



Linux: Managing Software

IT 4423

Unix/Linux Administration

J.G. Zheng
Spring 2012



Overview

- ◆ What are the ways to install/uninstall software in Linux/Unix?

Basic Approaches

◆ Package manager

- A package contains files needed for the software to function correctly. These files can be configuration files, binaries, and even pre- and postscripts to run while installing the software.
- A package manager supports
 - ◆ querying, installing, and uninstalling software
 - ◆ maintaining a database that stores various items of information about the packages

◆ Download source, compile, and install

Major Package Managers

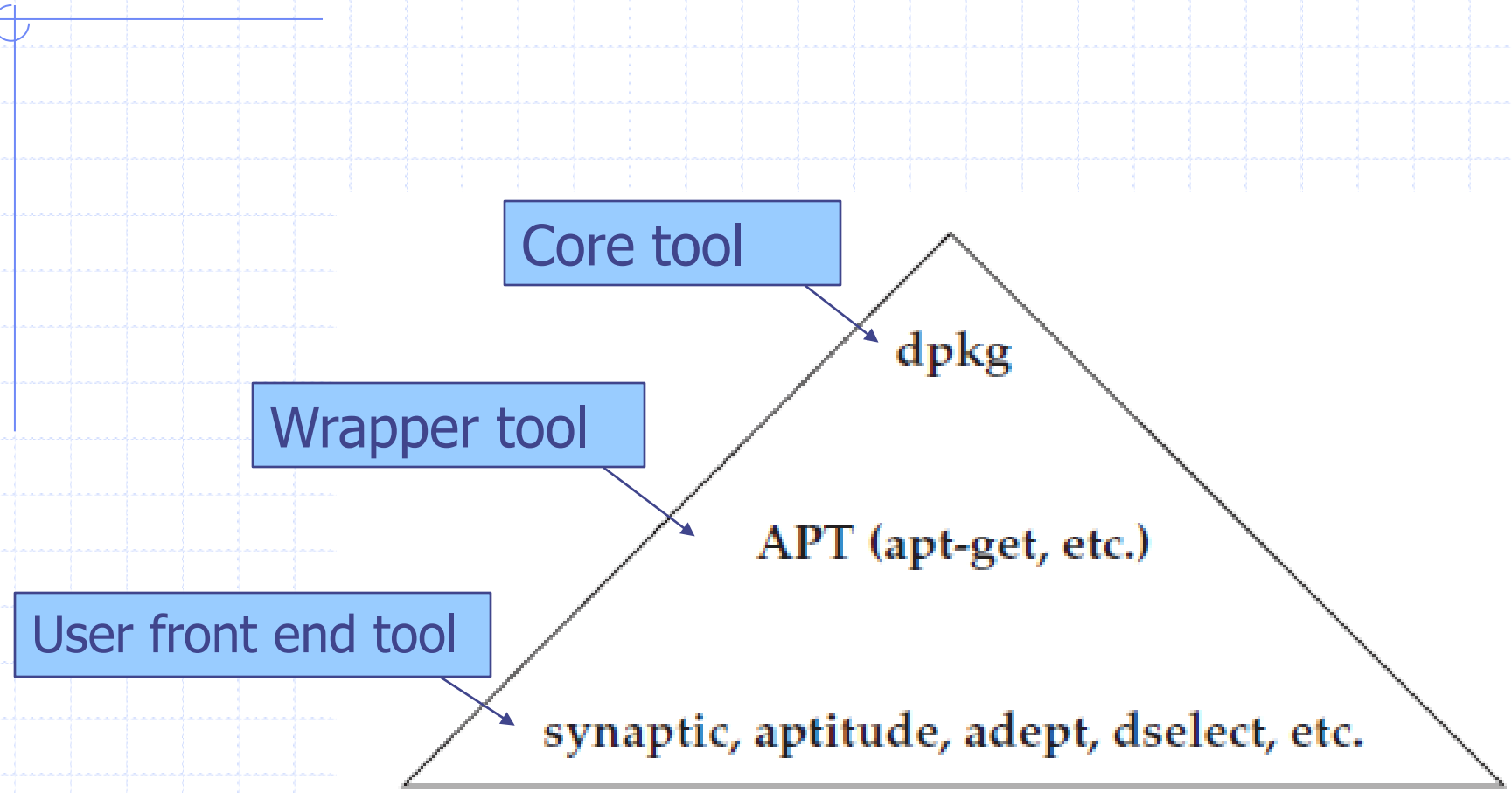
◆ RPM: Red Hat Package Manager

- Mainly used in Red Hat, OpenSuSE, Fedora, and other Red Hat derivatives

◆ DPMS: Debian Package Management System

- Used in Debian and Debian-like systems (such as Ubuntu).
- Package name ends with `.deb`

Tools used based on DPMS



APT: Advanced Packaging Tool

- ◆ APT is a wrapper tool that works with core libraries (DPMS) to handle the installation and removal of software on the Debian GNU/Linux distribution and its variants
- ◆ APT simplifies the process of managing software on Unix-like computer systems by automating the retrieval, configuration and installation of software packages.
 - Packages containing dependencies required by the package(s) specified for installation will be automatically retrieved and installed

Repository

- ◆ A repository is a place to store available software packages
 - APT can directly search and retrieve these packages

- ◆ A repository source configuration file (`/etc/apt/sources.list`) is used to locate and retrieve packages, and also obtain information about available (but uninstalled) packages.

