Customizing Your Cluster

Case Studies

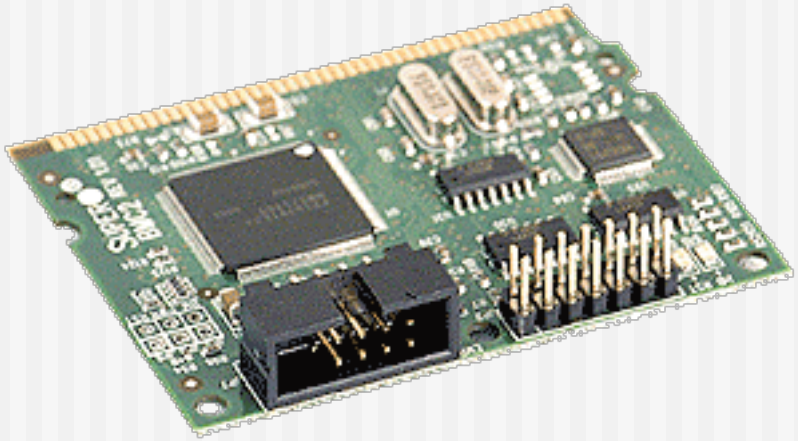
- Command Line
- Attributes
- Roll Innards

Case study in RCL

IPMI

IPMI – Intelligent Platform Management Interface

- Available (free or low cost) on most modern servers
- Passive monitoring of sensors (temp, fan speed)
- Active control of power (on/off/reset)
- ~~It's a Standard~~



Networking

- Dedicated NIC
  - Does you BMC have its own Ethernet port
  - Preferred since you can isolate network traffic.
- Shared NIC
  - You BMC will still have its own MAC address
  - Traffic will bridge over another Ethernet port
  - Bridging over eth0 (private network) makes sense

Step 1: Add a Network

- Every IPMI interface needs an IP address
- To isolate the traffic it should have a distinct subnet
- 192.168.0.0 / 255.255.0.0 is unused by default (your cluster may differ)

CONFIGURING IPMI

ADD NETWORK

```
rocks add network <network-name>
  <network>
  <subnet>

rocks add network ipmi
  192.168.1.0
  255.255.255.0
```

LIST NETWORK

```
NETWORK     SUBNET          NETMASK
private:    10.12.0.0       255.255.0.0
public:     169.228.3.0     255.255.255.240
ipmi:       192.168.1.0     255.255.255.0
```

Step 2: Add the interface

- Host first must be installed
- Then secondary NICs can be added
- After all hosts are configured just re-install

ADD HOST INTERFACE

```
rocks add host interface
  <host> <iface>
  ip=<address> subnet=<name>
  gateway=<address>
  name=<hostname>

rocks add host interface
  compute-0-0 ipmi
  ip=192.168.1.1 subnet=ipmi
  gateway=1 name=ipmi-0-0
```

LIST HOST INTERFACE

```
rocks list host interface compute-0-0
SUBNET    IFACE     MAC               IP            NETMASK       GATEWAY MODULE NAME
private   eth0      00:15:17:79:d3:c0 10.12.0.12    255.255.0.0   ------- e1000e compute-0-0
ipmi      ipmi      ----------------- 192.168.1.2   255.255.255.0 2       ------ ipmi-0-0
```

Module Parameter

- Is used to specify the IPMI channel
  - This will change in 5.4 (use new “channel” field)
- The channel indicates the NIC IPMI will use
- Channel 1 is `eth0`
- `You BMC may be different, read your motherboard docs`

```
rocks add host interface
  compute-0-0 ipmi
  ip=192.168.1.1 subnet=ipmi
  module=1 name=ipmi-0-0
```

Step 3: Re-install

- PXE Boot
  - Network Boot is first in BIOS boot order
  - Set Rocks Boot action to install
  - Reboot the host
- Otherwise use old rocks commands or just hard power cycle the host.

SET HOST BOOT

```
rocks set host boot
  <host>
  action=<boot-action>

rocks set host boot
  compute-0-0
  action=install
```

RUN HOST

```
rocks run host
  <host>
  <command>

rocks run host
  compute-0-0
  /sbin/init 6
```

/etc/sysconfig/ipmi-settings

```
ipmitool lan set 1 ipaddr 192.168.1.2
ipmitool lan set 1 netmask 255.255.255.0
ipmitool lan set 1 arp respond on
ipmitool user set password 1 admin
ipmitool lan set 1 access on
ipmitool lan set 1 user
ipmitool lan set 1 auth ADMIN PASSWORD
```

USING IPMITOOL

CHASSIS STATUS

```
ipmitool –H ipmi-0-0 -P admin chassis status
System Power             : on
Power Overload           : false
Power Interlock          : inactive
Main Power Fault         : false
Power Control Fault      : false
Power Restore Policy     : always-off
Last Power Event         : ac-failed
Chassis Intrusion        : active
Front-Panel Lockout      : inactive
Drive Fault              : false
Cooling/Fan Fault        : false
```

POWER

ipmitool –H ipmi-0-0 -P admin power off

ipmitool –H ipmi-0-0 -P admin power on

A case study in Attributes

COMMON USER TWEAKS TO ROCKS

ENABLING RSH

Don’t judge

- Enabling RSH is a common user request
- Requires
  - Minor XML changes
  - Rebuilding the distribution
  - Re-installing the nodes
- SSH-only is the Rocks default

Step 1: Set RSH == True

- Before you install any compute nodes
- Set the `rsh` attribute
- Compute nodes will install with rsh

- Still need to rsh-ify the frontend yourself

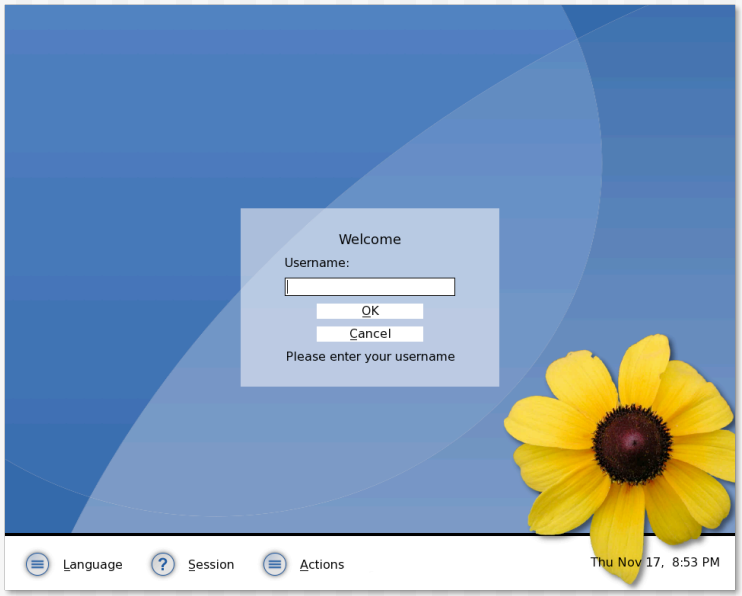
SET ATTR

```
rocks set attr
  <key> <value>

rocks set attr
  rsh true
```

ENABLING X11 ON COMPUTE HOSTS

Interactive Compute Nodes

- Another common request
- Good for computer labs
- Requires
  - Large XML changes
  - Rebuilding the distribution
  - Re-installing the nodes



Step 1: Set X11 == TRUE

- Before you install any compute nodes
- Set the `x11` attribute
- Compute nodes will install with X11

- This time we are only changing the compute nodes, not everything

SET APPLIANCE ATTR

```
rocks set appliance attr
  <appliance>
  <key> <value>

rocks set appliance attr
  compute
  x11 true
```

DISABLE SGE ON ONE RACK OF HARDWARE

Enable / Disable SGE

- Disable SGE and dedicate a rack to a single user
- Enable SGE on tile nodes on a Viz Wall
- Required a brand new appliance type!

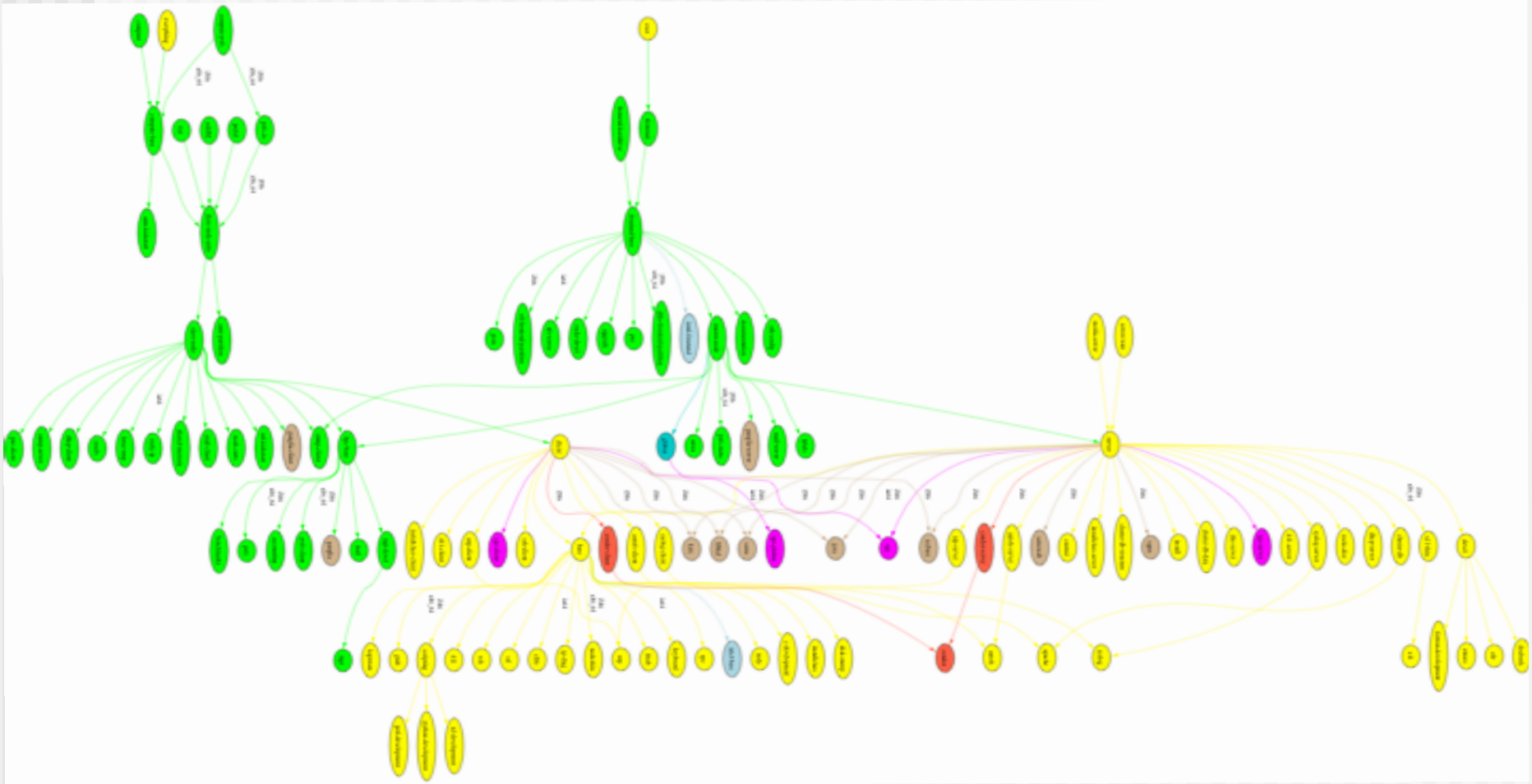
SET HOST ATTR

```
rocks set host attr
  <host(s)>
  <key> <value>

rocks set host attr
  rack0
  sge false
```

