

TOWARD OFFICIAL DUNE DAQ SOFTWARE DOCUMENTATION

JOHN FREEMAN
MAR-29-2021

CURRENT SITUATION

- For the last year, the de-facto homebase of the DUNE DAQ software documentation has been the appfwk GitHub Wiki
- Has provided versioned links to Wiki pages covering daq-buildtools ("Compiling and running") and daq-cmake ("Creating a new package") which have been regularly maintained by the Software Coordination Group
- Also contains links to numerous pages which became obsolete months ago and sometimes weren't even completed
- Documentation for other packages found in their READMEs and/or GitHub wikis
 - Not always versioned or kept up to date, and the purpose of the information has varied (notes for developers, meeting minutes, etc.)
- In this talk I'll describe the new, official home of the DUNE DAQ software documentation and how we can fill it out



Search or jump to...

Pull requests Issues Marketplace Explore



DUNE-DAQ / appfwk

Watch 16 Star 2 Fork 7

Code Issues 7 Pull requests 1 Actions Projects Wiki Security Insights Settings

Home

Edit New Page

jcfreeman2 edited this page yesterday · 22 revisions

JCF, Mar-24-2021: documentation for DUNE DAQ release dunedaq-v2.4.0 can be found [here](#). Only use this wiki for dunedaq-v2.3.0 and earlier

Welcome to the Dune DAQ wiki!

This area can be used as a collaborative working area for documentation that has not yet been finalized. Once the task force has agreed to adopt a document, it should be placed in the repository, and the links here should be updated accordingly.

This wiki lives in the **appfwk** but it supposed to serve as a guideline for all the project.

Links below which contain "under vX.Y.Z" deal strictly with packages as they were versioned "vX.Y.Z", X, Y and Z all integers. Without an "under vX.Y.Z", it should be assumed that the instructions cover the behavior of the latest commits to software packages in question.

Developers are encouraged to follow the instructions for the most recent frozen releases (dunedaq-v2.3.0 as of Mar-2-2021)

User Documentation

- [Compiling and running under v2.3.0](#)
- [Compiling and running under v2.2.0](#)
- [Compiling and running under v2.1.0](#)
- [Compiling and running under v2.0.0](#)
- [Compiling and running \(unreleased\)](#)
- [Compiling and running \(older versions\)](#)

Pages 43

Find a Page...

[Home](#)

[App Fwk v1 pre release checklist](#)

[Coding style](#)

[Compiling and running](#)

[Compiling and running old](#)

[Compiling and running under v1.0.0](#)

[Compiling and running under v1.1.0](#)

[Compiling and running under v1.2.0](#)

[Compiling and running under v1.2.1](#)

[Compiling and running under v2.0.0](#)

[Compiling and running under v2.1.0](#)

[Compiling and running under v2.2.0](#)

[Compiling and running under v2.3.0](#)

[Compiling and running under v2.4.0](#)

[Contributors Guide](#)

Show 28 more pages...

HTTPS://DUNE-DAQ-SW.READTHEDOCS.IO/EN/LATEST/

☰ DUNE DAQ Software Documentation Home 🔍 Search

DUNE DAQ Software Documentation Home

There are four DUNE DAQ software packages which are used to aid developers in creating packages designed for the DAQ itself:

To learn how to build an existing software package, read the [daq-buildtools documentation](#)

To learn how to create a new package or modify the build of an existing one, read the [daq-cmake documentation](#)

To learn about the standard development workflow, read the [daq-release documentation](#), in particular [this page](#)

To learn about the C++ coding guidelines which DUNE DAQ package developers should follow, read [the styleguide](#)

For the other packages, please click on one of the links below. To learn how to edit a package's documentation, click [here](#)

[appfwk](#) home of daq_application and tools for writing DAQModules

[cmdlib](#) interfaces for commanded objects ***Etc. for the rest of the packages***

- At the top, links to documentation for packages designed to help develop DAQ-focused packages
- Then alphabetical list of DAQ-focused packages
- Goal is that users can click on any package's link and get the info they need

CONCERNING THE TECHNOLOGY

- We thought about a couple of different technologies, and went with ReadTheDocs to display our documentation, using MkDocs to generate the html from Markdown
- You can use Markdown in your GitHub repo's page to edit your documentation (known syntax, ability to preview documentation)
- My favorite quotes from readthedocs.org:
 - "We will host your documentation for free, forever."
 - "We can host and build multiple versions of your docs."

