

# Table of Contents

<b>Bern 2.5D Model.....</b>	<b>1</b>
Source code.....	1
Getting started.....	1
Compile.....	2
Scripts.....	2
Run test case.....	2
Run a spinup case.....	3
Run a transient case.....	3
Run a spinup run followed by a transient run.....	3
Modify startfiles.....	3
Run thc with modified startfiles.....	3
Run a transient thc with on other forcing file.....	3
How the option -set defines the name of the startfile and the case name.....	4
More options.....	4
Create only the case directory.....	4
List thc related files.....	4
Run thc on brutus batch system.....	5
Run several thc in parallel on brutus.....	5
Setup for a crash.....	5
Benchmarks.....	6
Directory Layout.....	7
Debugging the model.....	7
(old description for original code) Compile and run on firebolt.....	8

# Bern 2.5D Model

-  How to run several thc-runs in parallel  [ProjectBern25Dparallel](#)
-  Bugs, limitations, ToDo's  [ProjectBern25Dbugs](#)

## Source code

- SVN repository <https://svn.iac.ethz.ch/repos/projects/bern2.5d> (classic view)
- Or browse the source code <https://svn.iac.ethz.ch/websvn/repos/wsvn/projects.bern2.5d> (nice view)
- Check out **latest version (trunk)** - the trunk is work in progress!

```
svn co https://svn.iac.ethz.ch/repos/projects/bern2.5d/trunk bern2.5d-trunk
```

- If you have once checked out the trunk version you can always **update your working copy** of the trunk to the latest trunk version on the svn server by executing inside your trunk folder `bern2.5d-trunk` the command

```
svn update
```

- View **change log** of latest version (trunk)  
<https://svn.iac.ethz.ch/websvn/repos/wsvn/projects.bern2.5d/?op=log&rev=0&isdir=1> or

```
svn log
```

- Check out version 1.0, original code from Kasper Plattner, compiles with gfortran, pgf90, ifort



```
svn co https://svn.iac.ethz.ch/repos/projects/bern2.5d/tags/1.0 bern2.5d-1.0
```

- Check out original code from Kasper Plattner

```
svn co https://svn.iac.ethz.ch/repos/projects/bern2.5d/tags/original_plattner_code b
```

or

```
svn export https://svn.iac.ethz.ch/repos/projects/bern2.5d/tar/bern2.5d_linux_knutt
```

-  For more information about how to work with subversion and the `svn` command see  [ServiceSvn](#).

## Getting started


- Checkout latest version

```
svn co https://svn.iac.ethz.ch/repos/projects/bern2.5d/trunk bern2.5d-trunk
```

- Read the provided readme files

```
cd bern2.5d-trunk
less README.txt
less COMPILER.txt
less MACHINES.txt
```

# Compile

-  See also **COMPILER.txt** and **MACHINES.txt**
- Define the following environmental variables. This can be for example done by **loading modules**, see below. Or with the `export` (bash) or `setenv` (tcsh) command.
  - ♦ **FC** : Fortran Compiler
  - ♦ **NETCDF** : path to your netcdf installation, libnetcdf.a should be in \$NETCDF/lib, include files in \$NETCDF/include
  - ♦ **FFLAGS** : additional compiler flags, for example set `FFLAGS=-g` for debugging
- On IAC systems you can use modules to set these variables correctly
- Environment for **pgf90**

```
module load pgi netcdf/3.6.3-pgf90
```

- Environment for **ifort**

```
module load ifort netcdf/3.6.3-ifort
```

- Environment for **gfortran**

```
module load gfortran netcdf/3.6.3
```

- Compile

```
make clean
make
```

# Scripts

- List of scripts

Scriptname	Purpose
thc.sh	<b>Run the Bern 2.5 D model</b>
thc_run.sh	Run a spinup run or transient run
create_case.sh	Create a spinup or transient case
create_restart_file.sh	Create a restart file
list_files.sh	List thc related files
edit_files.sh	Edit and show variables in spinup and transient startfiles
crestart.sh	Simple wrapper script to run crestart (obsolete)

- All scripts come with an internal **help**, just run the script with option `-h` or `-help`, for example

```
./list_files.sh -h
```

