Demonstration of the Integration of BOUT++ into the OMFIT Framework

by

T.M. Wilks[1]

with

O. Meneghini[2], S. Smith[2], C. Moynihan[4], O. Izacard[3], X. Q. Xu[5], J. W. Hughes[1]

[1]MIT-PSFC
[2]General Atomics
[3]PPPL
[4]UIUC
[5]LLNL

Presented at the
2018 BOUT++ Workshop
Livermore, CA
August 14, 2018

Overview of Tutorial

- OMFIT overview
- OMFIT setup options
- BOUT++ module overview
- Grid generation with Hypnotoad
- Running BOUT++
- Post-processing tools

Overview of Tutorial

- OMFIT overview
- OMFIT setup options
- BOUT++ module overview
- Grid generation with Hypnotoad
- Running BOUT++
- Post-processing tools

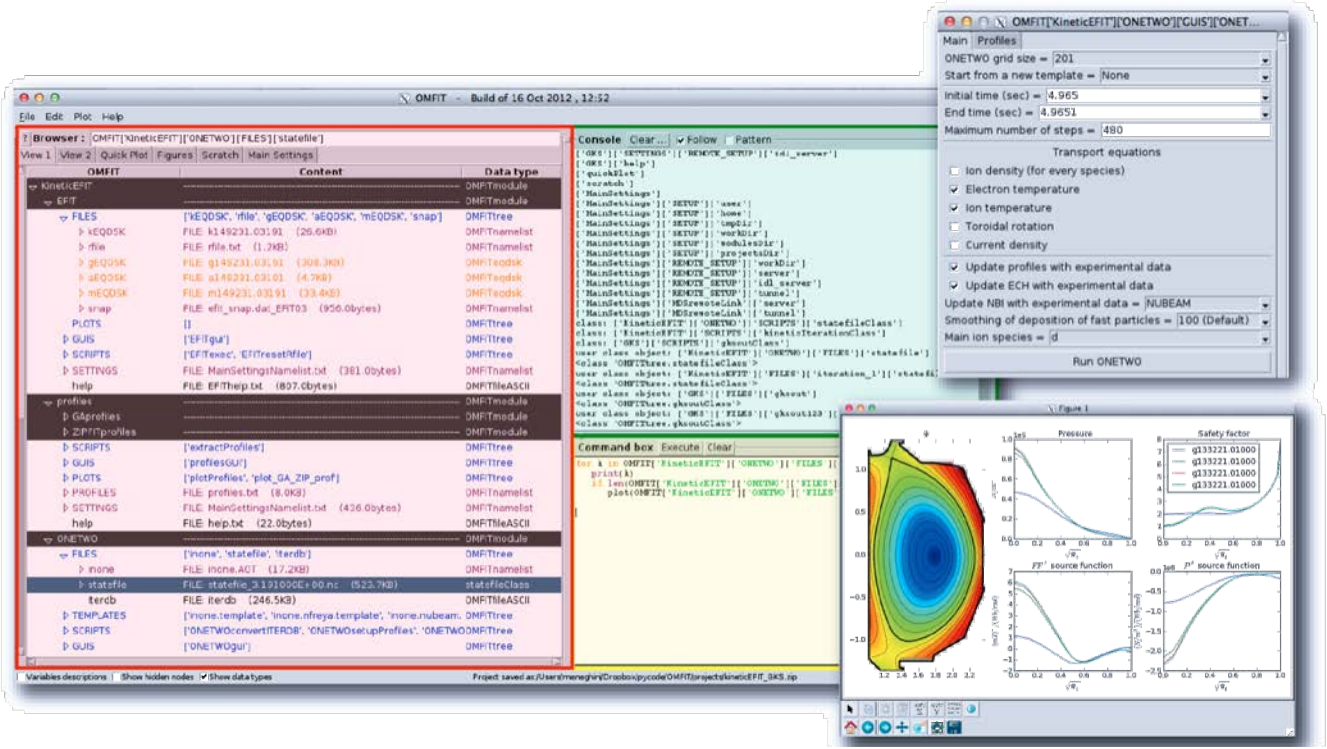
Active Participation

- ✓ NERSC user ID and password (train##)
- ✓ NoMachine or some Xming software

Overview of Tutorial

- OMFIT overview
- OMFIT setup options
- BOUT++ module overview
- Grid generation with Hypnotoad
- Running BOUT++
- Post-processing tools

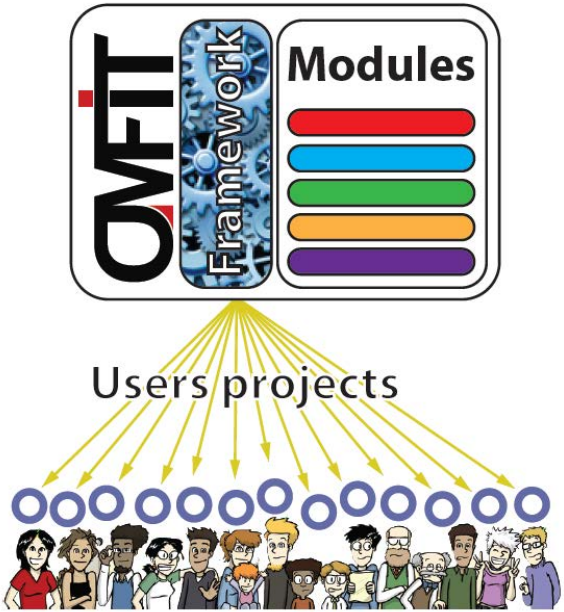
OMFIT - One Modeling Framework for Integrated Tasks



O. Meneghini, S. Smith, et. al. Nuclear Fusion 55, 083008 (2015)

Separation of framework, modules, and projects allows each to evolve independently

- Framework
  - Physics independent API to manage data and run codes
  - Defines common set of functionality and capabilities
- Physics module
  - Collection of scripts for the execution of physics codes and workflows
  - Re-used for different physics studies
- User project
  - Collection of modules and data for specific physics studies



