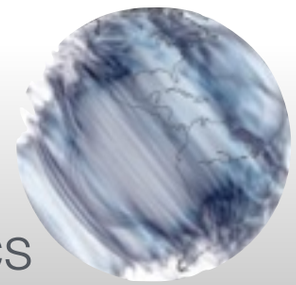
Spectral-element method

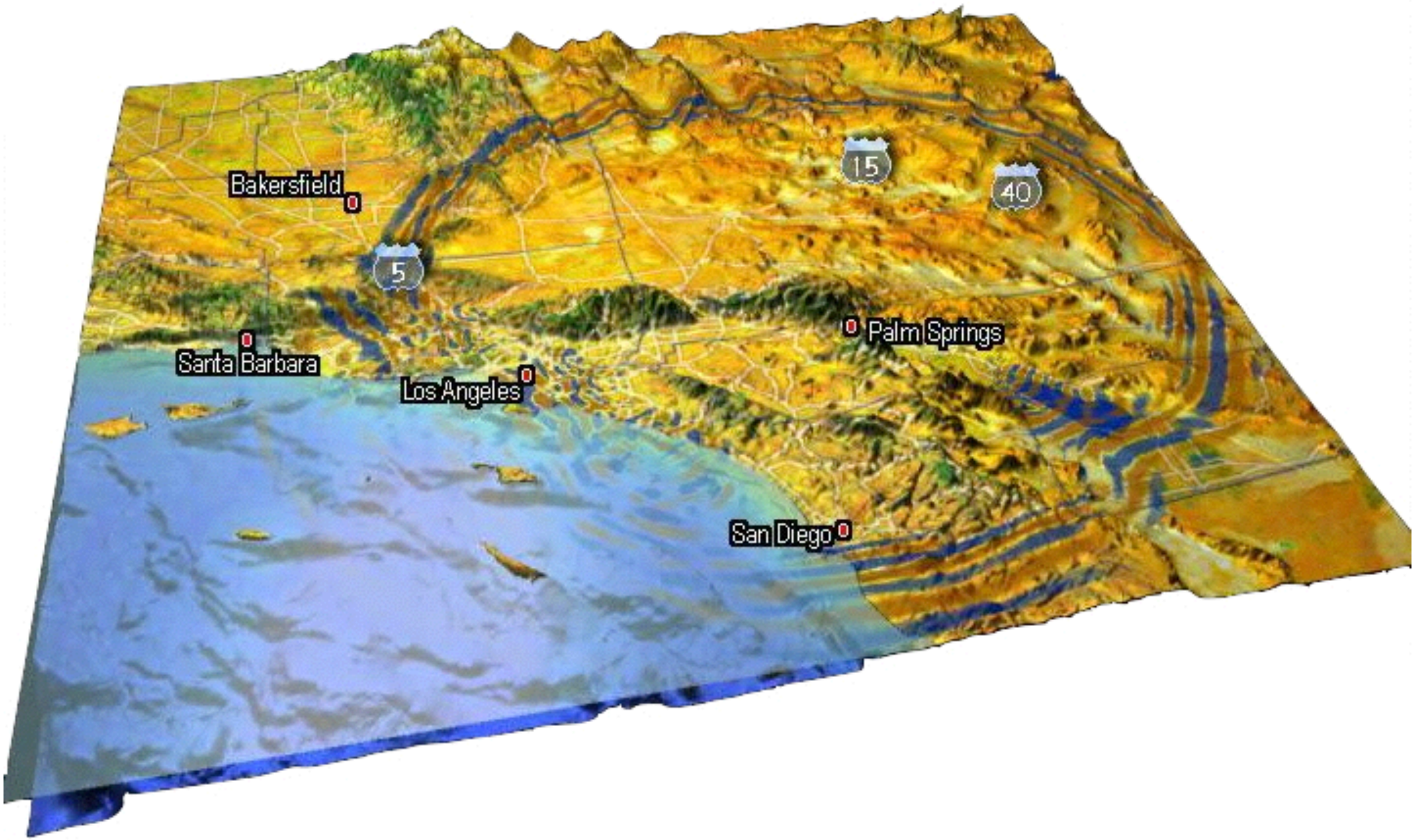
SPECFEM

Community Codes for Forward and Inverse Modeling



Computational Geophysics

SPECFEM3D_Cartesian

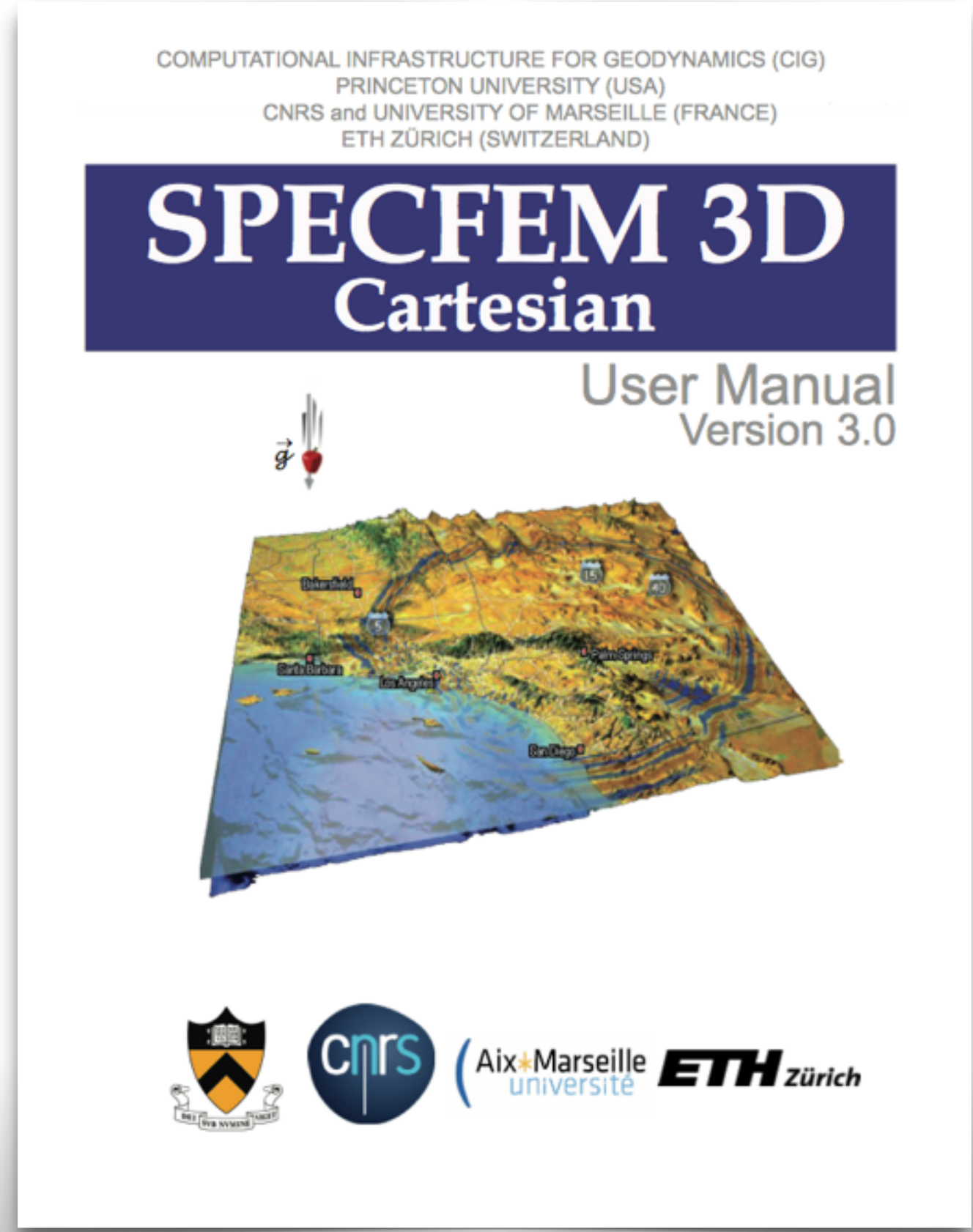


Open Source Community Software

SPECFEM3D_Cartesian

- Unstructured meshes (CUBIT)
- Load-balanced mesh partitioning
- Acoustic & elastic coupling
- Poroelasticity
- Anisotropy
- Attenuation
- Adjoint kernels

www.geodynamics.org



Installation

Installation

download:

- git checkout / web download

Installation

configuration & compilation:

- ./configure F90=ifort
- make

Quick start

EXAMPLES/

- homogeneous_halfspace
- layered_halfspace
- tomographic_model
- waterlayered_halfspace
- .... contribute your own models!

