Empacotando com o RPM

Hugo Cisneiros

hugo@devin.com.br
http://www.devin.com.br/eitch/

A pergunta que não quer calar é:

O que diabos é empacotamento???

...E o que é RPM???

Empacotando com o RPM

Uma breve história sobre o RPM:

- Inicialmente feito em Perl, pela Red Hat;
- Depois portado para C, utilizando banco de dados BDB;
- Teve seu nome mudado pela ampla adoção.

Os objetivos do RPM:

- Facilitar o trabalho do empacotador da distribuição;
- Trabalhar com arquiteturas múltiplas mais facilmente;
- Manter sempre em vista o que está instalado ou não no sistema.

Preparando o Ambiente

No Fedora/RedHat, um ambiente já se encontra pronto no diretório:

/usr/src/redhat

O problema é que apenas o root pode mexer neste diretório. E se quisermos criar nossos pacotes com um usuário comum (o que é recomendado)? Eis a solução! Primeiro crie a estrutura de diretórios necessária:

```
mkdir rpm
cd rpm
mkdir BUILD RPMS SOURCES SPECS SRPMS
cd RPMS
mkdir i386 i486 i586 i686 athlon x86_64 ppc noarch
```

Note que criamos diretórios para cada arquitetura. Um diretório como o PPC só será usado se você tiver um processador PowerPC, obviamente. O *noarch* é onde os pacotes sem dependência de arquitetura são armazenados.

Preparando o Ambiente

Agora que já temos o ambiente, precisamos falar ao RPM onde ele precisa trabalhar. Para isso edite o arquivo *.rpmmacros* dentro do diretório *HOME* do usuário, colocando o seguinte conteúdo (e adaptando):

```
%_topdir /home/boboalegre/rpm
%_tmppath /var/tmp
```

Agora o diretório *top* da construção de pacotes se localiza no diretório que criamos anteriormente. Não se esqueça que o diretório */var/*tmp também tem que ter permissões para o usuário que vai construir os pacotes. Geralmente esse diretório citado no exemplo é a opção mais usada e sempre tem permissões.

Criando o arquivo spec

- Que tal primeiro dizer o que é um arquivo spec?

Nosso alvo exemplo, o programa:

HELLO WORLD!

Criando o arquivo spec

O arquivo spec do nosso helloworld é dividido nas seguintes partes:

- Cabeçalhos com informações básicas do pacote
- Descrição do pacote (%description)
- Preparação para a compilação (%prep)
- A compilação do código-fonte em si (%build)
- A instalação do resultado ta compilação (%install)
- Limpeza dos arquivos depois da compilação (%clean)
- Scripts para antes/depois da instalação/desinstalação (%pre, %post, %preun, %postun)
- Lista de arquivos do pacote (%files)
- O *ChangeLog* do pacote (%changelog)

Agora cada uma destas partes!

Criando o arquivo spec

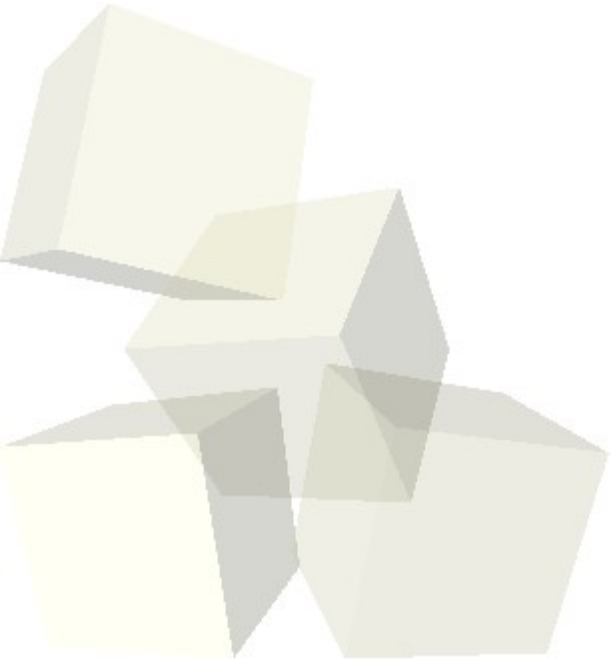
Cabeçalhos Iniciais:

```
Summary: Hello World Dummy Package
Name: helloworld
Version: 1.0
Release: 1.eitch
License: GPL
Group: System Environment/Base
URL: http://www.devin.com.br/eitch/
Source: helloworld-%{version}.tar.bz2
BuildRoot: %{_tmppath}/%{name}-%{version}-%{release}-buildroot
Requires: php >= 4.3, python >= 2.3, bash >= 3
BuildRequires: gcc
```