Graph XML and Rolls

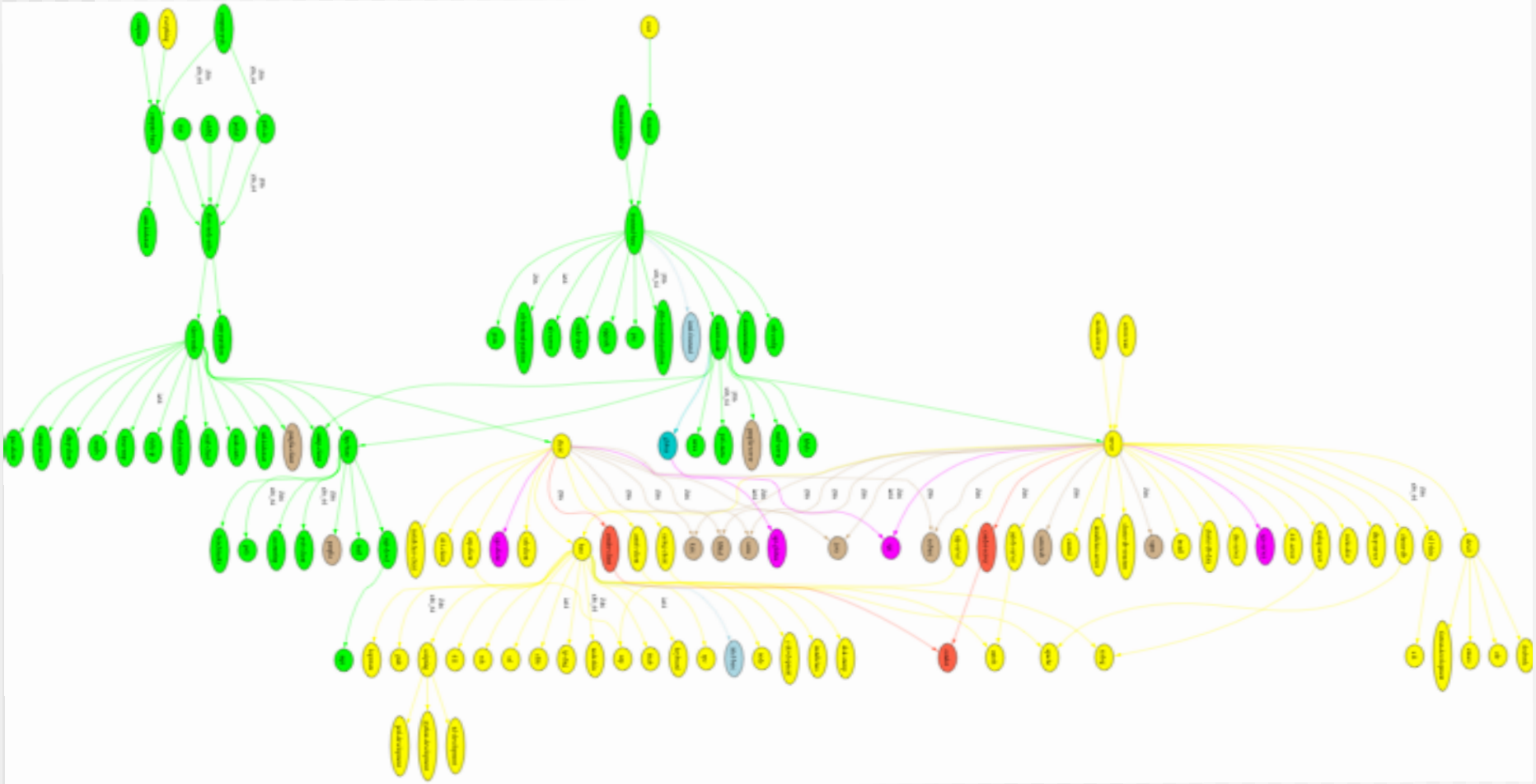
The Rocks engine

It looks something like this



GRAPH SYNTAX

Cluster DNA



Sample Node File

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE kickstart SYSTEM "@KICKSTART_DTD@" [<!ENTITY ssh "openssh">]>
<kickstart>
            <description>
            Enable SSH
            </description>

            <package>&ssh;</package>
            <package>&ssh;-clients</package>
            <package>&ssh;-server</package>
            <package>&ssh;-askpass</package>
<post>

cat &gt; /etc/ssh/ssh_config &lt;&lt; 'EOF' <!-- default client setup -->
Host *
        ForwardX11 yes
        ForwardAgent yes
EOF

chmod o+rx /root
mkdir /root/.ssh
chmod o+rx /root/.ssh

</post>
</kickstart>>
```

Sample Graph File

```
<?xml version="1.0" standalone="no"?>

<graph>
            <description>
            Default Graph for NPACI Rocks.
            </description>

            <edge from="base" to="scripting"/>
            <edge from="base" to="ssh"/>
            <edge from="base" to="ssl"/>
            <edge from="base" to=”grub"  arch="i386"/>
            <edge from="base" to="elilo" arch="ia64"/>
                        …
            <edge from="node" to="base"/>
            <edge from="node" to="accounting"/>
            <edge from="slave-node" to="node"/>
            <edge from="slave-node" to="nis-client"/>
            <edge from="slave-node" to="autofs-client"/>
            <edge from="slave-node" to="dhcp-client"/>
            <edge from="slave-node" to="snmp-server"/>
            <edge from="slave-node" to="node-certs"/>
            <edge from="compute" to="slave-node"/>
            <edge from="compute" to="usher-server"/>
            <edge from="master-node" to="node"/>
            <edge from="master-node" to="x11"/>
            <edge from="master-node" to="usher-client"/>
</graph>
```

Nodes XML Tools: Entities

- Get Attributes from Database
  - **`&Kickstart_PrivateGateway;`**
  - **`&hostname;`**

10.1.1.1
compute-0-0

Nodes XML Tools: <eval>

- Do processing on the frontend:
  - **<eval shell="bash">**
- To insert a fortune in the kickstart file:

```
<eval shell="bash">
/usr/games/fortune
</eval>
```

"Been through Hell? Whaddya bring back for me?"
-- A. Brilliant

Nodes XML Tools <file>

- Create a file on the system:

```
<file name="/etc/hi-mom" mode="append">
    How are you today?
</file>
```

- Used extensively throughout Rocks post sections
  - Keeps track of alterations automatically via RCS.

```
<file name="/etc/hi" perms="444">
How are you today?
I am fine.
</file>
```

```
…RCS checkin commands...
cat > /etc/hi << 'EOF'
How are you today?
I am fine.
EOF
chmod 444 /etc/hi-mom
…RCS cleanup commands…
```

Fancy <file>: nested tags

```
<file name="/etc/hi">

Here is your fortune for today:
<eval>
date +"%d-%b-%Y"
echo ""
/usr/games/fortune
</eval>

</file>
```

…RCS checkin commands...
cat > /etc/hi << 'EOF'

Here is your fortune for today:
13-May-2005

"Been through Hell? Whaddya bring back for me?"
-- A. Brilliant

EOF
…RCS cleanup commands…

Nodes Main

- Used to specify basic configuration:
  - timezone
  - mouse, keyboard types
  - install language
- Used more rarely than other tags
- Rocks main tags are usually a straight translation:

```
<main>

  <timezone>America/Mission_Beach
  </timezone>

</main>
```

```
…
timezone America/Mission_Beach
…
rootpw --iscrypted sndk48shdlwis
mouse genericps/2
url --url http://10.1.1.1/install/rocks-dist/..
```

Nodes Packages

- **`<package>java</package>`**
  - Specifies an RPM package. Version is automatically determined: take the *newest* rpm on the system with the name 'java'.
- **`<package arch="x86_64">java</package>`**
  - Only install this package on x86_64 architectures
- **`<package arch="i386,x86_64">java</package>`**

```
<package>newcastle</package>
<package>stone-pale</package>
<package>guinness</package>
```