Overview of Tutorial

- OMFIT overview
- OMFIT setup options
- BOUT++ module overview
- Grid generation with Hypnotoad
- Running BOUT++
- Post-processing tools

Launch OMFIT on NERSC (Edison)

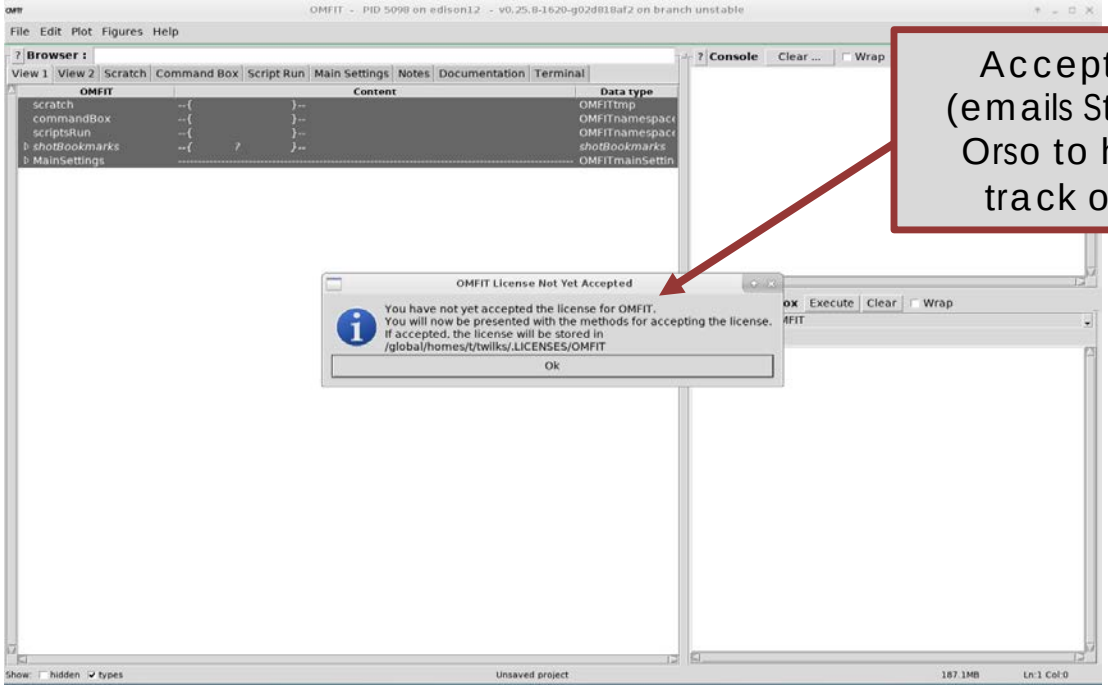
>> cd /global/project/projectdirs/atom/atom-install-edison/bin
>> ./omfit-dev

```
train41@edison08:~> cd /global/project/projectdirs/atom/atom-install-edison/bin/
train41@edison08:/global/project/projectdirs/atom/atom-install-edison/bin> ./omfit
This OMFIT installation assumes that you have private-public key
authentication set up by having generated an rsa key
(private key in ~/.ssh/id_rsa and public key in ~/.ssh/id_rsa.pub)
---
Because your current configuration does not satisfy this requirement
you will now be guided through that process.
When you are asked for a passphrase in the second question, please use
one.  Each time you start omfit, you might be asked for this password
to start the ssh-agent in the background (if it is not already
running), and you will not be asked for it again during that OMFIT
session.
Press <enter> to continue.
```

- Use suggested file location and id_rsa filename
- Press “enter” three times to leave password blank and continue launching OMFIT

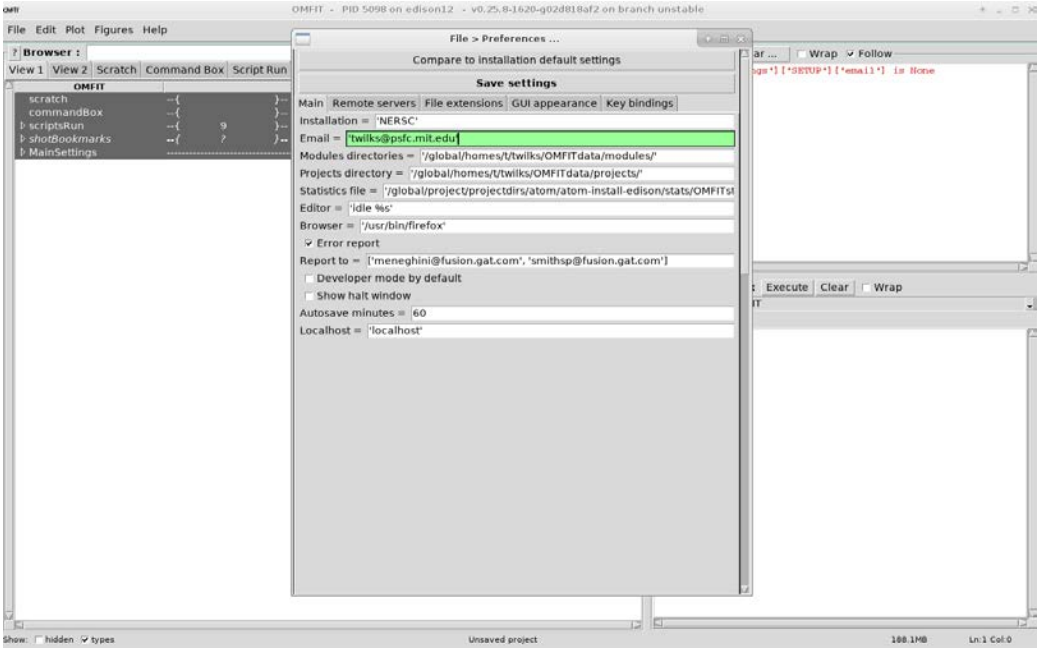
```
Running: ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/global/homes/t/train41/.ssh/id_rsa):
```

Welcome to OMFIT



Accept license: (emails Sterling and Orso to help keep track of usage)

Add your email and user names into preferences



This helps connect you to:

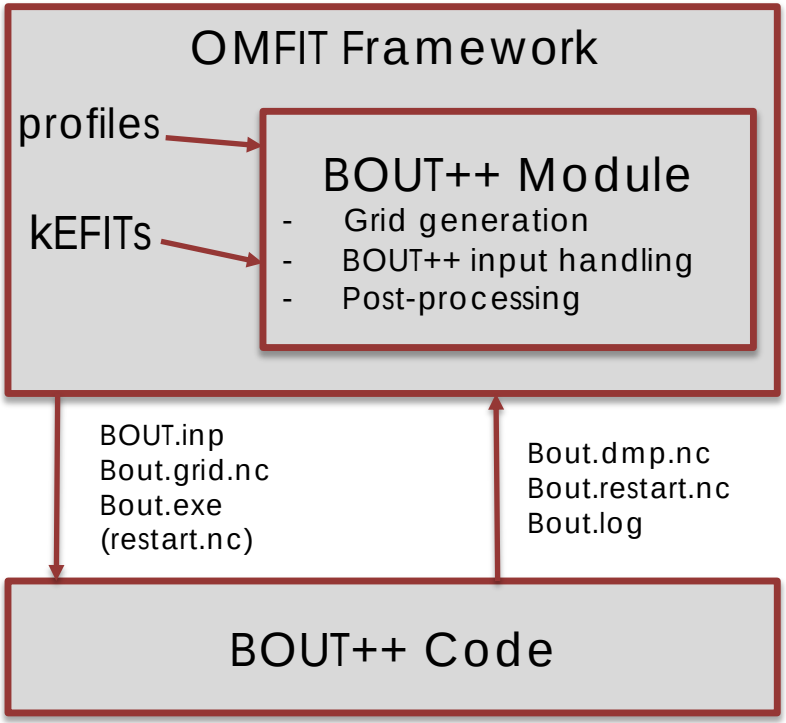
- the OMFIT community for help and error reports
- Other accounts like github and remote servers

Overview of Tutorial

- OMFIT overview
- OMFIT setup options
- BOUT++ module overview
- Grid generation with Hypnotoad
- Running BOUT++
- Post-processing tools

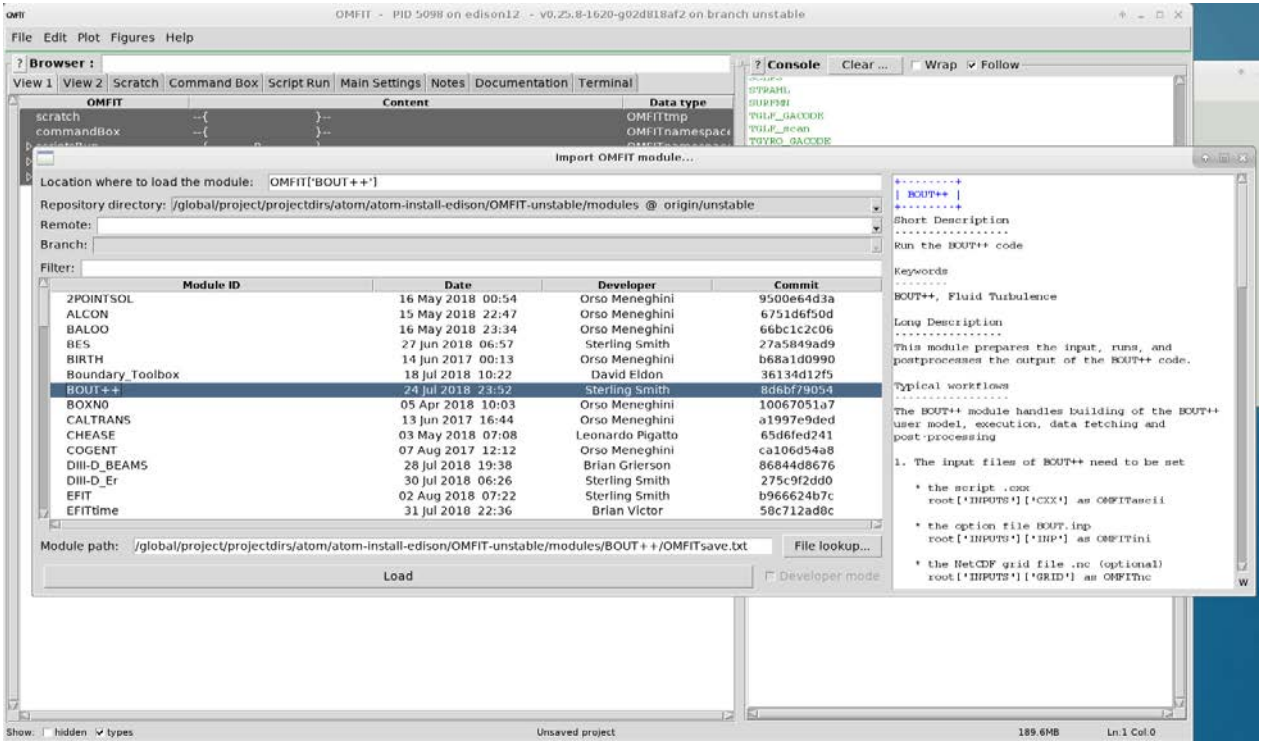
OMFIT allows easy setup and data management for BOUT++

- Goal is to reduce the learning curve for running complex codes like BOUT++
  - OMFIT handles details like input file setup, remote server connections, and data management
  - Allows user to *focus on physics* inputs and results and minimize user errors in setup