Nota: Cada distribuição possui sua hierarquia de grupos (Group)

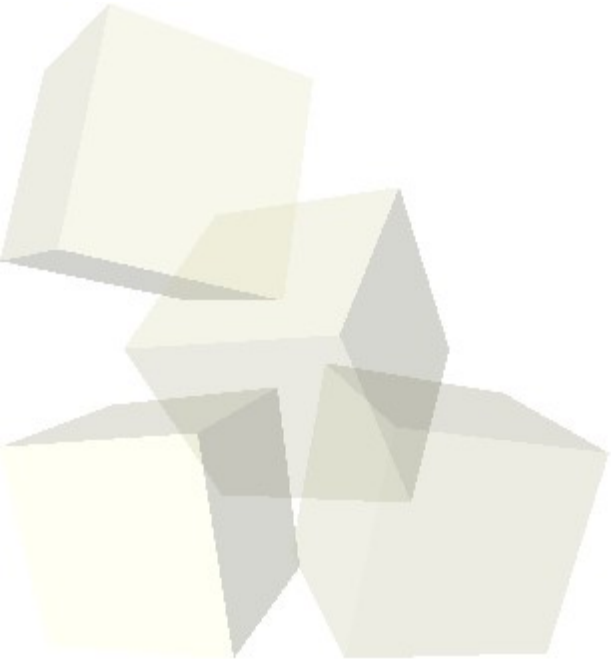
Descrição do pacote:

```
%description
Um pacote que não serve pra nada de útil no sistema, mas serve para
quem estiver usando o pacote aprender um pouco mais sobre
empacotamento de programas nas distribuições Linux
(principalmente com RPM).
```



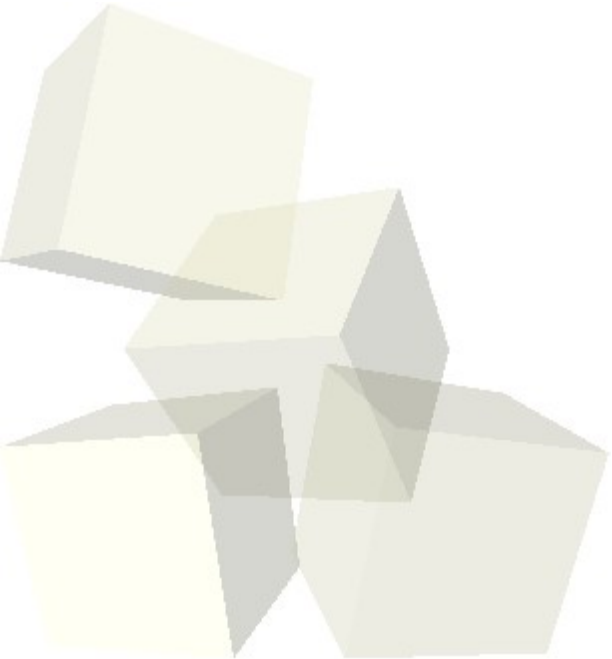
Preparação para a compilação:

```
%prep
%setup -q
```



A compilação do código-fonte em si:

```
%build
gcc helloworld.c -o helloworld-c
```