```
%packages
newcastle
stone-pale
guinness
```

Nodes Post

ntp-client.xml

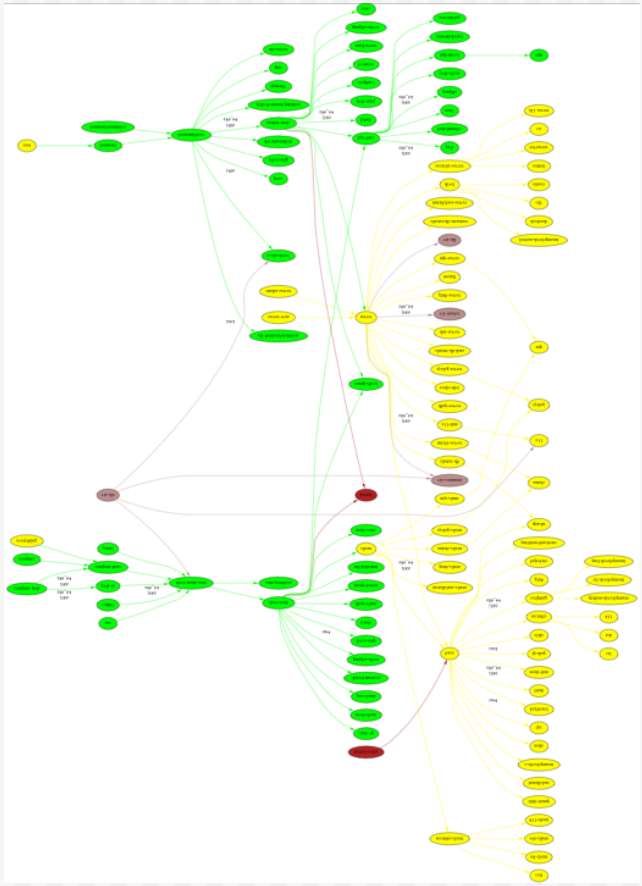
```
<post>

/bin/mkdir -p /etc/ntp
/usr/sbin/ntpdate &Kickstart_PrivateNTPHost;
/sbin/hwclock --systohc

</post>
```

```
%post

/bin/mkdir -p /etc/ntp
/usr/sbin/ntpdate 10.1.1.1
/sbin/hwclock --systohc
```

Case Study in Roll Development

VIZ ROLL

{ base, hpc, kernel, viz }



OptIPortal (SAGE)

One Node per Display

OptIPortal

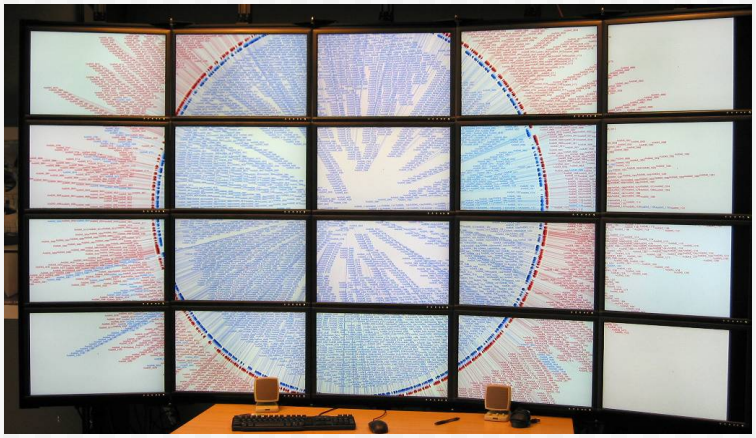
Nodes Behind the Wall

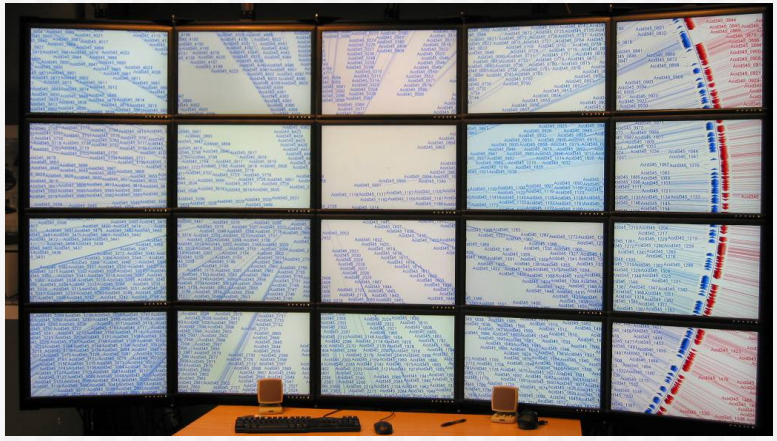
Use of OptIPortal to Interactively View Microbial Genome (CGView)

15,000 x 15,000 Pixels

Acidobacteria Bacterium Ellin345 (NCBI)

Soil Bacterium 5.6 Mb

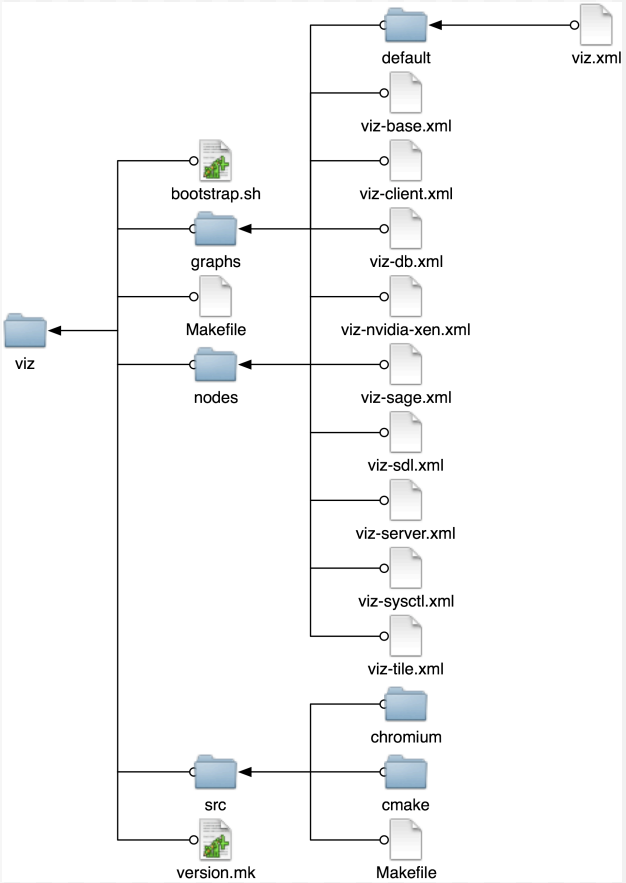




Source: Raj Singh, UCSD

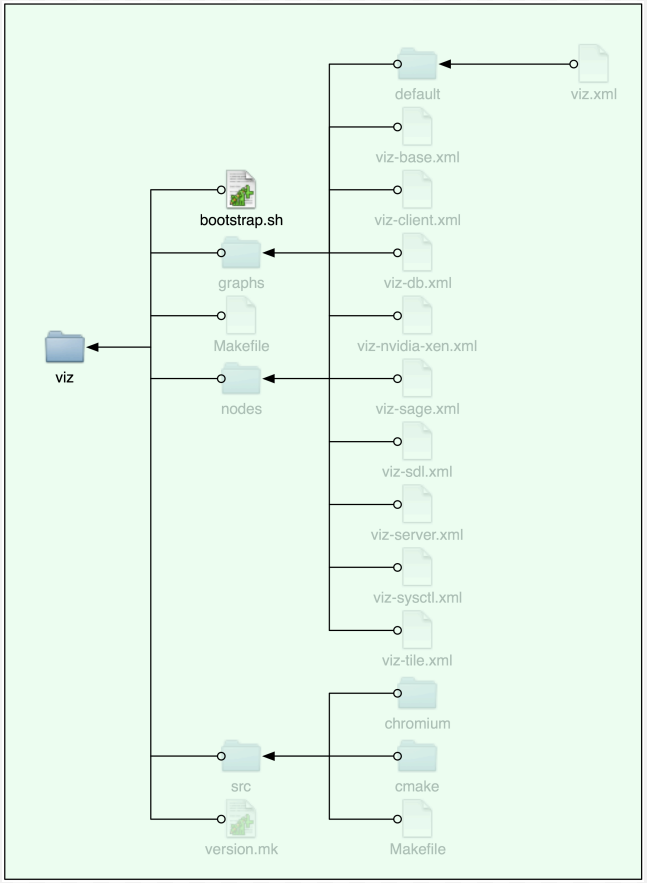
Acidobacteria bacterium Ellin345 (NCBI) Soil Bacterium 5.6 Mb

Viz Roll Source Tree



Bootstrapping

- Roll need to be build on a Rocks Frontend
- The Viz Roll needs to be build on a Rocks Frontend that has the Viz Roll install
- Is that a problem?



bootstap.sh

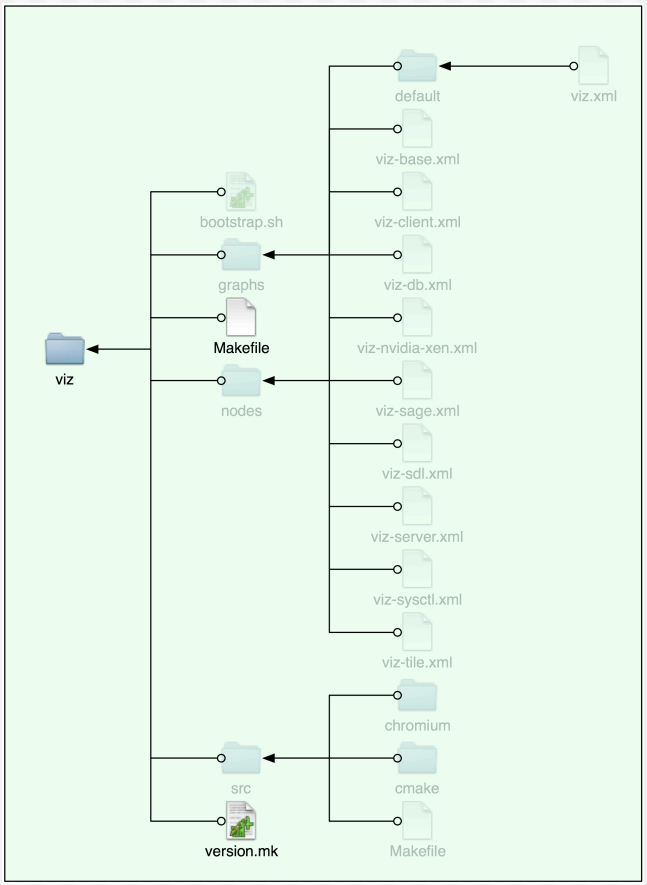
- Every Roll has one
- Used install building dependencies
  - From OS
  - From Roll install
- Simple shell script
- Easy to read

For Example

