

FeenoX

A cloud-first free no-fee no-X uniX-like finite-element(ish)
computational engineering tool

Contents

About FeenoX	1
News	2
Extents	2
Examples	2
Tutorials	2
Tests	2
Capabilities	3
Usefulness	4
Documentation	4
Quickstart	4
Download	4
Git repository	4
Licensing	5
Contributing	5
Further information	5
JOSS	10.21105/joss.05846

About FeenoX

FeenoX is a cloud-first free and open source tool to solve engineering-related problems using a computer (or many computers in parallel) with a particular design basis.

Choose your background for further details about the what, how and whys:

- Industry Engineer
- Unix Hacker

- Academic Professor

See the following paper for a brief summary (and to cite FeenoX in your work):

- Theler J, “FeenoX: a cloud-first finite-element(ish) computational engineering tool,” *Journal of Open Source Software*, vol. 9, no. 95, p. 5846, Mar. 2024, doi: 10.21105/joss.05846.

News

FeenoX is now in Debian! You can install FeenoX from APT

```
sudo apt install feenox
```

For now it is in Debian sid only, but eventually it will be part of Debian stable. And probably it will be available in Ubuntu as well. Follow the status at

<https://tracker.debian.org/pkg/feenox>

Open-source web-based UX for FeenoX. Take a look at SunCAE for an example of how to write a front end for FeenoX.

IB students Remember that your first read should be the 0-th tutorial, Setting up your workspace

Extents

Examples

`doc/examples-list.md`

Tutorials

`doc/tutorials-list.md`

Tests

The `tests` directory in the repository has hundreds of

- grep-able examples
- unit and regression tests,
- (crude) mathematical & code verification tests (as in “are we solving right the equations?”),
- subdirectories with further case studies
 - `mms`
 - * `thermal`
 - `2d`
 - `3d`
 - `nafems`
 - * `le10`

Capabilities

- Both *free* as in “free speech” and in “free beer”
- The problem to solve is defined through a syntactically-sugared self-descriptive English-like plain-text input file that should resemble the original human-friendly problem formulation as much as possible:
 - no need to recompile if the problem changes (FeenoX is a *program*, not a library)
 - nouns are definitions and verbs are instructions
 - simple problems need simple inputs
 - everything is an expression
 - 100%-defined user output (no PRINT nor WRITE_RESULTS instructions, no output)
- Cloud-first design (cloud friendliness is not enough)
- MPI parallelization
- Leverages high-quality well-established free and open source libraries to solve...
 - general mathematical problems using GNU GSL
 - sets of ODEs/DAEs using SUNDIALS
 - PDEs formulated with the finite element method
 - * reads mesh in Gmsh format
 - * uses PETSc/SLEPc to solve
 - linear systems (KSP)
 - non-linear systems (SNES)
 - time-dependent systems (TS)
 - generalized eigen-value problems (EPS)
 - * writes results in either Gmsh or VTK (Paraview) format
- Focuses on flexibility, especially when defining non-uniform multi-solid material properties from ASME tables
- Follows the Unix programming philosophy
 - rule of separation
 - rule of silence
 - rule of economy
- Each PDE (i.e. from Laplace downward in the list of examples) is implemented in a subdirectory within `src/pde` of the source tree
 - any subdirectory can be removed if a particular PDE is not needed
 - any subdirectory can be used as a template to add a new PDE to the capabilities
- Space, time and/or solution-dependent material properties and boundary conditions
 - algebraic expressions, and/or
 - point-wise interpolated data
- Command-line argument expansion for
 - increased flexibility,
 - parametric sweeps, and/or
 - optimization loops

- Steady-state, [quasi-static] and/or transient problems
- Linear and non-linear problems
- Possibility to verify the code using the Method of Manufactured Solutions
- Separate repository to profile and study code performance using Google's benchmark library

Usefulness

- CAEplex: a web-based thermo-mechanical solver running on the cloud
- Non-conformal mesh mapping
- ASME stress linearization for pressurized pipes and vessels
- Assessment of material properties from tabulated sources
- Environmentally-assisted fatigue analysis in dissimilar interfaces of nuclear pipes
- Neutron transport in the cloud
- Solving mazes without AI
- Parametric NAFEMS LE10 benchmark: comparison of resource consumption for different FEA programs
- Some Youtube videos

Documentation

Browse through the documentation index and/or the `doc` subdirectory of the Github repository for

- FAQs, including what FeenoX means
- Manual
- Description
- Software Design Requirements
- Software Design Specification
- Unix man page (accessible through `man feenox` after installation)
- History
- Compilation guide
- Programming guide

Quickstart

Download

`doc/downloads.md`

Git repository

`doc/git.md`

See the download page and the compilation guide for detailed information.

Licensing

FeenoX is distributed under the terms of the GNU General Public License version 3 or (at your option) any later version. The following text was borrowed from the Gmsh documentation. Replacing “Gmsh” with “FeenoX” (using Unix’s `sed`) gives:

```
doc/freesw.md  
doc/licensing.md
```

Contributing

```
doc/contributing-guidelines.md
```

Further information

Home page: <https://www.seamplex.com/feenox>
Repository: <https://github.com/seamplex/feenox>
Bug reporting: <https://github.com/seamplex/feenox/issues>
Discussions: <https://github.com/seamplex/feenox/discussions>
Follow us: YouTube LinkedIn Github

FeenoX is licensed under GNU GPL version 3 or (at your option) any later version.

FeenoX is free software: you are free to change and redistribute it.
There is **NO WARRANTY**, to the extent permitted by law.