

## Querying

### Searching Packages

`zypper search` or `zypper se`

Examples:

`zypper se -dC --match-words RSI` #look for RSI acronym (case-sensitively), also in summaries and descriptions  
`zypper se 'yast*'` #show all packages starting with 'yast'  
`zypper se -r upd` #list all packages from 'upd' repository  
`zypper se -i sqlite` #show all 'sqlite' installed packages  
`zypper se -t pattern -r upd` #list all patterns available in the 'upd' repository

### Getting Information about Packages

`zypper info` or `zypper if`

Examples:

`zypper info amarok`  
`zypper info -t patch amarok` #show info for 'amarok' patch  
`zypper patch-info amarok` #same as above  
`zypper info -t pattern lamp_server` #info 'lamp\_server' pattern

### Getting Information about Dependencies

`zypper what-provides` or `zypper wp`

Examples:

`zypper wp firefox`

## Utilities

### Verify Dependencies

`zypper verify` or `zypper ve`

Note:

This is useful in cases of a broken system

### Install New Recommended Packages

`zypper install-new-recommends` or `zypper inr`

## Package Locks

### Lock all Packages

`zypper addlock` or `zypper al`

Examples:

`zypper al 'yast2*'` #lock all packages starting with 'yast2'

### Remove Locks

`zypper removelock` or `zypper rl`

Examples:

`zypper rl 'yast2*'` #remove locks to all packages starting with 'yast2'

### List Locks

`zypper locks` or `zypper ll`

## Update Management

### Listing Needed Patches

`zypper list-patches` or `zypper lp`

### Applying Patches

`zypper patch`

### Listing All Patches

`zypper patches`

### Checking Patches

`zypper patch-check` or `zypper pchk`

### Getting Information About Patches

`zypper patch-info`

`zypper info -t patch`

### Packages Updates

`zypper list-updates` or `zypper lu`

`zypper update` or `zypper up`

## Distribution Upgrade

`zypper dist-upgrade` or `zypper dup`

Note:

When doing a distribution update, the best is to work only with the repositories of the distribution you want to install.

## Vocabulary

### Repositories

HTTP or FTP server, DVD, or a folder on a local disc. where a group or set of packages are located.

### Resource Identifiers (URI)

To specify locations of repositories or other resources (RPM files, .repo files) you can use any type of URIs supported by libzypp. See <http://en.opensuse.org/Libzypp/URI> for a complete list and usage examples.

### Refresh

Refreshing a repository means downloading metadata of packages from the medium (if needed), storing it in local cache (typically under `/var/cache/zypp/raw/<alias>` directory) and preparing the metadata into .solv files (building the solv cache), typically under `/var/cache/zypp/solv/<alias>`.

### Services

Services are one level above repositories and serve to manage repositories or to do some special tasks. Libzypp currently supports only one type of services, the Repository Index Service (RIS).

### Package Types

zypper works with several types of resource objects, called resolvables. A resolvable is a package, patch, pattern, or a product.

package - an ordinary RPM package

patch - update of one or more packages.

pattern - group of packages required or recommended to install some functionality

product - group of packages which are necessary to install a product