

# Build Instructions for the PNT Integrity Library

This repository contains the top level CMake project for building the IS4S PNT Integrity Library, User Interface and their dependencies.

## System Requirements

The PNT Integrity Library is written in a cross-platform manner using C++. While it is expected to work on a wide variety of platforms it has been tested with on the following operating systems:

```
* Ubuntu Linux 16.04, 18.04 & 20.04
* MacOS 10.15
* Windows 10
```

The following additional tools are needed to build the library:

```
* CMake 3.5 or greater
* C++14 compliant compiler (e.g. Clang 3.3+, GCC 4.7+, MSVC 2015+)
```

## Dependencies

The PNT Integrity Library is designed to require as few third party dependencies as possible to support building on a wide variety of platforms. Two dependencies are required in addition to the libraries provided in the package. The [Eigen \(https://eigen.tuxfamily.org\)](https://eigen.tuxfamily.org) C++ template library for linear algebra is required by the base PNT Integrity library. The [FFTW \(http://fftw.org\)](http://fftw.org) package is optionally required and is needed to use the acquisition check. [QT \(https://www.qt.io\)](https://www.qt.io) is required to build and run the user interface

Both packages can be installed following instructions on their respective websites. Eigen is a header-only package and can be installed by downloading a release from the project web site and extracting to a local folder. FFTW binaries are available for a range of platforms from the project web site.

Alternatively, a package manager can be used to install the dependencies. For MacOS the [Homebrew \(https://brew.sh\)](https://brew.sh) package manager is recommended. The [Chocolatey \(https://chocolatey.org\)](https://chocolatey.org) package manager is recommended for Windows. Instructions on installing the required and optional dependencies using package managers on the supported operating systems are provided in the following sections.

### Ubuntu / Debian

Install Eigen by running:

```
sudo apt install libeigen3-dev
```

Optionally install FFTW by running:

```
sudo apt install libfftw3-dev
```

Install Qt5 on Ubuntu by running:

```
sudo apt install qtdeclarative5-dev qtwebengine5-dev libqt5charts5-dev
```

## MacOS

Install Eigen by running:

```
brew install eigen
```

Optionally install FFTW by running:

```
brew install fftw
```

Install Qt5 on MacOS by running:

```
brew install qt
```

If you encounter CMake build errors when finding QT this may help:

```
export CMAKE_PREFIX_PATH=/usr/local/Cellar/qt/[version]/
```

## Windows

Install Eigen by running:

```
choco install eigen
```

Chocolatey does not support provide binaries for FFTW. They can be downloaded and installed directly from the project website: <http://www.fftw.org/install/windows.html>  
(<http://www.fftw.org/install/windows.html>)

For instructions on installing Qt on Windows platform click [here \(https://doc.qt.io/qt-5/windows.html\)](https://doc.qt.io/qt-5/windows.html).

## Building

Extract the release archive:

```
unzip release.zip
```

Generate build files using cmake

```
cd release  
mkdir build  
cmake ../
```

By default, this will generate Unix Makefiles for the package. Project files can be generated for other build systems or IDEs by selecting an alternative [CMake generator \(https://cmake.org/cmake/help/v3.15/manual/cmake-generators.7.html\)](https://cmake.org/cmake/help/v3.15/manual/cmake-generators.7.html).

Build the libraries by running:

```
make
```

The libraries can be optionally installed to the user's system by running:

```
make install
```