# Run test case

- Run the test case model:

```
./thc.sh -s test
```

## Run a spinup case

- Run default susciar4 spinup model :

```
./thc.sh -s susciar4_default
```

Note: susciar4\_default is equal to susciar4\_knum1\_iadv1

## Run a transient case

- Run transient model (the spinup run has to be done in advance):

```
./thc.sh -r susciar4_default -t ar4_sres_a2_ar4_default
```

In this case the transient run (-t) ar4\_sres\_a2\_ar4\_default is restarted (-r) from the spinup run susciar4\_default

## Run a spinup run followed by a transient run

- Do a spinup run followed by a transient run

```
./thc.sh -s susciar4_default -t ar4_sres_a2_ar4_default
```

## Modify startfiles

- See `edit_files.sh` script

```
./edit_files.sh -h
```

## Run thc with modified startfiles

- You can modify the startfiles within the thc.sh command

```
./thc.sh -set kvnum=1,khnum=1,iadv=1 -s susciar4
```

This will set kvnum=1, khnum=1 and iadv=1. Note: This is equal to

```
./thc.sh -s susciar4_default
```

## Run a transient thc with on other forcing file

- Place your forcing file in the folder **forcing**
- The name and path of the forcing file for a transient run is defined in the transient startup file in a line similar to

```
'forcing/start_ar4_sres_a2_ar4.dat' forcing (chfile_indus)
```

- Use option **-set** to change the name of the forcing file. **IMPORTANT:** Use a backslash \ to escape the ' sign !

```
./thc.sh -set forcing="\forcing/file1.dat\" -r susciar4_default -t ar4_sres_a2_ar4
```

## How the option **-set** defines the name of the startfile and the case name

- If you modify the startfiles with the option **-set**, a MD5-string will be appended to the name of startfile and therefore to the name of the new case.
- The MD5-string is calculate from the parameters given by the **-set** option. Run the script `./md5.sh` to find out the MD5 sum. For example

```
./md5.sh kvnum=1,khnum=1,iadv=1
```

- Note that the MD5 sum is independent of the order of the parameters

```
./md5.sh kvnum=1,khnum=1,iadv=1
MD5c3478ce6723a70b7fb3a4c2e57c97737
./md5.sh iadv=1,kvnum=1,khnum=1
MD5c3478ce6723a70b7fb3a4c2e57c97737
```

- If you don't want to choose the string which is appended, use the option **-add**, for example

```
./thc.sh -set kvnum=1,khnum=1,iadv=1 -add setup1 -s susciar4
```

This will create the case `susciar4_setup1.spinup` instead of `susciar4_MD5c3478ce6723a70b7fb3a4c2e57c97737.spinup`

## More options

- If you want to re-do a run, use the option **-f** (force) to overwrite everything

```
./thc.sh -f -s susciar4_default
```

- Run thc with nice 19 (**-n 19**)

```
./thc.sh -n 19 -s susciar4_default
```

- Run thc quiet (**-q**) - with no output

```
./thc.sh -q -s susciar4_default
```

- See all the available option of thc.sh by running

```
./thc.sh -h
```

## Create only the case directory

- Create case directory for a spinup case

```
./create_case.sh -s susciar4_default
```

- Create case directory for a transient case

```
./create_case.sh -r susciar4_default -t ar4_sres_a2_ar4_default
```

## List thc related files

- List available spinup start files

```
./list_files.sh -ls
```

- List available restart files (**-lr**) and available transient (**-lt**) start files

```
./list_files.sh -lr -lt
```

## Run thc on brutus batch system

- Compile thc, see also **MACHINES.txt**

```
module purge
module load pgi netcdf/4.0.1           # in case you want to use Portland compiler
module load intel netcdf/4.0.1        # in case you want to use Intel compiler
export FC=gfortran; module load netcdf/4.0.1 # in case you want to use GNU gfortran
make clean
make
```

- Create a case, for example for spinup susciar4\_default

```
./create_case.sh -s susciar4_default
```

- Submit batch job

```
bsub < ./cases/susciar4_default.spinup/susciar4_default.spinup.lsf
```

- Afterwards LSF log files are in the case folder, for example

```
cases/susciar4_default.spinup/susciar4_default.spinup-out.JOBID
cases/susciar4_default.spinup/susciar4_default.spinup-err.JOBID
```