Criando o arquivo spec

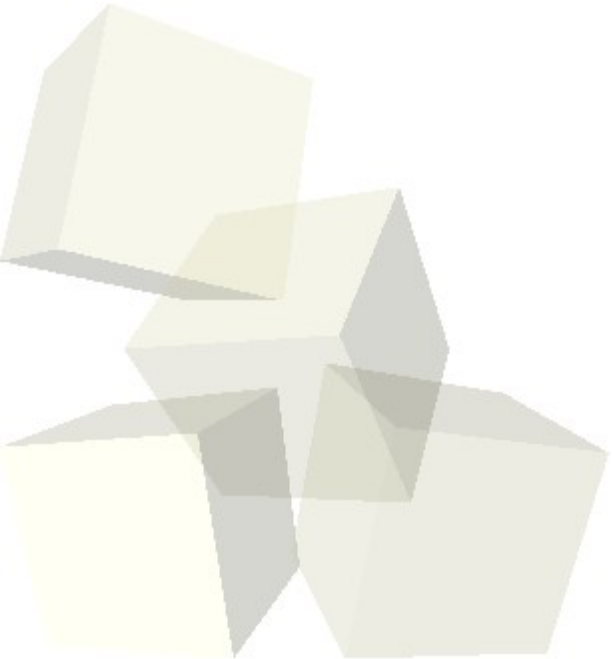
A instalação do resultado da compilação:

```
%install
mkdir -p %{buildroot}/usr/bin
install -m755 helloworld-c helloworld-php helloworld-sh \
%{buildroot}/usr/bin
mkdir -p %{buildroot}/usr/lib
install -m644 libhelloworld.so.1 libnaosirvopranada.so.1 \
%{buildroot}/usr/lib
ln -s /usr/lib/libhelloworld.so.1 \
%{buildroot}/usr/lib/libhelloworld.so
ln -s /usr/lib/libnaosirvopranada.so.1 \
%{buildroot}/usr/lib/libnaosirvopranada.so
mkdir -p %{buildroot}/etc
install -m644 helloworld.conf %{buildroot}/etc
mkdir -p %{buildroot}/usr/share/doc/%{name}-%{version}
install -m644 AUTHORS LICENSE README \
%{buildroot}/usr/share/doc/%{name}-%{version}
```

Criando o arquivo spec

Limpeza dos arquivos depois da compilação:

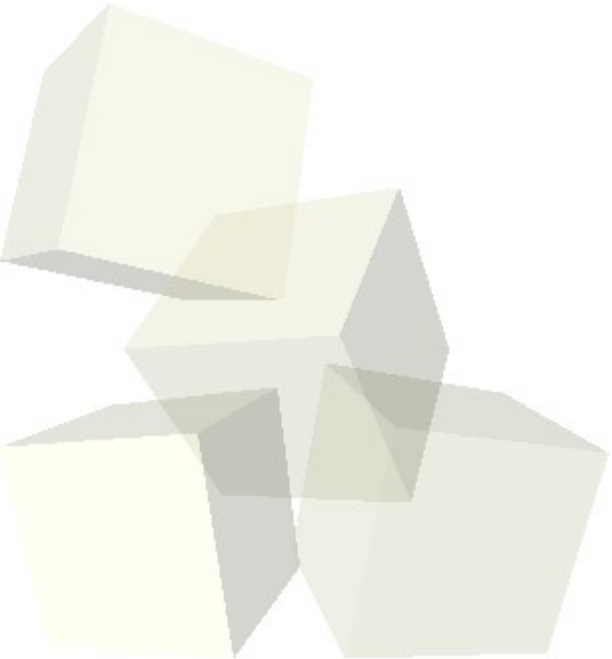
```
%clean
rm -rf $RPM_BUILD_ROOT
```



Criando o arquivo spec

Scripts para antes/depois da instalação/desinstalação:

```
(Não precisamos!)
```



Criando o arquivo spec

Lista de arquivos do pacote:

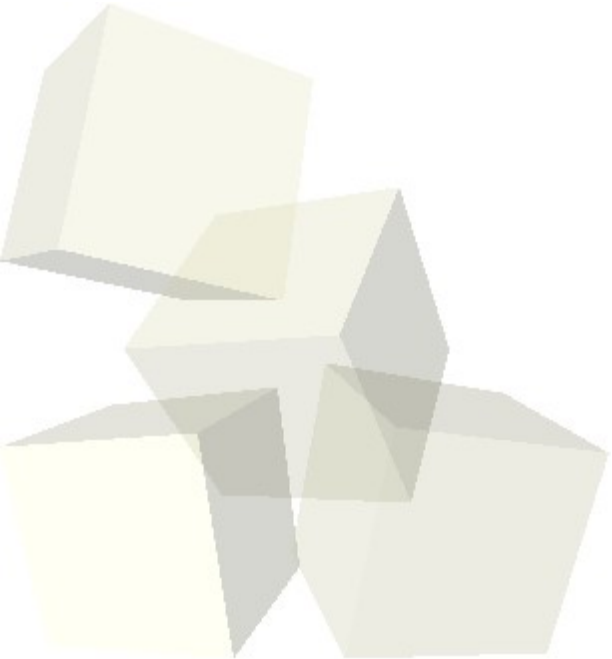
```
%files
%defattr(-,root,root)
/usr/bin/helloworld-c
/usr/bin/helloworld-php
/usr/bin/helloworld-sh
/usr/lib/libhelloworld.so.1
/usr/lib/libhelloworld.so
/usr/lib/libnaosirvopranada.so.1
/usr/lib/libnaosirvopranada.so
/etc/helloworld.conf
%doc AUTHORS
%doc LICENSE
%doc README
```

Criando o arquivo spec

O *ChangeLog* do pacote:

```
%changelog
* Tue May  3 2005 Zé Mané <zehmaneh@devin.com.br> - 1.0-1
- Criação do pacote

* Mon May  2 2005 Eitch <hugo@devin.com.br> - 0.9
- Criação do programa
```



Construindo o Pacote

Depois do arquivo spec pronto, é hora de colocar os arquivos nos seus devidos lugares, e montar o pacote. Para fazer isso, basta colocar:

- O arquivo *spec* dentro do diretório **SPECS** do *builddir* RPM;
- O fonte compactado (.tar.gz) e *patches* no diretório **SOURCES**;

E depois é a hora da verdade! Construa o pacote com o comando:

```
cd /home/boboalegre/rpm/SPECS
rpmbuild -ba helloworld.spec
```

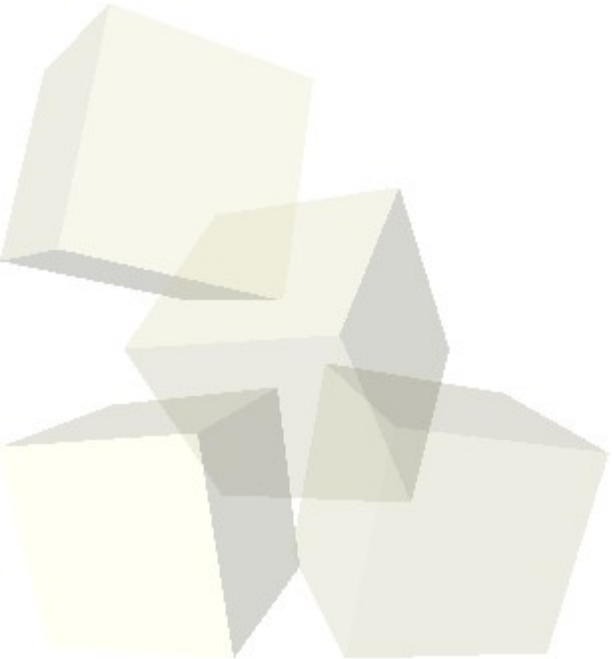
Observe atentamente o processo de construção, pois se der algum erro, você saberá onde parou e poderá depurar melhor para consertar.

O Resultado!

Tchan-nam! Se você fez tudo certinho, você terá construído dois pacotes: um binário e um SRPM. Suas localizações são:

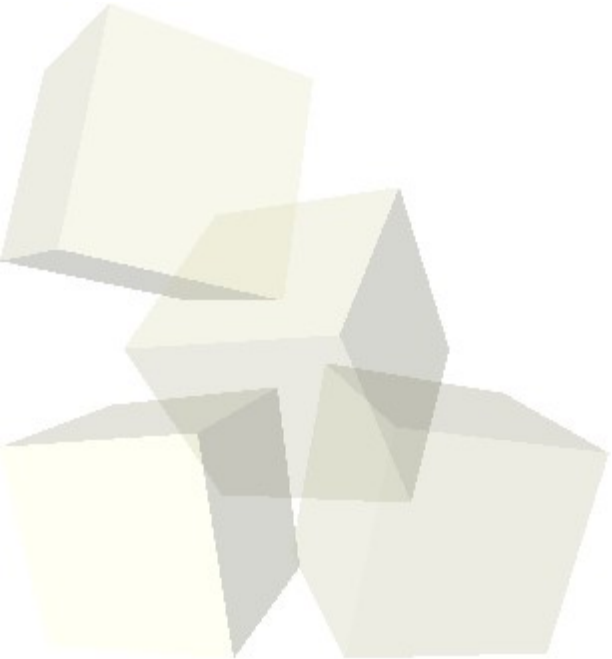
- **Binário:** rpm/RPMS/<*arquitetura*>
- **Fonte:** rpm/SRPMS