Quick start

README

```
------------------------------------------------------------------------
README
------------------------------------------------------------------------

step-by-step tutorial:

0. check that all software is available (or that modules are loaded):
     intel/openmpi, cubit, scotch, python, gnuplot


1. configure package:

   - From the SPECFEM3D root directory SPECFEM3D/
     configure the package, e.g. using intel's ifort compiler:
     > cd SPECFEM3D
     > ./configure F90=ifort

     If successful, this will generate the files:
     Makefile, constants.h, and precision.h, among others

   - copy two run scripts from SPECFEM3D/UTILS/Cluster/
     into SPECFEM3D/, e.g.,
     pbs/go_generate_databases_pbs.bash
     pbs/go_solver_pbs.bash


2. create mesh:

   - change to the examples directory SPECFEM3D/EXAMPLES/homogeneous_halfspace:
     > cd EXAMPLES/homogeneous_halfspace

   - open the cubit GUI:
     > claro (or cubit)

     then run meshing script:
     claro -> Menu "Tools" -> "Play Journal File" ... and select file: "block_mesh.py"

     if everything goes fine, this creates all the mesh files in a subdirectory MESH/:
        MESH/absorbing_surface_file_bottom
        MESH/absorbing_surface_file_xmax
        MESH/absorbing_surface_file_xmin
        MESH/absorbing_surface_file_ymax
        MESH/absorbing_surface_file_ymin
        MESH/free_surface_file
        MESH/materials_file
        MESH/mesh_file
        MESH/nodes_coords_file
        MESH/nummaterial_velocity_file

     the cubit graphics window should show a mesh similar to the file homogeneous.png
```

Mesh generation

meshing:

- CUBIT/Trelis
- in-house mesher xmeshfem3D

Mesh generation

cubit.sandia.gov

news | divers | code | platform | journals | seismology

Sandia National Laboratories

Employee Locator | Index | Site Map

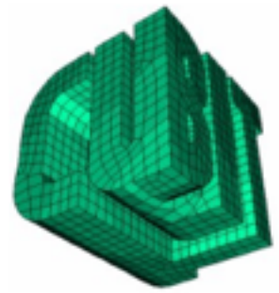
Search GO

About Sandia | Mission Areas | Newsroom | Careers | Doing Business | Education | Contact Us

CUBIT Toolkit

- Licensing
- Documentation
- Tutorials
- Other Tools
- Support
- Passwords needed:
  - Downloads
  - Developers' Pages

CUBIT



News of Note:

- CUBIT 15.2 Released July 19, 2016
- Next Cubit Tutorials November 15-16, 2016
- CUBIT 15.1 Released February 29, 2016
- CUBIT 15.0 Released April 16, 2015
- 25th International Meshing Roundtable will be held September 2016, Washington DC
- CUBIT 14.1 Released January 13, 2014

The CUBIT Geometry and Mesh Generation Toolkit

CUBIT is a full-featured software toolkit for robust generation of two- and three-dimensional finite element meshes (grids) and geometry preparation. Its main goal is to reduce the time to generate meshes, particularly large hex meshes of complicated, interlocking assemblies. It is a solid-modeler-based preprocessor that meshes volumes and surfaces for finite element analysis. Mesh generation algorithms include:

- Quadrilateral and triangular paving
- 2D and 3D mapping
- Hex sweeping and multi-sweeping
- Tet meshing
- Many special purpose primitives.

CUBIT also contains many algorithms for controlling and automating much of the meshing process, such as

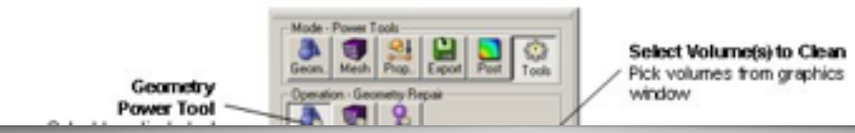
- Automatic scheme selection
- Interval matching
- Sweep grouping
- Sweep verification

And, of course, CUBIT also includes state-of-the-art smoothing algorithms.

CUBIT provides an extensive suite of tools for geometry decomposition and mesh generation.

More-extensive list of CUBIT Features...

See Cubit Licensing for information on obtaining the Cubit Geometry and Mesh Generation Toolkit. Licensed users may download the current release from this website.

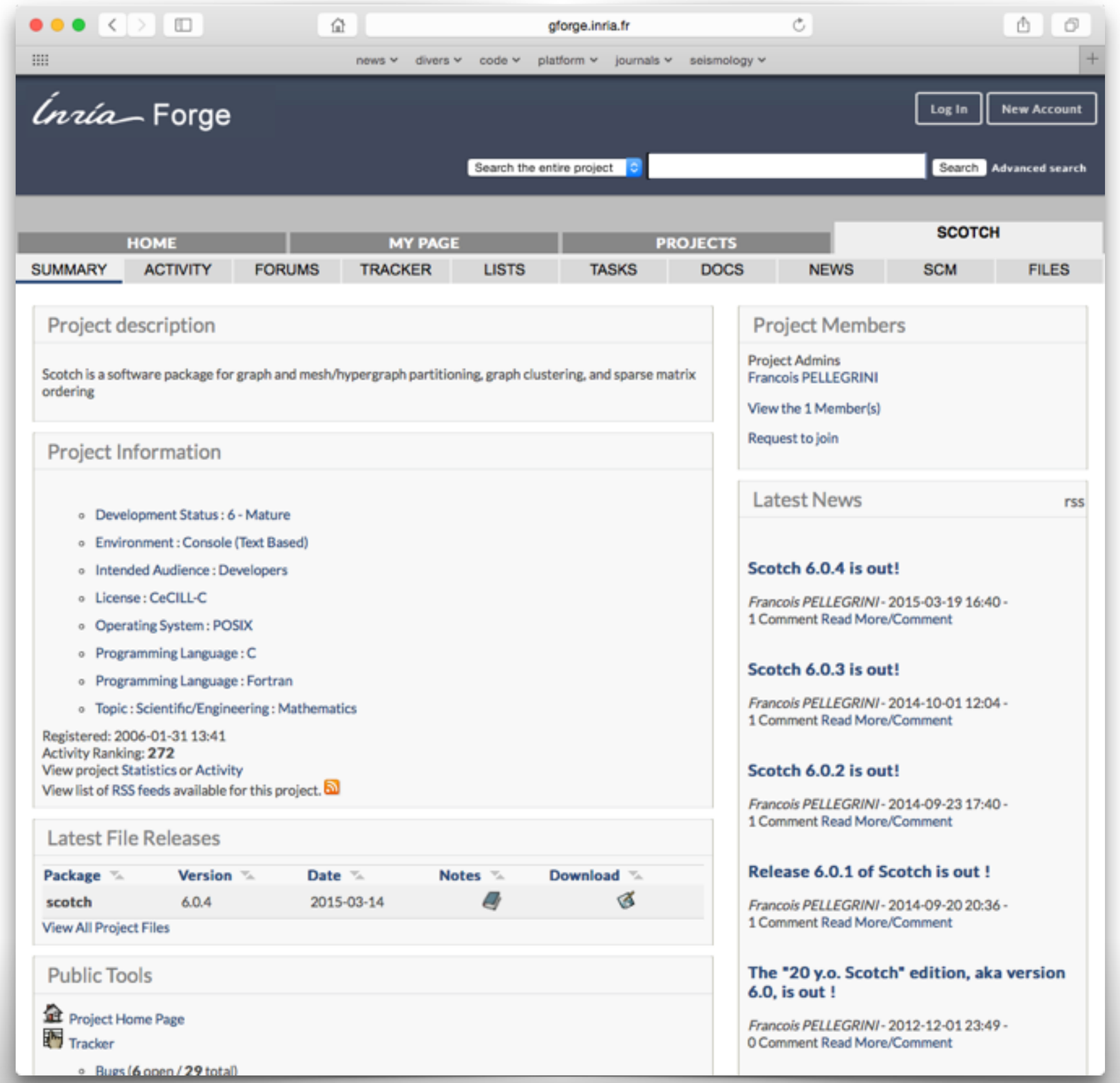


Mesh generation

partitioning:

- SCOTCH

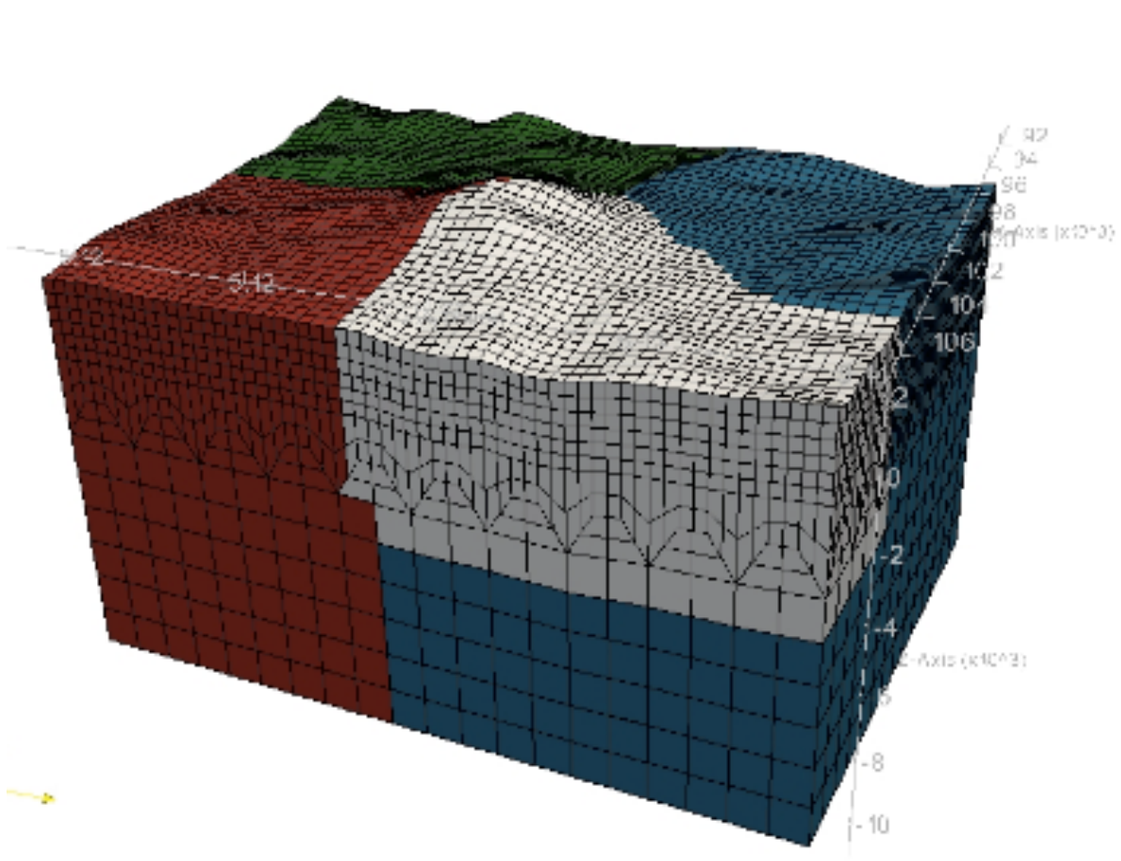
Mesh generation

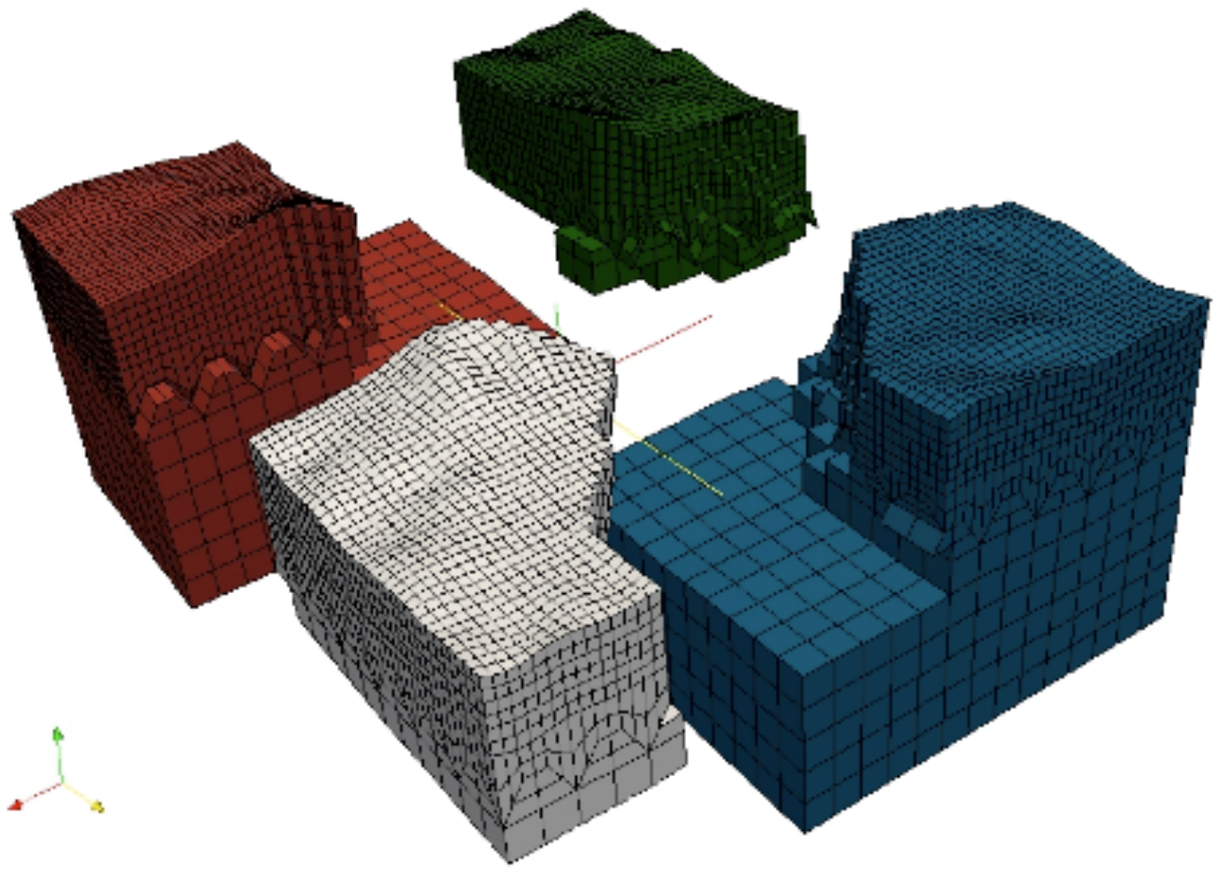


Mesh generation

partitioning:

- decompose_mesh_SCOTCH/





Input

CMT solutions:

- DATA/CMTSOLUTION

CMTSOLUTION

```
PDE  1999 01 01 00 00 00.00  67000 67000 -25000 4.2 4.2 hom_explosion
event name:        hom_explosion
time shift:        0.0000
half duration:     5.0
latitude:         67000.0
longitude:        67000.0
depth:           -25.0
Mrr:       1.000000e+23
Mtt:       1.000000e+23
Mpp:       1.000000e+23
Mrt:       0.000000
Mrp:       0.000000
Mtp:       0.000000
```

Input

stations:

- DATA/STATIONS

STATIONS

```
X1 DB 67000.00 0.000000 0.0 0.0
X2 DB 67000.00 1196.429 0.0 0.0
X3 DB 67000.00 2392.857 0.0 0.0
X4 DB 67000.00 3589.286 0.0 0.0
X5 DB 67000.00 4785.714 0.0 0.0
X6 DB 67000.00 5982.143 0.0 0.0
X7 DB 67000.00 7178.571 0.0 0.0
X8 DB 67000.00 8375.000 0.0 0.0
X9 DB 67000.00 9571.429 0.0 0.0
X10 DB 67000.00 10767.86 0.0 0.0
X11 DB 67000.00 11964.29 0.0 0.0
X12 DB 67000.00 13160.71 0.0 0.0
X13 DB 67000.00 14357.14 0.0 0.0
X14 DB 67000.00 15553.57 0.0 0.0
X15 DB 67000.00 16750.00 0.0 0.0
X16 DB 67000.00 17946.43 0.0 0.0
X17 DB 67000.00 19142.86 0.0 0.0
X18 DB 67000.00 20339.29 0.0 0.0
X19 DB 67000.00 21535.71 0.0 0.0
X20 DB 67000.00 22732.14 0.0 0.0
```

Input

forward simulation

Input

Par_file.default

```
# forward or adjoint simulation
SIMULATION_TYPE                 = 1    # 1 = forward, 2 = adjoint, 3 = both simultaneously
SAVE_FORWARD                    = .false.

# UTM projection parameters
UTM_PROJECTION_ZONE             = 11
SUPPRESS_UTM_PROJECTION         = .true.

# number of MPI processors
NPROC                           = 4

# time step parameters
NSTEP                           = 1000
DT                              = 0.05d0

# parameters describing the model
OCEANS                          = .false.
TOPOGRAPHY                      = .false.
ATTENUATION                     = .false.
USE_OLSEN_ATTENUATION           = .false.
ANISOTROPY                      = .false.

# absorbing boundary conditions for a regional simulation
ABSORBING_CONDITIONS            = .false.

# save AVS or OpenDX movies
MOVIE_SURFACE                   = .false.
MOVIE_VOLUME                    = .false.
NTSTEP_BETWEEN_FRAMES           = 200
CREATE_SHAKEMAP                 = .false.
SAVE_DISPLACEMENT               = .false.
USE_HIGHRES_FOR_MOVIES          = .false.
HDUR_MOVIE                      = 0.0

# save AVS or OpenDX mesh files to check the mesh
SAVE_MESH_FILES                 = .true.

# path to store the local database file on each node
LOCAL_PATH                      = DATABASES_MPI

# interval at which we output time step info and max of norm of displacement
NTSTEP_BETWEEN_OUTPUT_INFO      = 500

# interval in time steps for writing of seismograms
NTSTEP_BETWEEN_OUTPUT_SEISMOS   = 10000

# print source time function
PRINT_SOURCE_TIME_FUNCTION      = .false.
```

1: forward
2: pure adjoint
3: kernels

Input

cluster run: (template scripts in utils/Cluster)

- qsub go_generate_databases_pbs.bash
- qsub go_solver_pbs.bash

Input

adjoint simulation

Input

Par_file

```
# forward or adjoint simulation
SIMULATION_TYPE                 = 1    # 1 = forward, 2 = adjoint, 3 = both simultaneously
SAVE_FORWARD                    = .true.

# UTM projection parameters
UTM_PROJECTION_ZONE             = 11
SUPPRESS_UTM_PROJECTION         = .true.

# number of MPI processors
NPROC                           = 4

# time step parameters
NSTEP                           = 1000
DT                              = 0.05d0

# parameters describing the model
OCEANS                          = .false.
TOPOGRAPHY                      = .false.
ATTENUATION                     = .false.
USE_OLSEN_ATTENUATION           = .false.
ANISOTROPY                      = .false.

# absorbing boundary conditions for a regional simulation
ABSORBING_CONDITIONS            = .false.

# save AVS or OpenDX movies
MOVIE_SURFACE                   = .false.
MOVIE_VOLUME                    = .false.
NTSTEP_BETWEEN_FRAMES           = 200
CREATE_SHAKEMAP                 = .false.
SAVE_DISPLACEMENT               = .false.
USE_HIGHRES_FOR_MOVIES          = .false.
HDUR_MOVIE                      = 0.0

# save AVS or OpenDX mesh files to check the mesh
SAVE_MESH_FILES                 = .true.

# path to store the local database file on each node
LOCAL_PATH                      = DATABASES_MPI

# interval at which we output time step info and max of norm of displacement
NTSTEP_BETWEEN_OUTPUT_INFO      = 500

# interval in time steps for writing of seismograms
NTSTEP_BETWEEN_OUTPUT_SEISMOS   = 10000

# print source time function
PRINT_SOURCE_TIME_FUNCTION      = .false.
```