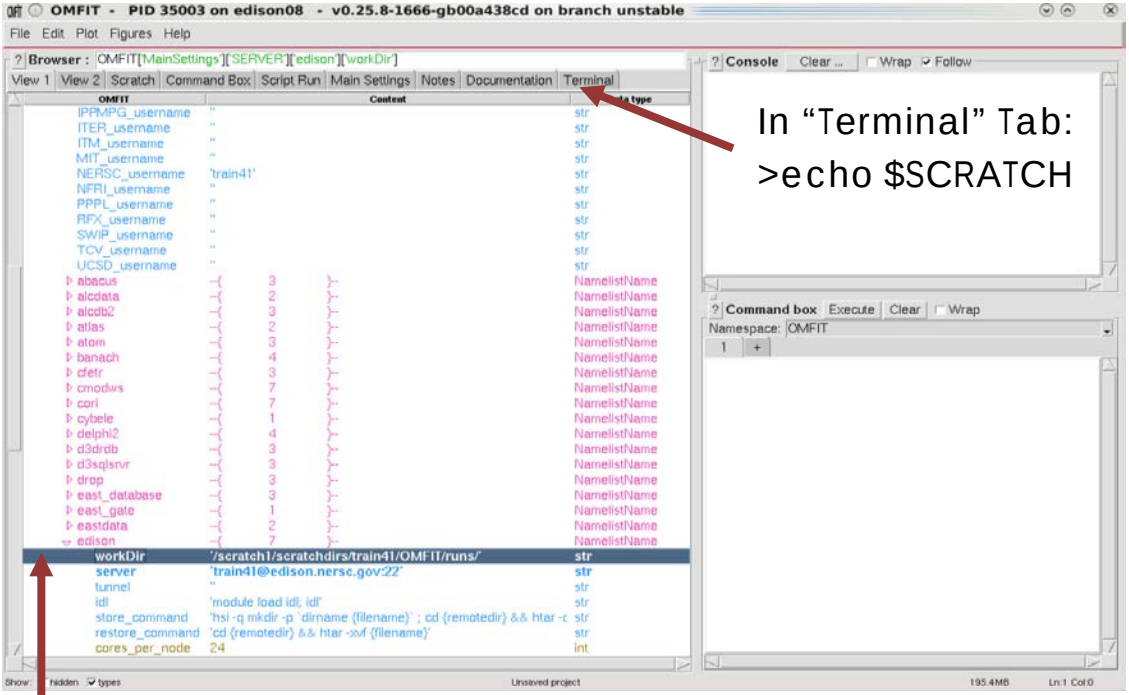
Server independent

Import BOUT++ Module



File→ Import module …

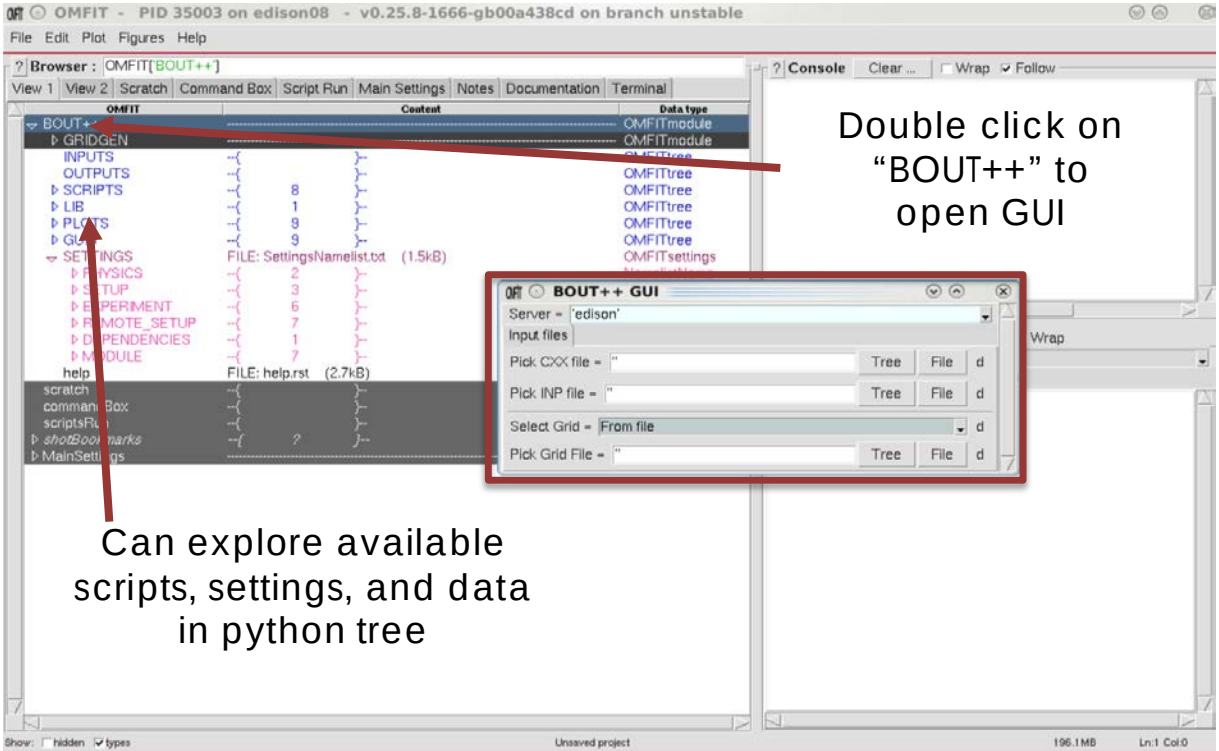
Double Check Scratch Folder is Set Correctly



OMFIT['MainSettings']['SERVER']['edison']['workdir'] = '/scratch#/scratchdirs/username/....

