Coding Style

============

The coding style to respect in this project is very similar to most

GLib projects. In particular, the following rules are largely adapted

from the PackageKit Coding Style.

\* 8-space tabs for indentation

\* Prefer lines of less than <= 80 columns

\* 1-space between function name and braces (both calls and macro

declarations)

\* If function signature/call fits in a single line, do not break it

into multiple lines

\* Prefer descriptive names over abbreviations (unless well-known)

and shortening of names. e.g `device` not `dev`

\* Single statements inside if/else should not be enclosed by '{}'

\* Use comments to explain why something is being done, but also avoid

over-documenting the obvious. Here is an example of useless comment:

// Fetch the document

fetch\_the\_document ();

\* Comments should not start with a capital letter or end with a full stop and

should be C-style, not C++-style, e.g. `/\* this \*/` not `// this`

\* Each object should go in a separate .c file and be named according

to the class

\* Use g\_autoptr() and g\_autofree whenever possible, and avoid `goto out`

error handling

\* Failing methods should return FALSE with a suitable `GError` set

\* Trailing whitespace is forbidden

\* Pointers should be checked for NULL explicitly, e.g. `foo != NULL` not `!foo`