## Run several thc in parallel on brutus

- Use the LSF jobfile **par\_thc.lsf** and the script **par\_thc.sh** to distribute thc-runs over several nodes. For more info see [ProjectBern25Dparallel](#)

## Setup for a crash

- Run spinup case susciar4

```
./thc.sh -s susciar4
```

- **1. Case:** Run transient case sres\_a2\_BernCC\_targwfb2.5progipccar4jan09

```
./thc.sh -r susciar4 -t sres_a2_BernCC_targwfb2.5progipccar4jan09
```

```
Time: 21021.2 yr      dt:  20.3 d      dta:  16.8 h
./thc.sh: line 167: 2139 Floating point exception$THC $ex_trans $ex_restart
ERROR: thc exit with an error.
```

- **2. Case:** Run transient case sres\_a2\_test

```
./thc.sh -r susciar4 -t sres_a2_test
```

```
Time: 20326.2 yr      dt:  20.3 d      dta:  16.8 h
./thc.sh: line 167: 2241 Floating point exception$THC $ex_trans $ex_restart
ERROR: thc exit with an error.
```

- **3. Case:** Run transient case ar4\_sres\_a2\_ar4\_gwfb\_3.2

```
./thc.sh -r susciar4 -t ar4_sres_a2_ar4_gwfb_3.2
```

```
Time: 20310.2 yr      dt:  20.3 d      dta:  16.8 h
./thc.sh: line 169: 25445 Floating point exception$THC $ex_trans $ex_restart
ERROR: thc exit with an error.
```

- **The crash can be avoided**, if you start with `spinup susciar4_iadv1`, `susciar4_knum1` or `susciar4_knum1_iadv1` instead of `susciar4` !

- **Crash Matrix**

Spinup	Transient	Runs without crash
susciar4	sres_a2_BernCC_targwfb2.5progipccar4jan09	😞
susciar4	sres_a2_test	😞
susciar4	ar4_sres_a2_ar4_gwfb_3.2	😞
susciar4_iadv1	sres_a2_BernCC_targwfb2.5progipccar4jan09	😄
susciar4_iadv1	sres_a2_test	😄
susciar4_iadv1	ar4_sres_a2_ar4_gwfb_3.2	😄
susciar4_knum1	sres_a2_BernCC_targwfb2.5progipccar4jan09	😄
susciar4_knum1	sres_a2_test	😄
susciar4_knum1	ar4_sres_a2_ar4_gwfb_3.2	😄
susciar4_knum1_iadv1	sres_a2_BernCC_targwfb2.5progipccar4jan09	😄
susciar4_knum1_iadv1	sres_a2_test	😄
susciar4_knum1_iadv1	ar4_sres_a2_ar4_gwfb_3.2	😄

## Benchmarks

- Summary: **Best performance** reached with **intel compiler**
- Single the run. **Minimum time** of

```
time ./thc.sh -f -s susciar4 >/dev/null
```

Compiler	Flags	System	Time
pgf90 9.0-1	-O3 -Ktrap=fp	brutus3 (login node)	14m 59s
ifort 10.1.018	-O3 -fpe0	brutus3 (login node)	10m 01s
ifort 10.1	-O3 -fpe0	firebolt	18m 26s
pgf90 7.2-3	-O3 -Ktrap=fp	firebolt	20m 10s
gfortran 4.1.2	-O3 -ffpe-trap=invalid,zero,overflow -fno-automatic	firebolt	21m 39s
ifort 10.1	-O3 -fpe0	fluffy	8m 19s
pgf90 7.2-3	-O3 -Ktrap=fp	fluffy	9m 25s
gfortran 4.3.2	-O3 -ffpe-trap=invalid,zero,overflow -fno-automatic	fluffy	11m 31s
ifort 10.1	-O3 -fpe0	i5-3570 3.40GHz	4m 15s
ifort 10.1	-O3 -fpe0	i7-2600 3.40GHz	4m 56s
ifort 10.1	-O3 -fpe0	iacdipl-2	11m 50s
pgf90 7.2-3	-O3 -Ktrap=fp	iacdipl-2	16m 19s
gfortran 4.3.2	-O3 -ffpe-trap=invalid,zero,overflow -fno-automatic	iacdipl-2	15m 09s
gfortran 4.8.1	-O3 -ffpe-trap=invalid,zero,overflow -fno-automatic	rasperypi	270m 28s
ifort 10.1	-O3 -fpe0	Xeon E3-1245 v3 3.40GHz (FS W540)	3m 57s 🍌
ifort 10.1	-O3 -fpe0	Xeon E3-1275 3.50 GHz	4m 08s
ifort 13.1.3	-O3 -fpe0	Xeon E5-2650 v2 2.60GHz	4m 38s
ifort 13.1.3	-O3 -fpe0	Xeon E5-2660 v2 2.20GHz	5m 17s

