photon-tools - Tools for analysis of single-photon measurement data

`photon-tools` is a collection of tools for the manipulation and analysis of photon timestamp data, particularly from FRET and FCS experiments.

Installation: the two-minute version

To install `photon-tools` on Ubuntu,

```
$ sudo apt-get install python python-numpy python-scipy python-matplotlib \
  build-essential cython libboost-all-dev
$ git clone git://github.com/bgamari/photon-tools.git
$ cd photon-tools
$ ./install.sh
```

Installation: the unabridged version

Many of these utilities are written in Python and generally require Python 2.6 or greater along with numpy. In particular, some optimized modules require Cython. Utilities capable of producing plots require the matplotlib python plotting library. On the whole, `photon-tools` depends on,

- Gnu make
- Python >= 2.6
- Numpy
- Scipy
- Matplotlib >= 1.2 (due to issue #1246)
- Cython >= 0.15
- Boost

The scripts and libraries included in `photon-tools` can be installed like any Python `distutils` package,

```
$ sudo ./setup.py install
```

Note that running scripts within the `photon-tools/` root directory will require that the Cython code is built in-place, due to limitations of Python's module name resolution scheme. To do this, one must run,

```
$ ./setup.py build_ext --inplace
```

Supported formats

Utilities requiring timestamp data as input accept data in the following formats,

- Raw 64-bit integer timestamps (read as little endian)
- Picoquant PT2
- Goldner FPGA timetagger `.timetag` files

In all of these cases, the utilities will attempt to figure out the period of the timebase (known as the jiffy) from whatever metadata is available in the format.

Tools

The tools that `photon-tools` provides are command-line utilities following typical UNIX argument conventions. That is, most arguments are delimited by a dash and have both a long form (`--output`) and a short form (`-o`).

Tools

The tools that `photon-tools` provides are command-line utilities following typical UNIX argument conventions. That is, most arguments are delimited by a dash and have both a long form (`--output`) and a short form (`-o`).

Below is a set of simple examples describing basic usage of the tools. These are, however, only basic examples and do not show all of the features of these tools. Full help for each utility is always available with from `--help`.