Case Studies of Existing Rolls

Building MGL roll

Rocks-A-Palooza I

May 2005

Nadya Williams nadya@sdsc.edu

ROCKS

Prepare for the build

◆ **Check out CVS Rocks distribution**

  ➲ cvs -
  d:pserver:anonymous@cvs.rocksclusters.org:/home/cvs/CVSROOT
  login

  ➲ cvs -
  d:pserver:anonymous@cvs.rocksclusters.org:/home/cvs/CVSROOT
  checkout -r ROCKS_3_3_0 rocks

◆ **Go to the roll's directory**

  ➲ **# cd rocks/src/roll/mgl**

  ➲ **# ls**

    CVS/   graphs/   Makefile   nodes/   src/   version.mk

File graphs/default/mgl.xml

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE kickstart SYSTEM "@GRAPH_DTD@">


<graph>
    <description>
    MGL Roll
    </description>

    <changelog>
    $Log: mgl.xml,v $
    </changelog>

    <edge from="server" to="mgl"/>
</graph>
```

color specified in version.mk

mgl

root → server

...

© UC Regents

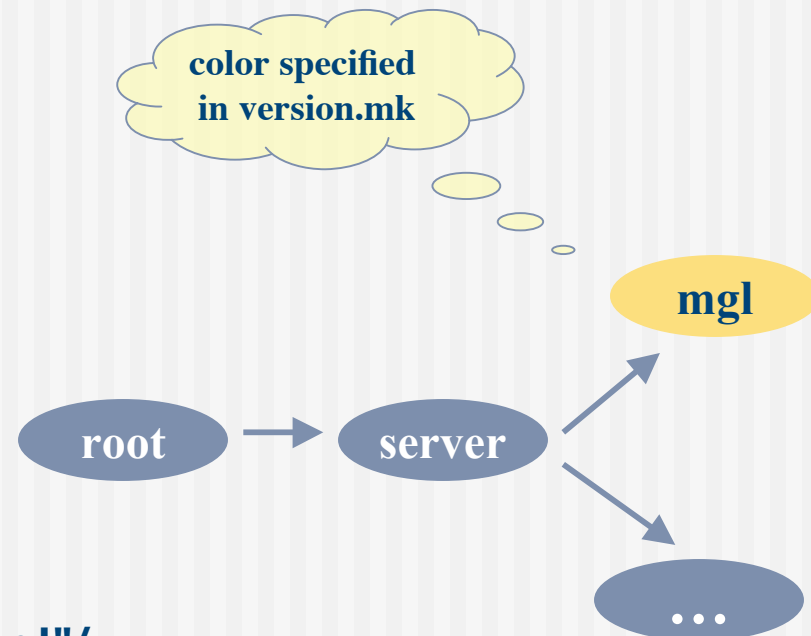# File nodes/mgl.xml

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE kickstart SYSTEM "@KICKSTART_DTD@">
<kickstart>
    <description>
    Python based software developed at the Molecular Graphics Lab (MGL)
    </description>

    <copyright></copyright>
    <changelog>$Log: mgl.xml,v $<changelog>

    <package>mgl-tools</package>
    <package>mgl-swig</package>
    <package>rocks-mgl</package>
    <package>roll-mgl-usersguide</package>

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

Contents of src/

- **# mkdir rocks**
  - user profile files
- **# mkdir tools**
  - mgl tarball
- **# mkdir swig**
  - prerequisite software for mgl tools
- **# mkdir usersguide**
  - users guide placeholder, used for real guide development

src/rocks/

- **create mgl.sh (and mgl.csh)**

  export MGL_ROOT="/opt/mgl"
  export PATH="$MGL_ROOT/bin:$PATH"
  if [-z "$LD_LIBRARY_PATH"]; then
    export LD_LIBRARY_PATH="$shlibdirs"
  else
    export LD_LIBRARY_PATH="$shlibdirs:$LD_LIBRARY_PATH"
  fi

- **Makefile**

  install::
        mkdir -p $(ROOT)/etc/profile.d
        install -m755 mgl.* $(ROOT)/etc/profile.d

- **version.mk**

  NAME = rocks-mgl
  RELEASE = 0
  VERSION = 3.3.0

© UC Regents

src/swig/

◆ **# ls src/swig**

 CVS/   Makefile   SWIG-1.3.22.tar.gz   mgl-swig.spec.in   version.mk

◆ **version.mk**

NAME = mgl-swig
RELEASE = 0
VERSION = 1.3.22
SWIG = SWIG

◆ **Makefile**

PKGROOT=/opt/mgl
install::
        mkdir -p $(ROOT)/$(PKGROOT)
        (cd $(SWIG)-$(VERSION); make prefix=$(ROOT)/$(PKGROOT) install;)
build:
        gunzip -c $(SWIG)-$(VERSION).tar.gz l tar -x
        (cd $(SWIG)-$(VERSION); ./configure --prefix=$(PKGROOT); make)

src/tools/

- **# ls src/tools**

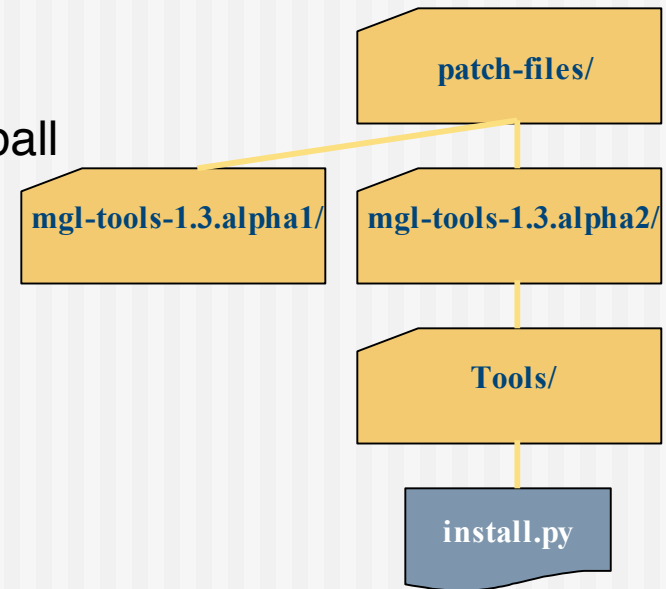 CVS/   Makefile.mgl.in   mgl-tools.spec.in
  Install.sh.in   mgl-tools-1.3.alpha1.tar.gz   patch-files/
  Makefile   mgl-tools-1.3.alpha2.tar.gz   version.mk

- **patch-files/**
  - drop-in files to substitute originals
  - same directory structure as in the tarball

- **version.mk**
  NAME = mgl-tools
  RELEASE = 0
  VERSION = 1.3.alpha2

patch-files/

mgl-tools-1.3.alpha1/   mgl-tools-1.3.alpha2/

Tools/

install.py

src/tools/Makefile