```
#!/bin/sh
#
# $Id: bootstrap.sh,v 1.18 2009/05/01 19:07:24 mjk Exp $

. $ROLLSROOT/etc/bootstrap-functions.sh

install_os_packages viz-server

install libdmx
install libdmx-devel

compile SRPMS
install glut
install glut-devel
install joy2key
install joystick

compute_and_install cmake

compile_and_install foundation-sdl
compile_and_install foundation-sdl-image
compile_and_install foundation-sdl-mixer
compile_and_install foundation-sdl-net
compile_and_install foundation-sdl-ttf

compile_and_install chromium
compile_and_install libraw1394
compile_and_install openal
compile_and_install freealut
compile_and_install plib
compile_and_install SimGear

compile_and_install libmad
compile_and_install libdvdread
compile_and_install faac
compile_and_install faad2
compile_and_install ffmpeg
compile_and_install mpeg2dec
compile_and_install wx
compile_and_install portaudio
compile_add_install glew

compile quanta
install QUANTA

```

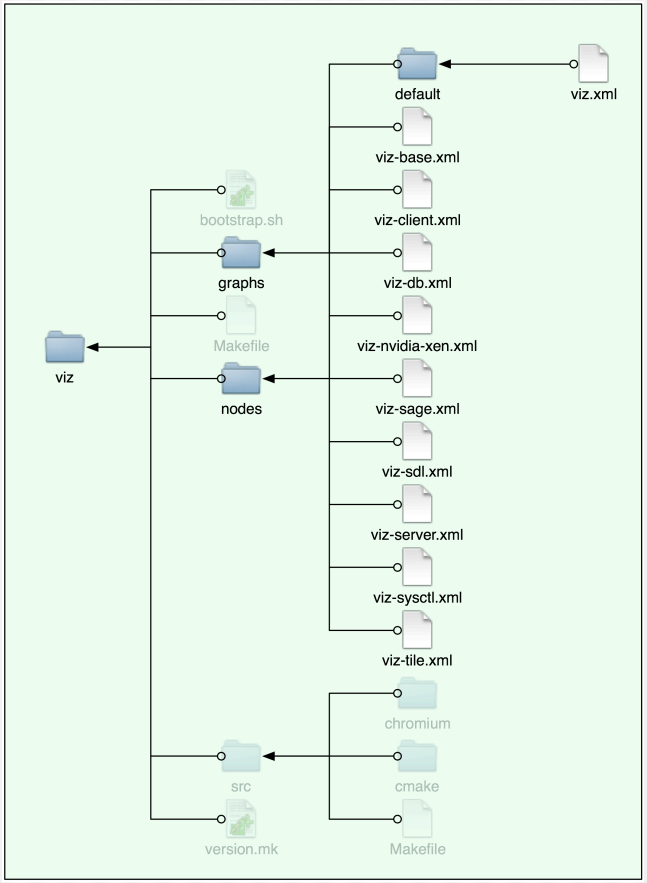
makefiles

- Every Rocks Frontend has all our build environment installed
- Roll Makefile almost empty
  - Just copy from another Roll
  - No need to change anything
- Version.mk can
  - Set version
  - Set color (graph)
  - A few other minor things



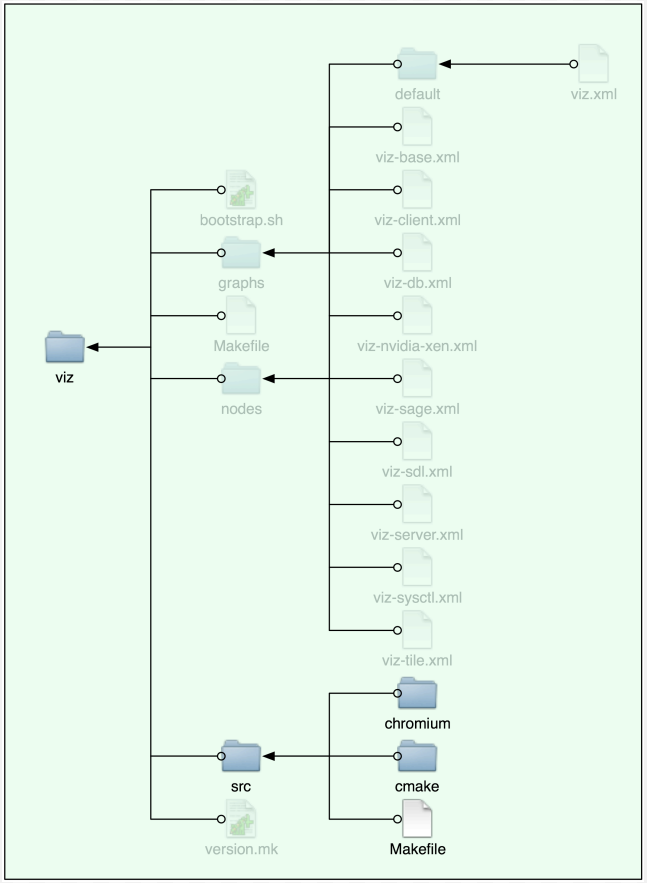
Viz Roll Sub-Graph

- Single Graph File
- 9 Nodes Files



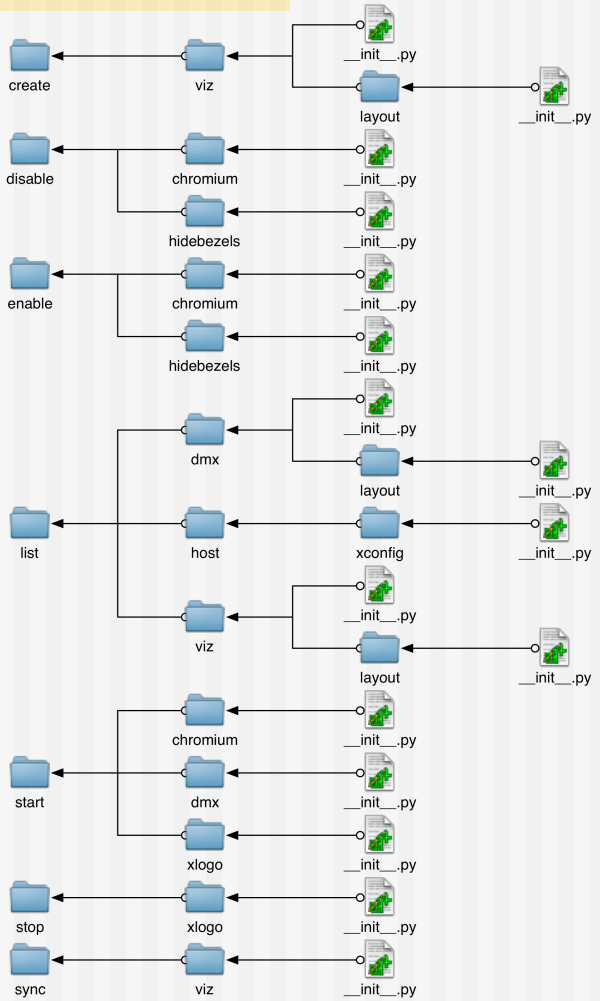
Source Code

- Almost 50 packages



Dozen+ Command

