

# Quick Guide to the Graphics Source#

---

## Get the Graphics source.#

If you have an account on Foundry27 you can do this:

```
svn checkout --username <userid> http://community.qnx.com/svn/repos/graphics/trunk
```

Alternately you can browse the [full source](#) tree. To compile the source you will need to checkout additional source from the BSP SVN -- [hardware/devi](#) and OS SVN -- [lib/compat](#), [lib/login](#), [lib/shutdown](#), [lib/util](#).

## Building the source#

Make sure that you have 6.4.x QNX Development environment installed. **If you don't properly create a staging area and suitable qconf-override.mk file, you risk the chance of corrupting your standard installation!** Read more on how building is done in QNX -- [Understanding the Neutrino Build Process](#) and [Building the OS Source](#) as an example of staging area handling. For QNX 6.4.0, if you haven't already done so, install the [srcversion patch](#) from the download section of the core os project as root.

```
# cd $QNX_TARGET/../../
# tar -xpf srcversion-patch-6.4.0.tar
```

On the top level (trunk/) do:

```
make hinstall
```

**NOTE:** You need to copy hardware/devi headers manually into your staging area:

**NOTE:** If you're attempting to build services/display with 6.4.1 libc headers, you'll require the following updated header [Gfx source/dispatch.h](#). It should be added to your stage directory under "/usr/include/sys/":

```
cp hardware/devi/private/sys/* <your_stage>/usr/include/sys
cp hardware/devi/public/sys/* <your_stage>/usr/include/sys
```

This will put all required public and private headers into your staging area, and now you are ready to build apps, services and libs.

**NOTE:** Windows applications: apps/phindows and tools/appbuilder won't build on Nto, so you need to exclude them from the build. You can do this by using this [handy script](#)

Do the build:

```
make install
```

If you need to build for specific platforms rather than all, you can use `export CPULIST="..."` to restrict builds to the specified platforms only before running make.

For example: `export CPULIST="x86"` will build for x86 only. Please note that older PhAB applications may not obey this envvar and will be build for all available platforms.

## Building apps/phindows – on Windows#

- Microsoft DirectX SDK must be installed, and these files: `d3d9.h`, `d3d9caps.h`, `d3d9types.h` and `ddraw.h` should be copied into the `apps/phindows/microsoft` directory (you need to create this directory yourself)
- The Cygwin Development Environment should be installed on your system.

## Sample apps#

Even though each of the posted modules could be used as a sample, we recommend using the designated sample modules. These modules are concise examples of graphics capabilities:

| Component | Location                        |
|-----------|---------------------------------|
| EGL       | <a href="#">apps/egl</a>        |
| GF        | <a href="#">apps/gf</a>         |
| GLES      | <a href="#">apps/gles</a>       |
| Photon    | <a href="#">apps/ph-samples</a> |

## Questions#

Please post questions related to the source code on [Graphics Source Code](#) forum.