```
PKGROOT = /opt/mgl
MGL_ROOT = $(PKGROOT)
SEDSPEC += \
    -e 's%@MGL_ROOT@%$(MGL_ROOT)%g'


install::
    mkdir -p $(ROOT)/$(PKGROOT)
    ( cd $(NAME)-$(VERSION); make prefix=$(ROOT)/$(PKGROOT) install; )


build:
    gunzip -c $(NAME)-$(VERSION).tar.gz | tar -x
    $(SED) $(SEDSPEC) Install.sh.in > $(NAME)-$(VERSION)/Install.sh
    $(SED) $(SEDSPEC) Makefile.mgl.in > $(NAME)-$(VERSION)/Makefile
    cd patch-files && find . -type f | grep -v CVS | cpio -pduv ../
```

Sedspec substitution

◆ Install.sh.in

…

export MGL_ROOT="@ROOT@/@MGL_ROOT@"
export MGL_PURE_ROOT="@MGL_ROOT@"

…

◆ Install.sh

export MGL_ROOT="<path-where-building>/opt/mgl"
export MGL_PURE_ROOT="/opt/mgl"

Building

- **Build rpms**
  - **# cd mgl/src/rocks; make rpm**
- **Build roll**
  - **# cd mgl; make roll**
- **Result**

mgl/RPMS/i386:
- mgl-swig-1.3.22-0.i386.rpm
- mgl-tools-1.3.alpha2-0.i386.rpm

mgl/RPMS/noarch:
- rocks-mgl-3.2.0-0.noarch.rpm
- roll-mgl-usersguide-3.3.0-0.noarch.rpm
- roll-mgl-kickstart-3.3.0-0.noarch.rpm

mgl:
- roll-mgl-3.3.0-0.i386.iso