

Installers and Package Managers

Definitions, Differences, Uses, Choices

Lewis Rosenthal, CNA, CLE, CLP, CWTS
Partner, Rosenthal & Rosenthal, LLC

Installer: What is it?

Simply put, something which places files and/or directories somewhere.

Simple Installer

Unzip

Why? It extracts something somewhere.

More Elaborate Installer

Single-purpose executable or script

Copies files to user-selectable destination, may create configuration files, may create desktop objects, and may launch the application and/or readme and/or registration utility.

Intelligent Installer

WarpIN

Analogous to Windows Installer; successor to venerable IBM Installer

WarpIN Features

- Supports system configuration (CONFIG.SYS, WPS classes, WPS objects)
- Ability to undo system changes
- Checks package dependencies
- Can execute external programs for system configuration after install
- REXX support built in
- Maintains database of installed applications

Package Manager

Wikipedia defines a package manager as:

“... a collection of software tools to automate the process of installing, upgrading, configuring, and removing software packages for a computer's operating system in a consistent manner. It typically maintains a database of software dependencies and version information to prevent software mismatches and missing prerequisites.”

Source - Package management system. (2013, August 5). In *Wikipedia, The Free Encyclopedia*. Retrieved September 12, 2013, from https://en.wikipedia.org/wiki/Package_management_system

Standard Package Manager

rpm (Red Hat Package Manager)

Checks prerequisites listed in the package to be installed, checks for existing versions (upgrade/downgrade), performs installation or removal.

Differences 1

Installer

- One-way (install)
- Usually ignorant of conflicts; prerequisites
- No package database

Package Manager

- Two-way (install/remove)
- Aware of conflicts & prerequisites
- Maintains database of installed applications (installed by it)

Differences 2

WarpIN

- Scriptable
- Can be used as a stub, providing self-installing packages
- Binary, OS/2-style INI package database
- Packaging can be tedious

rpm

- Not scriptable
- Used as a stand-alone application only
- BerkeleyDB package database
- Packaging more widely known

Differences 3

WarpIN

- Bundled with eCS
- User-friendly UI
- Destinations may be user-configurable at time of install

rpm

- Must be installed
- Commandline tool
- Destinations set in package, but package may be flagged as “relocatable”

Package Management Utility

Wrapper or supplement to a package manager, providing additional function (file repository locations; package download); front end to the package manager.

Standard Utility

yum (Yellowdog Updater, Modified)

What a Utility Does

- Maintains list of locations for obtaining packages (repositories, or repos)
- Allows enabling and disabling of configured repos
- Provides information on requested package(s) (version available, which repo, prerequisites, what the package does). This data is stored in the package metadata.

Differences

rpm

- Maintains database of applications installed by it
- Installs & removes packages
- Can query details of installed or locally available packages

yum

- Maintains database of repos
- Does not install (calls rpm)
- Can query details of multiple packages, installed or not

Example: Unzip

```
40% / 2 (WINDOW)
[c:\>]unzip j:\DOWNLOAD\os2\assoed212.zip
Archive:  j:/DOWNLOAD/os2/assoed212.zip
  inflating: assoedit/ASSOEDIT.exe
  inflating: assoedit/assoedit.hlp
  inflating: assoedit/ASSOEDIT.ICO
  inflating: assoedit/ASSOEDIT.TXT
 extracting: assoedit/source.zip      (160 bytes EAs)
  inflating: assoedit/WPTOOLS.DLL    (40 bytes EAs)

[c:\>]dir assoed*

Volume in drive C is OS2BOOT          Serial number is 7C9A:990C
Directory of  C:\assoed*

9-12-13  17:44          <DIR>          124  assoedit
          0 bytes in 0 files and 1 dir
          3,492,589,568 bytes (3,330MB) free

[c:\>]
```


Example: REXX Installer

```
INSTALL.COM
[j:\download\os2\ecalc2prev31]install
eCALC installation

Creating WPS objects...
Installation done. Hit ENTER to continue.
```

Example: WarpIN

The screenshot displays the WarpIN installer interface. On the left, a 'DRIVE INFORMATION' window shows a table of drive details. Below it, the 'INSTALL STATUS' window lists the current step as 'Collecting information'. The main 'WarpIN' window shows a 'Packages' list with one item, 'bootAble (V6.9.1.0, 853 KB)', selected. The 'Install path' is set to 'C:\TOOLS\bootAble'.

