COMPUTATIONAL INFRASTRUCTURE for GEODYNAMICS

Day 1 Instructions:
https://bit.ly/ASPECT-day1

A Tutorial on ASPECT

Wolfgang Bangerth, Juliane Dannberg,
Menno Fraters, Rene Gassmoeller, Anne Glerum,
Timo Heister, Lorraine Hwang, John Naliboff

With material from many other ASPECT contributors

What is ASPECT?

COMPUTATIONAL INFRASTRUCTURE for GEODYNAMICS

ASPECT: The "Advanced Solver for Problems in Earth's ConvecTion"

- Originally intended to simulate *mantle convection*:
  - Simulation of viscous fluid flow

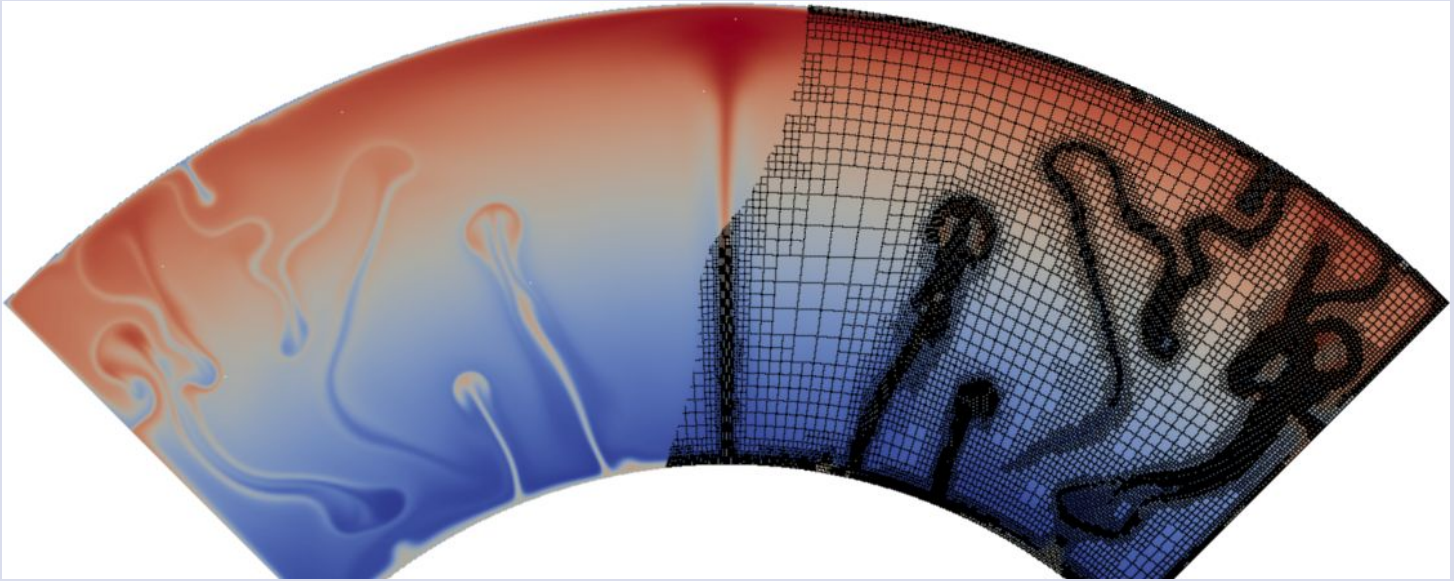
- Now also a code for *tectonics modeling*:
  - Also elastoviscoplastic models
  - Also melting, free surfaces, ...

What is ASPECT?

COMPUTATIONAL INFRASTRUCTURE for GEODYNAMICS

ASPECT: The “Advanced Solver for Problems in Earth’s ConvecTion”

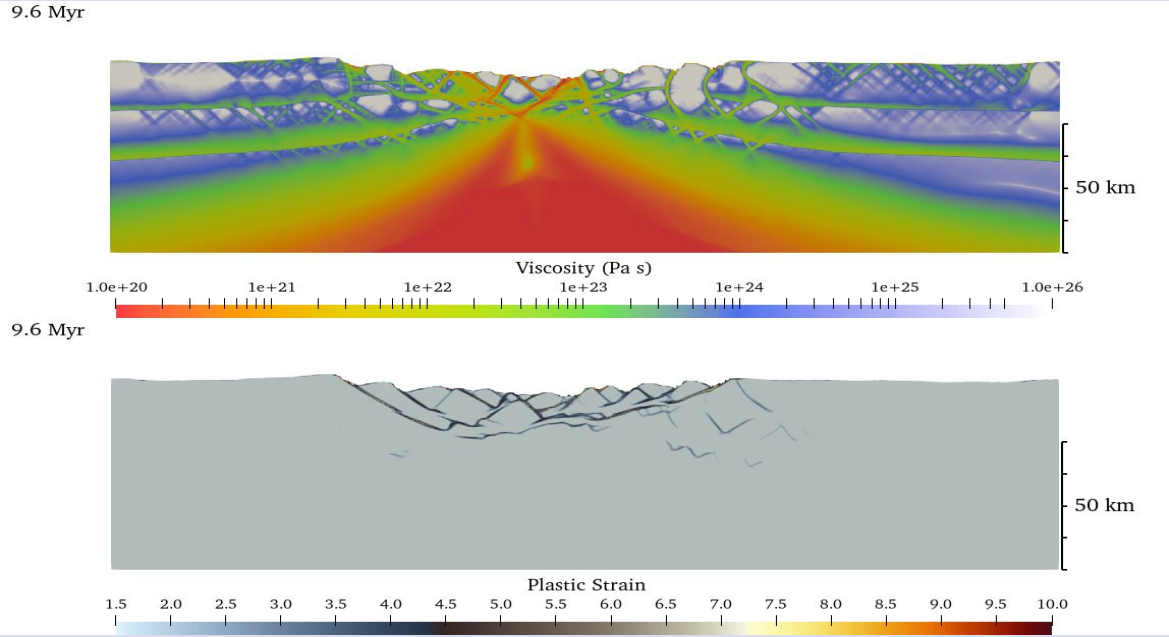
Examples:



What is ASPECT?

ASPECT: The “Advanced Solver for Problems in Earth’s ConvecTion”

Examples:

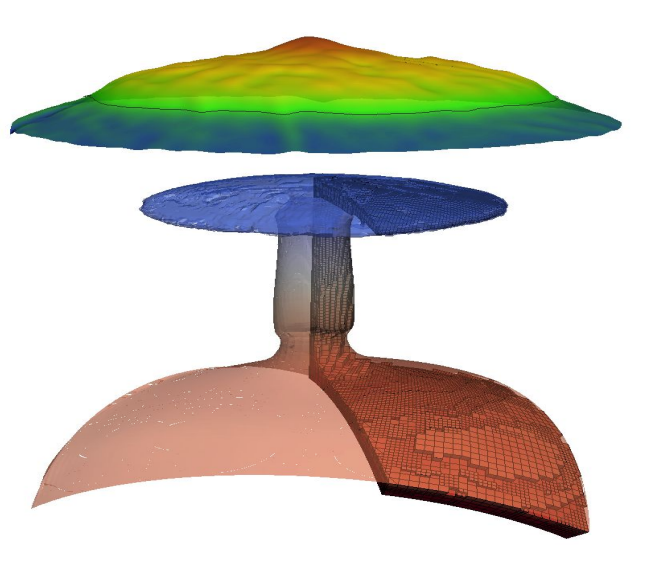


What is ASPECT?

