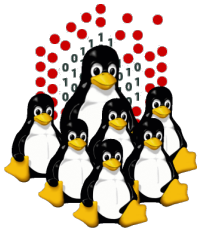
  (vs. default “ksinclude.partition”, ...)

- post-installation script executed at the end, if available:
  - .../scripts/custom.*ictp2011*



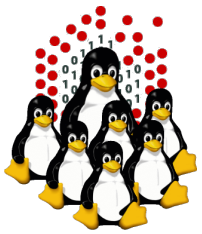
Summary

- **Profiles**
  - master, nodes, iosrv, ...
  - → EPICO_PROFILE=<PROFILE>
  - ➢ /distro/epico/profiles.d/<PROFILE>/
- **Subprofiles**
  - master@ICTP, master@TEMPLE, master@AAU, ...
  - → EPICO_KSEXT=<EXTENSION> EPICO_KS*=...
    EPICO_KSPART={<EXTENSION>|ask} EPICO_KSNET={<EXTENSION>|ask}
    EPICO_KSPASS=<PASSWORD> EPICO_KSTMZ=<TIMEZONE>
  - ➢ /distro/epico/include/.../ksinclude.*.<EXTENSION>
- **Hosts**
  - iosrv, node01, storage03, ...
  - → `hostname -s` or EPICO_HOST=<HOSTNAME>
  - ➢ /distro/epico/include/profiles.d/<PROFILE>/hosts.d/<HOSTNAME>/
- **Pools**
  - GPU, gpu01 –> GPU, gpu02 –> GPU
  - → `hostname -s` or EPICO_HOST=<HOSTNAME>
  - ➢ /distro/epico/include/profiles.d/<PROFILE>/hosts.d/<POOL>/
    /distro/epico/include/profiles.d/<PROFILE>/hosts.d/<HOSTNAME> –> <POOL>/



Customization layers

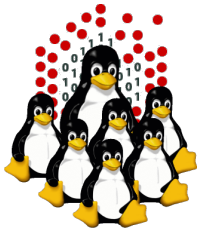
- common kickstart file
- %pre and %post externalized in a logical tree based on profiles/hosts or common defaults
- pxe configuration, EPICO keywords provided as kernel cmdline arguments in order to define profiles/subprofiles or force specific hosts
- DNS configuration (hostnames and TXT) to define pools, subset or customize by hostname/IP
- scripts that will be executed or not depending on a task list specific for each profile or host
- routines that check for hardware availability (e.g. presence of infiniband card)



How it works

From the first stage of the network boot-up, the EPICO server supplies:

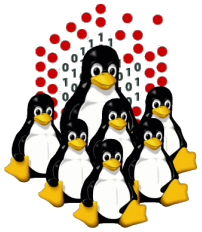
- DHCP information
- PXE configuration file
- kernel/initrd + kernel cmdline options
- Kickstart file
- Kickstart includes based on IP/hostname/profile
- Packages repository (base + extras)
- pre-installation (%pre) and post-installation (%post) scripts for the customization based on IP/hostname/profile
- Post-boot procedure (startup script executed at each boot)
- RAMDISK integration (diskless nodes)



Main services involved

- PXE: network booting
- DHCP: IP binding + NBP (pxelinux.0)
- TFTP: pxe configuration file (pxelinux.cfg/<HEXIP>), alternative boot-up images (memtest, UBCD, ...)
- NFS: kickstart + RPM repository (with little modification can be adapted to FTP/HTTP(S) based repos)

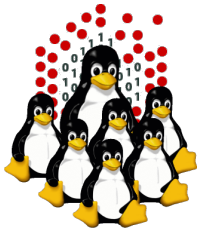
- POST-BOOT: uses port-knocking, ssh, c3-tools
- CONFIGURATION/PACKAGE UPDATE: uses ssh, c3-tool



Web resources and download

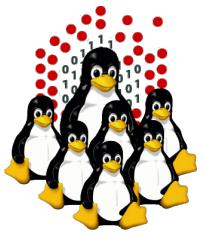
- http://epico.escience-lab.org
- http://eforge.escience-lab.org/gf/project/epico/

```
svn co --username anonymous --password anonymous \
https://eforge.escience-ab.org/svn/epico/trunk/distro
```