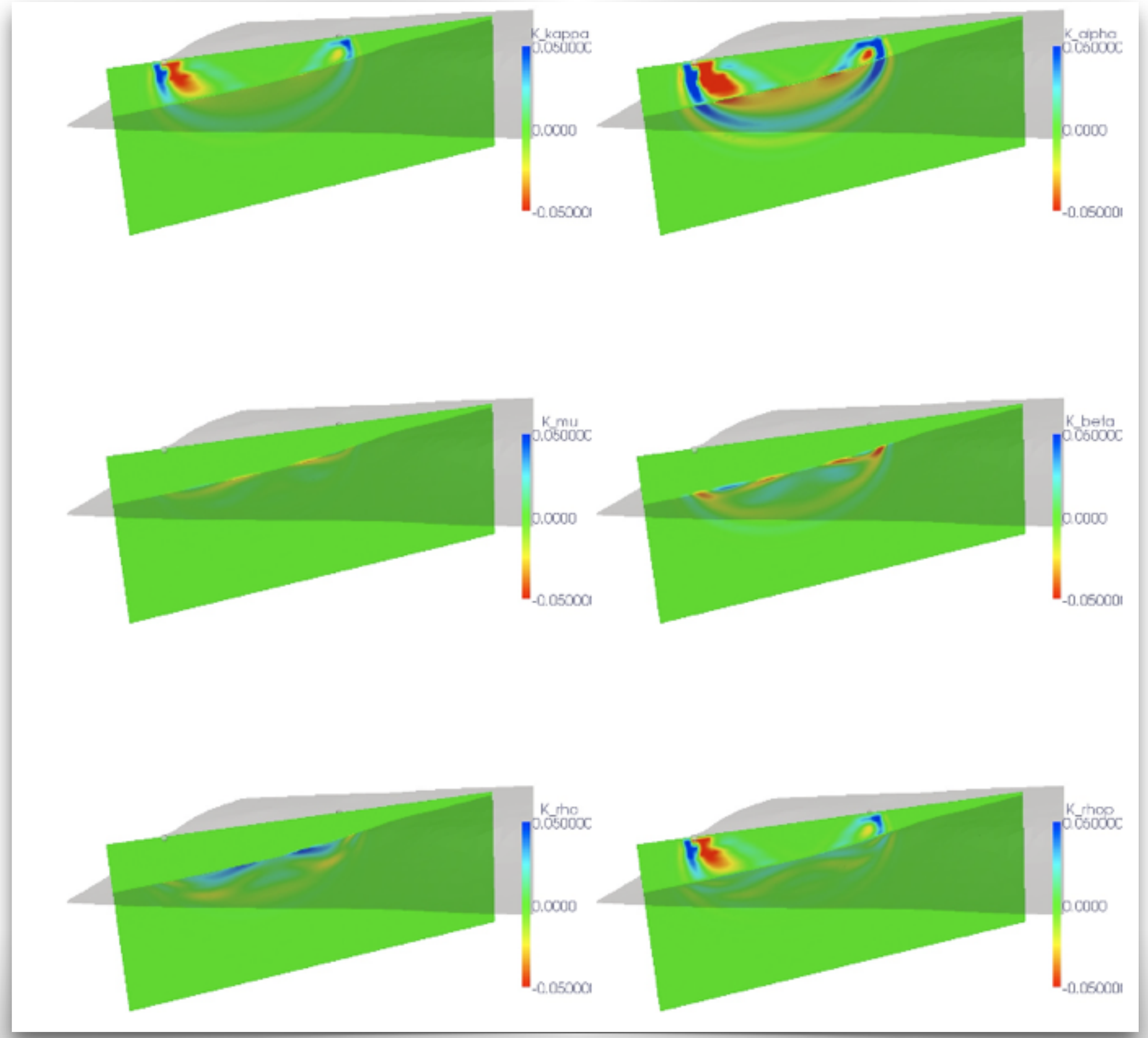
1. run: forward type=1, SAVE_FORWARD

2.run: both type=3

./change_simulation_type.pl -F/-b

Output

fluid-solid
sensitivity kernels



Output

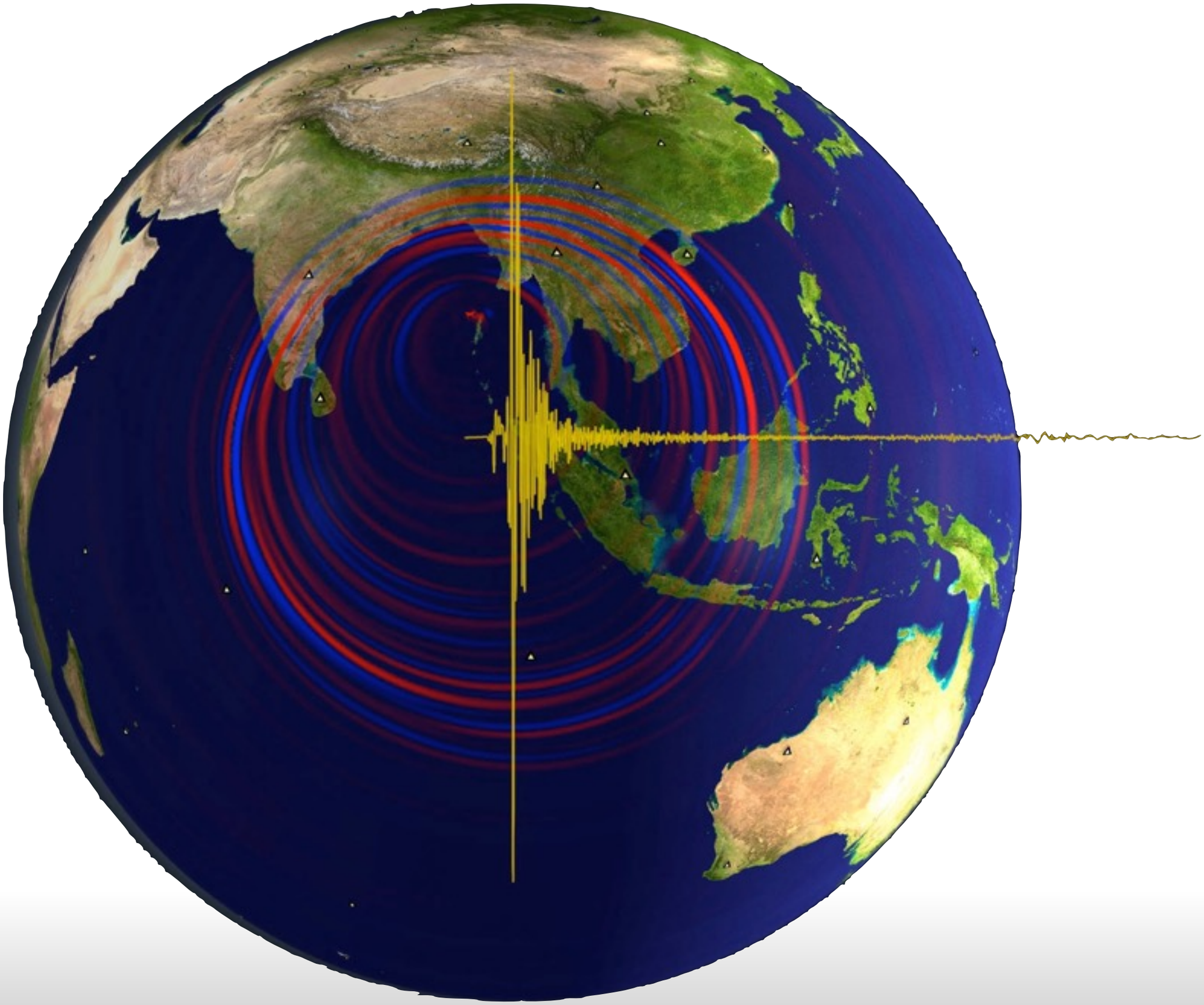
visualizations

- Paraview
- GMT
- openDX

Outreach

Southern California
shakemovie

SPECFEM3D_GLOBE

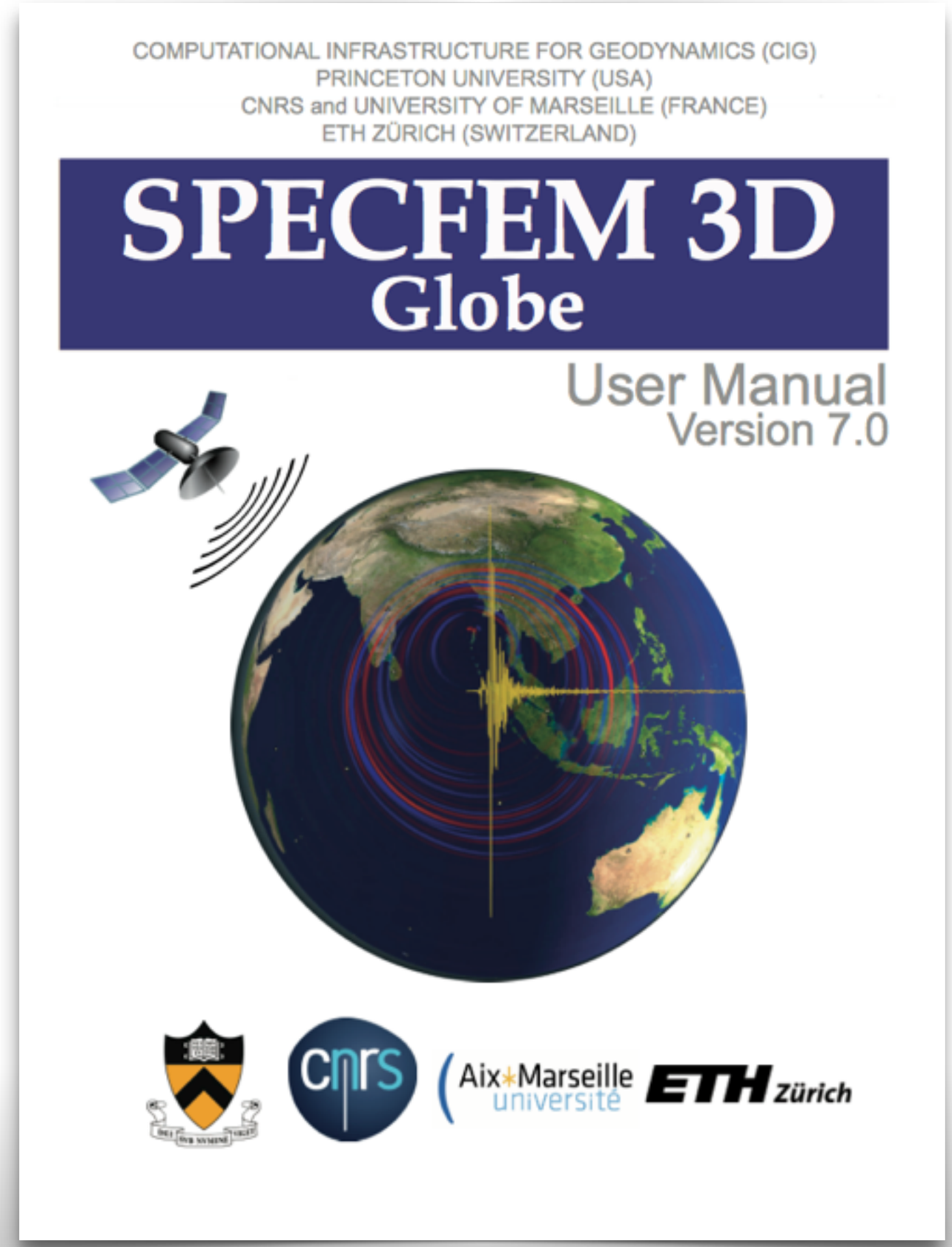


Open Source Community Software

SPECFEM3D_GLOBE

- 3D crust and mantle models
- Topography & Bathymetry
- Rotation
- Ellipticity
- Gravitation
- Anisotropy
- Attenuation

www.geodynamics.org



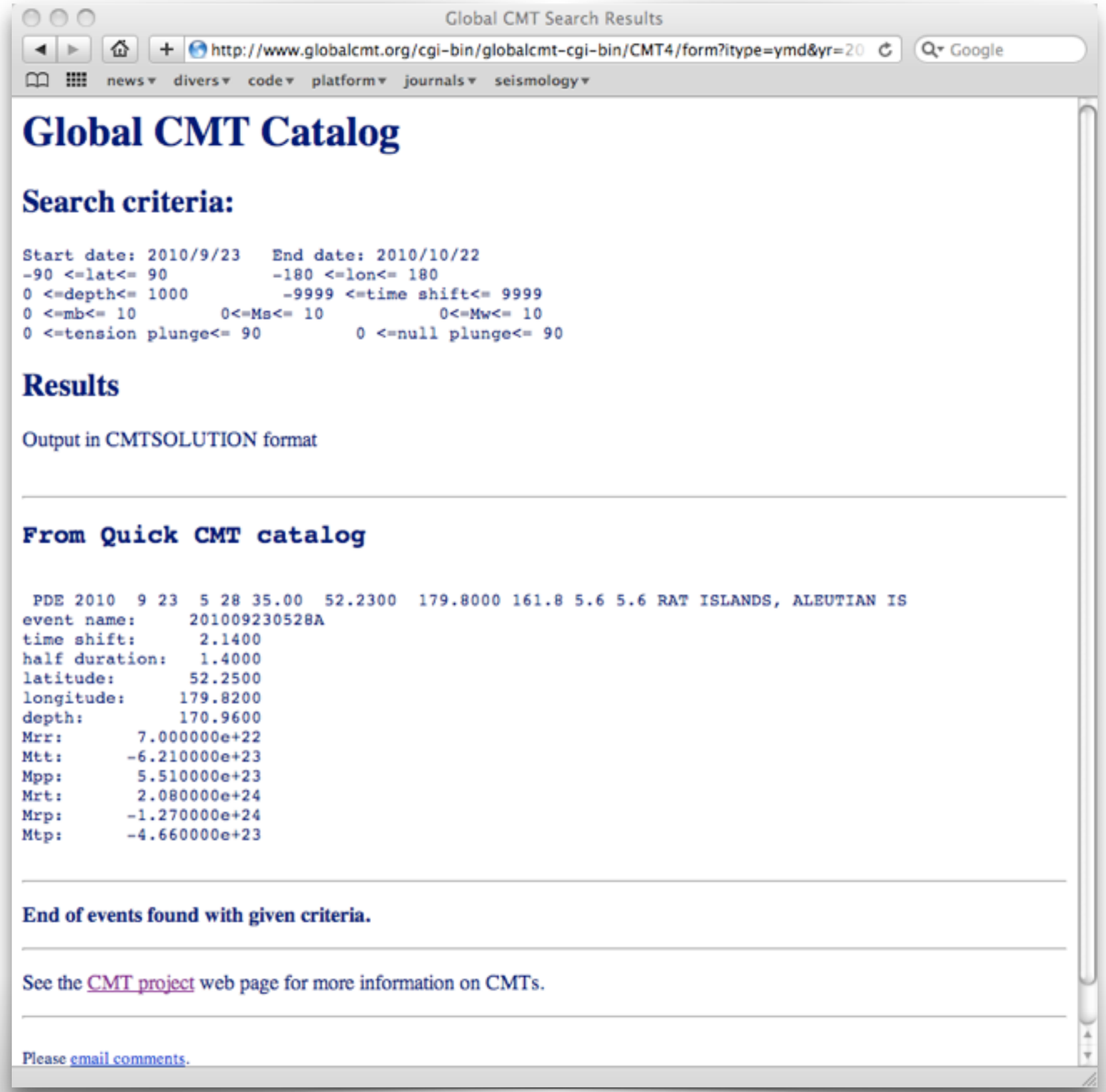
Installation

Input

CMT solutions:

- globalcmt.org
- kinematic rupture simulation

Input


Global CMT Search Results
http://www.globalcmt.org/cgi-bin/globalcmt-cgi-bin/CMT4/form?itype=ymd&yr=20
Google
news
divers
code
platform
journals
seismology
Global CMT Catalog
Search criteria:
Start date: 2010/9/23 End date: 2010/10/22
-90 <=lat<= 90 -180 <=lon<= 180
0 <=depth<= 1000 -9999 <=time shift<= 9999
0 <=mb<= 10 0<=Ms<= 10 0<=Mw<= 10
0 <=tension plunge<= 90 0 <=null plunge<= 90
Results
Output in CMTSOLUTION format
From Quick CMT catalog
PDE 2010 9 23 5 28 35.00 52.2300 179.8000 161.8 5.6 5.6 RAT ISLANDS, ALEUTIAN IS
event name: 201009230528A
time shift: 2.1400
half duration: 1.4000
latitude: 52.2500
longitude: 179.8200
depth: 170.9600
Mrr: 7.000000e+22
Mtt: -6.210000e+23
Mpp: 5.510000e+23
Mrt: 2.080000e+24
Mrp: -1.270000e+24
Mtp: -4.660000e+23
End of events found with given criteria.
See the CMT project web page for more information on CMTs.
Please email comments.

Input

CMTSOLUTION_POLET_YORBA_LINDA

```
PDE 2002  9  3  7  8 51.00  33.9190 -117.7640   7.0 4.3 4.3 Yorba Linda (JASCHA)
event name:     9818433
time shift:      0.0000
half duration:   1.0000
latitude:       33.9190
longitude:    -117.7640
depth:           7.0000
Mrr:       -0.124000e+23
Mtt:       -0.219000e+23
Mpp:        0.343000e+23
