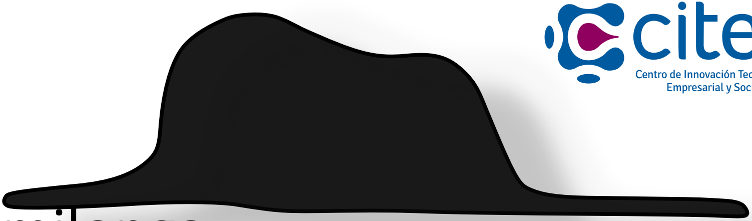
Workshop: Hands on milonga

Germán Theler

cites
Centro de Innovación Tecnológica
Empresarial y Social



milonga a free nuclear reactor core analysis code

II Reunión Grupo Argentino de Cálculo de Reactores
November 25th 2015
Buenos Aires, Argentina

4913f17b378d:24:2015-11-25

Milonga is...

- a core-level neutronic code

Milonga is...

- a core-level neutronic code
- work in progress, not yet usable for production

Milonga is...

- a core-level neutronic code
- work in progress, not yet usable for production
- designed over sound rationales (if not, please tell me why & where!), with a rather different design basis (with respect to usual nuclear codes)

Milonga is...

- a core-level neutronic code
- work in progress, not yet usable for production
- designed over sound rationales (if not, please tell me why & where!), with a rather different design basis (with respect to usual nuclear codes)
- free & open released under the GNU General Public License v3
  - follows (or tries to follow) UNIX rules
  - make extensive use of free & open libraries

Milonga is...

- a core-level neutronic code
- work in progress, not yet usable for production
- designed over sound rationales (if not, please tell me why & where!), with a rather different design basis (with respect to usual nuclear codes)
- free & open released under the GNU General Public License v3
  - follows (or tries to follow) UNIX rules
  - make extensive use of free & open libraries
- a *glue layer* between a mesh generator and an eigenvalue solver

Milonga is...

- a core-level neutronic code
- work in progress, not yet usable for production
- designed over sound rationales (if not, please tell me why & where!), with a rather different design basis (with respect to usual nuclear codes)
- free & open released under the GNU General Public License v3
    - follows (or tries to follow) UNIX rules
    - make extensive use of free & open libraries
- a *glue layer* between a mesh generator and an eigenvalue solver
- spelled in lowercase unless it starts a sentence

Milonga is...

- a core-level neutronic code
- work in progress, not yet usable for production
- designed over sound rationales (if not, please tell me why & where!), with a rather different design basis (with respect to usual nuclear codes)
- free & open released under the GNU General Public License v3
  - follows (or tries to follow) UNIX rules
  - make extensive use of free & open libraries
- a *glue layer* between a mesh generator and an eigenvalue solver
- spelled in lowercase unless it starts a sentence
- hosted on Bitbucket at `https://bitbucket.org/gtheler/milonga`

Milonga is...

- a core-level neutronic code
- work in progress, not yet usable for production
- designed over sound rationales (if not, please tell me why & where!), with a rather different design basis (with respect to usual nuclear codes)
- free & open released under the GNU General Public License v3
  - follows (or tries to follow) UNIX rules
  - make extensive use of free & open libraries
- a *glue layer* between a mesh generator and an eigenvalue solver
- spelled in lowercase unless it starts a sentence
- hosted on Bitbucket at `https://bitbucket.org/gtheler/milonga`

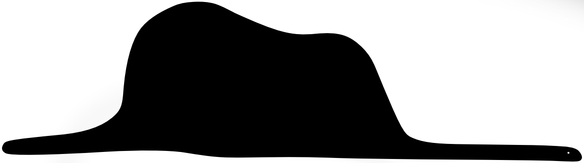
A piece of advise

It is really worth any amount of time and effort to get away from Windows if you are doing computational science.

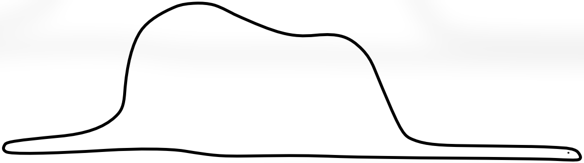
Matthew Knepley, Jul 22 2015, PETSc Users mailing list

The logo

- pure black



- transparent with black outline



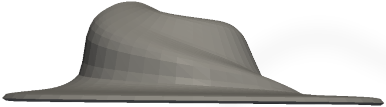
- try to use vector formats, convert to PNG if needed but never ever to JPG!
- ask for the SVGs if you need them (they are in the repository though)

Hat or snake?