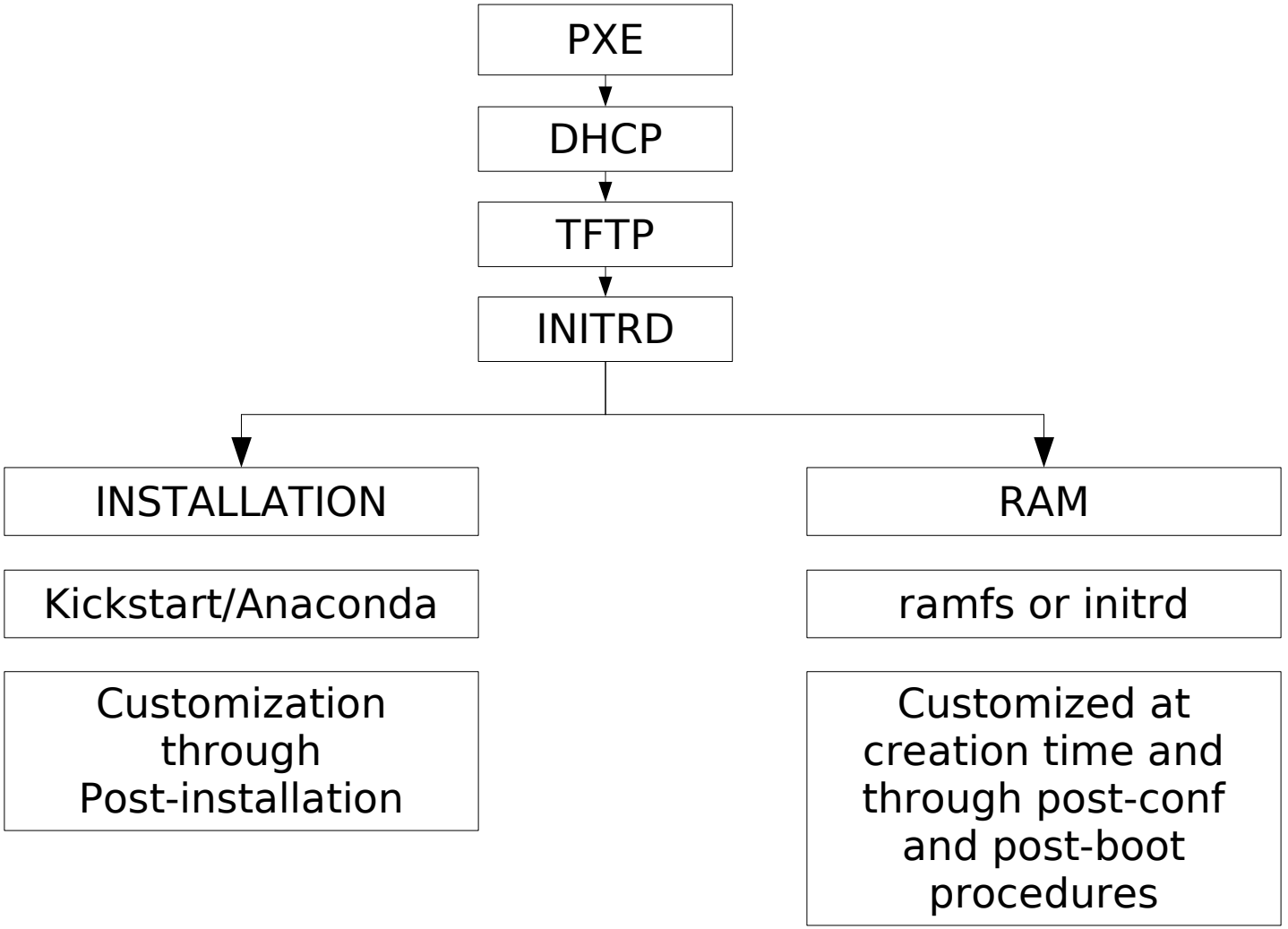


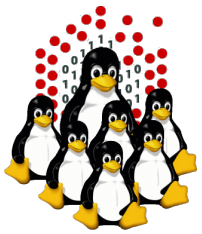
How to install an EPICO server

- USB key + RH/CentOS DVD (boot+RPM repo)
- USB key + RH/CentOS CD (boot) + RPM REPO
- USB key + PXE (boot) + RPM REPO
- from an already available EPICO SERVER (PXE boot + EPICO over NFS + RPM REPO)
- direct deployment on any LINUX machine (if configured properly), whatever the distro