```
# rocks list roll command viz
COMMAND
create viz layout
disable chromium
disable hidebezels
enable chromium
enable hidebezels
list dmx layout
list host xconfig
list viz layout
start chromium
start dmx
start xlogo
stop xlogo
sync viz
```



LCD Bezels

rocks enable hidebezels

- Draws pixels behind the bezels (mullions) of the LCD monitors
- Calculated offset for TwinView and normal modes
- Reset the X11 configuration on all nodes
- Great mode for moving graphics

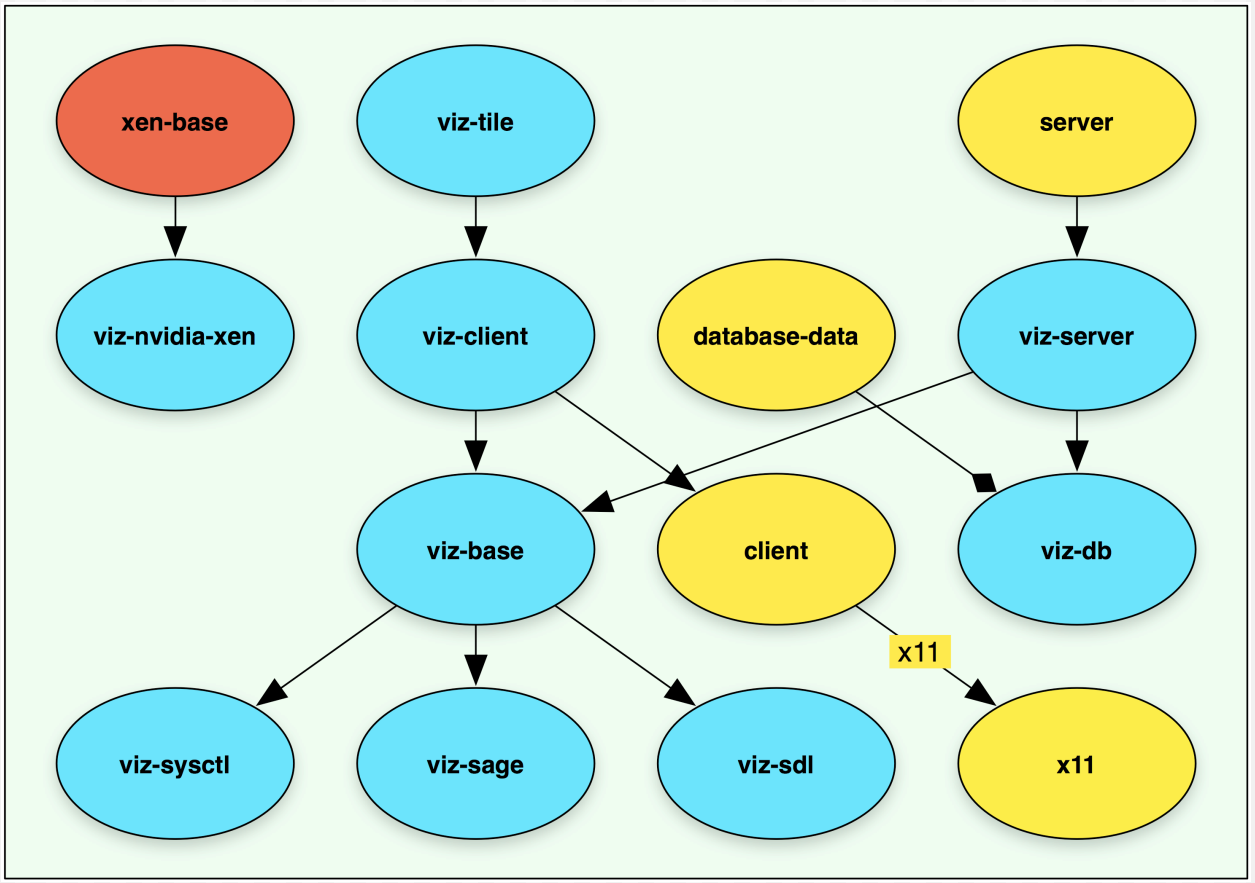
```
 1: import rocks.commands.enable
 2: import os
 3:
 4: class Command(rocks.commands.enable.command):
 5:         """
 6:         Enable Bezel Hiding mode.
 7:
 8:         <example cmd="enable hidebezels">
 9:         </example>
10:         """
11:
12:         MustBeRoot = 0
13:
14:         def run(self, params, args):
15:
16:                 os.system('touch ~/.hidebezels')
17:
18:                 # If the database videowall layout has two (or more) entries
19:                 # for the same host and card we know we are in twinview
20:                 # mode.  In this case we need to reconfigure and restart
21:                 # X11 for this host.
22:
23:                 self.db.execute("""select n.name, v.cardid
24:                         from nodes n, videowall v where v.node=n.id""")
25:                 dict     = {}
26:                 for key in self.db.fetchall():
27:                         if dict.has_key(key):
28:                                 dict[key] = 1   # TwinView host
29:                         else:
30:                                 dict[key] = 0   # NonTwinView host (so far)
31:
32:                 for (host, card) in dict.keys():
33:                         if dict[(host, card)]:
34:                                 os.system('ssh -f '
35:                                         '%s /usr/X11R6/bin/xrandr -d :0 -s 1'
36:                                         % host)
```

rocks disable hidebezels

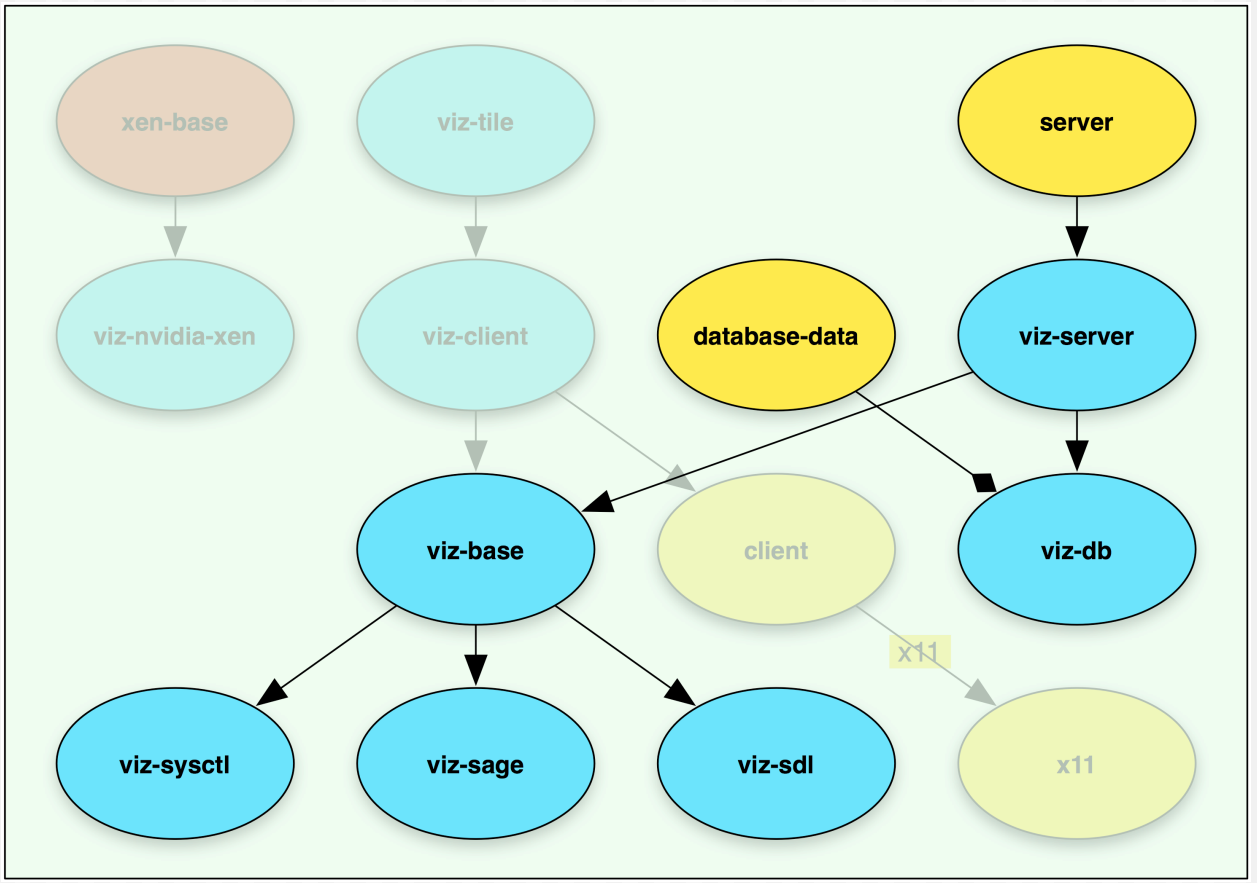
- All pixels are drawn and the bezels break apart the image
- Removes any offset from previous mode
- Resets the X11 configuration on all nodes
- Great for static images and text

```
import rocks.commands.enable
import os

class Command(rocks.commands.disable.command):
        """
        Disable Bezel Hiding mode.

        <example cmd="disable hidebezels">
        </example>
        """

        MustBeRoot = 0

        def run(self, params, args):

                os.system('/bin/rm ~/.hidebezels')

                # If the database videowall layout has two (or more) entries
                # for the same host and card we know we are in twinview
                # mode.  In this case we need to reconfigure and restart
                # X11 for this host.

                self.db.execute("""select n.name, v.cardid
                        from nodes n, videowall v where v.node=n.id""")
                dict     = {}
                for key in self.db.fetchall():
                        if dict.has_key(key):
                                dict[key] = 1   # TwinView host
                        else:
                                dict[key] = 0   # NonTwinView host (so far)

                for (host, card) in dict.keys():
                        if dict[(host, card)]:
                                os.system('ssh -f '
                                        '%s /usr/X11R6/bin/xrandr -d :0 -s 0'
                                        % host)

```

Viz Roll Sub-Graph

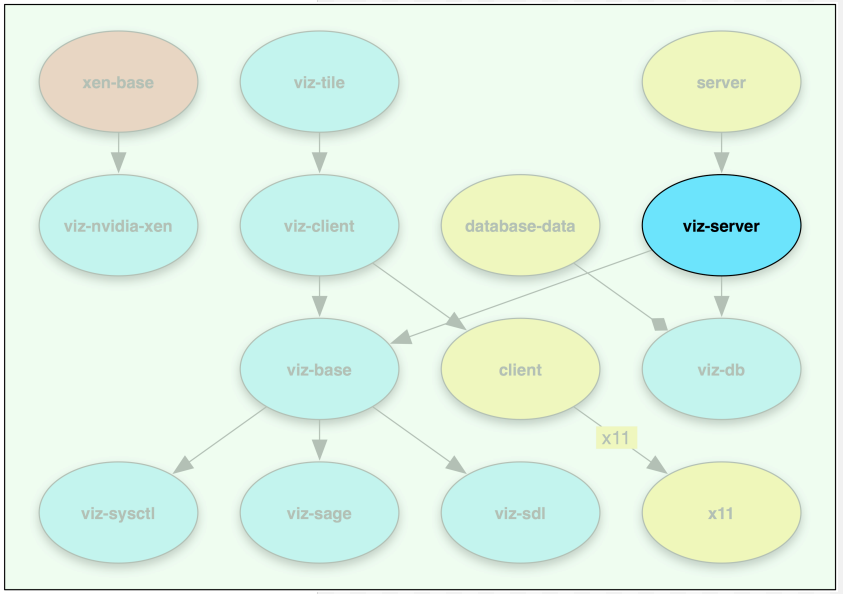


Frontend Profile



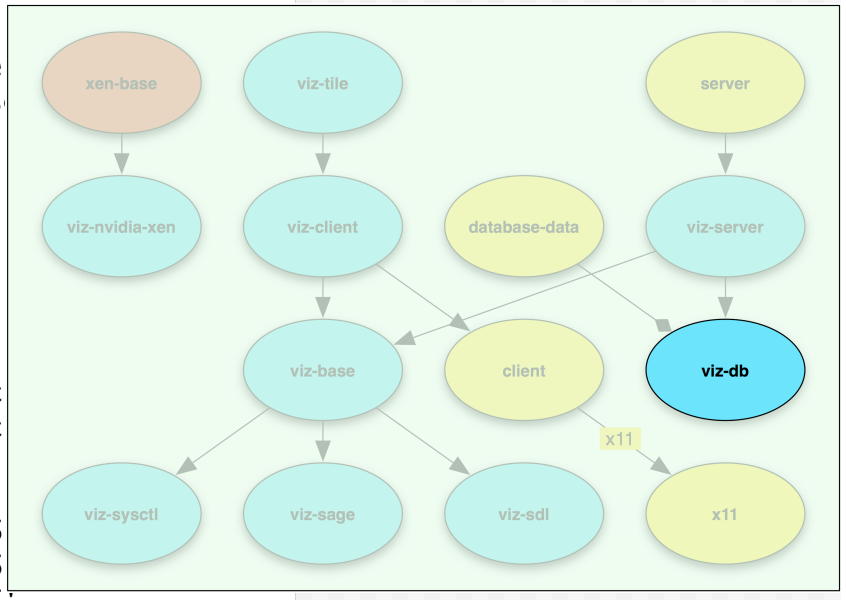
viz-server