COMPUTATIONAL INFRASTRUCTURE for GEODYNAMICS

ASPECT: The “Advanced Solver for Problems in Earth’s ConvecTion”

Examples:



ASPECT: Methods

CIG COMPUTATIONAL INFRASTRUCTURE for GEODYNAMICS

- **Modern numerical methods:**
  adaptive mesh refinement, linear and nonlinear solvers, higher-order discretizations, stabilization schemes
- **Usability and extensibility:**
  manual: 570+ pages, 50+ cookbooks and examples, plugin architecture
- **Parallel scalability**
- **Building on others' work:**
  tested foundation, smaller codebase, automatic improvements

ASPECT: Community

- **ASPECT was always intended as a community code**
  GPL, developed in the open
  Encourage contributions, be welcoming
- **Start:** 2011
  Timo Heister
  Wolfgang Bangerth

ASPECT: Community

- **ASPECT was always intended as a community code**
  GPL, developed in the open
  Encourage contributions, be welcoming
- **Later:**
  + Juliane Dannberg
  + Rene Gassmoeller

ASPECT: Community

- **ASPECT was always intended as a community code**
  GPL, developed in the open
  Encourage contributions, be welcoming
- **Now:**
  8 principal developers
  60+ contributors over the years
  Development during “hackathons”

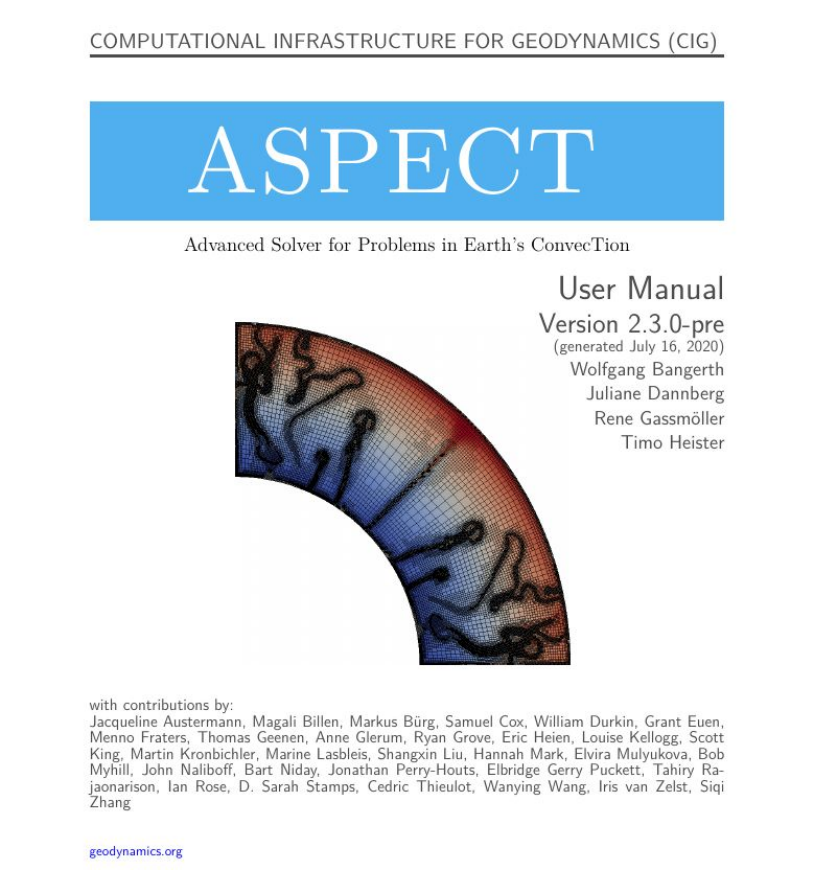
Resources

Website and manual:

https://aspect.geodynamics.org

Publications:

- Kronbichler, Heister, Bangerth: *"High Accuracy Mantle Convection Simulation through Modern Numerical Methods"* Geophysical Journal International, 2012.
- Heister, Dannberg, Gassmoeller, Bangerth: *"High Accuracy Mantle Convection Simulation through Modern Numerical Methods. II: Realistic Models and Problems"* Geophysical Journal International, 2017.
- + 70 using ASPECT



COMPUTATIONAL INFRASTRUCTURE FOR GEODYNAMICS (CIG)

ASPECT

Advanced Solver for Problems in Earth's ConvecTion

User Manual
Version 2.3.0-pre
(generated July 16, 2020)
Wolfgang Bangerth
Juliane Dannberg
Rene Gassmöller
Timo Heister

with contributions by:
Jacqueline Austermann, Magali Billen, Markus Bürg, Samuel Cox, William Durkin, Grant Euen, Menno Fraters, Thomas Geenen, Anne Glerum, Ryan Grove, Eric Heien, Louise Kellogg, Scott King, Martin Kronbichler, Marine Lasbleis, Shangxin Liu, Hannah Mark, Elvira Mulyukova, Bob Myhill, John Naliboff, Bart Niday, Jonathan Perry-Houts, Elbridge Gerry Puckett, Tahiry Rajaonarison, Ian Rose, D. Sarah Stamps, Cedric Thieulot, Wanying Wang, Iris van Zelst, Siqi Zhang

geodynamics.org

Using ASPECT

Using ASPECT

- **ASPECT is a command line tool (demonstrated later)**
  - driven by text-based input files (.prm)
  - extended by C++ “plug-ins”

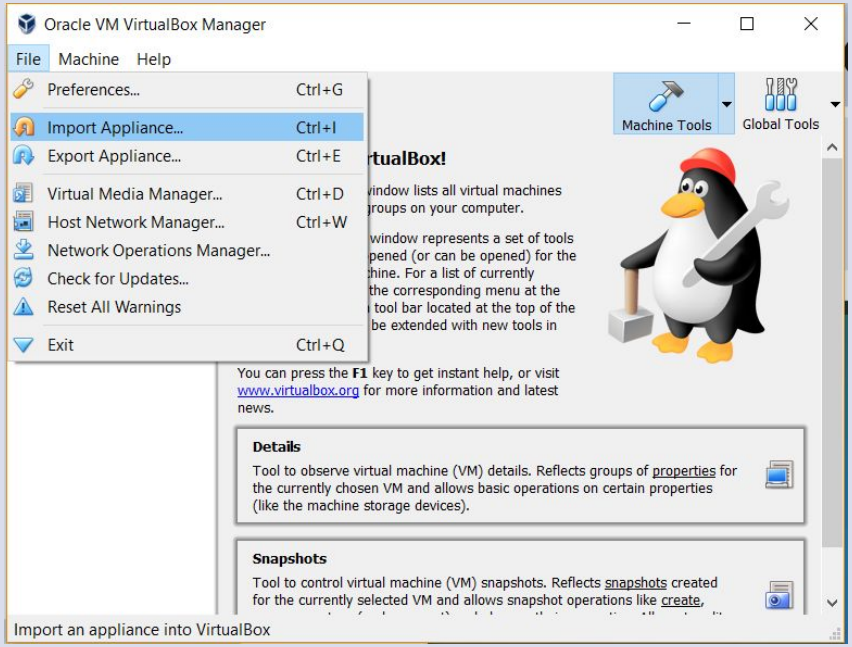
- **Outputs (demonstrated later)**
  - text-based “statistics” files
  - files that allow visualizing simulations

- **Installation**
  - from source on Linux, Mac, Windows
  - easier: via a “virtual machine” using VirtualBox

