

Part VI. GTK+ Tools: GTK+ 3 Reference Manual

Table of Contents

[gtk3-demo](#) — Demonstrate GTK+ widgets

[gtk3-demo-application](#) — Demonstrate GtkApplication

[gtk3-widget-factory](#) — Demonstrate GTK+ widgets

[gtk3-icon-browser](#) — List themed icons

[gtk-query-immodules-3.0](#) — Input method module registration utility

[gtk-update-icon-cache](#) — Icon theme caching utility

[gtk-encode-symbolic-svg](#) — Symbolic icon conversion utility

[gtk-builder-tool](#) — GtkBuilder file utility

[gtk-launch](#) — Launch an application

[gtk-query-settings](#) — Utility to print name and value of all GtkSettings properties

[broadwayd](#) — Broadway display server

Changes in GTK+ 3.22

The CSS parser has gotten a bit more selective in what it accepts as valid values for the font: shorthand. Following the CSS specification, at least a size and a family name are required now. If you want to change an individual facet of the font, like the weight, use the individual CSS properties: font-weight, font-size, font-family, etc.

The CSS parser now warns about some more constructs that are not according to the CSS spec, such as gradients with a single color stop. Instead, you can just use `image(<color>)`.

The “[ignore-hidden](#)” property has not been working properly for a long time, and we’ve not documented it as broken and deprecated. The recommended alternative for reserving space of widgets that are not currently shown in the UI is to use a [GtkStack](#) (with some 'filler' widget, e.g. an empty [GtkBox](#), in another page).

`GtkHeaderBar` now respects the `hexexpand` property for its custom title widget and its packed children. This change may inadvertently cause the layout of those children to change, if they unintentionally had `hexexpand` set before.

The behavior of the `expand` flag in `GtkTables` [GtkAttachOptions](#) has been changed to (again) match the behavior in [GtkBox](#) and in GTK+ 2.x. These options don't cause the table itself to expand.

The way `GtkPopover` behaved during a call to `gtk_widget_hide()` violated some of the internal assumptions GTK+ makes about widget visibility. `gtk_popover_popup()` and `gtk_popover_popdown()` have been introduced to show or hide the popover with a transition, while `gtk_widget_show()` and `gtk_widget_hide()` on a `GtkPopover` now work the same way they do on any other widget and immediately hide (or show) the popover.

gtk3-demo

gtk3-demo — Demonstrate GTK+ widgets

Synopsis

```
gtk3-demo [--help] [--list] [--run EXAMPLE] [--autoquit]
```

Description

gtk3-demo is a collection of examples. Its purpose is to demonstrate many GTK+ widgets in a form that is useful to application developers.

The application shows the source code for each example, as well as other used resources, such as ui files and icons.

Options

The following options are understood:

<code>-h, --help</code>	Show help options
<code>--list</code>	List available examples.
<code>--run EXAMPLE</code>	Run the named example. Use <code>--list</code> to see the available examples.
<code>--autoquit</code>	Quit after a short timeout. This is intended for use with <code>--run</code> , e.g. when profiling.

gtk3-demo-application

gtk3-demo-application — Demonstrate GtkApplication

Synopsis

gtk3-demo-application

Description

gtk3-demo-application is an example application used by **gtk3-demo**. There is no need to call it manually.

gtk3-widget-factory

gtk3-widget-factory — Demonstrate GTK+ widgets

Synopsis

gtk3-widget-factory [--help]

Description

gtk3-widget-factory is a collection of examples. Its purpose is to demonstrate many GTK+ widgets in a form that is useful to GTK+ theme developers.

The application shows widgets in different, typical combinations and states.

Options

The following options are understood:

-h, --help

Show help options

gtk3-icon-browser

gtk3-icon-browser — List themed icons

Synopsis

gtk3-icon-browser [--help]

Description

gtk3-icon-browser is a utility to explore the icons in the current icon theme. It shows icons in various sizes, their symbolic variants where available, as well as a description of the icon and its context.

Options

The following options are understood:

<code>-h, --help</code>	Show help options
-------------------------	-------------------

gtk-query-immodules-3.0

gtk-query-immodules-3.0 — Input method module registration utility

Synopsis

gtk-query-immodules-3.0 [--update-cache] [MODULE...]

Description

gtk-query-immodules-3.0 collects information about loadable input method modules for GTK+ and writes it to the default cache file location, or to `stdout`.

If called without arguments, it looks for modules in the GTK+ input method module path.

If called with arguments, it looks for the specified modules. The arguments may be absolute or relative paths.

Normally, the output of **gtk-query-immodules-3.0** is written to `libdir/gtk-3.0/3.0.0/immodules.cache`, where GTK+ looks for it by default. If it is written to some other location, the `GTK_IM_MODULE_FILE` environment variable can be set to point GTK+ at the file.

Options

<code>--update-cache</code>	Write the output to the default cache location instead of <code>stdout</code>
-----------------------------	---

Files

<code>libdir/gtk-3.0/3.0.0/immodules.cache</code>	The default im cache file used by GTK+ applications
---	---

Environment

<u>GTK_PATH</u>	Prepends directories to the input method module path
<u>GTK_IM_MODULE_FILE</u>	Specifies an alternative im module cache for GTK+ applications

gtk-update-icon-cache

gtk-update-icon-cache — Icon theme caching utility

Synopsis

```
gtk-update-icon-cache [--force] [--ignore-theme-index] [ --index-only | --include-image-data ] [--source  
NAME ] [--quiet] [--validate] PATH
```

Description

gtk-update-icon-cache creates mmapable cache files for icon themes.