Right click → edit tree entry

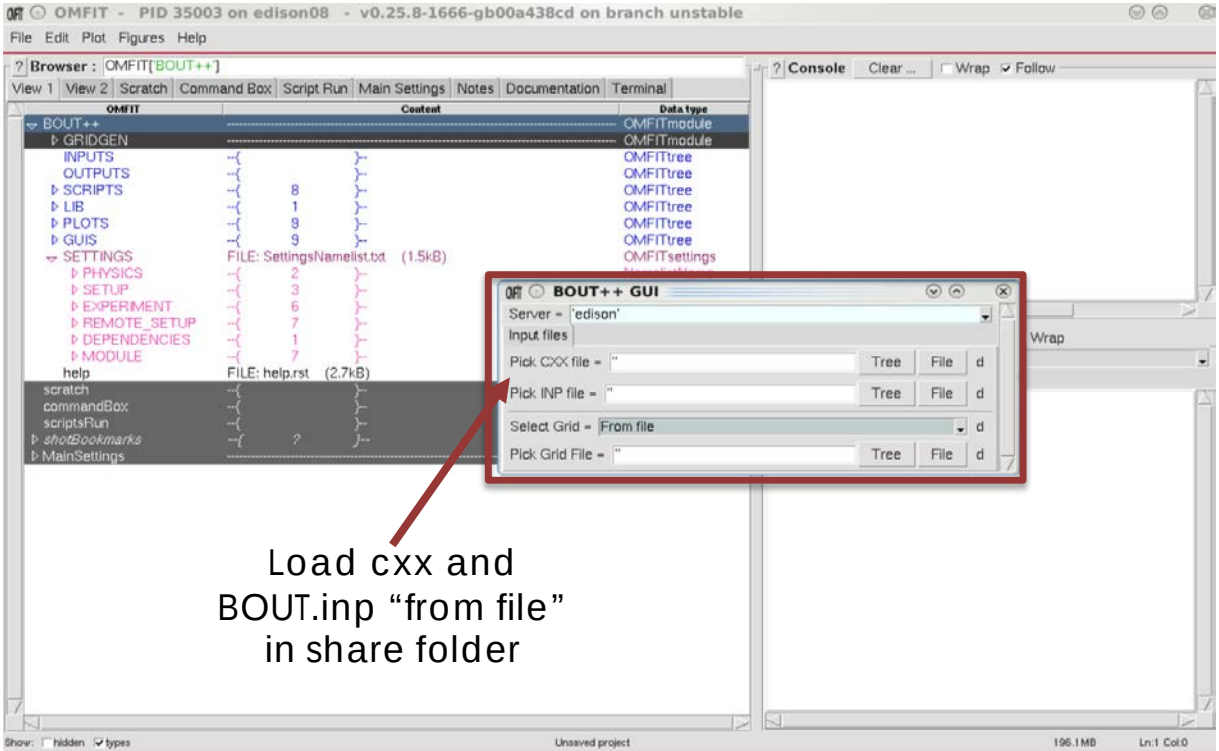
Open main GUI



Double click on "BOUT++" to open GUI

Can explore available scripts, settings, and data in python tree

Load example input files into project



Load cxx and
BOUT.inp "from file"
in share folder

Shared folder:
/global/homes/t/twilks/Share/BOUTworkshop18/

Compile the cxx file

OMFIT - PID 35003 on edison08 - v0.25.8-1666-gb00a438cd on branch unstable

File Edit Plot Figures Help

Browser : OMFIT['BOUT++']

View 1 | View 2 | Scratch | Command Box | Script Run | Main Settings | Notes | Documentation | Terminal

OMFIT	Content	Data type
BOUT++		OMFITmodule
GRIDGEN		OMFITmodule
INPUTS	2	OMFITtree
CXX	FILE: elm_pb.cxx (130.8kB)	OMFITascii
INP	FILE: BOUT.inp (8.9kB)	OMFITini
OUTPUTS		OMFITtree
SCRIPTS	8	OMFITtree
LIB	1	OMFITtree
PLOTS	9	OMFITtree
GUIS	9	OMFITtree
SETTINGS	FILE: SettingsNamelist.txt (1.5kB)	
PHYSICS	2	
SETUP	3	
EXPERIMENT	6	
REMOTE_SETUP	7	
DEPENDENCIES	1	
MODULE	7	
help	FILE: help.rst (2.7kB)	
scratch		
commandBox		
scriptsRun		
shotBookmarks	?	
MainSettings		

Console Clear ... Wrap Follow

```
Clearing local directory: /tmp/train41/OMFIT/OMF
Changed to working directory: /tmp/train41/OMFIT
Clearing remote directory: /scratch1/scratchdirs
Local copy of: /tmp/train41/OMFIT/OMFIT_2018-08-
Local execute of:
module swap PrgEnv-intel PrgEnv-gnu
module load cray-fftw
make in:
/scratch1/scratchdirs/train41/OMFIT/runs/project
edison08
/scratch1/scratchdirs/train41/OMFIT/runs/project
local_PID=61925

  Compiling  elm_pb.cxx
```

BOUT++ GUI

Server = 'edison'

Input files | Reset

Pick a different CXX file

Pick a different INP file

Compile CXX

Select Grid = From file

Pick Grid File = '' | Tree | File

Execution of OMFIT workflow...

OMFIT['BOUT++']['SCRIPTS']['makeBOUTmodule']

Running locally:

```
module swap PrgEnv-intel PrgEnv-gnu
module load cray-fftw
make
```

Kill local | Open terminal | Abort

Compile!

Show: hidden types | Unsaved project | 196.4MB | Ln:1 Col:0

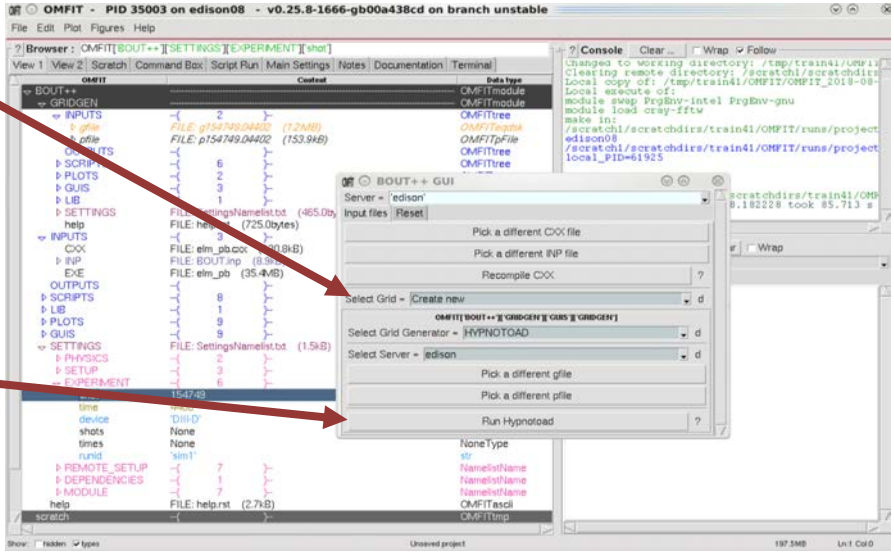
Overview of Tutorial

- OMFIT overview
- OMFIT setup options
- BOUT++ module overview
- Grid generation with Hypnotoad
- Running BOUT++
- Post-processing tools