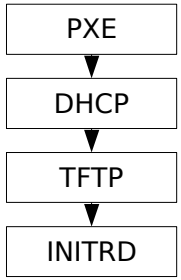


Network-based Distributed Installation
Overview

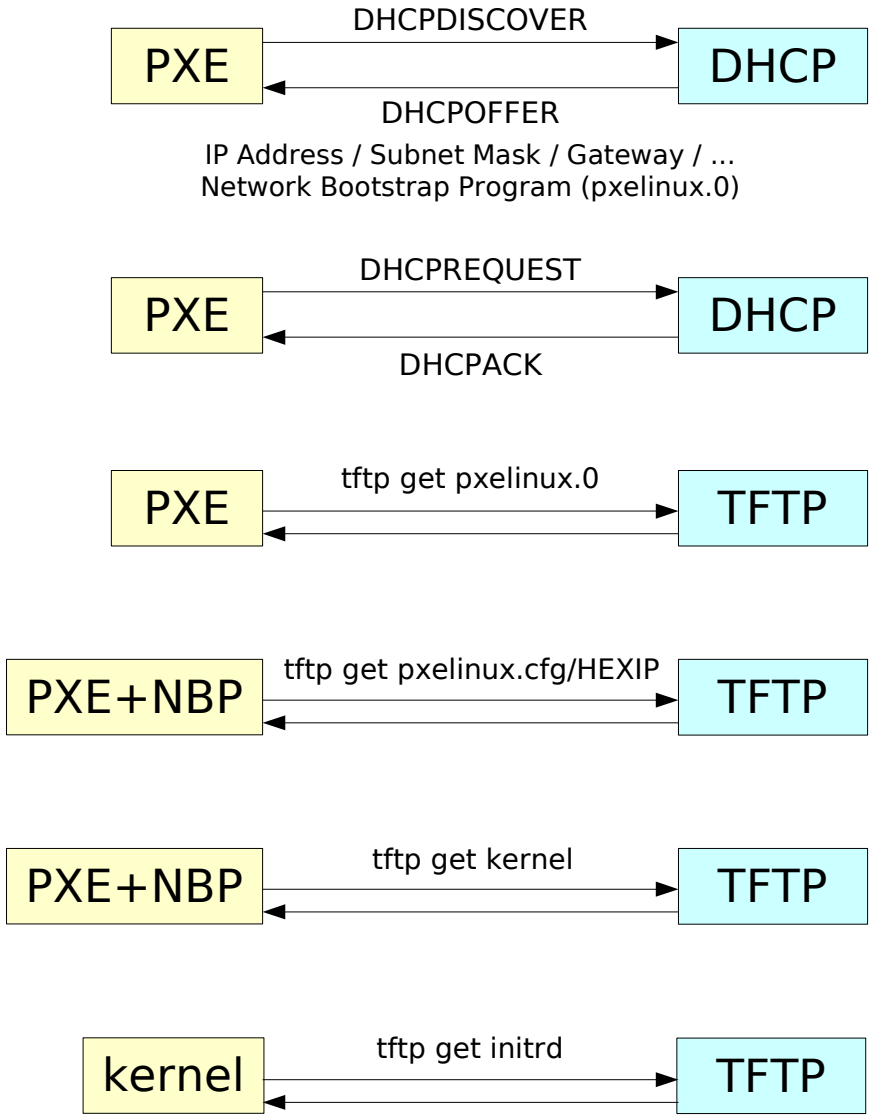




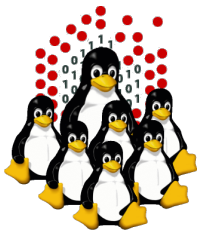
Network booting (NETBOOT)
PXE + DHCP + TFTP + KERNEL + INITRD