Mrt:       -0.094000e+23
Mrp:       -0.105000e+23
Mtp:       -0.170000e+23
```

CMTSOLUTION_Northridge

```
PDE 2003  7  7 23 59 17.78  34.0745 -118.3792   6.4 4.2 4.2 FICTITIOUS
event name:     9903873
time shift:     0.5447953452769341E+01
half duration:   0
latitude:       34.2438031511517800
longitude:    -118.7368191131799000
depth:       20.4269026324769500
Mrr:     0.3741044075662543E+24
Mtt:    -0.3117105402966140E+24
Mpp:    -0.6239386726964047E+23
Mrt:    -0.8591611017176262E+23
Mrp:    -0.1301384990281119E+23
Mtp:     0.1473147100031584E+24
PDE 2003  7  7 23 59 17.78  34.0745 -118.3792   6.4 4.2 4.2 FICTITIOUS
event name:     9903873
time shift:     0.5020661887758161E+01
half duration:   0
latitude:       34.2373686758751600
longitude:    -118.7239386070113000
depth:       20.4269026324769500
Mrr:     0.4980058767550058E+24
Mtt:    -0.4156140537288186E+24
Mpp:    -0.8319182302618730E+23
Mrt:    -0.1145548135623502E+24
Mrp:    -0.1735179987041493E+23
Mtp:     0.1964196133375445E+24
PDE 2003  7  7 23 59 17.78  34.0745 -118.3792   6.4 4.2 4.2 FICTITIOUS
event name:     9903873
time shift:     0.4599210132417263E+01
half duration:   0
latitude:       34.2309327808731100
longitude:    -118.7110600237053000
depth:       20.4269026324769500
Mrr:     0.6431970516051390E+24
Mtt:    -0.5359233850713713E+24
Mpp:    -0.1072736665337679E+24
Mrt:    -0.1477154174882937E+24
Mrp:    -0.2237468930658767E+23
Mtp:     0.2532779224615705E+24
PDE 2003  7  7 23 59 17.78  34.0745 -118.3792   6.4 4.2 4.2 FICTITIOUS
event name:     9903873
time shift:     0.4185136269172839E+01
half duration:   0
latitude:       34.2244954671005500
longitude:    -118.6981833632119000
depth:       20.4269026324769500
Mrr:     0.3806676427867150E+24
Mtt:    -0.3171791462667300E+24
Mpp:    -0.6348849651998505E+23
Mrt:    -0.8742341035021457E+23
Mrp:    -0.1324216305900087E+23
Mtp:     0.1498991785997050E+24
```