It expects to be given the `PATH` to a icon theme directory containing an `index.theme`, e.g. `/usr/share/icons/hicolor`, and writes a `icon-theme.cache` containing cached information about the icons in the directory tree below the given directory.

GTK+ can use the cache files created by **gtk-update-icon-cache** to avoid a lot of system call and disk seek overhead when the application starts. Since the format of the cache files allows them to be mmaped shared between multiple applications, the overall memory consumption is reduced as well.

Options

<code>--force, -f</code>	Overwrite an existing cache file even if it appears to be uptodate.
<code>--ignore-theme-index, -t</code>	Don't check for the existence of <code>index.theme</code> in the icon theme directory. Without this option, gtk-update-icon-cache refuses to create an icon cache in a directory which does not appear to be the toplevel directory of an icon theme.
<code>--index-only, -i</code>	Don't include image data in the cache.
<code>--include-image-data</code>	Include image data in the cache.
<code>--source, -c</code>	Output a C header file declaring a constant <code>NAME</code> with the contents of the icon cache.
<code>--quiet, -q</code>	Turn off verbose output.
<code>--validate, -v</code>	Validate existing icon cache.

gtk-encode-symbolic-svg

gtk-encode-symbolic-svg — Symbolic icon conversion utility

Synopsis

```
gtk-encode-symbolic-svg [OPTION...] PATH WIDTHxHEIGHT
```

Description

gtk-encode-symbolic-svg converts symbolic svg icons into specially prepared png files. GTK+ can load and recolor these pngs, just like original svgs, but loading them is much faster.

`PATH` is the name of a symbolic svg file, `WIDTHxHEIGHT` are the desired dimensions for the generated png file.

To distinguish them from ordinary pngs, the generated files have the extension `.symbolic.png`.

Options

-o DIRECTORY, --output DIRECTORY

Write png files to DIRECTORY instead of the current working directory.

gtk-builder-tool

gtk-builder-tool — GtkBuilder file utility

Synopsis

gtk-builder-tool [COMMAND] [OPTION...] FILE

Description

gtk-builder-tool can perform various operations on GtkBuilder .ui files.

You should always test the modified .ui files produced by **gtk-builder-tool** before using them in production.

Commands

The following commands are understood:

validate
simplify

Validates the .ui file and report errors to stderr.

Simplifies the .ui file by removing properties that are set to their default values and write the resulting XML to stdout, or back to the input file.

enumerate

Lists all the named objects that are created in the .ui file.

preview

Preview the .ui file. This command accepts options to specify the ID of an object and a .css file to use.

Simplify Options

The `simplify` command accepts the following options:

--replace

Write the content back to the .ui file instead of stdout.

Preview Options

The `preview` command accepts the following options:

--id= ID

The ID of the object to preview. If not specified, `gtk-builder-tool` will choose a suitable object on its own.

--css= FILE

Load style information from the given .css file.

gtk-launch

gtk-launch — Launch an application

Synopsis

gtk-launch [OPTION...] APPLICATION [URI...]

Description

gtk-launch launches an application using the given name. The application is started with proper startup notification on a default display, unless specified otherwise.

gtk-launch takes at least one argument, the name of the application to launch. The name should match application desktop file name, as residing in /usr/share/application, with or without the '.desktop' suffix.

If called with more than one argument, the rest of them besides the application name are considered URI locations and are passed as arguments to the launched application.

Options

The following options are understood:

-?, --help	Prints a short help text and exits.
--version	Prints the program version and exits.

gtk-query-settings

gtk-query-settings — Utility to print name and value of all GtkSettings properties

Synopsis

gtk-query-settings [PATTERN]

Description

gtk-query-settings prints both name and value of all properties available in the GtkSettings class. Optionally, you can filter which properties to list by specifying a PATTERN.

broadwayd

broadwayd — Broadway display server

Synopsis

```
broadwayd [--port PORT] [--address ADDRESS] [--unixsocket ADDRESS] [:DISPLAY]
```

Description

broadwayd is a display server for the Broadway GDK backend. It allows multiple GTK+ applications to display their windows in the same web browser, by connecting to **broadwayd**.

When using **broadwayd**, specify the display number to use, prefixed with a colon, similar to X. The default display number is 0.

```
broadwayd :5
```

Then point your web browser at `http://127.0.0.1:8085`. Start your applications like this:

```
GDK_BACKEND=broadway BROADWAY_DISPLAY=:5 gtk3-demo
```

You can add password protection for your session by creating a file in `$XDG_CONFIG_HOME/broadway.passwd` or `$HOME/.config/broadway.passwd` with a crypt(3) style password hash. A simple way to generate it is with `openssl`:

```
openssl passwd -1 > ~/.config/broadway.passwd
```

Options

<code>--port</code>	Use <code>PORT</code> as the HTTP port, instead of the default 8080 + (<code>DISPLAY</code> - 1).
<code>--address</code>	Use <code>ADDRESS</code> as the HTTP address, instead of the default <code>http://127.0.0.1:PORT</code> .
<code>--unixsocket</code>	Use <code>ADDRESS</code> as the unix domain socket address. This option overrides <code>--address</code> and <code>--port</code> . It is available only on Unix-like systems.