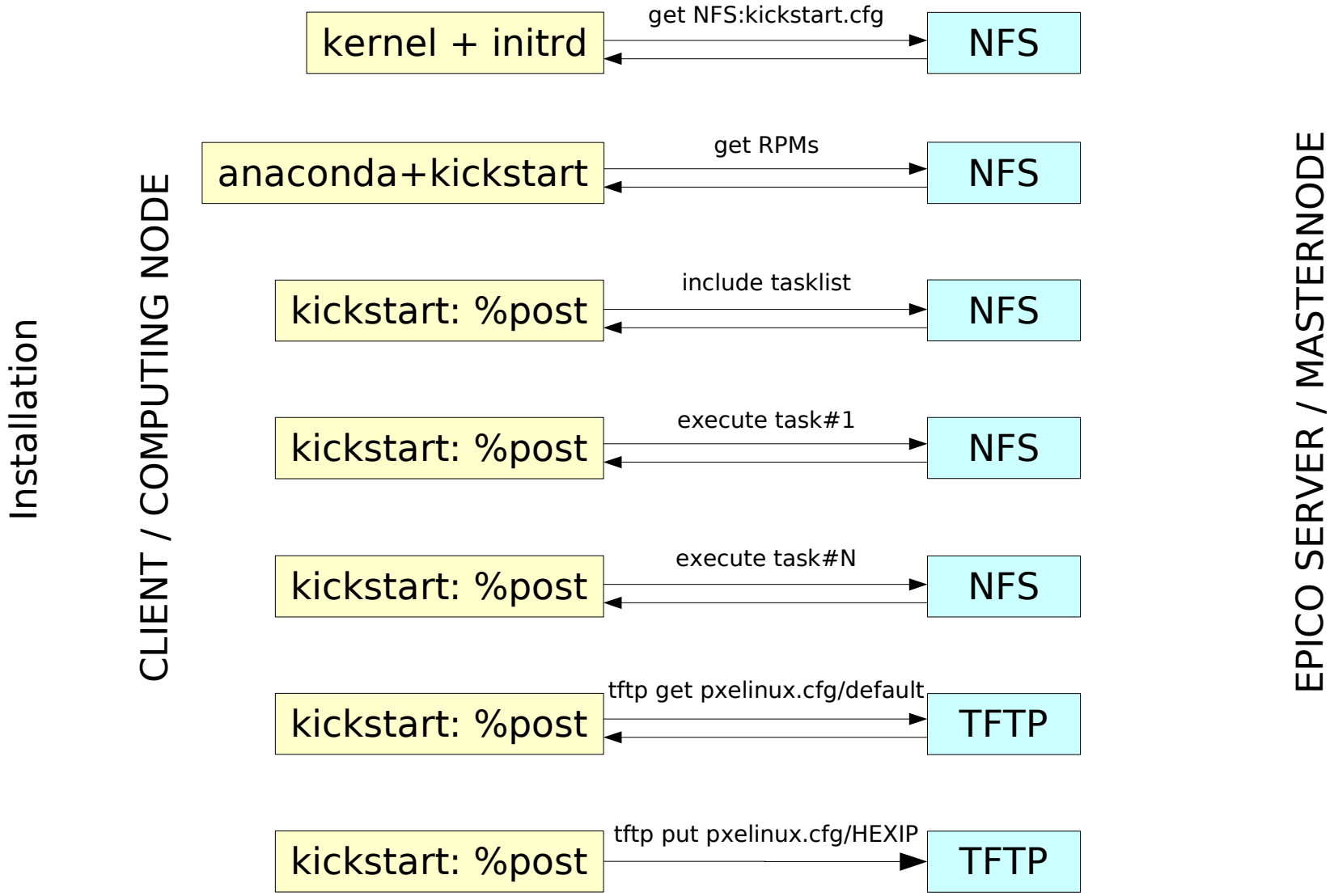
CLIENT / COMPUTING NODE

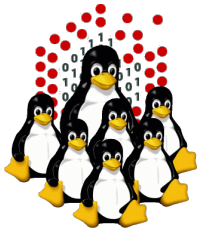


EPICO SERVER / MASTERNODE



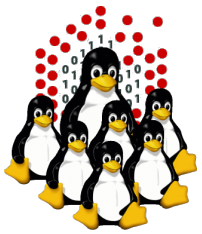
Network-based Distributed Installation
NETBOOT + KICKSTART INSTALLATION