Drive	Free(KB)	Required(KB)
C:	3,411,192	853
D:	4,120,887	0
F:	81,356,424	0
G:	81,354,752	0
J:	12,087,992	0
L:	11,011,936	0
N:	81,356,424	0
P:	11,011,936	0
S:	11,011,936	0
Total:	296,723,483	853

INSTALL STATUS

- Loading archive
- ▶ Collecting information**
- Removing packages
- Unpacking files
- Updating COI INF: SYS
- Registering Workplace Shell classes
- Creating Workplace Shell objects
- Updating database

WarpIN Archive Selections Help

Packages

Please select the package which is to be installed.

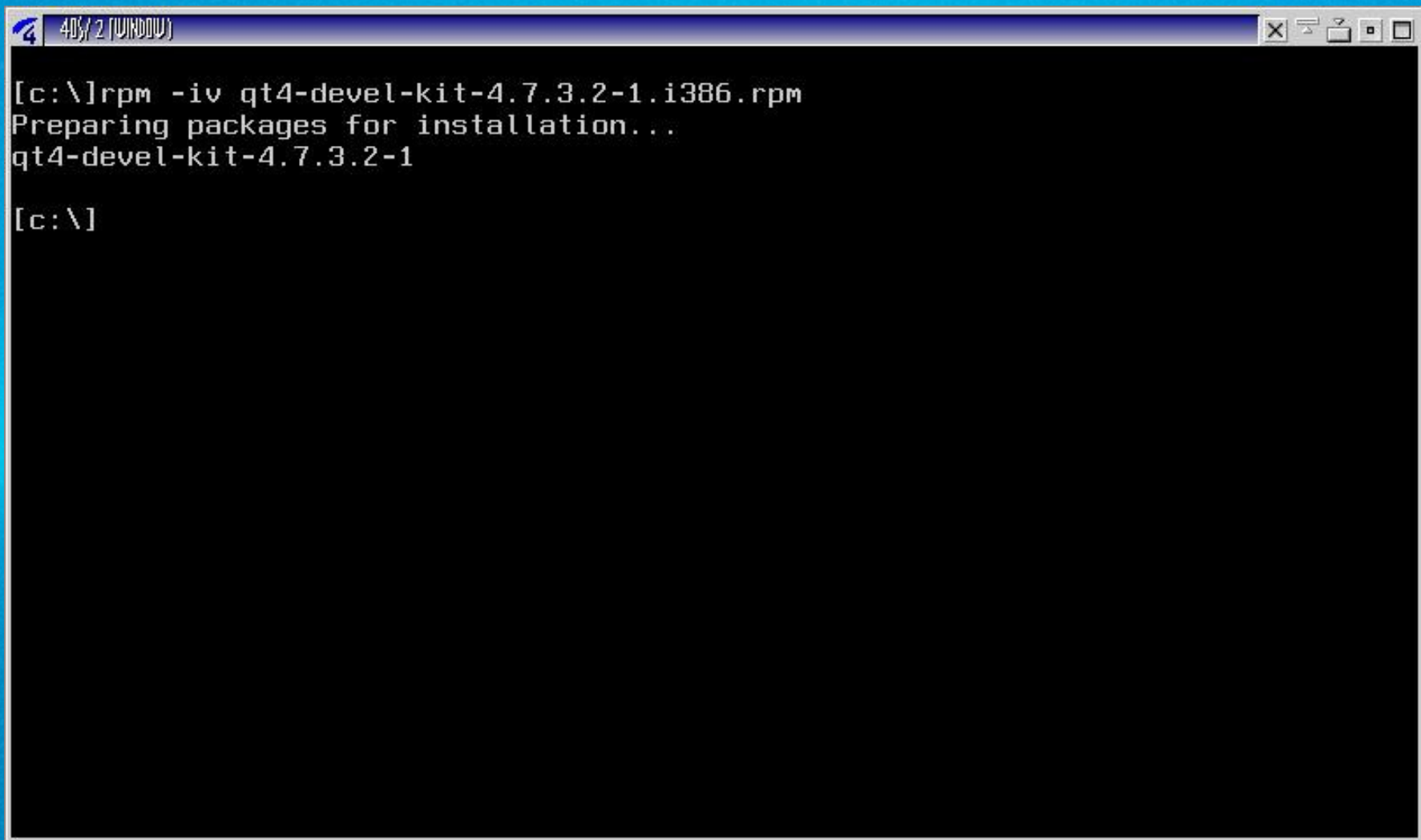
bootAble
(V6.9.1.0, 853 KB)