Start Oracle VM VirtualBox and import VM

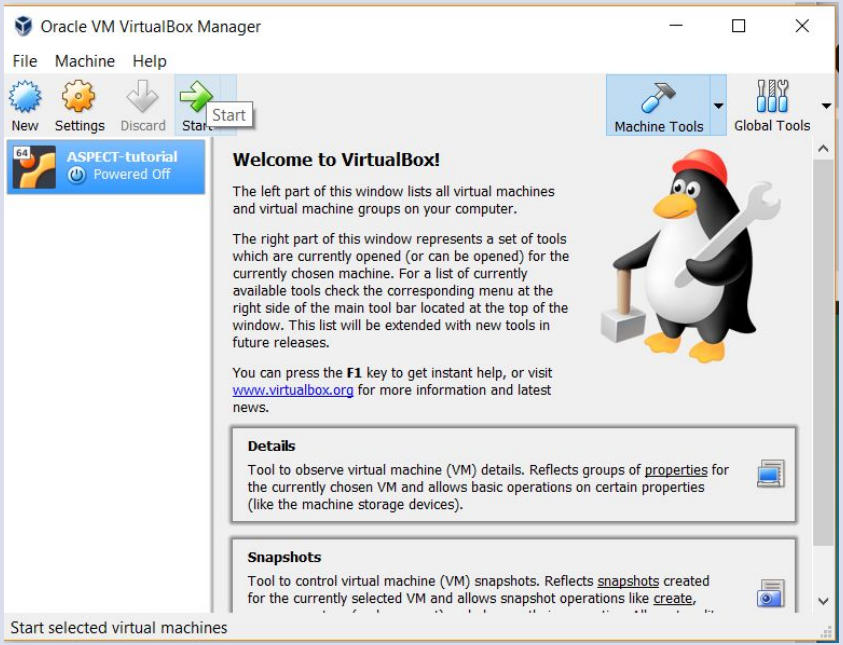
- Under machine settings, choose appropriate number of cores and RAM
- The machine is available at

  https://www.math.clemson.edu/~heister/dealvm/



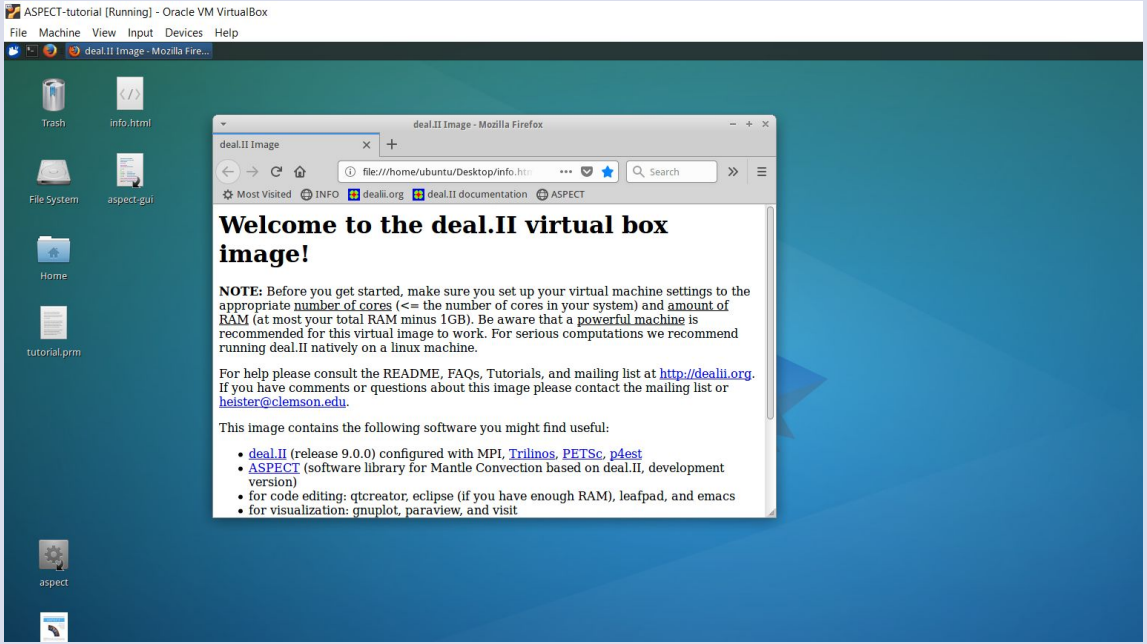
VirtualBox

Start ASPECT-tutorial virtual machine



VirtualBox

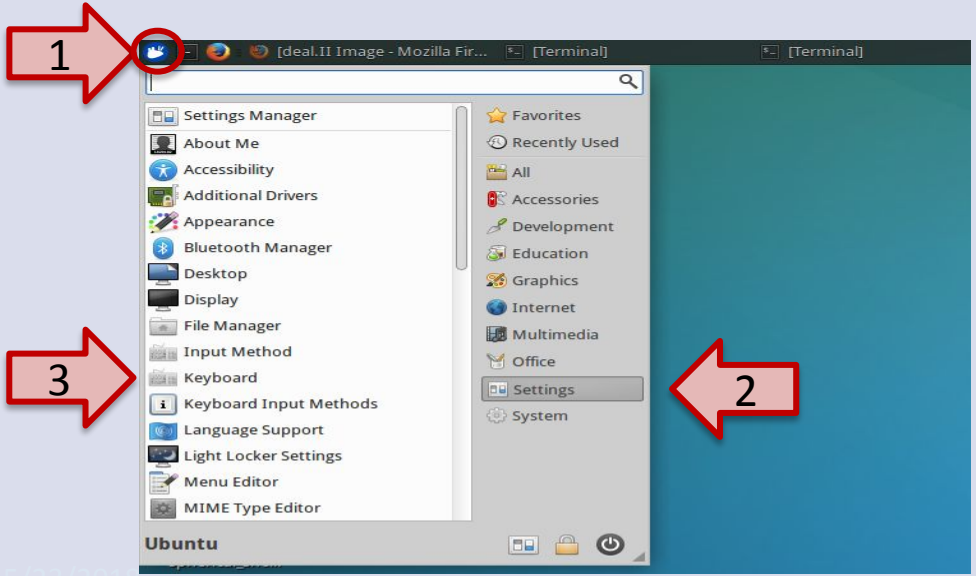
You should see a screen like this:

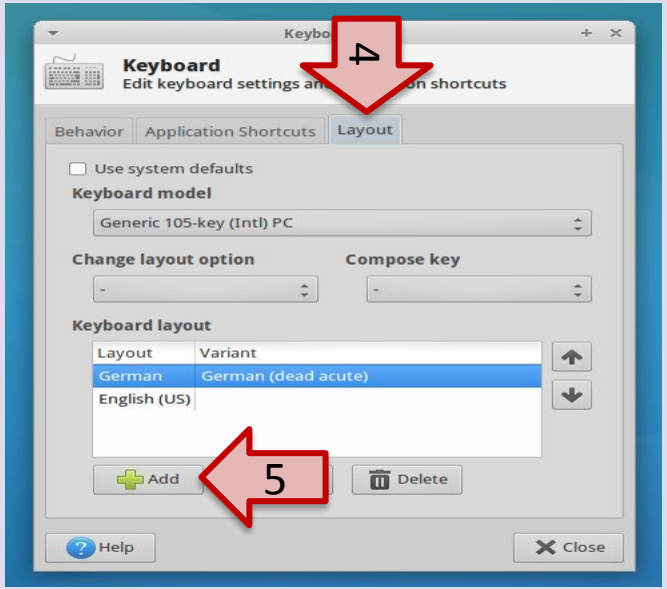


VirtualBox: Keyboard layout

COMPUTATIONAL INFRASTRUCTURE for GEODYNAMICS

If you have a different keyboard layout than the standard English(US)…





Demonstration

- ASPECT models are described through parameter files (e.g. tutorial.prm)
- Parameter files determine (i) discretization details, (ii) initial conditions, (iii) boundary conditions, (iv) …

Day 1 Instructions:
https://bit.ly/ASPECT-day1

- By the end of today, you should be able to:
  – Run aspect from the command line
  – Understand the basic layout of parameter files
  – Be able to visualize simulation results in ParaView.