Input

stations:

- FDSN network stations

Input

FDSN: Station Information

http://www.fdsn.org/stations/FDSNstations.htm

news ▾ divers ▾ code ▾ platform ▾ journals ▾ seismology ▾

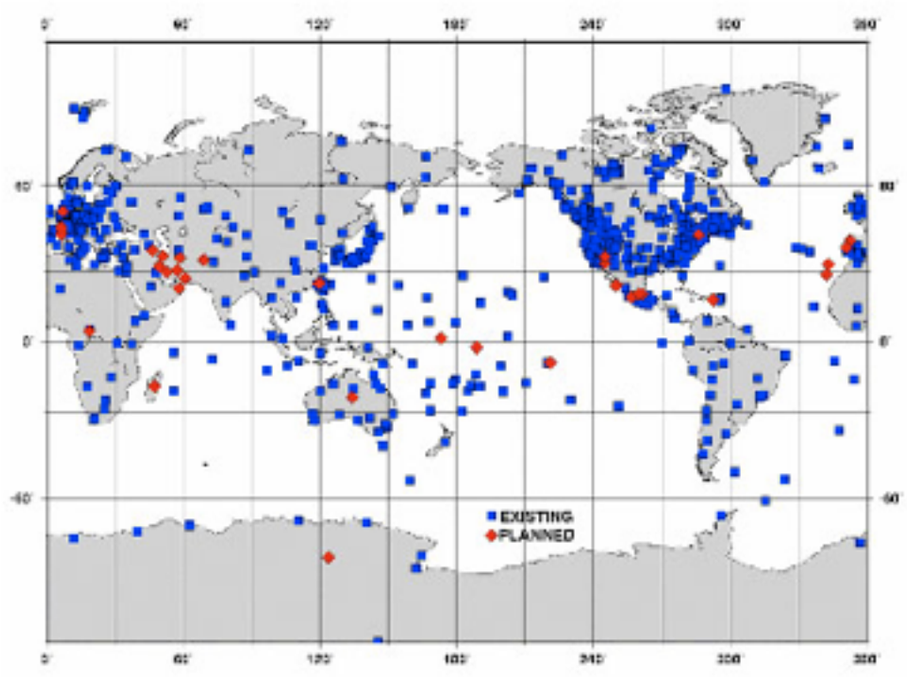
DIGITAL BROADBAND SEISMOGRAPH STATIONS
by Region

search

- North America
- South America & Atlantic Ocean
- Antarctica and Temporary Stations
- Europe
- Asia
- Africa & Indian Ocean
- Australia & Pacific Ocean
- Acronyms and Information

2009 FDSN Network List in Microsoft Excel format

Station Map in Adobe Acrobat (.pdf) format



[click to zoom map]

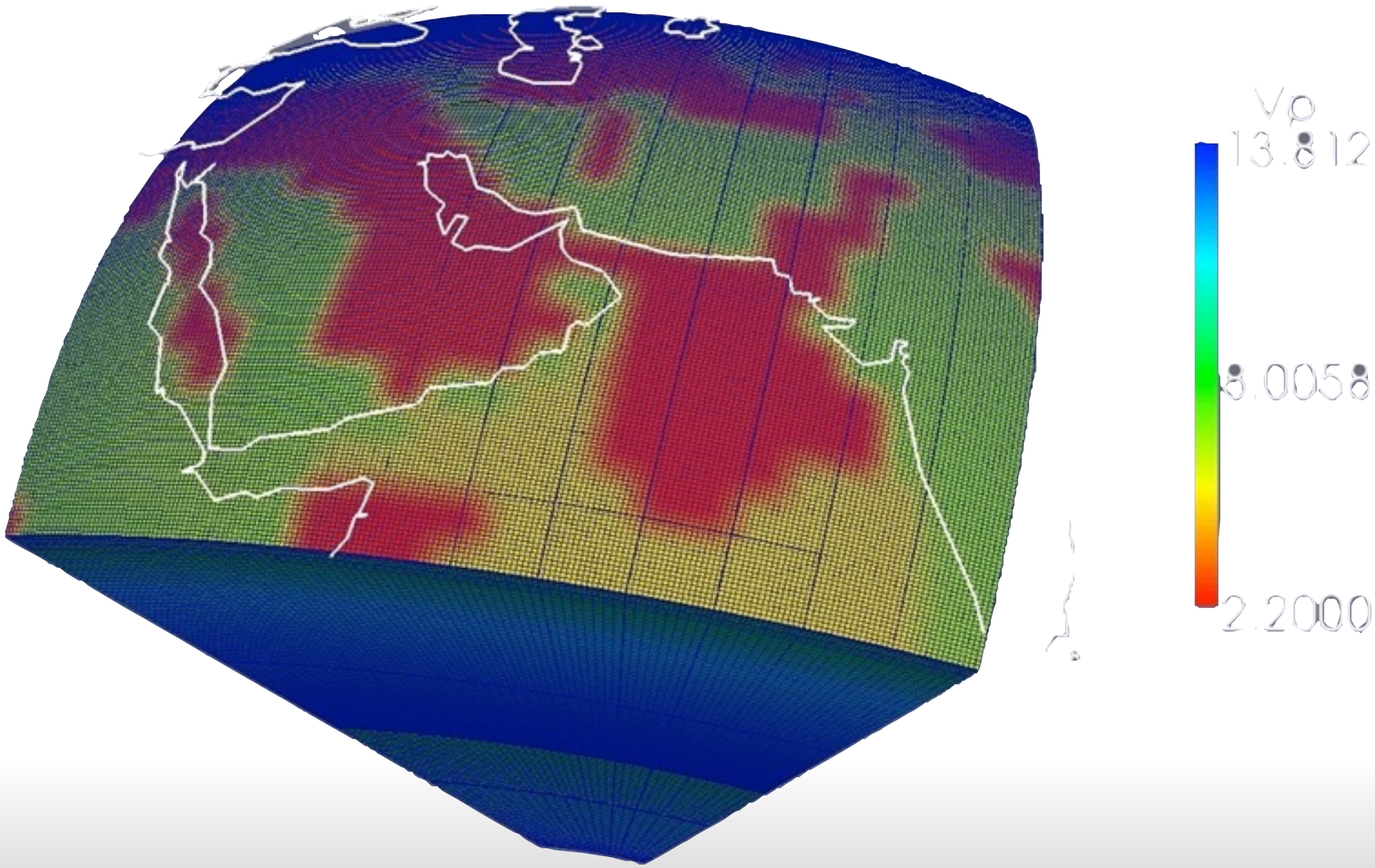
STATIONS_ALL

ASBS	AZ	33.6208	-116.4664	0.0	0.0
BZN	AZ	33.4915	-116.6670	0.0	0.0
CRY	AZ	33.5654	-116.7373	0.0	0.0
ELKS	AZ	33.5813	-116.4496	0.0	0.0
FRD	AZ	33.4947	-116.6022	0.0	0.0
GLAC	AZ	33.6014	-116.4781	0.0	0.0
KNW	AZ	33.7141	-116.7119	0.0	0.0
LVA2	AZ	33.3516	-116.5615	0.0	0.0
MONP	AZ	32.8927	-116.4225	0.0	0.0
PFO	AZ	33.6117	-116.4594	0.0	0.0
RDM	AZ	33.6300	-116.8478	0.0	0.0
SHUM	AZ	33.6327	-116.4445	0.0	0.0
SMTC	AZ	32.9449	-115.7999	0.0	0.0
SND	AZ	33.5519	-116.6129	0.0	0.0
SOL	AZ	32.8410	-117.2480	0.0	0.0
THSB	AZ	32.8788	-117.2269	0.0	0.0
TRO	AZ	33.5234	-116.4257	0.0	0.0
WMC	AZ	33.5736	-116.6747	0.0	0.0
YAQ	AZ	33.1666	-116.3539	0.0	0.0
CMB	BK	38.0346	-120.3865	0.0	0.0
CMB	BK	38.0346	-120.3865	0.0	0.0
CVS	BK	38.3453	-122.4584	0.0	0.0
HOPS	BK	38.9935	-123.0723	0.0	0.0
JCC	BK	40.8175	-124.0296	0.0	0.0
MHC	BK	37.3416	-121.6426	0.0	0.0
ORV	BK	39.5545	-121.5004	0.0	0.0
SAO	BK	36.7640	-121.4472	0.0	0.0
WDC	BK	40.5799	-122.5411	0.0	0.0
YBH	BK	41.7320	-122.7104	0.0	0.0
G405	CE	33.7870	-118.3560	0.0	0.0
J732	CE	34.1900	-117.3300	0.0	0.0
K400	CE	34.0370	-118.1780	0.0	0.0
K851	CE	34.0700	-118.3460	0.0	0.0
K853	CE	34.0770	-118.2860	0.0	0.0
ADO	CI	34.5505	-117.4339	0.0	0.0

Input

regional simulations:

- 1-chunk / 2-chunk run



Input

Par_file

```
# forward or adjoint simulation
SIMULATION_TYPE                 = 1
SAVE_FORWARD                    = .false.  # save last frame of forward simulation or not

# number of chunks (1,2,3 or 6)
NCHUNKS                         = 1

# angular width of the first chunk (not used if full sphere with six chunks)
ANGULAR_WIDTH_XI_IN_DEGREES    = 45.d0      # angular size of a chunk
ANGULAR_WIDTH_ETA_IN_DEGREES   = 40.d0
CENTER_LATITUDE_IN_DEGREES     = 29.d0
CENTER_LONGITUDE_IN_DEGREES    = 55.d0
GAMMA_ROTATION_AZIMUTH         = 0.d0

# number of elements at the surface along the two sides of the first chunk
# (must be multiple of 16 and 8 * multiple of NPROC below)
NEX_XI                          = 480
NEX_ETA                         = 480

# number of MPI processors along the two sides of the first chunk
NPROC_XI                        = 12
NPROC_ETA                       = 12

# 1D models with real structure:
# 1D_isotropic_prem, 1D_transversely_isotropic_prem, 1D_iasp91, 1D_1066a, 1D_ak135, 1D_ref, 1D_ref_iso
#
# 1D models with only one fictitious averaged crustal layer:
# 1D_isotropic_prem_onecrust, 1D_transversely_isotropic_prem_onecrust, 1D_iasp91_onecrust, 1D_1066a_onecrust, 1D_ak135_onecrust
#
# fully 3D models:
# transversely_isotropic_prem_plus_3D_crust_2.0, 3D_anisotropic, 3D_attenuation,
# s20rts, s362ani, s362iso, s362wmani, s362ani_prem, s362ani_3DQ, s362iso_3DQ,
# s29ea, s29ea,sea99_jp3d1994,sea99,jp3d1994,heterogen
MODEL                           = s29ea

# parameters describing the Earth model
OCEANS                          = .true.
ELLIPTICITY                     = .true.
TOPOGRAPHY                      = .true.
GRAVITY                         = .true.
ROTATION                        = .true.
ATTENUATION                     = .true.

# absorbing boundary conditions for a regional simulation
ABSORBING_CONDITIONS            = .true.

# record length in minutes
RECORD_LENGTH_IN_MINUTES        = 22.0d0
```

Input

global simulations:

- 6-chunk run

Input

Par_file

```
# forward or adjoint simulation
SIMULATION_TYPE                 = 1
SAVE_FORWARD                    = .false.  # save last frame of forward simulation or not

# number of chunks (1,2,3 or 6)
NCHUNKS                         = 6

# angular width of the first chunk (not used if full sphere with six chunks)
ANGULAR_WIDTH_XI_IN_DEGREES    = 45.d0      # angular size of a chunk
ANGULAR_WIDTH_ETA_IN_DEGREES   = 40.d0
CENTER_LATITUDE_IN_DEGREES     = 29.d0
CENTER_LONGITUDE_IN_DEGREES    = 55.d0
GAMMA_ROTATION_AZIMUTH         = 0.d0

# number of elements at the surface along the two sides of the first chunk
# (must be multiple of 16 and 8 * multiple of NPROC below)
NEX_XI                          = 480
NEX_ETA                         = 480

# number of MPI processors along the two sides of the first chunk
NPROC_XI                        = 12
NPROC_ETA                       = 12

# 1D models with real structure:
# 1D_isotropic_prem, 1D_transversely_isotropic_prem, 1D_iasp91, 1D_1066a, 1D_ak135, 1D_ref, 1D_ref_iso
#
# 1D models with only one fictitious averaged crustal layer:
# 1D_isotropic_prem_onecrust, 1D_transversely_isotropic_prem_onecrust, 1D_iasp91_onecrust, 1D_1066a_onecrust, 1D_ak135_onecrust
#
# fully 3D models:
# transversely_isotropic_prem_plus_3D_crust_2.0, 3D_anisotropic, 3D_attenuation,
# s20rts, s362ani, s362iso, s362wmani, s362ani_prem, s362ani_3DQ, s362iso_3DQ,
# s29ea, s29ea,sea99_jp3d1994,sea99,jp3d1994,heterogen
MODEL                           = s362ani

# parameters describing the Earth model
OCEANS                          = .true.
ELLIPTICITY                     = .true.
TOPOGRAPHY                      = .true.
GRAVITY                         = .true.
ROTATION                        = .true.
ATTENUATION                     = .true.

# absorbing boundary conditions for a regional simulation
ABSORBING_CONDITIONS            = .false.

# record length in minutes
RECORD_LENGTH_IN_MINUTES        = 22.0d0
```

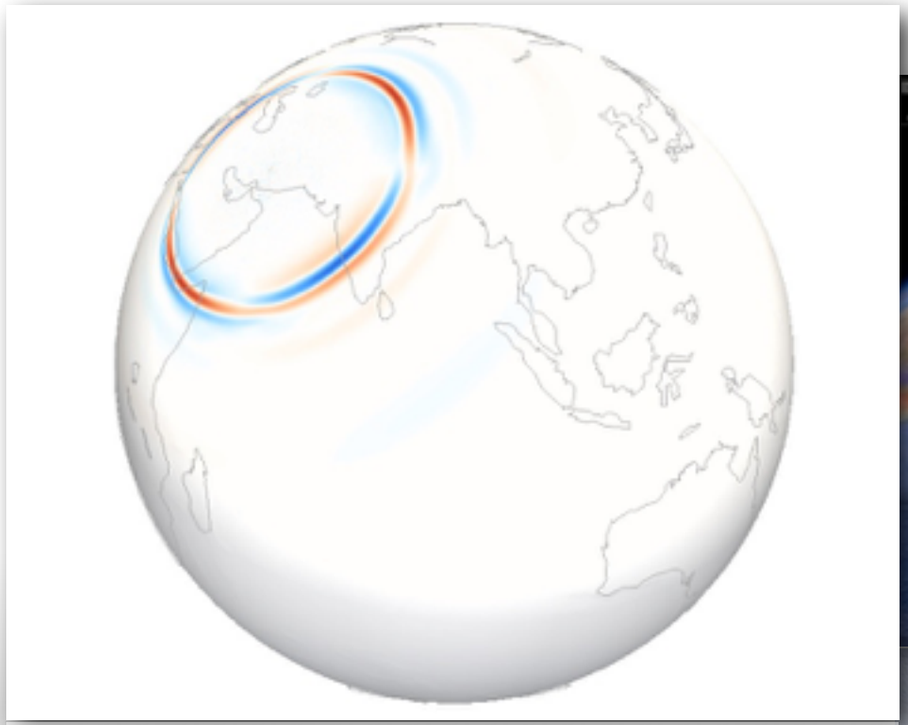
Output

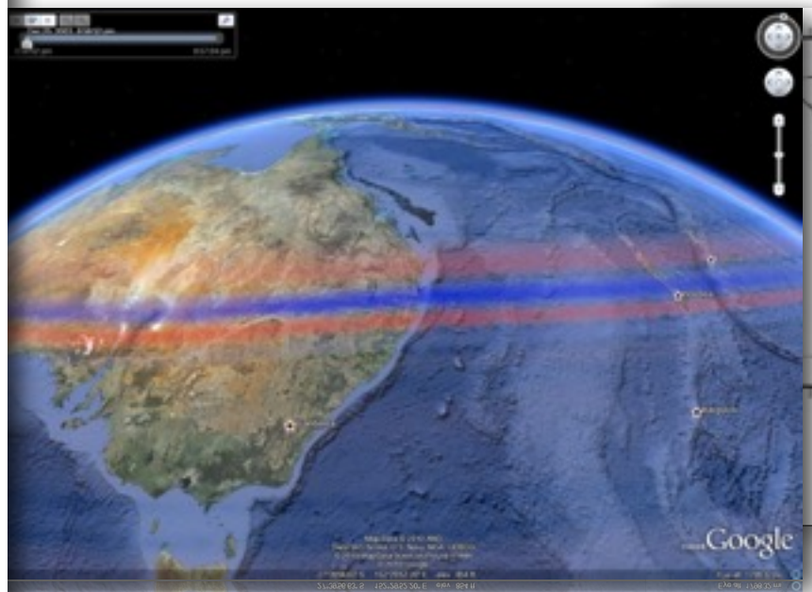
utils/:

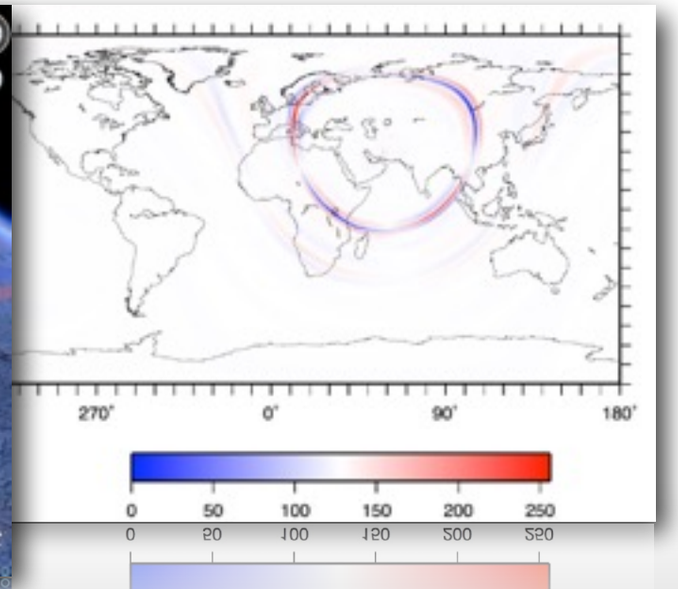
- data & synthetics processing
- visualizations:

  Paraview / VTK, GMT,

  Google Earth...