Runs IDL procedure
by B. Dudson

Generate BOUT++ gridfile with Hypnotoad

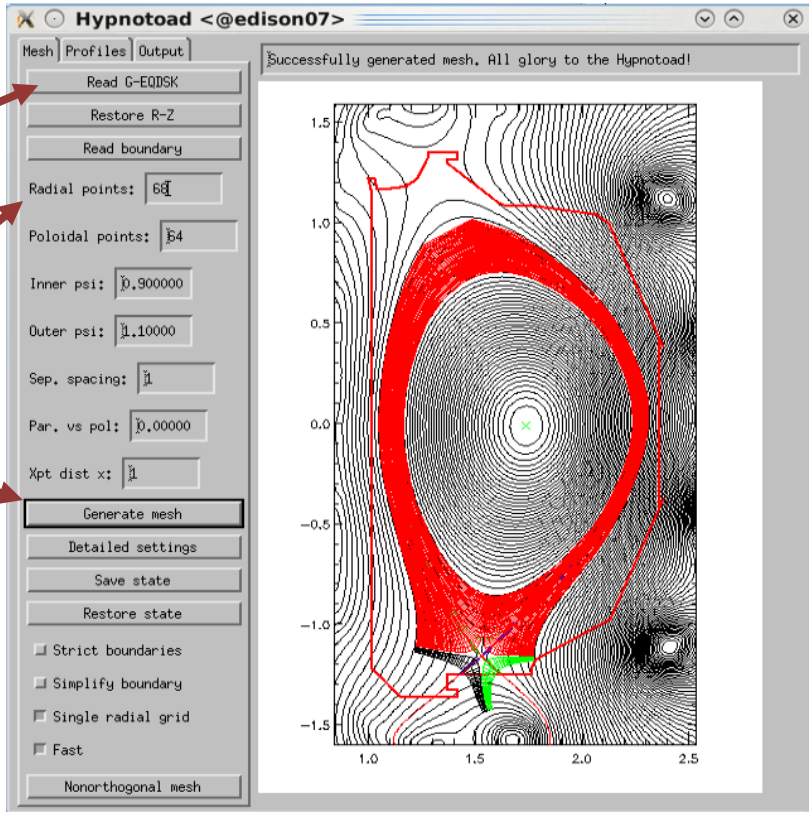
- Change Grid File (create new)
  - Hypnotoad Grid generator
- Load gfile and pfile from share folder
- Run hypnotoad



Shared folder:
/global/homes/t/twilks/Share/BOUTworkshop18/

Generate Mesh with Hypnotoad

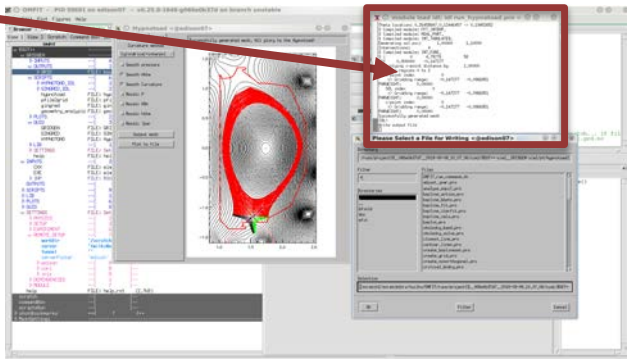
- Read geqdsk (will be in the same folder)
- Set number of radial and poloidal points
- Generate mesh



Output Mesh and load into OMFIT tree

- Need to interface with the terminal to exit out of Hynotoad

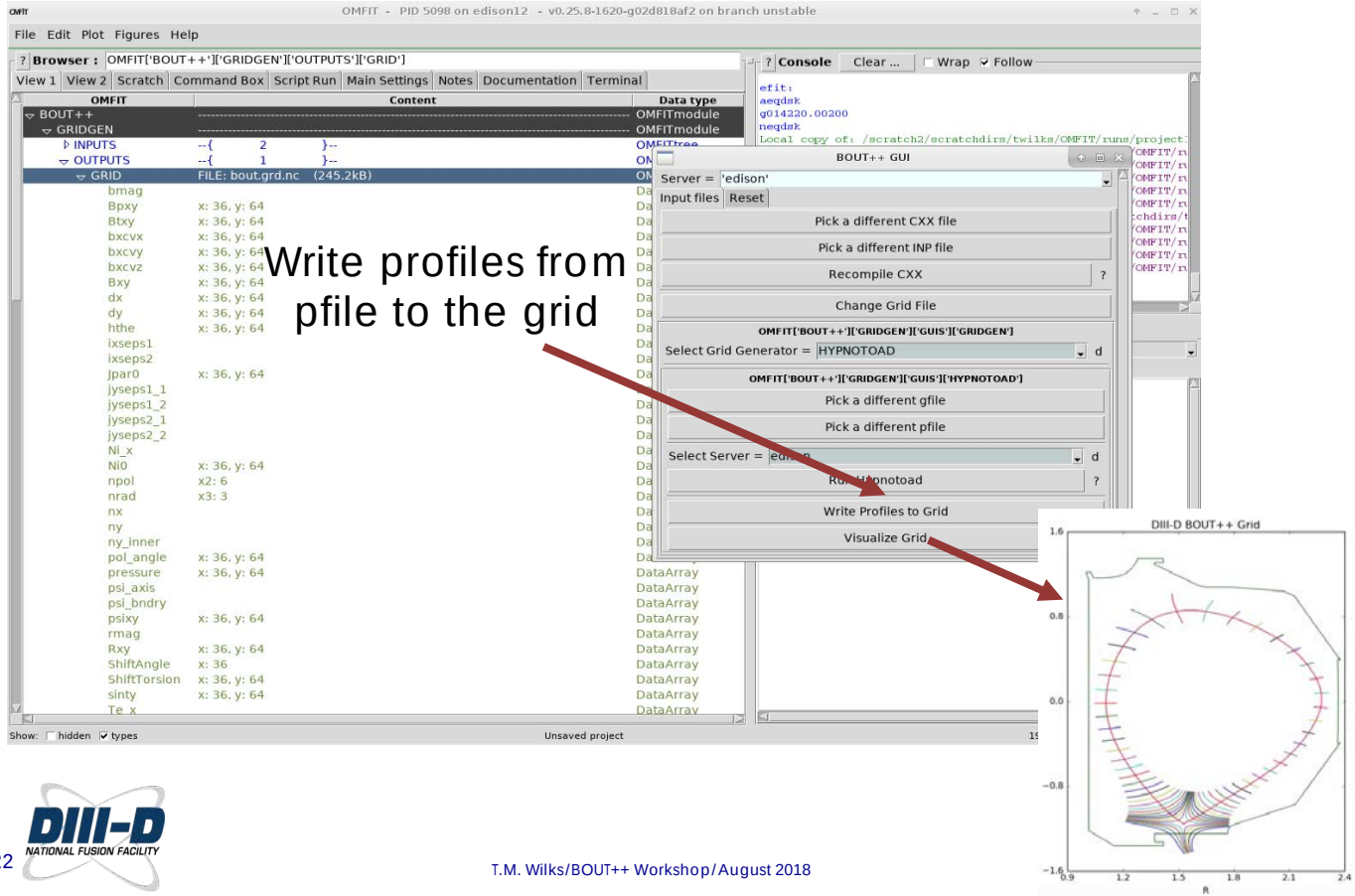
  Profile option: 1
  Temperature (eV): 1000
  Is this ok? Yes
  IDL>exit