Demonstration

- Begin by opening a “terminal” (ctrl + alt + t)
- Download files for the tutorial:

```
git clone https://github.com/gassmoeller/aspect-tutorials
```

- This creates a folder “aspect-tutorials” in your home directory. To go there in your terminal:

```
cd ~/aspect-tutorials/2020-tectonics-modeling-tutorial/day1/
```

Day 1 Instructions:
https://bit.ly/ASPECT-day1

Demonstration

- Change to the appropriate directory and run:

```
cd ~/aspect-tutorials/2020-tectonics-modeling-tutorial/day1/
~/aspect/aspect tutorial.prm
```

- This takes ~30 seconds. Once finished, look at "log.txt" in the "output" folder:

```
leafpad output/log.txt
```

- There are many other files in "output" (we will get to that later)

Day 1 Instructions:
https://bit.ly/ASPECT-day1

Visualizing Results with ParaView

Visualization with ParaView

- To visualize the simulation results, we will use *ParaView*
- Used to visualize large data sets
- Supports visualization data via isosurfaces, slices, streamlines, volume rendering, …

- Already installed on the virtual machine
- Open it by typing “paraview” in a terminal

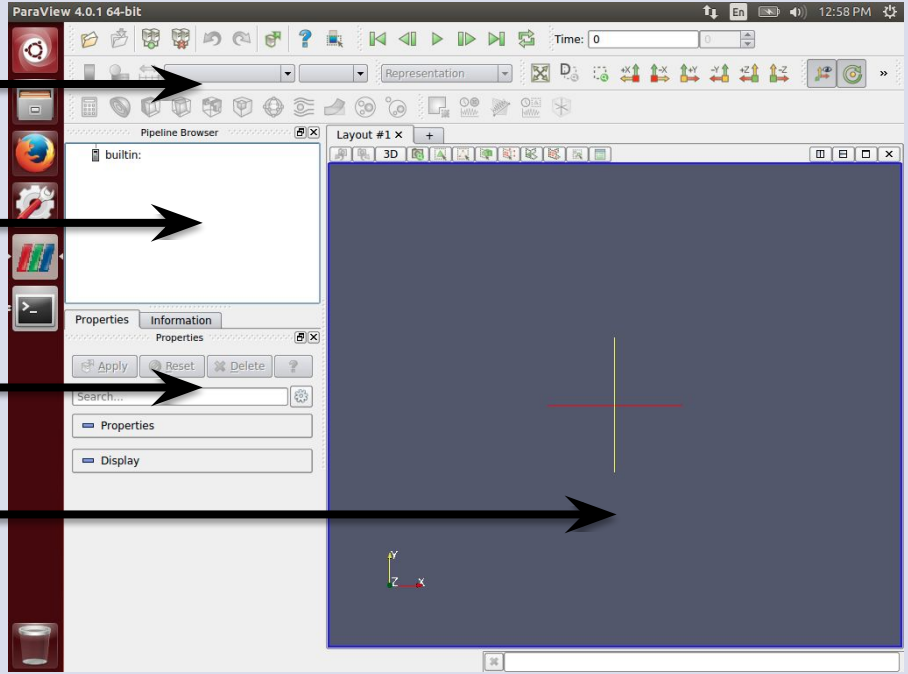
Visualization with ParaView

Toolbars

Pipeline Browser

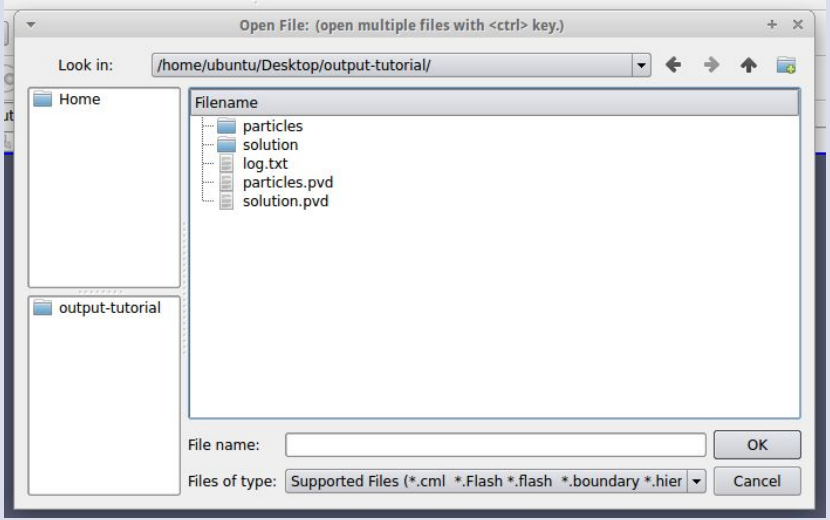
Object Inspector

2D/3D View



Visualization with ParaView

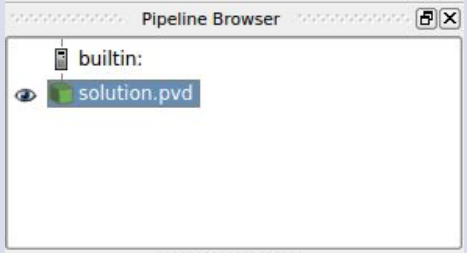
- Start by opening solution.pvd which was created by ASPECT
- You can choose “Open” from the File menu or use the Open icon (toolbar)

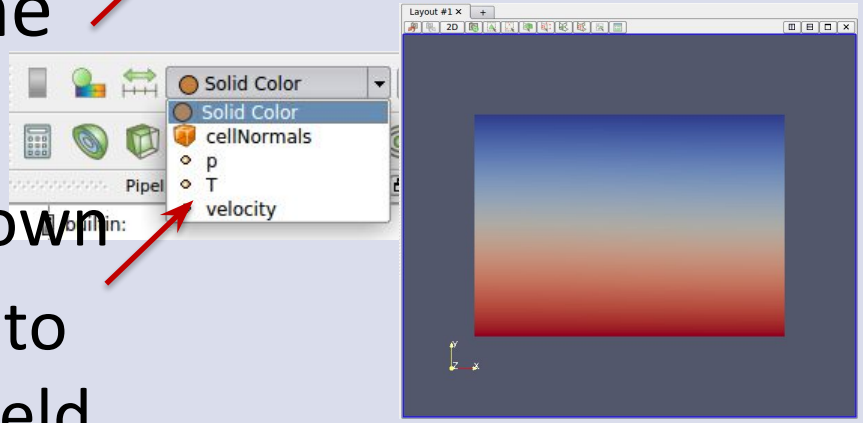


- The file is in /home/ubuntu/aspect-tutorial/2020-tectonics-modeling-tutorial/day1/output/

Visualization with ParaView

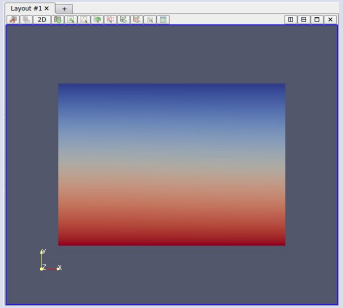
- The file will appear in the pipeline browser
  - Make sure this is solution.pvd
- Click “Apply” to show the field in the view area
  - By default, no field is shown
  - Select “T” in the toolbar to show the temperature field



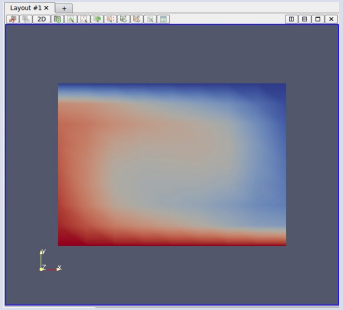


Visualization with ParaView

- The top toolbar has buttons to change the time, shown below
  - Click the play button and watch how the temperature field changes
  - Near the end, is the temperature field static? Is the velocity field static? Is material moving?