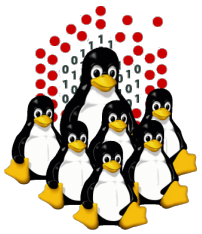
Main tree

/distro/epico/	top dir
/distro/epico/include/	profiles, ksincludes, rpmlists, tasklists, scripts
/distro/epico/ks/	kickstart files
/distro/epico/rc/	scripts loaded @ %pre and %post
/distro/epico/tars/	partial trees for services and configurations to be installed
/distro/epico/bin/	utils and wrappers (addnode.sh, show-install.sh, debug-stage.sh, lshex, ...)
/distro/epico/sbin/	post-boot master daemon
/distro/epico/post-boot/	post-boot procedure
/distro/epico/doc/	various READMEs
/distro/epico/ramdisk/	ramdisk creation, utils, ...
/distro/epico/queue/	torque, maui, pbsacct/quart files
/distro/extras/	additional RPM packages



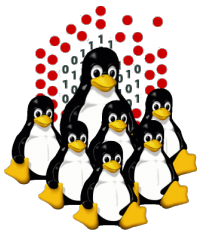
Other fs involved (see tars directory)

/etc/dhcpd.*	DHCP configuration
/etc/sysconfig/dhcpd	dhcpd options
/etc/named.*	DNS main configuration
/var/named	DNS zones configuration
/etc/resolv.conf	hostnames/IPs resolution
/etc/xinetd.d/tftp	in.tftpd startup handled by xinetd
/tftpboot/	PXE/TFTP related files and boot images
/distro/centos/	RPM repository
/etc/exports	NFS exports
/etc/sysconfig/iptables	firewall and NAT configuration
/etc/ntp.*	NTP configuration



Kickstart %include & post-install

ksinclude.partition	hard drives partitioning, bootloader
ksinclude.network	network settings, firewall
ksinclude.passwd	superuser's password
ksinclude.timezone	timezone (CET, Europe/Rome, Africa/Addis_Ababa)
ksinclude.misc	xconfig, optional ks parameters
rpmlist.install	list of rpm packages and groups to install
rpmlist.extras	extra packages
rpmlist.remove	unwanted packages to be removed
post-install.list	tasklist, list of post-configuration scripts to execute
post-install.env	some environment variables that might affect scripts behavior
scripts/	bash scripts to execute



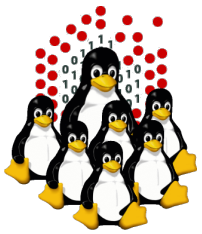
Fallback procedure

ks include files, rpmlists, tasklist, scripts

1. ad-hoc (subprofile, ksinclude extension)
2. host/pool specific
   (include/profiles.d/<PROFILE>/hosts.d/<HOST|POOL>)
3. default by profile
   (include/profiles.d/<PROFILE>/default/)
4. common (include/common/)

/distro/epico/bin/epico-show-install.sh

wrapper to verify install fallbacks as used during installation
(based on profile and host)



Default hardcoded configuration

Management “.sp” network: 10.1.0.0/16

Gigabit “.nfs” network: 10.2.0.0/16

Infiniband “.ib” network: 10.3.0.0/16

External interface: eth0

Internal interface: eth1

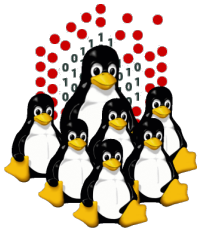
Masternode IP/hostname:

10.2.0.1/master.nfs
10.3.0.1/ib-master.ib
10.1.0.1/master-sp.sp
10.1.0.255/sp-master.sp (IPMI interface)

Predefined nodes:

node01-node08, 10.{1,2,3}.1.{1-8}, TXT=blade

(of course, everything can be modified)



Debug and Troubleshooting

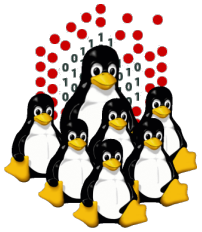
$ telnet <HOSTNAME/IP> 9000
username: *epico*
password: *debug*

$ nc -v <HOSTNAME/IP> 9001
epico-debug

epicoshell (interactive), if EPICO_DEBUG=1 given

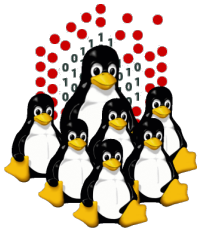
epico-debug-stage.sh (ping / tcpdump / tshark)

logs (DHCP, TFTP, NFS mount requests)