Install/Remove Software

- ◆ dpkg can be used to install or remove a .deb package
 - **dpkg --install lynx_2.8.6-2ubuntu2_i386.deb**
 - **dpkg --remove lynx**
- ◆ apt-get is also used to install any software available in the repositories on the Internet or locally (CD/DVD ROM, file system, etc).
- ◆ Using **apt-get** to install/remove software is a little easier
 - APT will usually take care of any dependency issues
 - Users only need to know a part of the name of the software; no need to know the exact version number.
 - No need to manually download the software before installing.

GUI tools

◆ Aptitude

- Aptitude is a text-based user interface used for console

◆ Synaptic

- This is a GUI tool in Gnome to manage software

◆ Ubuntu Software Center

- Intended to replace synaptic

Compile and Install from Source

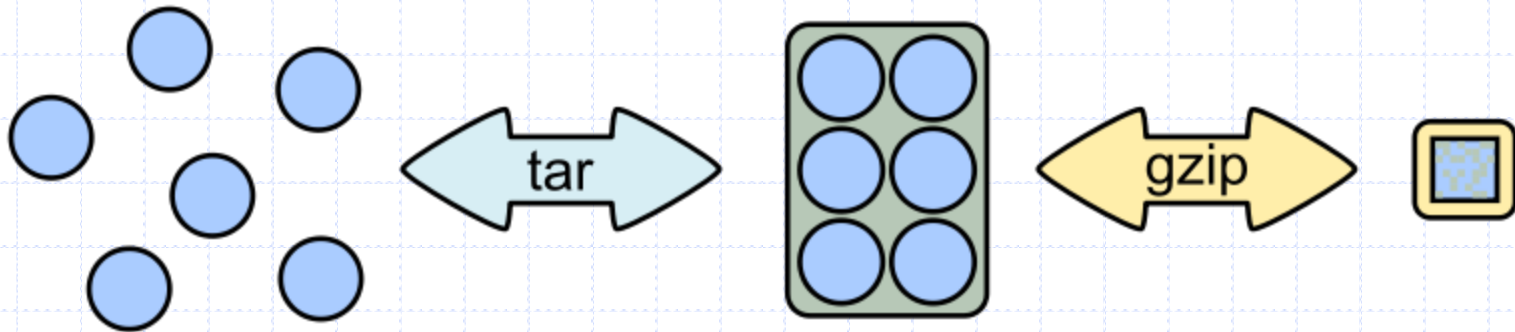
◆ General steps

- Get the source package (frequently a compressed TAR file)
- Extract all files in a local directory
- View the installation file
 - ◆ INSTALL or ReadMe
- Configure it for compilation and installation
 - ◆ Usually there is a "configure" executable file
- Compile and install it use the "make" command
 - ◆ type "make install"

TAR

◆ Source code is usually wrapped by in a .tar.gz file

- tar wraps separate files
- gzip compresses the file



◆ Use the "tar" tool to extract files

dpkg

- ◆ list all currently installed
 - **dpkg -l**
- ◆ Get basic information about selected packages
 - **dpkg -l nano**
 - **dpkg -l bas***
- ◆ The command to get more detailed information about the bash package is
 - **dpkg -p nano**
- ◆ To view the list of files that comes with the bash package
 - **dpkg-query -L firefox**
- ◆ Query for specific information about a package
 - **dpkg-query -W --showformat='\${Package} \${Architecture}\n' bash**

apt-get

- ◆ Use "apt-cache" to Search Programs

apt-cache search "vi improved"

- ◆ Install a package

apt-get install vim

- ◆ Remove software

apt-get remove vim

tar

◆ Basic commands

- c --- to create a tar file
- t --- table of contents, see the names of all files or those specified in other command line arguments.
- x --- extract (restore) the contents of the tar file.
- f --- specifies the filename (which follows the f) used to tar into or to tar out from
- z --- use zip/gzip to compress the tar file or to read from a compressed tar file
- v --- verbose output, show, e.g., during create or extract, the files being stored into or restored from the tar file.

◆ Quick examples

- Create a tar file
 - ◆ tar cf my-file.tar *
- Extract
 - ◆ tar xvf hello.2.3.tar.gz
- See content
 - ◆ tar tf my-file.tar

◆ Reference

- <http://www.computerhope.com/unix/utar.htm>

Summary

◆ Key concepts and terms

- Software package
- RPM, APT, DPMS
- Software repository
- Synaptic, Aptitude, TAR

◆ Key practices

- Use dpkg command to view installed packages
- Use apt-get to install and remove packages
- Use GUI tools such as Synaptic to manage software packages