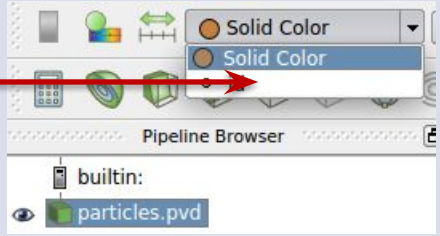
Frame 0



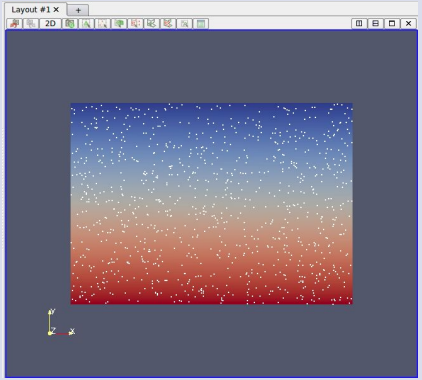
Frame 231

Visualization with ParaView

- Open the file particles.pvd and click "Apply"
  - The tracer particles from the simulation now appear on the temperature field
  - By default they are uniformly colored
  - Change the coloring scheme to "id" to show each particle in a different color
  - Click play again to see how material is flowing with the tracer particles
  - Even when the temperature field is static, is material flowing?
  - How would you characterize this flow pattern? Where is the upwelling material? The downwelling material?



Change the coloring scheme to "id"



Temperature field with tracer particles

Break + play time

Day 1 Instructions:
https://bit.ly/ASPECT-day1

Try this:

– Open a terminal and enter the following commands:

```
git clone https://github.com/gassmoeller/aspect-tutorials
cd aspect-tutorials/2020-tectonics-modeling-tutorial/day1/
~/aspect/aspect tutorial.prm
```

– Inspect “tutorial.prm”

– Open “log.txt” in the “output/” folder

– Open “solution.pvd” in ParaView to visualize

Setting up models in ASPECT

How to describe physical situations in ASPECT

Setting up models in ASPECT

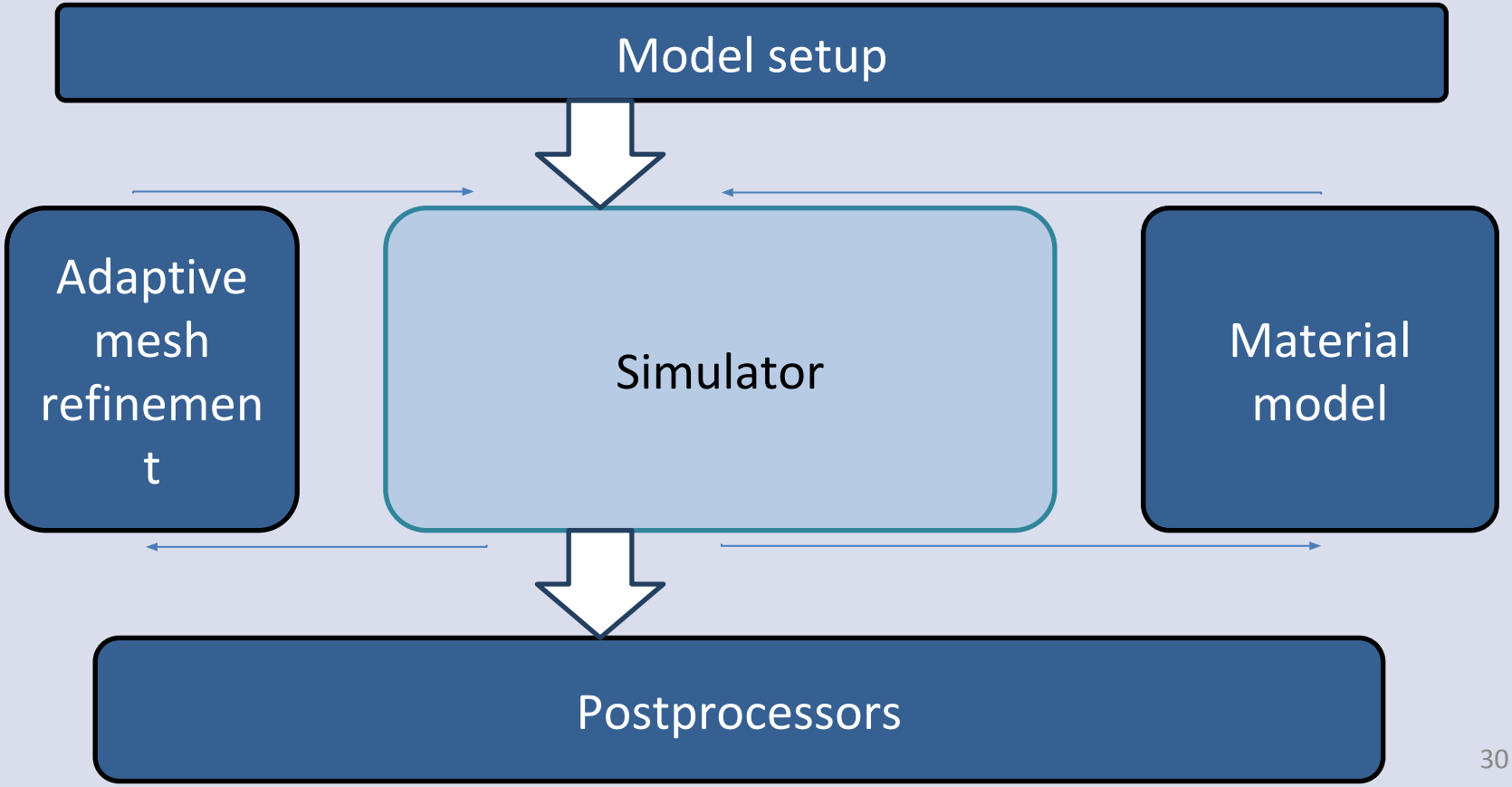
- PDE Models consist of several key components:
  1. The equations for the model
  2. Coefficients in the model
  3. Dependent and independent variables
  4. The initial state of the model
  5. The boundary conditions

- We need to discretize the model (mesh/finite element choices)

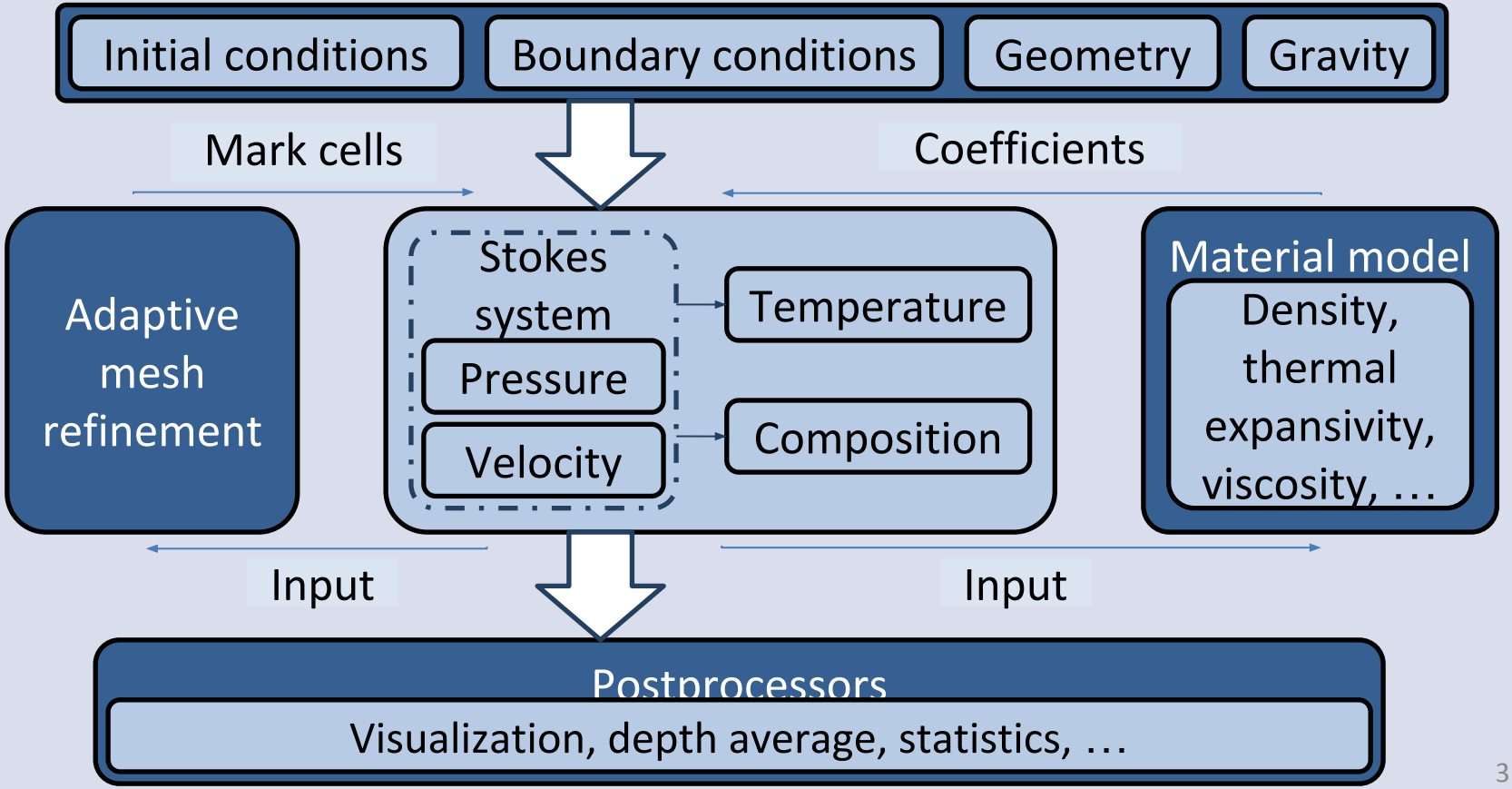
- We need to say what we want to do with a computed solution