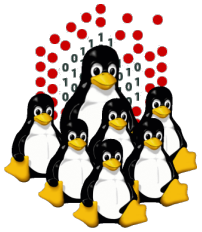
3rd party and contrib

torque	http://www.clusterresources.com/downloads/torque/
gold	http://www.clusterresources.com/downloads/gold/
maui	http://www.clusterresources.com/ (can't be downloaded without registration)
openmpi	http://www.open-mpi.org/
ganglia	http://ganglia.sourceforge.net/
nagios	http://www.nagios.org/
g95	http://ftp.g95.org/
UBCD	http://www.ultimatebootcd.com/
QUART	http://eforge.escience-lab.org/gf/project/quart/
C4	http://eforge.escience-lab.org/gf/project/c-4/
LazyBuilder	http://eforge.escience-lab.org/gf/project/lazybuilder/



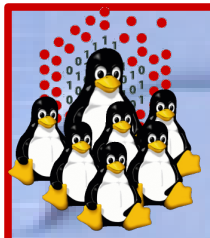
Open issues

- Complex structure, allow flexibility but some knowledge about the services involved is a requirement, as well as scripting experience
- Some part of the configuration is hard-coded and must be manually modified, configurator/wrapper needed
- Switching internal and external name of the masternode lead to some issues for torque/maui (need reconfiguration, a wrapper might help)
- Redistribution of 3rd party and contribs (non-open licenses)



TODO

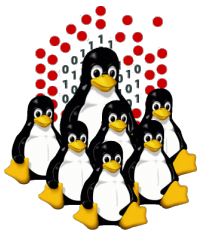
- Documentation, documentation, documentation
- Configurator, installer, wrappers
- RH/CentOS 6 support
- Support for other distributions (Fedora, Debian, Ubuntu)
- EPICO as RPM package(s)
- BOOT from usbkey (self-consistent installer + pkgs repo)
- Improve HD/USB disk-overy and autopartitioning
- ...
- Text-based and graphical UI