```
1: <kickstart interface="public">
2:
3:         <description>
4:         Visualization Cluster Frontend support
5:         </description>
6:
7:         <package>curl</package>
8:         <package>curl-devel</package>
9:
10:        ...
11:
12:        <package arch="i386">cmake</package>
13:
14: <post>
15:
16: /opt/rocks/bin/rocks add appliance tile \
17:         membership=Tile node=viz-tile
18:
19: /opt/rocks/bin/rocks set attr HideBezels false
20:
21: /opt/rocks/bin/rocks set appliance attr tile x11 true
22:
23: </post>
24:
25: </kickstart>
```



viz-db

```
<kickstart>

        <description>
        Visualization Cluster Database Tables
        </description>

        <copyright>
        Copyright (c) 2000 - 2009 The Regents of the
        All rights reserved. Rocks(r) v5.2 www.rocks
        </copyright>

<post>

<file name="/tmp/viz-tables.sql">
DROP TABLE IF EXISTS videowall;
CREATE TABLE videowall (
        Node            int(11) NOT NULL default '0'
        Display         varchar(8)  NOT NULL default
        Resolution      varchar(32) NOT NULL default
        X               int(11) NOT NULL default '0'
        Y               int(11) NOT NULL default '0'
        LeftBorder      int(11) NOT NULL default '75
        RightBorder     int(11) NOT NULL default '75
        TopBorder       int(11) NOT NULL default '75',
        BottomBorder    int(11) NOT NULL default '75'
) TYPE=MyISAM;
</file>

/opt/rocks/bin/mysql --user=root --password='' cluster &lt; /tmp/viz-
tables.sql

</post>

</kickstart>
```

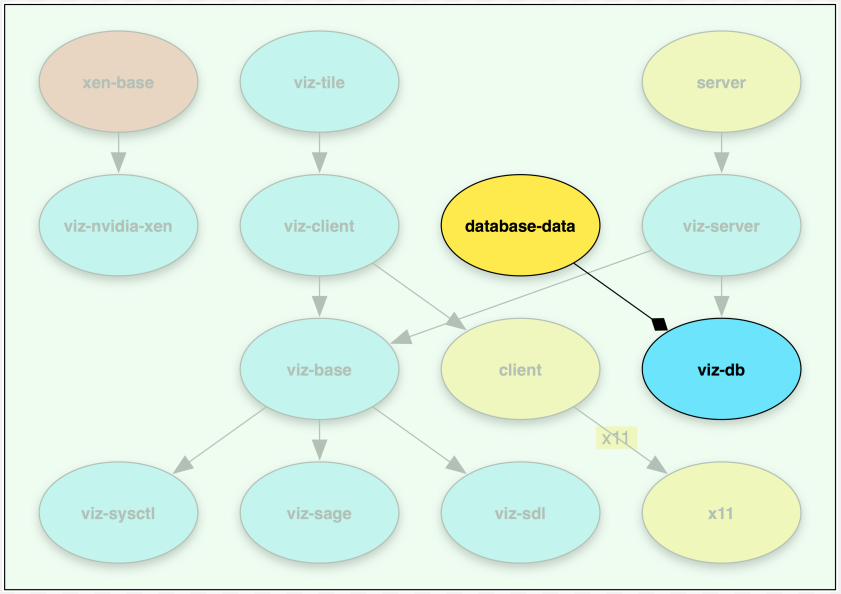


Resume @ 10:45

Ordering

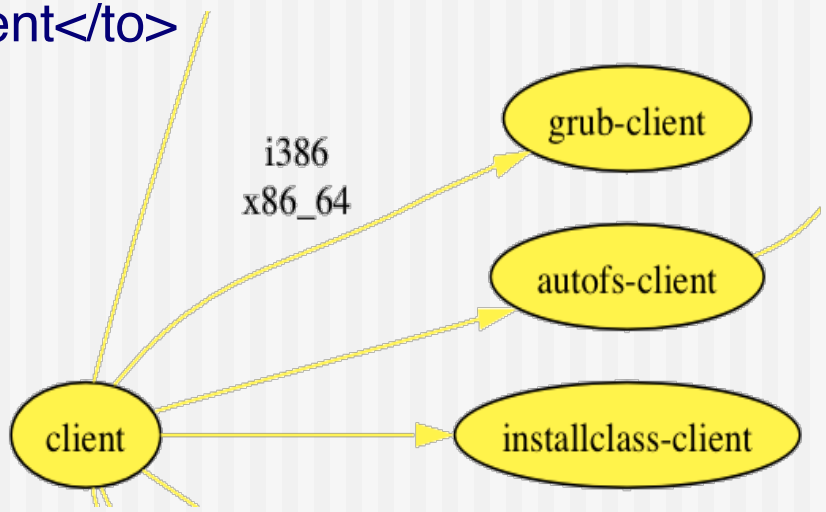
- Added DB Schema
- But, need DB up first

```
<order head="database-data">
   <tail>viz-db</tail>
</order>
```



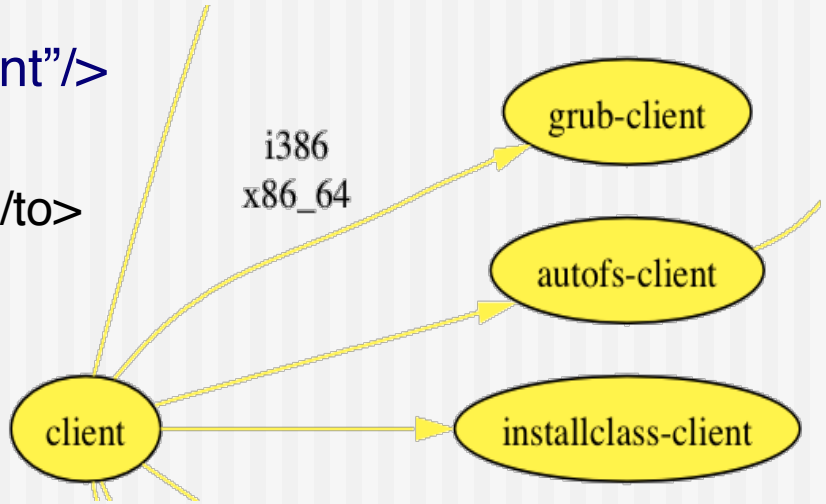
Another Look at XML

```
<graph>

    <edge from="client">
        <to arch="i386,x86_64">grub-client</to>
        <to>autofs-client</to>
        <to>installclass-client</to>
    </edge>

</graph>
```



Partial Ordering

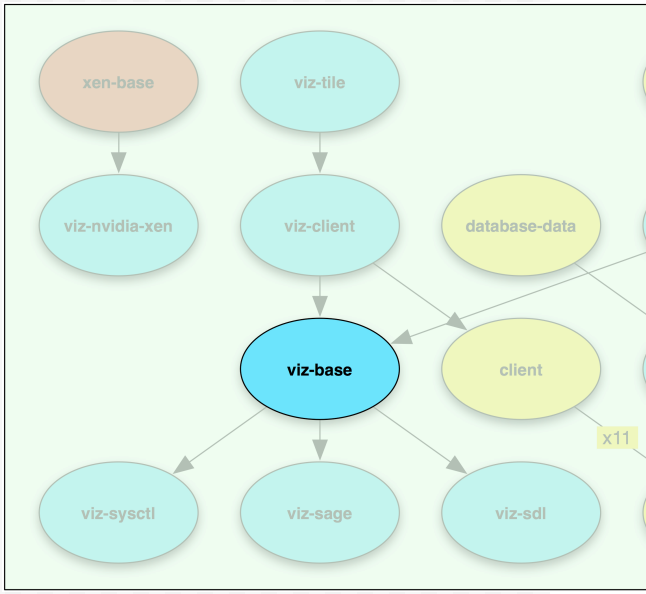
```
<graph>
    <order head="autofs-client" tail="client"/>
    <edge from="client">
        <to arch="i386,x86_64">grub-client</to>
        <to>autofs-client</to>
        <to>installclass-client</to>
    </edge>

</graph>
```



- Forces autofs-client <post> section to run before client's <post> section
- In order graph traversal enforces a partial ordering
- Applying standard graph theory to system installation

viz-base

```
<kickstart interface="public">


        <description>
        Visualization Cluster support
        </description>

        <package>rocks-viz</package>
        <package>fvwm</package>
        <package>nvidia-driver</package>
        <package>freeglut</package>
        <package>chromium</package>
        <package>Cg</package>
        <package>wx</package>
        ...