All of this is described in the input files.

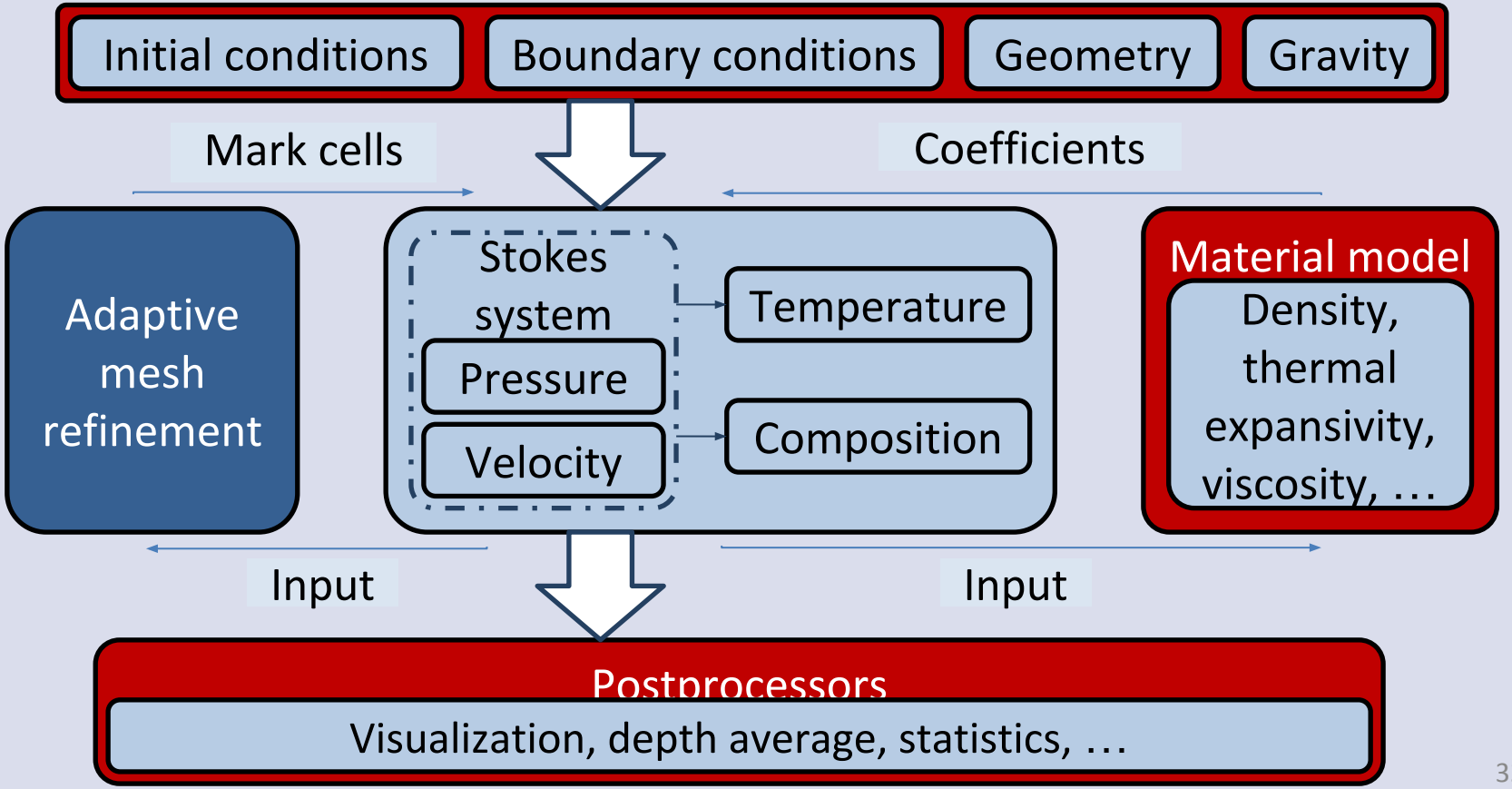
Setting up models in ASPECT



Setting up models in ASPECT



Setting up models in ASPECT



Setting up models in ASPECT

- All of these pieces are represented in the input "parameter" file
- Related parameters are grouped into "subsections"
- Parameter files can (should be!) documented

Input file in `leafpad`

tutorial.prm

File Edit Search Options Help

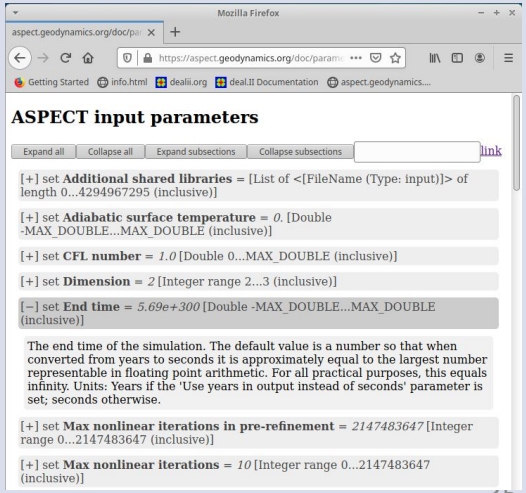
```
# At the top, we define the number of space dimensions we would like to work
# in:
set Dimension                                = 2

# There are several global variables that have to do with what
# time system we want to work in and what the end time is. We
# also designate an output directory.
set Use years in output instead of seconds = true
set End time                                 = 5e10
set Output directory                         = output


# Then come a number of sections that deal with the setup
# of the problem to solve. The first one deals with the
# geometry of the domain within which we want to solve.
# The sections that follow all have the same basic setup
# where we select the name of a particular model (here,
# the box geometry) and then, in a further subsection,
# set the parameters that are specific to this particular
# model.
subsection Geometry model
```

What parameters exist?

- Website with all parameters at aspect.geodynamics.org -> “Parameters” or aspect.geodynamics.org/doc/parameter_view/parameters.xml
- Searchable
- Click on [+] to expand descriptions



General parameters

- There is a small set of “global” or “general” parameters for the simulation

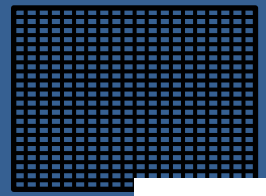
3	set Dimension	= 2
8	set Use years in output instead of seconds	= true
9	set End time	= 3e10
10	set Output directory	= output

- **Note:** Internally, calculations will always use seconds, but *externally* we can use years.
- *End time* has been set to 3×10^{10} years. (We use “computer notation”: 2×10^3 is written as 2e3.)
- Output will be stored in the directory named “output”.