- This ends your OMFIT command that is running IDL and brings the gridfile back into the OMFIT tree

root[‘GRIDGEN’][‘OUTPUTS’][‘GRID’]

Prepare gridfile to be passed to BOUT++



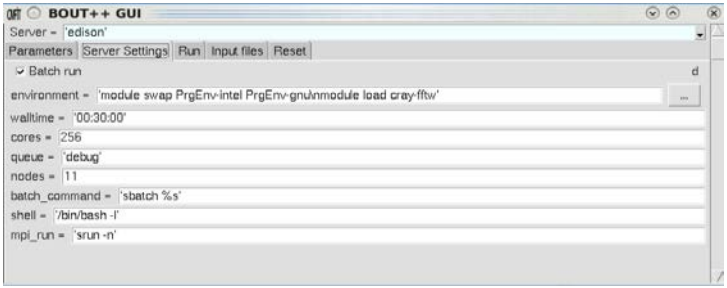
Write profiles from pfile to the grid

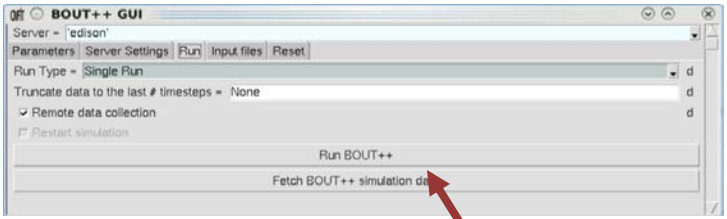
Overview of Tutorial

- OMFIT overview
- OMFIT setup options
- BOUT++ module overview
- Grid generation with Hypnotoad
- Running BOUT++
- Post-processing tools

Prepare BOUT++ simulation inputs

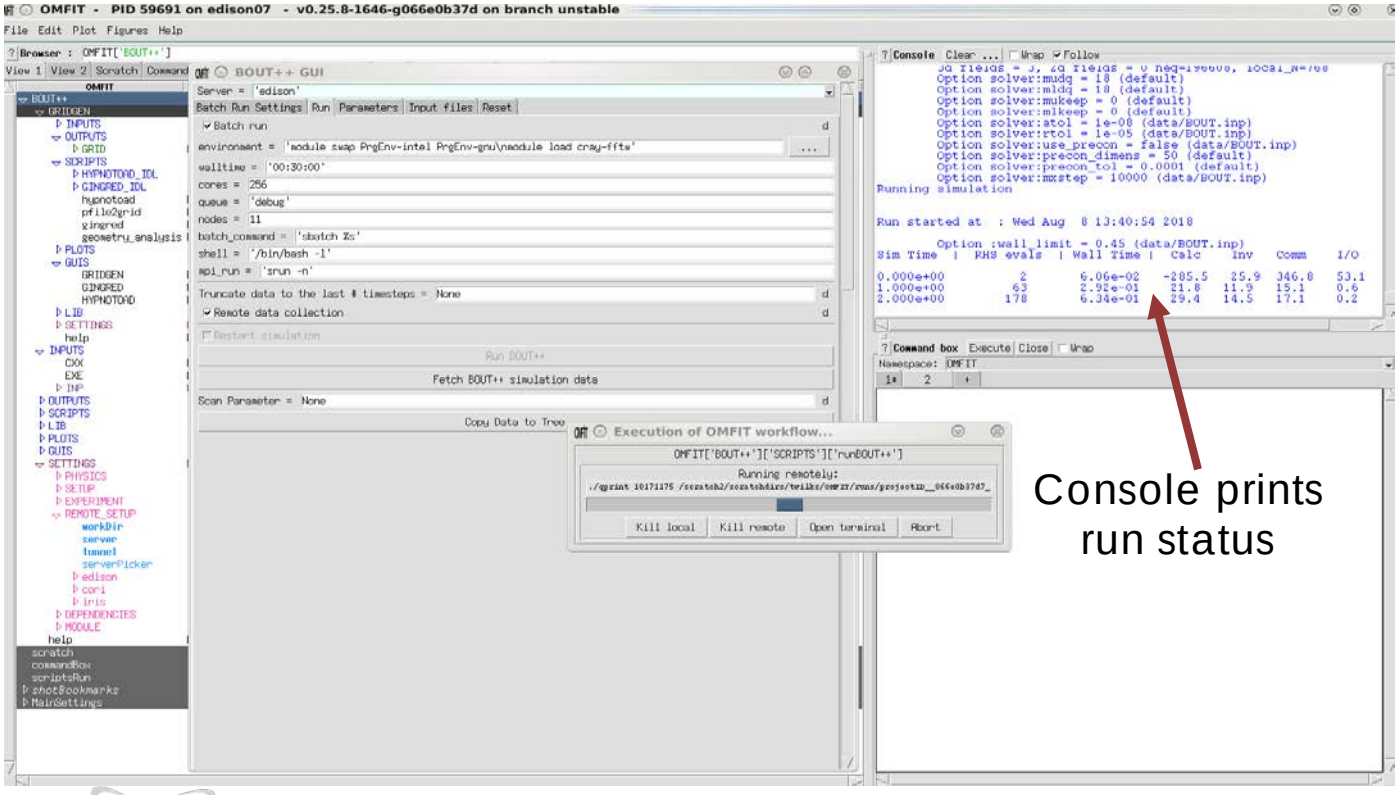
- "Parameters" tab handles BOUT++ input file options for physics assumptions
- "Server Settings" tab handles batch run settings
- "Run Settings" tab handles data fetching and convergence studies (still under development)