<post>

<!-- Create the viz account on each node (avoid 411) -->
/usr/sbin/useradd -M -u402 -c "Viz X11 Owner" -d /opt/viz/etc viz

chown -R viz.viz /opt/viz/etc

/sbin/chkconfig --add  nvidia

<file name="/etc/inittab"/>
gawk \
        '/id:3:initdefault:/ { print "id:5:initdefault:"; next; } \
        { print; }' /etc/inittab &gt; /tmp/inittab
mv /tmp/inittab /etc/inittab

</post>

</kickstart>
```



useradd

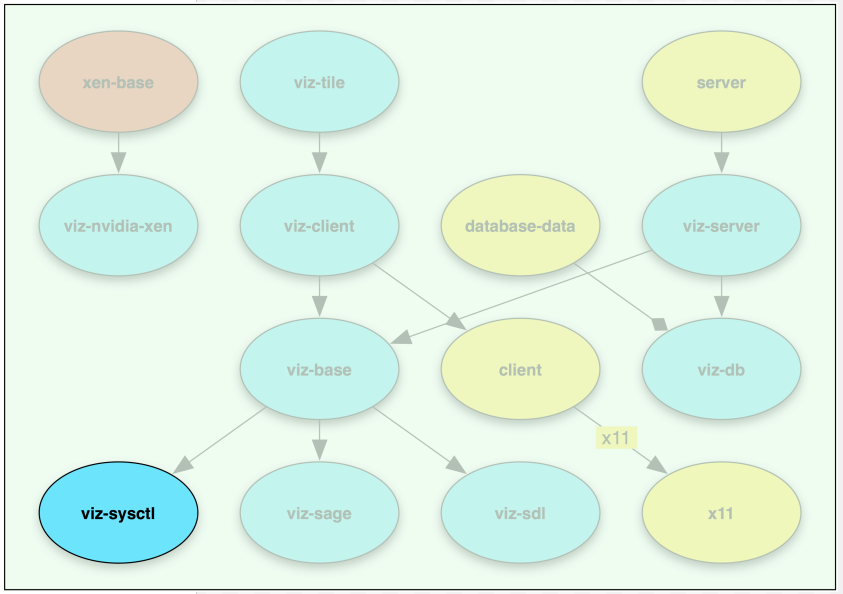
- Create a new user named viz
- Used to auto-login X11 and xhost+ tiles
- Home is /opt/viz/etc
  - Directory is local to every node
  - Required for .xauthority files
- UID is hard coded
  - Avoids 411 lookups during install
  - Roll developers can request a RESERVED-UID

rocks/src/roll/RESERVED-UIDS

```
# $Id: RESERVED-UIDS,v 1.17 2009/04/21 00:32:49 anoop Exp $
#
# Register all UIDs created by Rolls here.  Third party developers may
# request an ID allocation from devel@rocksclusters.org.

ID  NAME              SOURCE ROLL          NOTES
400 sge               sge
401 globus grid
402 viz               viz                  owner of X11 session
403 Rocksdb           base                 foundation mysql user
404 Biouser           bio                  user associated with the bioroll
405 jboss             java                 jboss not started by default
406 gxmap             gx-map
407 Condor condor                 condor master user
408 pbs               pbs                  pbs batch
409 postgres          postgres8  postgres version 8 roll
410 cvs               cvs-server
411 cvsanon           cvs-server
412 tomcat java                   tomcat not started by default
413 nagios nagios
414 ssoadmin          camera               SSO admin user
415 wsuser camera                 ws user for dropbox
416 hg                ------               mercurial user
417 boinc             boinc                boinc user
418 moab              moab                 moab scheduler user
419 zfsuser           thumper-conf         ZFS Replication user
```

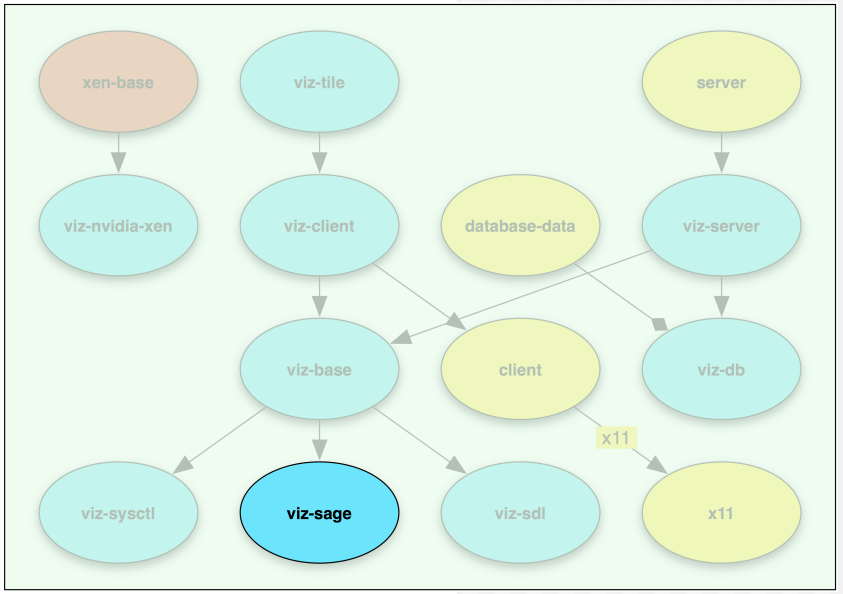
viz-sysctl

```
<kickstart>


        <description>
        Increase network buffers
        </description>

<post>

<file name="/etc/sysctl.conf"/>
<file name="/tmp/sysctl.awk">
/net\.core\.rmem_max/           { next; }
/net\.core\.wmem_max/           { next; }
/net\.ipv4\.tcp_rmem/           { next; }
/net\.ipv4\.tcp_wmem/           { next; }
/net\.ipv4\.route\.flush/       { next; }
/net\.core\.rmem_max/           { next; }
{ print; }
END {
        print "net.core.rmem_max = 16777216";
        print "net.core.wmem_max = 16777216";
        print "net.ipv4.tcp_rmem = 4096 87380 16777216";
        print "net.ipv4.tcp_wmem = 4096 16384 16777216";
        print "net.ipv4.route.flush = 1";
}
</file>
awk -f /tmp/sysctl.awk /etc/sysctl.conf &gt; /tmp/sysctl.conf
cp /tmp/sysctl.conf /etc/sysctl.conf
rm /tmp/sysctl.awk /tmp/sysctl.conf

</post>

</kickstart>
```



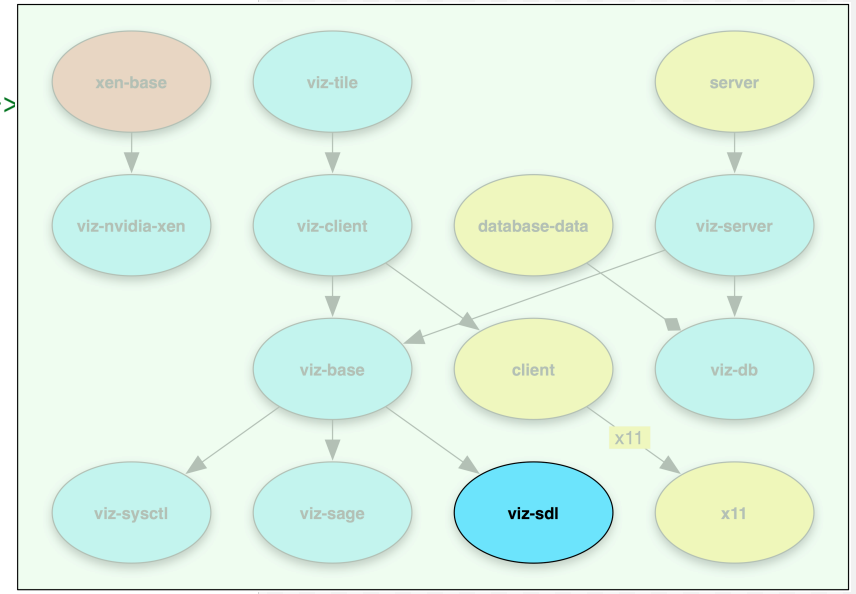
viz-sage

```
1: <kickstart>
2:
3:         <description>
4:         Common Setup for all SAGE nodes
5:         </description>
6:
7:
8:         <package>wx</package>
9:
10:        <package>libtool-libs</package>
11:        <package>libtool</package>
12:
13:        <package>mpeg2dec</package>
14:        <package>libraw1394</package>
15:        <package>libiec61883</package>
16:        <package>QUANTA</package>
17:        <package>sage</package>
18:
19: </kickstart>
```

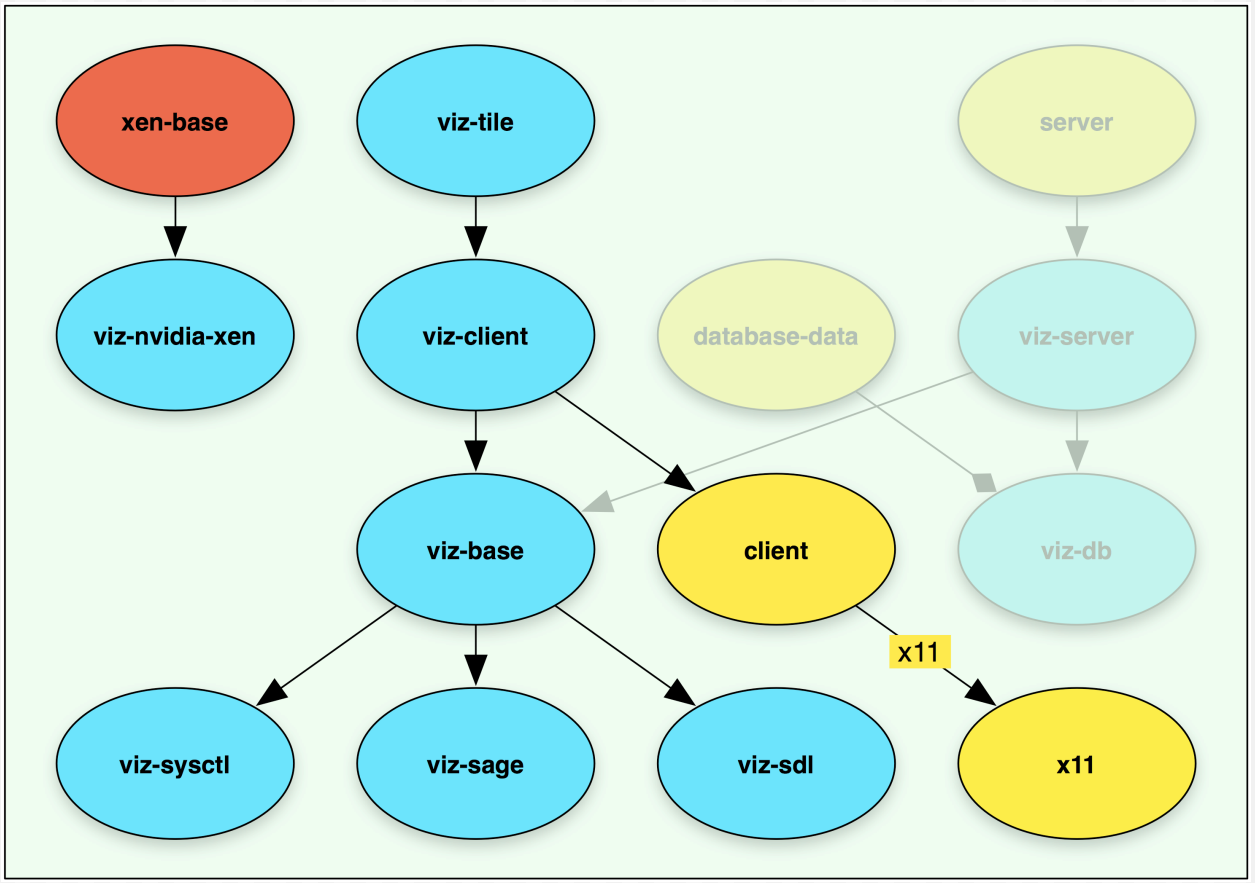


viz-sdl