Install path: C:\TOOLS\bootAble

Installed version:

◀ Back Next ▶ Cancel

Example: rpm



```
[c:\>] rpm -iv qt4-devel-kit-4.7.3.2-1.i386.rpm
Preparing packages for installation...
qt4-devel-kit-4.7.3.2-1

[c:\>]
```

Example: yum info

```
4 40% / 2 (WINDOW)
[c:\]yum info qt4-devel-kit
Installed Packages
Name       : qt4-devel-kit
Arch      : i386
Version   : 4.7.3.2
Release   : 1
Size      : 0.0
Repo      : installed
Summary   : Qt 4 development kit
URL       : http://www.qtsoftware.com/
License   : LGPLv2 with exceptions or GPLv3 with exceptions
Description: Qt is a software toolkit for developing applications.
:
: This meta package installs all individual packages containing
: various Qt development components, including the Qt headers and
: libraries, the Linguist, Assistant and Designer applications,
: development tools, example Qt applications and the API
: documentation.

[c:\]
```

Example: yum whatprovides

```
40% 2 (WINDOW)
[c:\]yum whatprovides */qt4
libqt4-4.7.3.2-1.i386 : Qt 4 runtime libraries
Repo      : netlabs-rel-2rosenthals
Matched from:
Filename  : /@unixroot/usr/share/doc/qt4
Filename  : /@unixroot/usr/share/qt4
Filename  : /@unixroot/usr/lib/qt4

libqt4-4.7.3-1.i386 : Qt 4 runtime libraries
Repo      : netlabs-rel-2rosenthals
Matched from:
Filename  : /@unixroot/usr/share/doc/qt4
Filename  : /@unixroot/usr/share/qt4
Filename  : /@unixroot/usr/lib/qt4

[c:\]
```

Example: yum update

```
[c:\]yum update
Setting up Update Process
Resolving Dependencies
--> Running transaction check
---> Package coreutils.i386 0:8.6-10.oc00 set to be updated
---> Package findutils.i386 1:4.4.2-5.oc00 set to be updated
---> Package gcc.i386 0:4.4.6.17-1.oc00 set to be updated
---> Package gcc-wlink.i386 0:4.4.6.17-1.oc00 set to be updated
---> Package gcc-wrc.i386 0:4.4.6.17-1.oc00 set to be updated
---> Package libc.i386 0:0.6.5-18.oc00 set to be updated
---> Package libc-devel.i386 0:0.6.5-18.oc00 set to be updated
---> Package libgcc446.i386 0:4.4.6.17-1.oc00 set to be updated
---> Package libssp.i386 0:4.4.6.17-1.oc00 set to be updated
---> Package libstdc++.i386 0:4.4.6.17-1.oc00 set to be updated
---> Package libsupc++.i386 0:4.4.6.17-1.oc00 set to be updated
---> Package os2-base.i386 0:0.0.0-7.oc00 set to be updated
---> Package zip.i386 0:3.0-6.oc00 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved
```

