FeenoX documentation index

2024-06-14

Contents

Every bit of FeenoX documentation is written in Pandoc-flavored Markdown in the `doc` directory of the FeenoX repository. It is then converted to HTML, PDF, Unix manpage or Texinfo as needed. FeenoX documentation is released under the terms of the GNU Free Documentation License v1.3, or any later version.

1 Manual and description

As per the GNU Coding Standards, "a manual should serve both as tutorial and reference." Due to the formatting restrictions, the Texinfo version contains only the description and not the full reference.

- FeenoX manual [PDF]
- FeenoX description [PDF] [Texinfo]

1.1 Design

1. The FeenoX project starts as an offer to an imaginary "request for quotations" that defines software requirements specifications for an open source computational tool.
   - Software Requirements Specification (Fictitious RFQ) [PDF]
2. Then a fictitious "offer" to the above tender is given in a software design specifications document that explains the design decisions and features included in FeenoX.
   - Software Design Specification (Imaginary FeenoX's offer) [PDF]

1.2 Other documents

- Unix manpage (accessible with `man feenox` after (global) installation with `sudo make install`)
- Frequently Asked Questions PDF
- Compilation Guide PDF
- Programming Guide PDF
- History PDF

1.3 Youtube videos

- FeenoX hands-on tutorial
- FeenoX, a cloud-first free and open source finite-element(ish) tool
- Why FeenoX works the way it works
- Verification of PDE solvers using the Method of Manufactured Solutions
- FeenoX: a free and open source computational tool for solving (nuclear-related) DAEs in the cloud
- CAEplex: finite elements in the cloud
- Thermal finite elements in the cloud in 1 minute
- Add CAE to your Onshape CAD with CAEplex
- First 10 natural modes of a cylinder in less than 1 min. with CAEplex
- Democratizing CAE

1.4 Markdown sources

The Markdown sources are in the `doc`:

- `feenox-manual`

- `feenox-desc`
- `srs`
- `sds`
- `FAQ`
- `feenox.1`
- `history`
- `compilation`
- `programming`

They are compiled by running the `make.sh` script.

2 Hands on

Go directly to the point and see how to solve problems with FeenoX. Everything needed (input files, geometry files, meshes, data, scripts, etc.) is included in the FeenoX repository.

2.1 Examples

Annotated examples can be found in the examples directory of the Github repository.

- Online annotated examples
  - Basic mathematics
  - Systems of ODEs/DAEs
  - Laplace's equation
  - Heat conduction
  - Linear elasticity
  - Modal analysis
  - Neutron diffusion
  - Neutron SN
- Github examples directory

2.2 Tutorials

1. Setting up your workspace

2.2.1 General tutorials

1. Overview: the tensile test case
2. Fun & games: solving mazes with PDES instead of AI

2.2.2 Detailed functionality

1. Input files, expressions and command-line arguments
2. Static & transient cases
3. Functions & functionals
4. Vectors & matrices
5. Differential-algebraic equations
6. Meshes & distributions

2.2.3 Physics tutorials

1. The Laplace equation
2. Heat conduction
3. Linear elasticity
4. Modal analysis
5. Thermo-mechanical analysis
6. Neutron diffusion
7. Neutron transport

2.3 Tests

The `tests` directory in the repository has hundreds of

- `grep`-able examples,
- unit tests,
- regression tests, and/or
- (crude) mathematical verification tests.

3 Contributing

3.1 Asking questions & reporting bugs

- Use Github discussions to ask for help,
- Use the Github issue tracker to report bugs.

3.2 Contributing guidelines

Contributions from hackers and/or academics are welcome, especially new types of PDEs and new formulations of existing PDEs. For elliptic operators feel free to use the Laplace equation at `src/pdes/laplace` as a template.

1. Read the Programming Guide.
2. Browse Github discussions and open a new thread explaining what you want to do and/or asking for help.
3. Fork the Git repository under your Github account
4. Create a pull request, including
   - code,
   - documentation, and
   - tests.
5. Follow up the review procedure.

Note that

- It is mandatory to observe the Code of Conduct.
- The contributed code has to be compatible with the GPLv3+ license.
- Each author keeps the copyright of the contribution.
- You can ask!

3.3 Adding a new PDE to `src/pdes`

To be explained.