



*Full credit is given to the above companies including the OS that this PDF file was generated!*

## **Red Hat Enterprise Linux Release 9.2 Manual Pages on 'flatpak-update.1' command**

**\$ man flatpak-update.1**

FLATPAK UPDATE(1) flatpak update FLATPAK UPDATE(1)

### NAME

flatpak-update - Update an application or runtime

### SYNOPSIS

flatpak update [OPTION...] [REF...]

flatpak update [OPTION...] --appstream [REMOTE]

### DESCRIPTION

Updates applications and runtimes. REF is a reference to the application or runtime to update. If no REF is given, everything is updated, as well as appstream info for all remotes.

Each REF argument is a full or partial identifier in the flatpak ref format, which looks like "(app|runtime)/ID/ARCH/BRANCH". All elements except ID are optional and can be left out, including the slashes, so most of the time you need only specify ID. Any part left out will be matched against what is installed, and if there are multiple matches an error message will list the alternatives.

By default this looks for both apps and runtimes with the given REF, but you can limit this by using the --app or --runtime option, or by supplying the initial element in the REF.

Normally, this command updates the application to the tip of its branch. But it is possible to check out another commit, with the --commit option.

If the configured remote for a ref being updated has a collection ID

configured on it, flatpak will search mounted filesystems such as USB drives as well as Avahi services advertised on the local network for the needed refs, in order to support offline updates. See `ostree-find-remotes(1)` for more information.

Note that updating a runtime is different from installing a different branch, and runtime updates are expected to keep strict compatibility. If an application update does cause a problem, it is possible to go back to the previous version, with the `--commit` option.

Unless overridden with the `--user`, `--system` or `--installation` option, this command updates any matching refs in the standard system-wide installation and the per-user one.

## OPTIONS

The following options are understood:

`-h, --help`

Show help options and exit.

`--user`

Update a per-user installation.

`--system`

Update the default system-wide installation.

`--installation=NAME`

Updates a system-wide installation specified by NAME among those defined in `/etc/flatpak/installations.d/`. Using

`--installation=default` is equivalent to using `--system`.

`--arch=ARCH`

The architecture to update for. See `flatpak --supported-arches` for architectures supported by the host.

`--subpath=PATH`

Install only a subpath of the ref. This is mainly used to install a subset of locales. This can be added multiple times to install multiple subpaths. If this is not specified the subpaths specified at install time are reused.

`--commit=COMMIT`

Update to this commit, instead of the tip of the branch. You can

find commits using flatpak remote-info --log REMOTE REF.

--no-deploy

Download the latest version, but don't deploy it.

--no-pull

Don't download the latest version, deploy whatever is locally available.

--no-related

Don't download related extensions, such as the locale data.

--no-deps

Don't update or install runtime dependencies when installing.

--app

Only look for an app with the given name.

--appstream

Update appstream for REMOTE, or all remotes if no remote is specified.

--runtime

Only look for a runtime with the given name.

--sideload-repo=PATH

Adds an extra local ostree repo as source for installation. This is equivalent to using the sideload-repos directories (see flatpak(1)), but can be done on a per-command basis. Any path added here is used in addition to ones in those directories.

-y, --assumeeyes

Automatically answer yes to all questions (or pick the most prioritized answer). This is useful for automation.

--noninteractive

Produce minimal output and avoid most questions. This is suitable for use in non-interactive situations, e.g. in a build script.

--force-remove

Remove old files even if they're in use by a running application.

-v, --verbose

Print debug information during command processing.

--ostree-verbose

Print OSTree debug information during command processing.

## EXAMPLES

```
$ flatpak --user update org.gnome.gedit
```

## SEE ALSO

flatpak(1), flatpak-install(1), flatpak-list(1), ostree-find-remotes(1)

flatpak

FLATPAK UPDATE(1)