- depends whether you are a kid or a grown-up

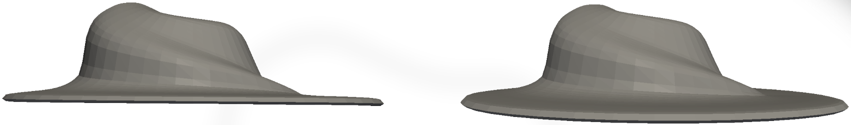
Hat or snake?

- depends whether you are a kid or a grown-up
- depends on the topology (i.e. 2D or 3D)



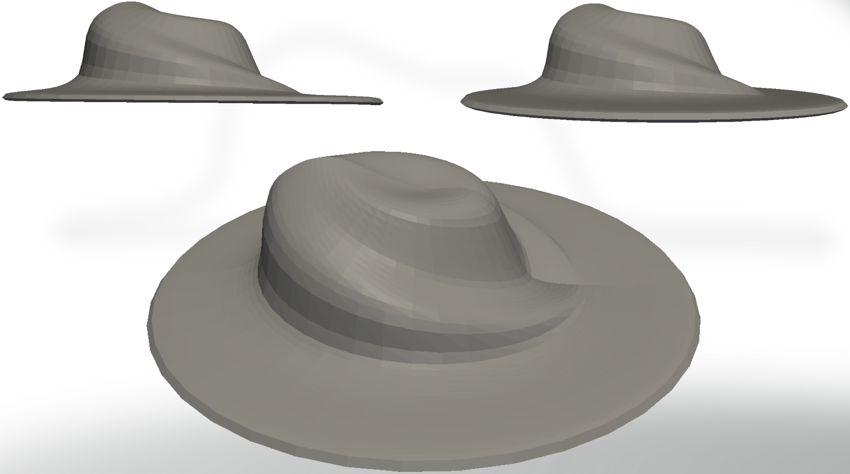
Hat or snake?

- depends whether you are a kid or a grown-up
- depends on the topology (i.e. 2D or 3D)

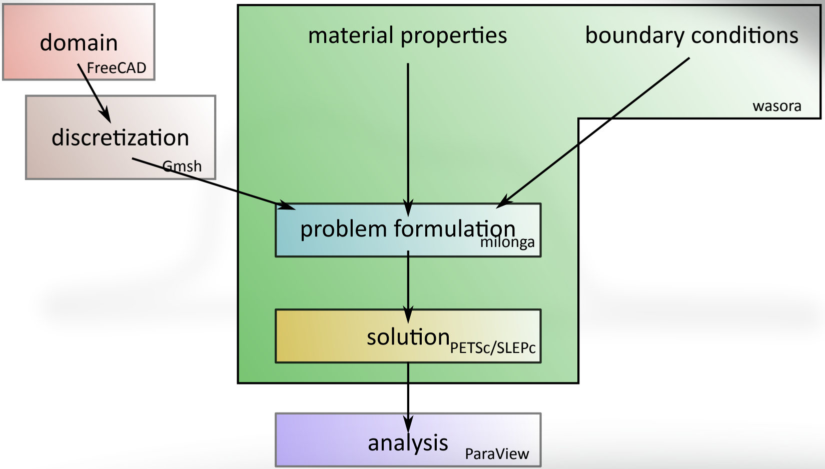


Hat or snake?

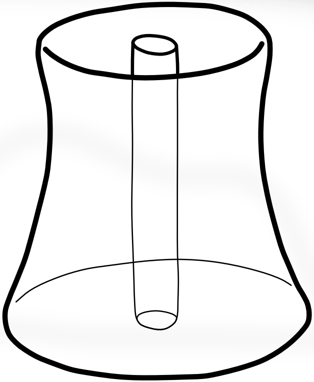
- depends whether you are a kid or a grown-up
- depends on the topology (i.e. 2D or 3D)



Solving PDEs



Wasora



wasora’s an advanced suite for optimization & reactor analysis

`http://www.talador.com.ar/jeremy/wasora/`

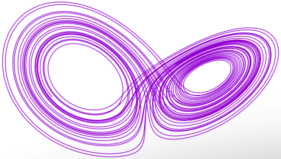
`https://bitbucket.org/gtheler/wasora/`

Wasora

Wasora should be seen as a *syntactically-sweetened* way to ask a computer to perform a certain mathematical calculation:

The Lorenz system:

$$\dot{x} = \sigma\left(y - x\right)$$
$$\dot{y} = x\left(r - z\right) - y$$
$$\dot{z} = xy - bz$$



```
# lorenz' seminal dynamical system solved with wasora
PHASE_SPACE x y z
end_time = 40

# parameters that lead to chaos
sigma = 10
r = 28
b = 8/3

# initial conditions
x_0 = -11
y_0 = -16
z_0 = 22.5

# the dynamical system (note the dots before the '=' sign)
x_dot .= sigma*(y - x)
y_dot .= x*(r - z) - y
z_dot .= x*y - b*z

# write the solution to the standard output
PRINT t x y z
```