Hands-on Laboratory Session

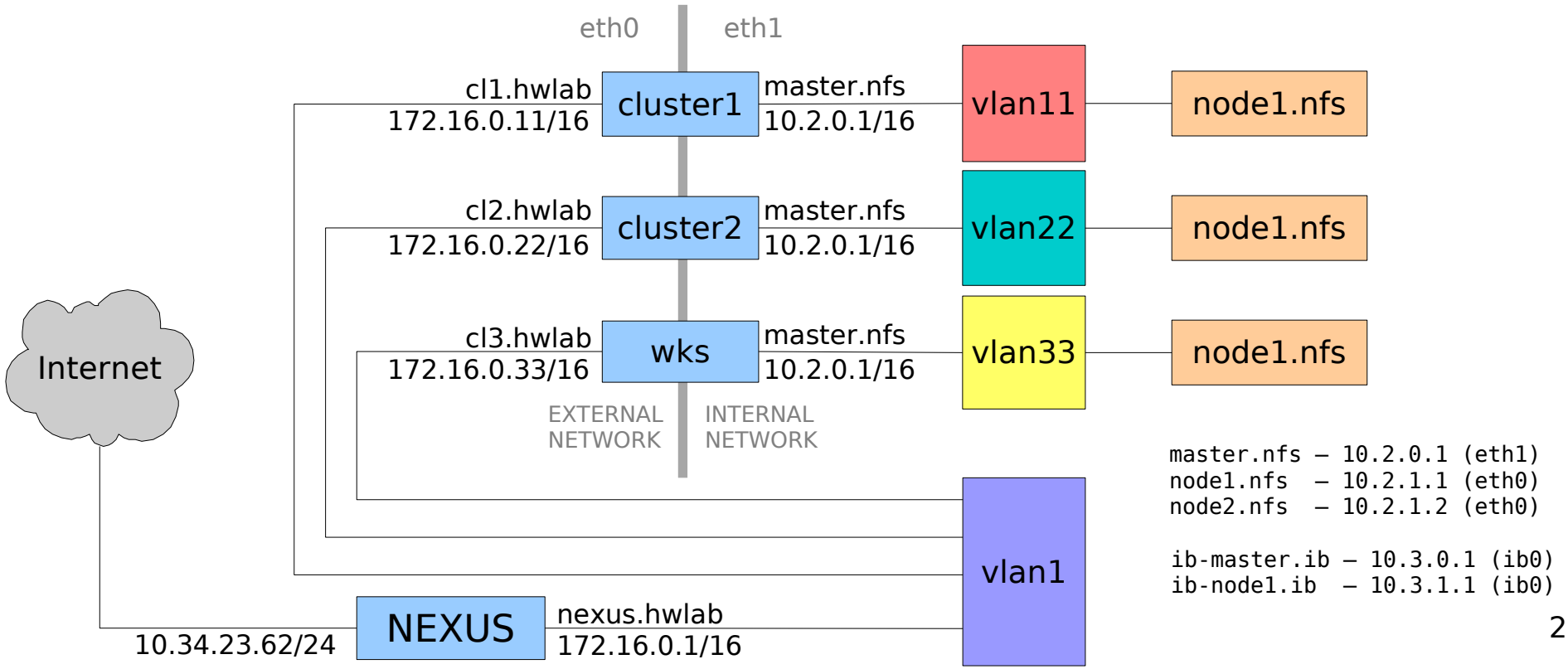
http://epico.escience-lab.org

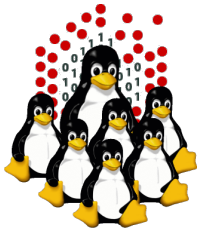
http://eforge.escience-lab.org



Hands-on Laboratory Session

- 2 clusters made of 1 masternode (IBM) + 1 computing node each (SUN v20z)
- 1 workstation + 1 client/computing node
- 1 storage cluster made of 1 masternode, 3 computing nodes, 4 storage nodes as frontend to a SAN

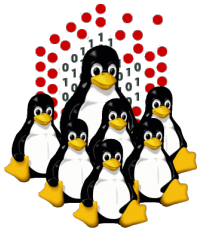




That's All Folks!


```
( questions ; comments ) | mail -s uheilaaa baro@democritos.it

( complaints ; insults ) &>/dev/null
```



REFERENCES AND USEFUL LINKS

Cluster Toolkits:

- EPICO – eLab Procedure for Installation and COnfiguration
  http://epico.escience-lab.org
- OSCAR – Open Source Cluster Application Resources
  http://oscar.openclustergroup.org/
- NPACI Rocks
  http://www.rocksclusters.org/
- Scyld Beowulf
  http://www.beowulf.org/
- CSM – IBM Cluster Systems Management
  http://www.ibm.com/servers/eserver/clusters/software/
- xCAT – eXtreme Cluster Administration Toolkit
  http://www.xcat.org/
- Warewulf/PERCEUS
  http://www.warewulf-cluster.org/ http://www.perceus.org/

Installation Software:

- SystemImager http://www.systemimager.org/
- FAI http://www.informatik.uni-koeln.de/fai/
- Anaconda/Kickstart http://fedoraproject.org/wiki/Anaconda/Kickstart

Management Tools:

- C3 tools – The Cluster Command and Control tool suite
  http://www.csm.ornl.gov/torc/C3/
- PDSH – Parallel Distributed SHell
  https://computing.llnl.gov/linux/pdsh.html
- DSH – Distributed SHell
  http://www.netfort.gr.jp/~dancer/software/dsh.html.en
- ClusterSSH
  http://clusterssh.sourceforge.net/
- C4 tools – Cluster Command & Control Console
  http://gforge.escience-lab.org/projects/c-4/

Monitoring Tools:

- Ganglia http://ganglia.sourceforge.net/
- Nagios http://www.nagios.org/

Network traffic analyzer:

- tcpdump http://www.tcpdump.org
- wireshark http://www.wireshark.org

RFC: (http://www.rfc.net)

- RFC 1350 – The TFTP Protocol (Revision 2)
  http://www.rfc.net/rfc1350.html
- RFC 2131 – Dynamic Host Configuration Protocol
  http://www.rfc.net/rfc2131.html
- RFC 2132 – DHCP Options and BOOTP Vendor Extensions
  http://www.rfc.net/rfc2132.html
- RFC 4578 – DHCP PXE Options
  http://www.rfc.net/rfc4578.html
- RFC 4390 – DHCP over Infiniband
  http://www.rfc.net/rfc4390.html

- PXE specification
  http://www.pix.net/software/pxeboot/archive/pxespec.pdf
- SYSLINUX http://syslinux.zytor.com/