Dependendo do seu processador, e/ou flags de compilação do programa, substitua o <*arquitetura*> por i386, i686, x86_64, ppc, etc. Caso seu programa não for binário, estará no *noarch*.



man rpm

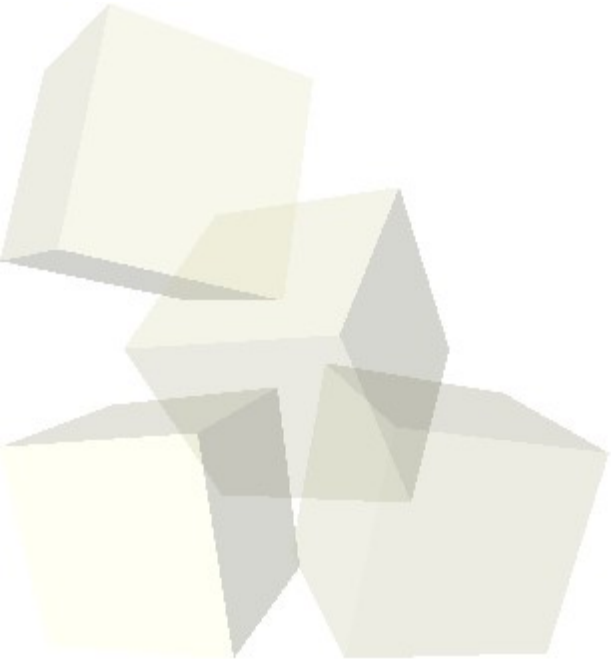
:-)



Outras Notas

O RPM é uma poderosa ferramenta, que contém muita coisa além do que vimos aqui.

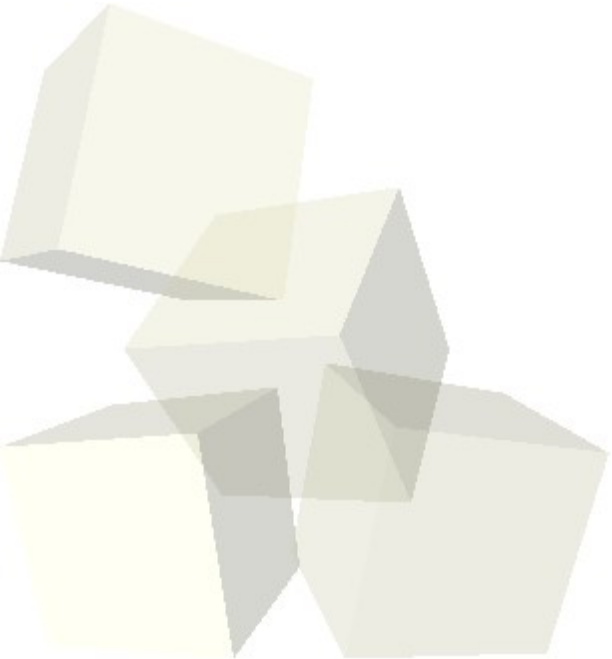
- Macros;
- Extensões em linguagens diversas (ex. Python);
- Triggers;
- Controle de assinaturas;
- etc



Documentação

Boas referências para se aprender mais sobre RPM:

- http://www.rpm.org
- http://www.rpm.org/max-rpm/
- http://www.linuxdoc.org/HOWTO/RPM-HOWTO/
- https://moin.conectiva.com.br/GuiaCriacaoSpecsCL



Linux na pele!!!

...Até a próxima!