Features (mostly a high-level interface for GSL)

- evaluation of algebraic expressions
- one and multi-dimensional function interpolation
- scalars, vectors and matrices operations
- numerical integration, differentiation and root finding of functions
- possibility to solve iterative and/or time-dependent problems
- adaptive integration of systems of differential-algebraic equations
- I/O from files and shared-memory objects (with optional synchronization using semaphores)
- execution of arbitrary code provided as shared object files
- parametric runs using quasi-random sequence numbers to efficiently sweep a sub-space of parameter space
- non-linear fit of scattered data to one or multi-dimensional functions
- non-linear multidimensional optimization

Extending wasora with plugins

Should a particular calculation be needed, wasora's features may be extended by the implementation of dynamically-loaded plugins, for example:

- Free (GPLv3)

besssugo	builds scientific videos out of wasora computations
fino	solves partial differential equations using the finite element method
milonga	core-level multigroup neutronic code
waspy	runs Python code within wasora sharing variables, vectors and matrices
xdfrrpf	eXtracts Data From RELAP Restart-Plot Files

- Private (not for distribution)

dynetz	runs DYNETZ-like Fortran codes
pcex	re-implementation of neutronic code PCE

Regla 70-20-10

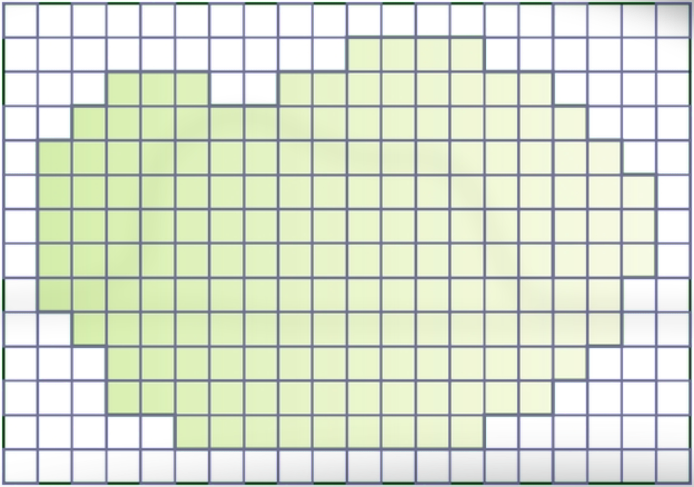
http://www.talador.com.ar/jeremy/wasora/realbook/

Milonga’s problems (design basis)

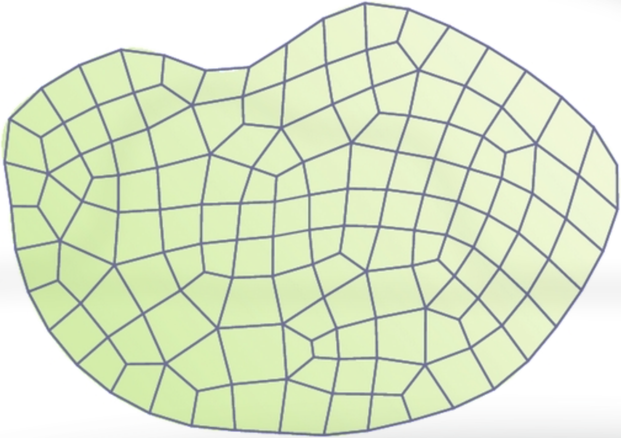
- Steady-state multigroup core-level neutronic problems
  - academic cases
    - problems with analytical solution, i.e. bare homogeneous geometries with one neutron energy group
    - problems without analytical solution but simple enough to illustrate the physics, i.e one-dimensional reflected semi-homogeneous slabs with two energy groups
  - benchmark tests
    - two and three-dimensional few-group problems with different materials (each one with uniform cross sections) and mixed boundary conditions
    - sensitivity studies using different meshes and numerical schemes
  - industrial problems
    - full three-dimensional reflected geometry with an arbitrary number of energy groups using homogenized macroscopic cross sections that depend on the distribution of other properties (temperatures and densities, boron, xenon, control rods, etc.)
    - fuel management optimization
    - coupled transient operational and safety calculations