Compiler	Flags	System	Time
ifort 13.1.3	-O3 -fpe0	Xeon E5-2670 v2 2.50GHz	4m 46s
ifort 13.1.3	-O3 -fpe0	Xeon E5-2680 v2 2.80GHz	4m 23s
ifort 10.1	-O3 -fpe0	Xeon E5-2690 2.90GHz (kryo)	6m 30s
ifort 13.1.3	-O3 -fpe0	Xeon E5-2697 v2 2.70GHz	5m 32s
ifort 10.1	-O3 -fpe0	Xeon X5690 3.47GHz (atmos)	6m 02s

#### • Benchmarks of 400 parallel runs

```
./thc.sh -set iadv=1 -add iadv_1 -s susciar4
for i in $( seq 0 399 ); do
    var=$( printf "%03d\n" $i)
    echo "./thc.sh -f -set gwfb=2.$var -r susciar4_iadv_1 -t ar4_sres_a2_ar4"
done > joblist; echo END >> joblist
time ./par_thc.sh -g 2
```

Compiler	Flags	System	cores	Time	Time * core
ifort 10.1	-O3 -fpe0	fluffy	8	8m 27s	68m 🍌
ifort 10.1	-O3 -fpe0	iacdipl-2	16	6m 47s	109m
ifort 10.1	-O3 -fpe0	firebolt	8	21m 0s	168m

## Directory Layout

```
|-- cdf_2d                (source code of cdf_2d.a)
|-- src                  (source code of thc)
|-- thc.sh               (script to run thc *** main script ***)
|-- thc_run.sh           (script to run spinup and transient cases)
|-- create_case.sh       (script to create a new case)
|-- create_restart_file.sh (script to create a restart file)
|-- list_files.sh        (script to list thc related files)
|-- edit_files.sh        (edit/show starfiles)
|-- crestart.sh          (points to old src/crestart)
|-- data                 (directory of data files, ???)
|-- forcing              (directory of data files, forcing)
|-- start_files          (directory of start files)
|-- cases                (directory containing runs/cases)
|   |-- susciar4.spinup  (case directory for susciar4 case, spinup run)
|   |   |-- input        (input data)
|   |   |-- output       (output data)
|   |-- susciar4.sres_a2_test (case directory for transient run sres_a2_test,
|   ...                 spinup from susciar4 run)
|
|-- batch-template.lsf    (template for batch job script)
|-- COMPILER.txt         (read how to compile the model)
|-- MACHINES.txt         (machine specific documentation)
|-- README.txt           (general readme file)
`-- Makefile              (main Makefile)
```

## Debugging the model

- For debugging pgf90 compiled code, use Portland Group Debugger, see </usr/local/pgi/linux86/7.2-3/doc/pgi72tools.pdf>
- or <http://www.pgroup.com/doc/pgitools.pdf>
- Load the compiler and corresponding the netcdf library, if not yet done

```
module load pgi netcdf/3.6.3-pgf90
```



- Compile with option `-g`

```
FFLAGS="-g"
make clean
make
```

- Run `thc` inside `pgdbg`

```
pgdbg src/thc
```

- Press **Run** inside the `pgdbg` gui

## (old description for original code) Compile and run on firebolt

 ProjectBern25Dold.

---





*Access*

- Set `DENYTOPICVIEW =`

Edit | Attach | Print version | History: *%REVISIONS%* | Backlinks | Raw View | More topic actions  
 Topic revision: r59 - 14 Feb 2014 - 14:24:15 - UrsBeyerle

- ☐ Climphys

- Log In

-  **Climphys Web**
-  Create New Topic
-  Index
-  Search
-  Changes
-  Notifications
-  RSS Feed
-  Statistics
-  Preferences
- **Webs**
- ☐ Public
- ☐ System

- 
- 



Copyright © by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding Wiki? Send feedback