Geometry model

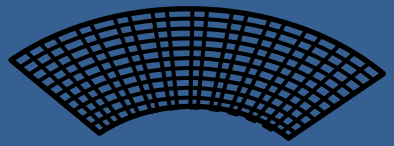
2D or 3D?

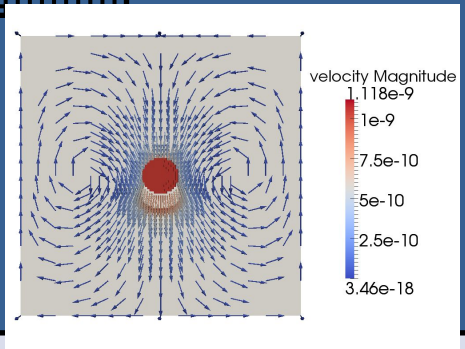
Geometry model

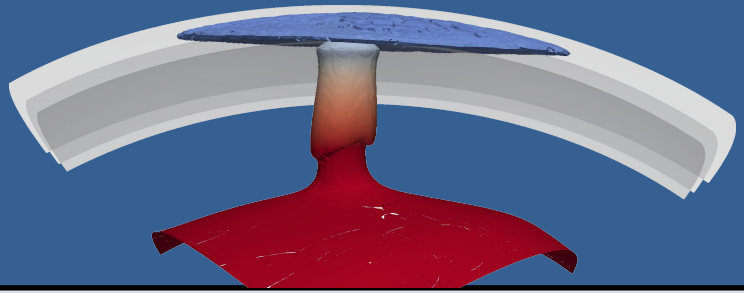


Box

Spherical shell

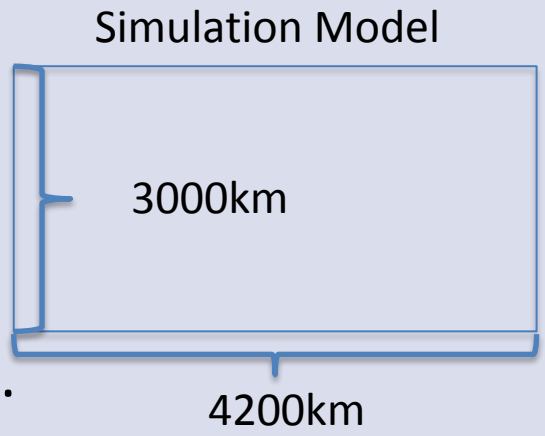






Geometry model

- Aspect has many built in geometry models (e.g., “box” and “spherical shell”).
- A box is a rectangle (2D) or a cuboid (3D).
- Width (X extent) of the box is 4.2×10^6 meters and depth (Y extent) is 3×10^6 meters.
- ASPECT uses SI units: meter, kg, seconds/years

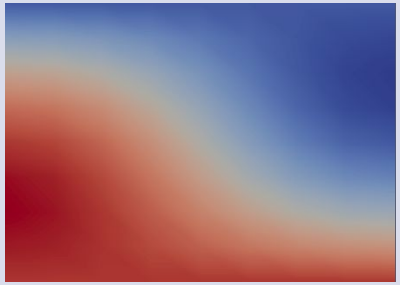



```
subsection Geometry model
    set Model name = box
    subsection Box
        set X extent = 4.2e6
        set Y extent = 3e6
    end
end
```

Initial temperature

Initial condition models specify the initial temperature

- "function" model lets us specify the initial temperature as a formula, with user defined constants.
- Here we are specifying a sinusoidal perturbation of a linear temperature profile.



Initial temperature field (p=-0.5)

$$T(x,y) = T_{top} + (T_{bottom} - T_{top})(1 - \frac{y}{D} - pcos(\frac{k\pi x}{L})sin(\frac{\pi y}{D})$$

```
subsection Initial temperature model
    set Model name = function
    subsection Function
        set Variable names = x,y
        set Function constants = p=-0.01, L=4.2e6, D=3e6,
            pi=3.1415926536, k=1, T_top=273, T_bottom=3600
        set Function expression = T_top + (T_bottom-T_top)*
            (1-(y/D)-p*cos(k*pi*x/L)*sin(pi*y/D)))
    end
end
```

Temperature at the boundary

- Temperature at the bottom of the box is fixed at 3600 K, top is 273K
- Depending on the model, left and right options can be similarly specified (and front/back in 3D)
- Unspecified boundaries default to no heat flux (i.e., an insulated boundary)


```
subsection Boundary temperature model
    set Fixed boundary indicators    = top, bottom
    set Model name                   = box
    subsection Box
        set Bottom temperature       = 3600
        set Top temperature          = 273
    end
end
```

Velocity at the boundary

- One can prescribe the velocity at boundaries
- Or choose prescribed tractions
- Or choose “free slip” boundaries

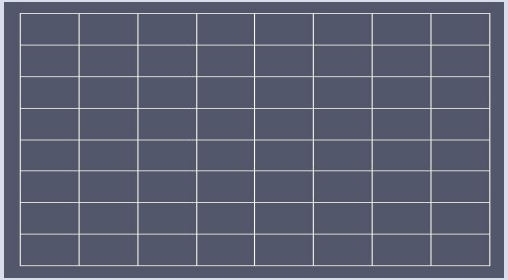
- Here: Free slip all around

```
subsection Boundary velocity model
    set Zero velocity boundary indicators          =
    set Tangential velocity boundary indicators    = left, right, top, bottom
end
```

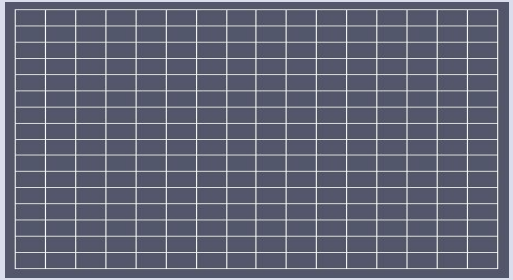
Discretization

Initial global refinement specifies the mesh “grid spacing”

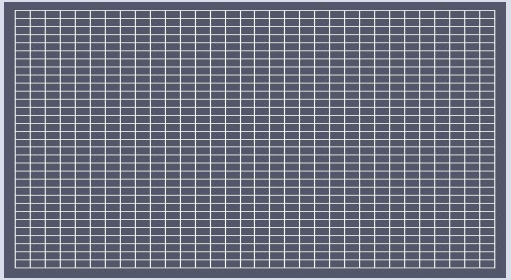
- For this tutorial, REFINE=3 or 4 or 5 (no “adaptive” mesh refinement for now).



REFINE=3 (8x8 cells)



REFINE=4 (16x16 cells)



REFINE=5 (32x32 cells)

```
subsection Mesh refinement
        set Initial global refinement                    = <REFINE>
        set Initial adaptive refinement                  = 0
        set Time steps between mesh refinement           = 0
end
```

Postprocessing

- This section says how to analyze the simulation data
- Here: Heat flux and other statistics, visualization
- Graphical output is generated every 1e7 simulated years
- Also add particles to better understand the flow

```
subsection Postprocess
    set List of postprocessors = velocity statistics, temperature statistics, heat flux statistics, ...
                                 ...visualization, particles, basic statistics
    subsection Visualization
        set Time between graphical output    = 1e7
        set Output format                    = vtu
        set List of output variables         = material properties
    end
    subsection Particles
        set Number of particles              = 1000
        set Time between data output         = 1e7
        set Data output format               = vtu
    end
end
```

Errors in the input file…

```
Terminal
File  Edit  View  Terminal  Tabs  Help
-----------------------------------------------------------------------------
-- This is ASPECT, the Advanced Solver for Problems in Earth's ConvecTion.
--      . version 2.0.0-pre
--      . running in OPTIMIZED mode
--      . running with 1 MPI process
--      . using Trilinos
-----------------------------------------------------------------------------

    Line <1> of file <input string>: No entry with name <Output diectory> was declared in the current subsection.



----------------------------------------------------
Exception 'dealii::ExcMessage ("Invalid input parameter file.")' on rank 0 on processing:

--------------------------------------------------------
An error occurred in line <343> of file </home/ubuntu/aspect/source/main.cc> in function
    void parse_parameters(const string&, dealii::ParameterHandler&)
The violated condition was:
    false
Additional information:
    Invalid input parameter file.
--------------------------------------------------------

Aborting!
----------------------------------------------------
```

- The error message often already tells you what the problem is!