THE DOCS REPO

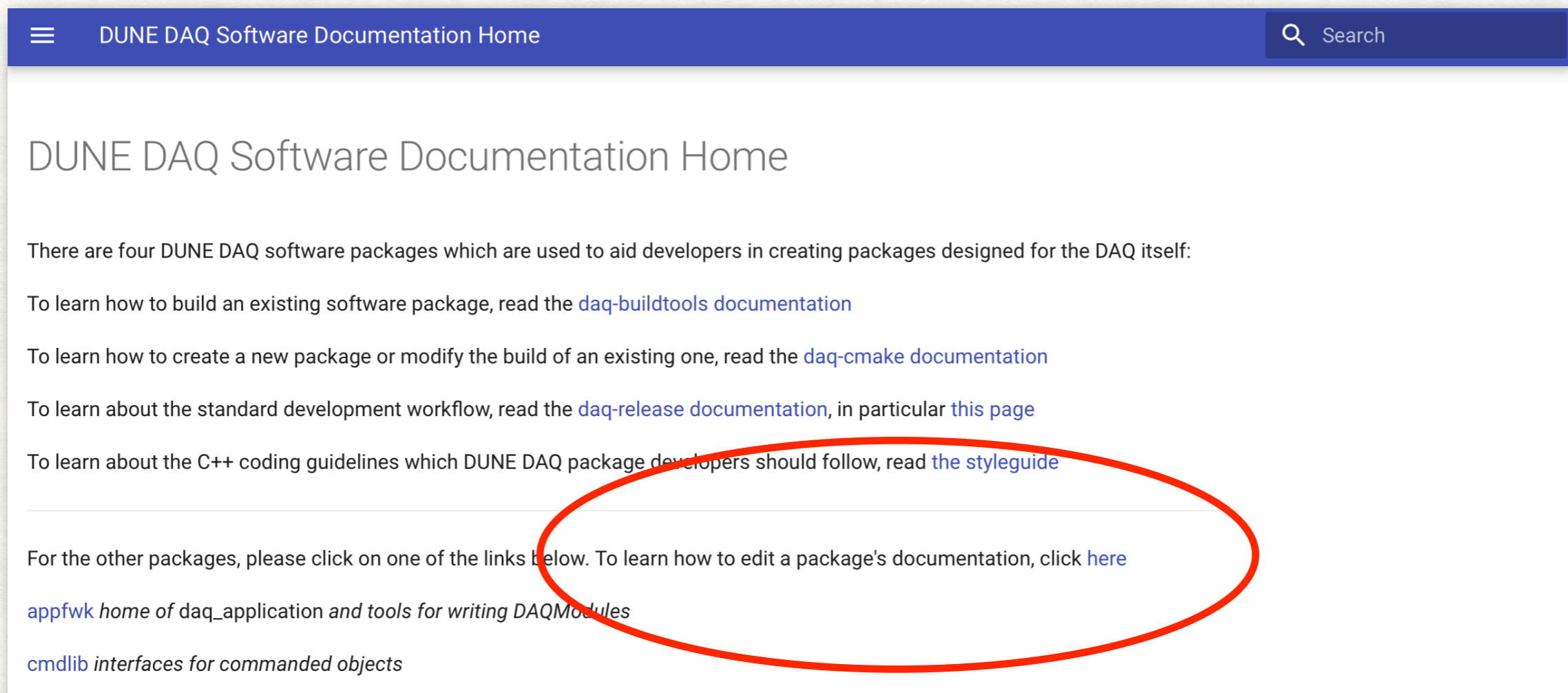
- A couple of weeks ago, we created a new repo called "docs"
- Whenever a change gets pushed to docs, ReadTheDocs can (via MkDocs) re-generate the documentation
- We run a script which grabs the documentation from each repo and sticks it in docs → this is how your edits become the per-package documentation on the official website
- Can be versioned. Technically, the website is currently the head of docs' develop branch, but we can also have versions in the future (e.g., <https://dune-daq-sw.readthedocs.io/en/dunedaq-v2.4.1>, <https://dune-daq-sw.readthedocs.io/en/dunedaq-v2.6.0>, etc.)

THIS IS WHERE YOU COME IN

- The script expects that:
 - Your repo has a subdirectory called `docs/` (not to be confused with the repo named docs)
 - Its homepage is in `docs/README.md` → will display in GitHub
 - Other pages are Markdown files in `docs/`; they can be linked to from `docs/README.md`
- **Step 1 (timescale one week): move any existing README.md into your (new?) docs/ directory and copy your GitHub Wiki pages into it (Issues will be filed)**
- Can see, e.g., the `johnfreeman/dont-readme` branch of `minidaqapp` for an example of Step 1
- Can perform edits of `docs/*.md` files on your repo's GitHub page and hit "Preview" to check for syntax errors

MORE DETAILS ON HOW TO DOCUMENT

- From <https://dune-daq-sw.readthedocs.io/en/latest/>, click here:



☰ DUNE DAQ Software Documentation Home 🔍 Search

DUNE DAQ Software Documentation Home

There are four DUNE DAQ software packages which are used to aid developers in creating packages designed for the DAQ itself:

To learn how to build an existing software package, read the [daq-buildtools documentation](#)

To learn how to create a new package or modify the build of an existing one, read the [daq-cmake documentation](#)

To learn about the standard development workflow, read the [daq-release documentation](#), in particular [this page](#)

To learn about the C++ coding guidelines which DUNE DAQ package developers should follow, read [the styleguide](#)

For the other packages, please click on one of the links below. To learn how to edit a package's documentation, click [here](#)

[appfwk](#) *home of daq_application and tools for writing DAQModules*

[cmdlib](#) *interfaces for commanded objects*

THIS IS WHERE YOU COME IN, PT. 2

- Step 2 (timescale a couple of weeks, for dunedaq-v2.4.1 release): provide the first official version of your package's documentation for users
- Things to think about:
 - "Have I minimized the number of emails I'll receive from users with questions about this package?"
 - "Is this existing page more appropriate for the official documentation (user-facing), or for the GitHub wiki (developer-facing)?"
 - "Do I want this link on my page to still point to that package's GitHub page, or should it now point to its official page?"
- ...

CONCLUSIONS

- For each package, for each version, we want a reliable source of information for its users
- For those writing their package's documentation, steps 1 and 2 await. This talk, https://dune-daq-sw.readthedocs.io/en/latest/editing_package_documentation/, and myself will all be resources which can help.