Unstructured grids

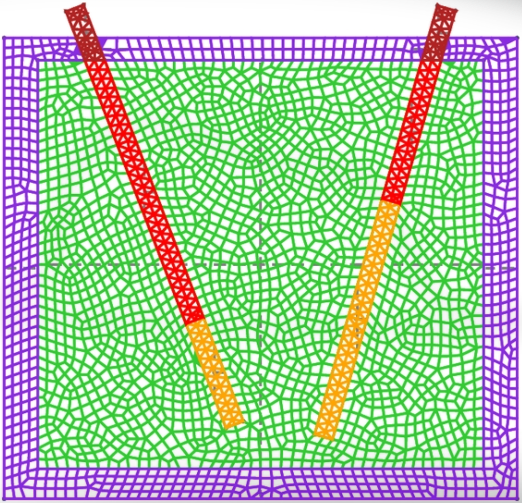
Unstructured grids



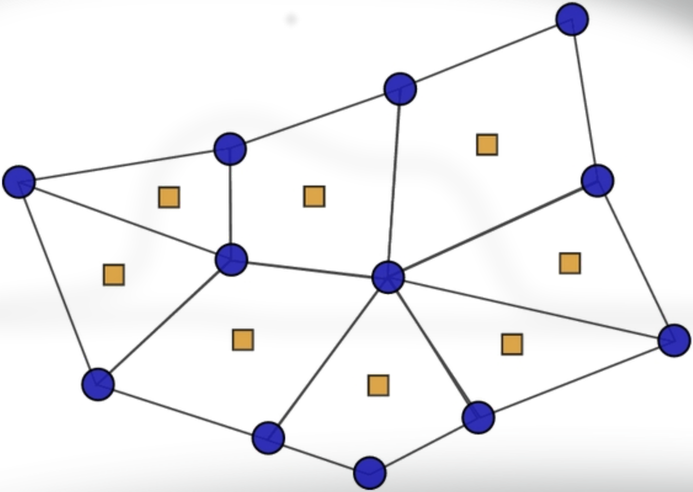
Unstructured grids



Unstructured grids



FVM vs. FEM



Hands on milonga

Me lo contaron y lo olvidé; lo vi y lo entendí; lo hice y lo aprendí.

```
$ milonga -v
milonga 0.4.18  (deab1b33a29a 2015-11-24 22:22 -0300)
free nuclear reactor core analysis code

 rev hash deab1b33a29ad05da8a015b631dd1dcdc72225ad
 last commit on 2015-11-24 22:22 -0300 (rev 239)
 compiled on 2015-11-24 22:46:06 by gtheler@ralph (linux-gnu x86_64)
 with gcc (Debian 4.9.2-10) 4.9.2 using -O2 linked against
  SLEPc Release Version 3.6.2, Nov 03, 2015
  Petsc Release Version 3.6.2, Oct, 02, 2015  arch-linux2-c-opt
 running on Linux 3.16.0-4-amd64 #1 SMP Debian 3.16.7-ckt11-1+deb8u3 (2015-08-04) x86_64
 4  Intel(R) Core(TM) i5-3317U CPU @ 1.70GHz


 milonga is copyright (c) 2010-2015 jeremy theler
 licensed under GNU GPL version 3 or later.
 milonga is free software: you are free to change and redistribute it.
 There is NO WARRANTY, to the extent permitted by law.


-----------          ---------         ---------      -----    ----
wasora 0.4.20  (8c90e2cdbbc4 2015-11-18 08:25 -0300)
wasora’s an advanced suite for optimization & reactor analysis

 rev hash 8c90e2cdbbc42979082d7a2025539d5da297440e
[...]
```

Hands on milonga

- one dimension
  - uniform XS to illustrate differences between diffusion & S_N using FVM & FEM
  - reflected two-group slab to plot the thermal shoulder
  - two-zone slab solved using both structured and unstructured grids parametrically
  - slab with non-uniform XS, perhaps xenon feedback?
  - …
- two dimensions
  - the fish problem
  - square core with circular reflector
  - 2D PWR IAEA Benchmark
  - …
- three dimensions
  - Stanford Bunny
  - 3D PWR IAEA Benchmark
  - …

Question

What information would you like to get in the debug markdown output (`MILONGA_DEBUG`)?

Future works (help appreciated)

- Evangelization
  - Use it
  - Give feedback and report bugs
  - Spread the word (especially with your students!)
  - …
- Documentation
  - Description
  - Reference
  - Real Book (i.e. examples)
  - Benchmarks
  - …
- Development
  - Higher harmonics (very easy!)
  - Compute adjoint flux (easy!)
  - Fix memory leaks (hard & tedious)
  - Add new formulations (collision probabilities? P_N?)
  - Add new discretizations (FEM-like discretization in Ω?)
  - Solve non-linear (over the eigenvector) eigenproblems
  - Uncertainty analysis
  - Use PETSc's DMPLEX
  - Implement a parallelization scheme
  - Python wrapper
  - Solve transient problems (moving meshes? multipoint kinetics?)
  - …