Playtime:

Convection in a 2D Box and the Rayleigh-Nusselt number

Nusselt-Rayleigh Relationship

A simple example of convection modeling with ASPECT:

- Consider a box heated below and cooled above
  -> convection transports heat from bottom to top

- Reasonable assumption:
  If we had a less viscous fluid
  -> more vigorous convection
  -> more heat is transported convectively

Nusselt-Rayleigh Relationship

- Physically, what matters is not actually the viscosity:
  - a cubic millimeter of honey will not convect for a temperature difference of 100 K
  - but a cubic kilometer of honey may
- Also:
  - a cubic kilometer of honey will not convect for a temperature difference of 0.1 K
  - but it may with a temperature difference of 100 K

- Thus, the expected degree of convection is described by the "Rayleigh number"

$$Ra = \frac{\rho_0 g \alpha \triangle T D^3}{\eta \kappa}$$

Nusselt-Rayleigh Relationship

The mathematical statement of the example is therefore:

- Consider a box heated below and cooled above
  -> convection transports heat from bottom to top

- Reasonable assumption:
  Larger Rayleigh number
  -> more vigorous convection
  -> more heat is transported convectively

- Measure conductive heat transported via the “Nusselt number” (proportional to the surface heat flux)

Nusselt-Rayleigh Relationship

COMPUTATIONAL INFRASTRUCTURE for GEODYNAMICS

Rayleigh number

$$Ra = \frac{\rho_0 g \alpha \triangle T D^3}{\eta \kappa}$$

Known empirical relationship:

$$Nu \sim Ra^{1/3}$$

Let's confirm this by playing with one of the parameters (viscosity)!

Our parameters:

$$\rho_0 = 3300, g = 9.81, \alpha = 2 \cdot 10^{-5}, \triangle T = (3600 - 273) = 3327,$$

$$D = 3 \cdot 10^6, k = 4.7, c_p = 1250, \kappa = \frac{k}{\rho_0 c_p}$$

(See also https://en.wikipedia.org/wiki/Nusselt_number#Free_convection_from_horizontal_plates)

Convection in a 2D Box

1. Other output is shown in “output/statistics”. Open this file and see what sort of values are stored here.

```
xdg-open output/statistics
```

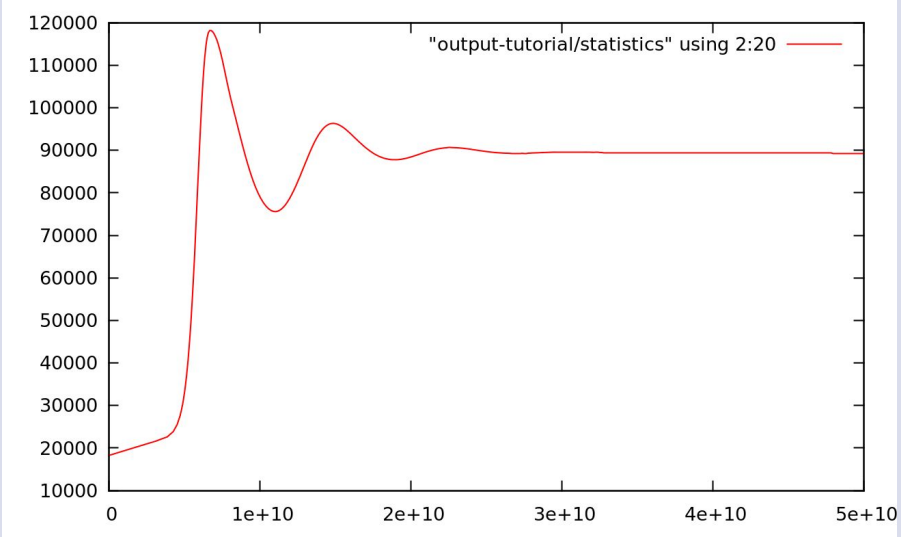
2. We want to see how heat flux changes over time. Plot the results in gnuplot showing simulation time vs. heat flux

```
gnuplot
plot “output/statistics” using 2:20 with lines
```

3. What is the surface heat flux at the end of this run?

Heat Flux / Nusselt No.

Example of visualizing time (column 2) vs heat flux (column 20):



Commands:

```
gnuplot
plot "output/statistics" using 2:20 with lines
```

Day 1 Instructions:
https://bit.ly/ASPECT-day1

Nusselt-Rayleigh Relationship

- Link to spreadsheet: https://bit.ly/NusseltSpreadsheet

Ra	4000	20000	100000	500000
end time	1.00E+12	2.00E+11	3.00E+10	5.00E+09
viscosity	1.28E+25	2.55E+24	5.10E+23	1.02E+23
Refine 2				
Refine 3				
Refine 4				
Refine 5				
Ra^1/3	9.52E+04	1.63E+05	2.79E+05	4.76E+05

Break + play time

Day 1 Instructions:
https://bit.ly/ASPECT-day1

Try this:

- Plot heat flux over time
- Change geometry
- Change boundary conditions

- Open manual and go through the list of cookbooks (input files are in ~/aspect/cookbooks)

More about ASPECT

ASPECT can run in “debug” or “optimized” mode:

- “debug” mode:
  - lots of internal checks to verify correctness of algorithms in deal.II, ASPECT, and user-provided plugins
  - no compiler optimizations to make debugging simpler
  - slow
- “optimized” (or “release”) mode:
  - most internal checks are switched off
  - use all available compiler optimizations
  - fast: about 4-10 times faster
- The default is **debug** mode

More about ASPECT

Guidance for debug vs. optimized mode:

- *Always* test all new setups, models, and plugins in debug mode
  - This makes finding bugs *much much simpler*!

- Run production runs with
  - more mesh refinement
  - optimized mode

- *Never* do production runs in debug mode
  - It is a waste of CPU time
  - Remember: 1 CPU hour = $0.10

More about ASPECT

- To create an ASPECT executable in release mode type in a terminal:

```
mkdir ~/aspect/build-release
cd ~/aspect/build-release
cmake -DCMAKE_BUILD_TYPE=Release ~/aspect
make -j2
```
(may take 30 min)

(more on this tomorrow)

- To run release mode use:

```
~/aspect/build-release/aspect tutorial.prm
```

- Verify this by looking at the first lines of output, timing information that is output every 100 time steps

Running ASPECT in parallel

ASPECT can run in parallel on a single machine:

- Several ASPECT executables working together in parallel

- To try this:

```
mpirun -np 2 ~/aspect/aspect tutorial.prm
```

(“np” = “number of processes”)

- Verify this by looking at the first lines of output, timing information that is output every 100 time steps

Running ASPECT in parallel

General guideline:

- Using more processors is faster if every processor has at least 30,000 degrees of freedom
  - find the number of freedom at the top of *log.txt*
- If you do have a large problem, use
  - as many processors as you have
  - but no more than *#DoFs / 30,000*
- For example:
  - *tutorial.prm* with 3 global refinements has 948 DoFs
  - *tutorial.prm* with 5 global refinements has 13,764 DoFs
- Neither of these benefits much from parallelization: The cost of communication is larger than the gain due to parallelization!