Example: yum update

```
=====  
Package           Arch      Version           Repository         Size  
=====  
Installing:  
libssp            i386      4.4.6.17-1.oc00  netlabs-rel-2rosenthals 7.1 k  
  replacing gcc-stack-protector.i386 4.4.6-15.oc00  
libstdc++         i386      4.4.6.17-1.oc00  netlabs-rel-2rosenthals 220 k  
  replacing gcc-stdc++-shared-library.i386 4.4.6-15.oc00  
libsupc++         i386      4.4.6.17-1.oc00  netlabs-rel-2rosenthals 37 k  
  replacing gcc-supc++-shared-library.i386 4.4.6-15.oc00  
Updating:  
coreutils        i386      8.6-10.oc00      netlabs-rel-2rosenthals 5.2 M  
findutils        i386      1:4.4.2-5.oc00   netlabs-rel-2rosenthals 722 k  
gcc               i386      4.4.6.17-1.oc00  netlabs-rel-2rosenthals 14 M  
gcc-wlink        i386      4.4.6.17-1.oc00  netlabs-rel-2rosenthals 3.7 k  
gcc-wrc          i386      4.4.6.17-1.oc00  netlabs-rel-2rosenthals 3.7 k  
libc             i386      0.6.5-18.oc00    netlabs-rel-2rosenthals 927 k  
libc-devel       i386      0.6.5-18.oc00    netlabs-rel-2rosenthals 30 M  
libgcc446        i386      4.4.6.17-1.oc00  netlabs-rel-2rosenthals 25 k  
os2-base         i386      0.0.0-7.oc00     netlabs-rel-2rosenthals 6.2 k  
zip              i386      3.0-6.oc00       netlabs-rel-2rosenthals 321 k  
  
Transaction Summary  
=====  
Install          3 Package(s)  
Upgrade          10 Package(s)  
  
Total download size: 52 M  
Is this ok [y/N]:
```

Example: yum repolist

```
[c:\]yum repolist
repo id                repo name                status
netlabs-exp-2rosenthals Netlabs experimental repository - USA Rosenthal M  444
netlabs-rel-2rosenthals Netlabs Stable Repository - USA Rosenthal Mirror  1,139
repolist: 1,583

[c:\]
```


FAQs

- How do I know which to use?

The package will determine which tool is best suited for the job.

- How do I install yum?

You need rpm, as yum by itself won't do much (and it has a dependency on rpm). To install rpm and yum, there is a WarpIN bootstrap. See: <http://trac.netlabs.org/rpm/wiki/RpmInstall>

FAQs

- Can WarpIN download new packages?

No, WarpIN is a package manager, not a package management utility.

- Can WarpIN read my rpm database?

No, WarpIN can only read its own database. WarpIN is unaware of any packages installed either by unzip or rpm.

FAQs

- Can rpm download packages?

No, rpm is a package manager, not a package management utility.

- Can rpm read my WarpIN database?

No, rpm can only read its own database. rpm is unaware of any packages installed either by unzip or WarpIN.

FAQs

- Can yum download packages?

Yes.

- Can yum install packages?

Not by itself, no. yum relies on rpm to handle installing and removing packages.

FAQs

- Can yum read my WarpIN and/or rpm database?

No, but this is irrelevant. Yum only needs to maintain a database of repositories, not packages.

What's Still Needed for eCS?

Conflicts can occur when an application has been packaged in different ways, and package managers are mixed. A bridge between WarpIN and rpm would be a major improvement.

Conflict Example: Odin

Odin installed via unzip to O:\Libraries\Odin

Odin installed via yum install libodin, which installs to %UNIXROOT%\var\lib\odin

libodin is not relocatable:

```
[j:\]rpm -qpi libodin-0.8.9-1.i386.rpm | grep Relocations
```

```
Name      : libodin          Relocations: (not relocatable)
```

Conflict Example: Odin

Result: Two separate installations

What's Still Needed for eCS?

FastestMirror plugin for yum. As Rosenthal & Rosenthal provides a US-based mirror of the both Netlabs repositories (experimental and release), users now must manually choose which to enable. FastestMirror would allow both sets of repositories to be enabled, and would test connecting to each, choosing the most responsive for that session.

References

- WarpIN – Netlabs, Jens Bläckman, Ulrich Möller, Paul Ratcliffe
- rpm – community, Red Hat
- yum – Seth Vidal
- rpm and yum for eCS – Netlabs, Yuri Dario