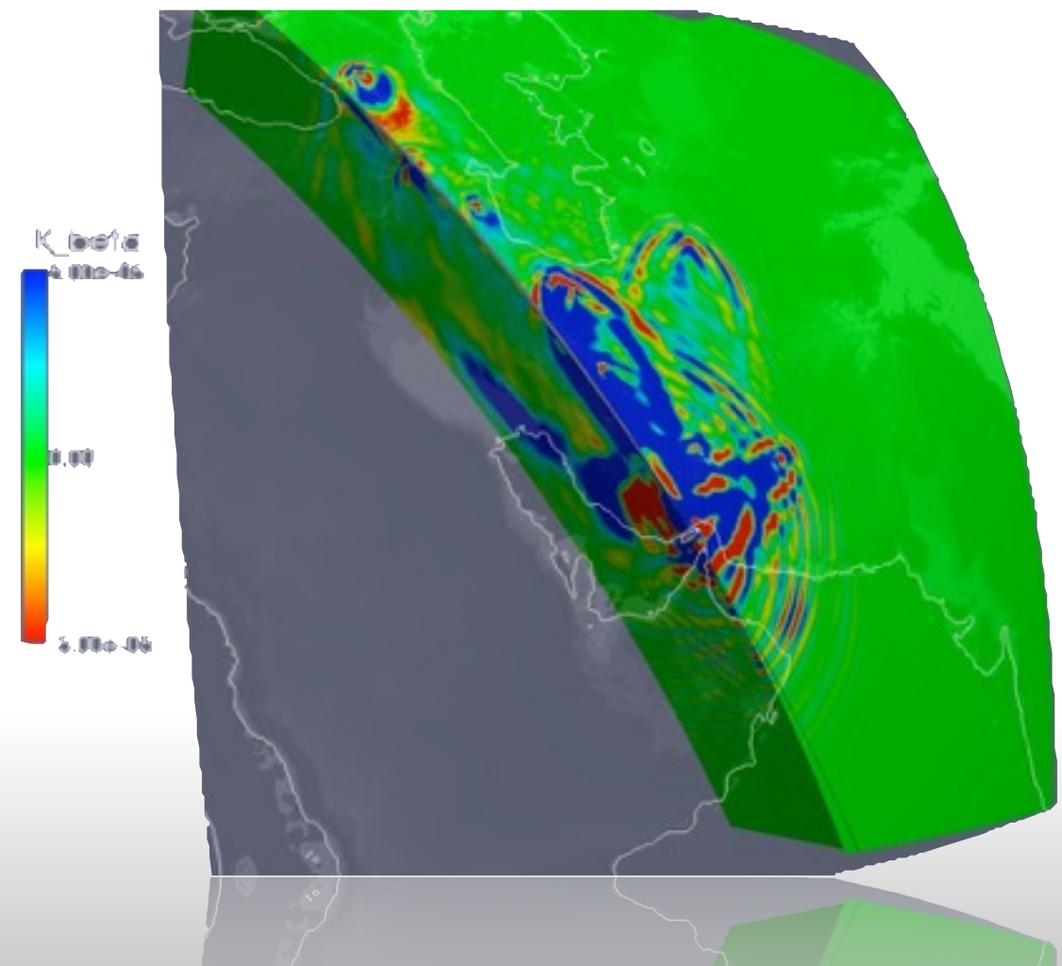




Output

Paraview visualizations:

- kernel run -> proc***alpha_kernel.bin
- combine_vol_data -> reg_1_alpha_kernel.mesh
- mesh2vtu -> reg_1_alpha_kernel.vtu



Outreach

global shakemovie

Modifications

create new crust/mantle model

Modifications

- meshfem3D_models.f90
- model_1dref.f90
- model_1066a.f90
- model_ak135.f90
- model_aniso_inner_core.f90
- model_aniso_mantle.f90
- model_atten3D_QRFSI12.f90
- model_attenuation.f90
- model_crust.f90
- model_crustmaps.f90
- model_eucrust.f90
- model_gll.f90
- model_heterogen_mantle.f90
- model_iasp91.f90
- model_jp1d.f90
- model_jp3d.f90
- model_ppm.f90
- model_prem.f90
- model_s20rts.f90
- model_s40rts.f90
- model_s362ani.f90
- model_sea1d.f90
- model_sea99_s.f90
- model_sb4l18.f90
- model_sb10l18.f90
- model_topo_bathy.f90

fortran code:
search & follow comments

```
!---
!
! ADD YOUR MODEL HERE
!
!---
```

Code optimizations

High-order methods for incompressible fluid flow, M.O. Deville, P.M. Fischer, E.H. Mund, 2002, Cambridge University Press

Software optimizations for high performance computing: creating faster applications, I.L. Crawford & K.R. Wadleigh, 2003, Prentice Hall Professional

Bentley's rules, J. Bentley, http://www.hipecc.wichita.edu/bentley.htm

Code optimizations

matrix-vector multiplications:

```
do j=1,m2
  do i=1,m1
    C1_m1_m2_5points(i,j) = hprime_xx(i,1)*B1_m1_m2_5points(1,j) + &
                           hprime_xx(i,2)*B1_m1_m2_5points(2,j) + &
                           hprime_xx(i,3)*B1_m1_m2_5points(3,j) + &
                           hprime_xx(i,4)*B1_m1_m2_5points(4,j) + &
                           hprime_xx(i,5)*B1_m1_m2_5points(5,j)

    C2_m1_m2_5points(i,j) = hprime_xx(i,1)*B2_m1_m2_5points(1,j) + &
                           hprime_xx(i,2)*B2_m1_m2_5points(2,j) + &
                           hprime_xx(i,3)*B2_m1_m2_5points(3,j) + &
                           hprime_xx(i,4)*B2_m1_m2_5points(4,j) + &
                           hprime_xx(i,5)*B2_m1_m2_5points(5,j)

    C3_m1_m2_5points(i,j) = hprime_xx(i,1)*B3_m1_m2_5points(1,j) + &
                           hprime_xx(i,2)*B3_m1_m2_5points(2,j) + &
                           hprime_xx(i,3)*B3_m1_m2_5points(3,j) + &
                           hprime_xx(i,4)*B3_m1_m2_5points(4,j) + &
                           hprime_xx(i,5)*B3_m1_m2_5points(5,j)
  enddo
enddo
do j=1,m1
  do i=1,m1
    ! for efficiency it is better to leave this loop on k inside, it leads to slightly faster code
    do k = 1,NGLLX
      tempx2(i,j,k) = dummyx_loc(i,1,k)*hprime_xxT(1,j) + &
                      dummyx_loc(i,2,k)*hprime_xxT(2,j) + &
                      dummyx_loc(i,3,k)*hprime_xxT(3,j) + &
                      dummyx_loc(i,4,k)*hprime_xxT(4,j) + &
                      dummyx_loc(i,5,k)*hprime_xxT(5,j)
```

compute_forces_crust_mantle_Dev.f90

Code optimizations

do-loops:

```
! way 1:
!           do i_sls = 1,N_SLS
!             R_xx_val = R_memory(1,i_sls,i,j,k,ispec)
!             R_yy_val = R_memory(2,i_sls,i,j,k,ispec)
!             sigma_xx = sigma_xx - R_xx_val
!             sigma_yy = sigma_yy - R_yy_val
!             sigma_zz = sigma_zz + R_xx_val + R_yy_val
!             sigma_xy = sigma_xy - R_memory(3,i_sls,i,j,k,ispec)
!             sigma_xz = sigma_xz - R_memory(4,i_sls,i,j,k,ispec)
!             sigma_yz = sigma_yz - R_memory(5,i_sls,i,j,k,ispec)
!           enddo

! way 2:
! note: this should help compilers to pipeline the code and make better use of the cache;
!       depending on compilers, it can further decrease the computation time by ~ 30%.
!       by default, N_SLS = 3, therefor we take steps of 3
          do i_sls = 1,mod(N_SLS,3)
            R_xx_val1 = R_memory(1,i_sls,i,j,k,ispec)
            R_yy_val1 = R_memory(2,i_sls,i,j,k,ispec)
            sigma_xx = sigma_xx - R_xx_val1
            sigma_yy = sigma_yy - R_yy_val1
            sigma_zz = sigma_zz + R_xx_val1 + R_yy_val1
            sigma_xy = sigma_xy - R_memory(3,i_sls,i,j,k,ispec)
            sigma_xz = sigma_xz - R_memory(4,i_sls,i,j,k,ispec)
            sigma_yz = sigma_yz - R_memory(5,i_sls,i,j,k,ispec)
          enddo

          do i_sls = mod(N_SLS,3)+1,N_SLS,3
            R_xx_val1 = R_memory(1,i_sls,i,j,k,ispec)
            R_yy_val1 = R_memory(2,i_sls,i,j,k,ispec)
```

compute_forces_crust_mantle_Dev.f90

Code optimizations

file I/O:

```
/* fastest performance on nehalem nodes:

Linux 2.6.18-164.11.1.el5 #1 SMP Wed Jan 20 10:04:55 EST 2010 x86_64 x86_64 x86_64 GNU/Linux

achieved with 16 KB buffers: */

//#define MAX_B 65536 // 64 KB
//#define MAX_B 32768 // 32 KB
#define MAX_B 16384 // 16 KB
//#define MAX_B 8192 // 8 KB

// absorbing files: instead of passing file descriptor, we use the array index
//                  first 0 - 3 indices for crust mantle files
//                  last 4 - 8 indices for outer core files
#define ABS_FILEID 9

// file points
static FILE * fp_abs[ABS_FILEID];
// file work buffers
static char * work_buffer[ABS_FILEID];


//void
//FC_FUNC_(open_file_abs_r_fbin,OPEN_FILE_ABS_R_FBIN)(int *fid, char *filename,int *length, int *filesize){
void open_file_abs_r_fbin(int *fid, char *filename,int *length, int *filesize){

// opens file for read access

//This sequence assigns the MAX_B array work_buffer to the file pointer
// to be used for its buffering. performance should benefit.
  char * fncopy;
  char * blank;
  FILE *ft;

  // checks filesize
  if( *filesize == 0 ){
    perror("Error file size for reading");
    exit(EXIT_FAILURE);
  }
```

write_c_binary.c

More details

CIG - https://geodynamics.org/cig/software/specfem3d/

CIG wiki - https://wiki.geodynamics.org/software:specfem3d:start

github - https://github.com/geodynamics/specfem3d

github wiki - https://github.com/geodynamics/specfem3d/wiki