```
1: <kickstart>
2:
3:         <description>
4:         SDL stuff
5:         </description>
6:
7:         <!-- SDL needs stuff we trim off the base CD -->
8:         <package>arts</package>
9:         <package>qt</package>
10:
11:        <!-- SDL is now in the foundation -->
12:        <package>foundation-sdl</package>
13:        <package>foundation-sdl-image</package>
14:        <package>foundation-sdl-mixer</package>
15:        <package>foundation-sdl-net</package>
16:        <package>foundation-sdl-ttf</package>
17:
18: </kickstart>
```

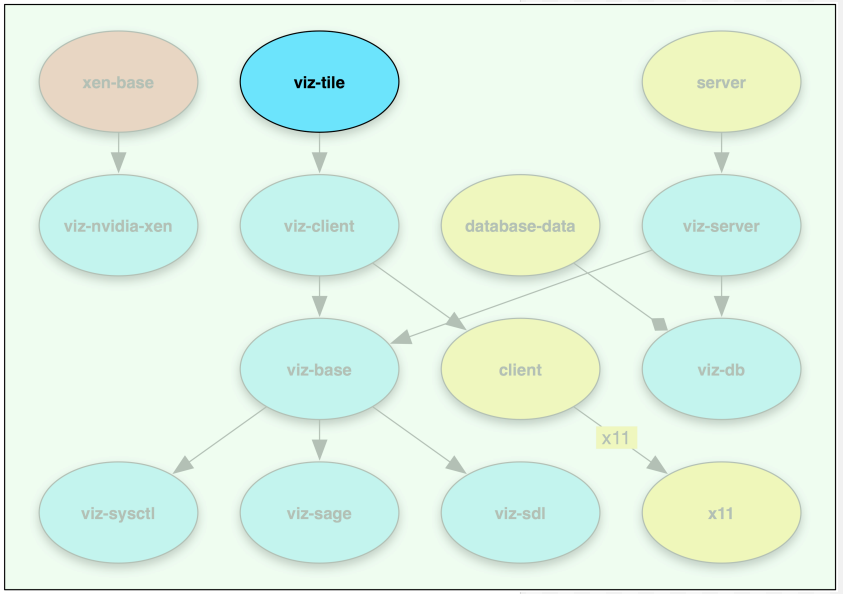


Tile Profile

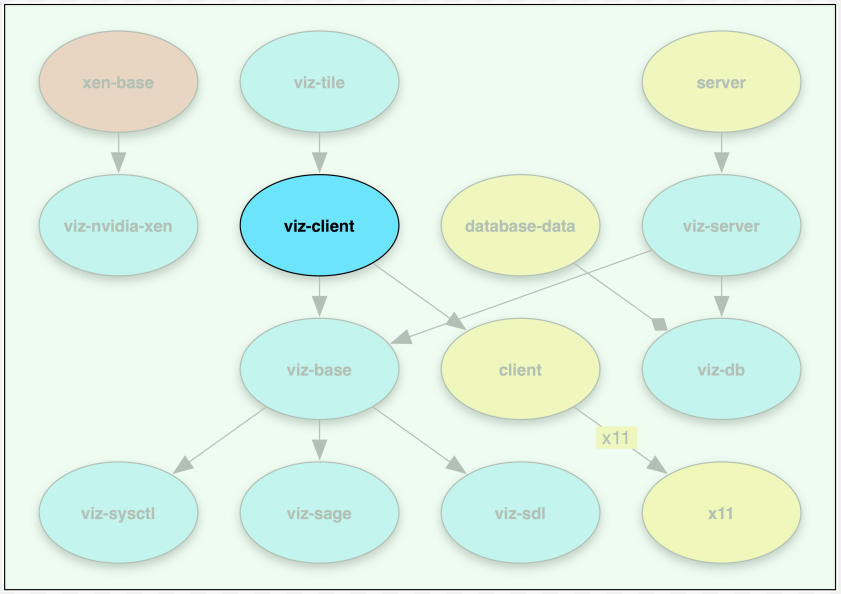


viz-tile

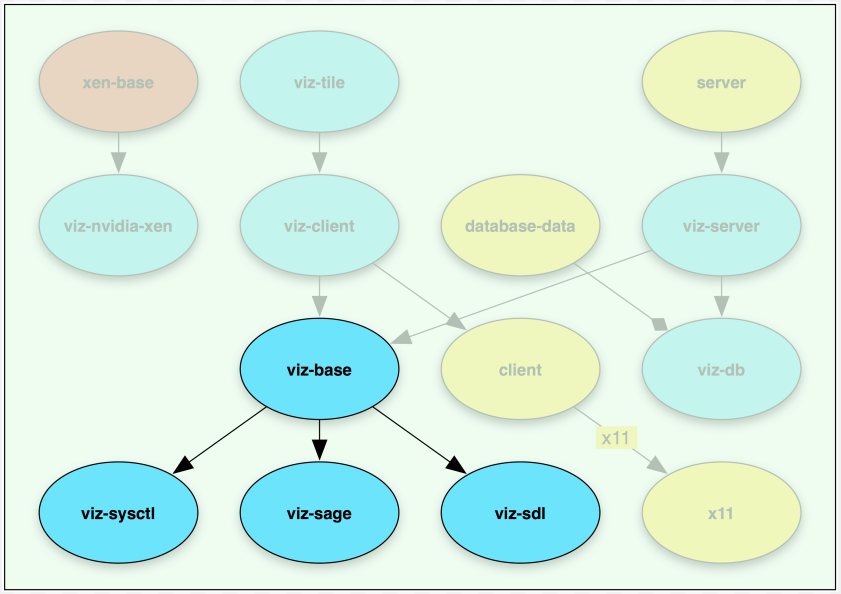
```
<kickstart interface="public">

        <description>
        Display Node
        </description>

<post>

<!-- setup autologin for viz user -->
<file name="/etc/gdm/custom.conf">
[daemon]
TimedLoginEnable=true
TimedLogin=viz
TimedLoginDelay=5

[security]
DisallowTCP=false
</file>

</post>

</kickstart>
```



viz-client



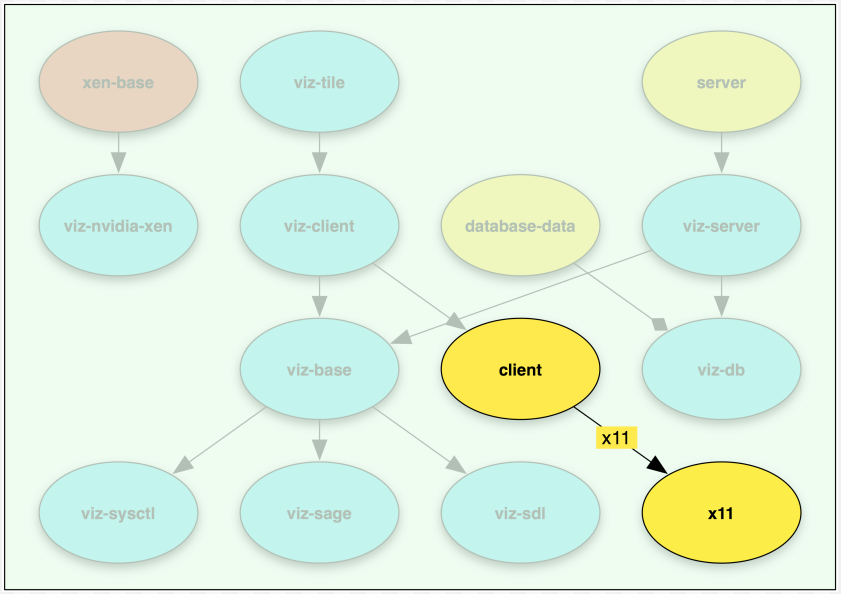
viz-base …



client -> x11

- Conditional Edge from Base Roll
- Uses x11 attribute
- Set in viz-server

```
<edge from="client"
   to="x11" cond="x11"/>
```



xen-base

```
<kickstart>

        <description>
        Patch the nVidia driver to make it work in dom0
        </description>

        <package>nvidia-xen</package>

<post>

<file name="/opt/viz/drivers/nvidia/install-driver" perms="755">
<![CDATA[
#!/bin/bash
if [ -e /usr/src/linux-2.6 ]
then
    ln -s /usr/src/linux-2.6 /usr/src/linux
elif [ -e /usr/src/linux-2.4 ]
then
    ln -s /usr/src/linux-2.4 /usr/src/linux
fi

/opt/viz/drivers/nvidia/NVIDIA*run --no-network -x > /dev/null 2>&1

cd NVIDIA*/usr/src/nv
zcat /opt/viz/drivers/nvidia/nvidia_xenpatch.gz | \
        patch --forward -p4 --quiet > /dev/null 2>&1

make SYSSRC=/usr/src/linux module > /dev/null 2>&1

install -D -o root -g root -m 0644 nvidia.ko \
        /lib/modules/`uname -r`/video/nvidia.ko

depmod -a

cd ../../../
./nvidia-installer --no-kernel-module -s > /dev/null 2>&1
]]>
</file>

</post>

</kickstart>
```