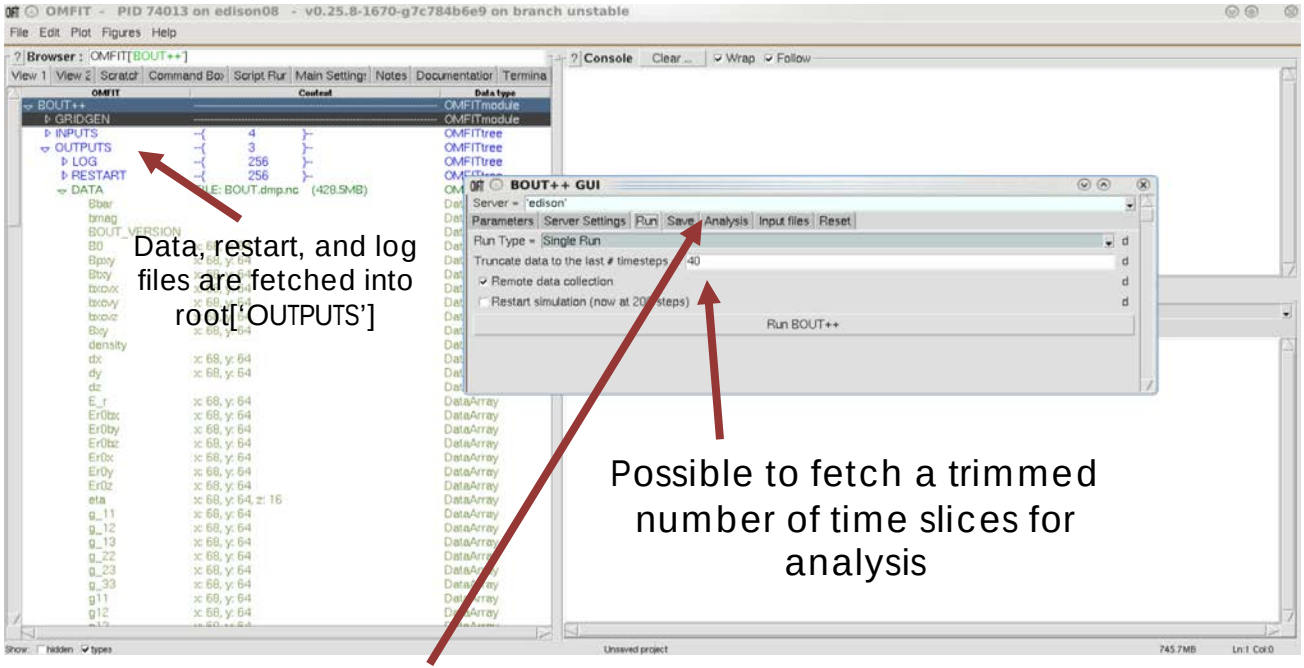


Run BOUT++

Run BOUT++


OMFIT - PID 59691 on edison07 - v0.25.8-1646-g066e0b37d on branch unstable
BOUT++ GUI
Execution of OMFIT workflow...
Console prints run status

OMFIT fetches the BOUT++ data back to the python tree



Data, restart, and log files are fetched into root['OUTPUTS']

Possible to fetch a trimmed number of time slices for analysis

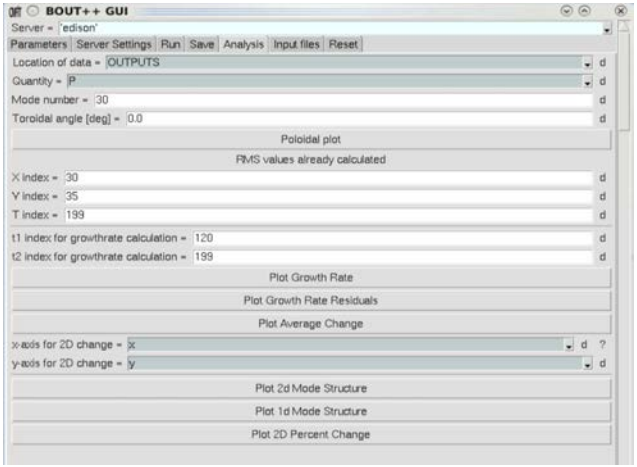
With data available in "OUTPUTS", the "Analysis" tab becomes available

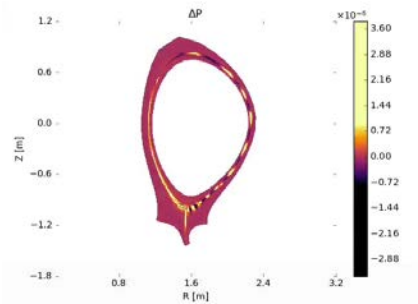
Overview of Tutorial

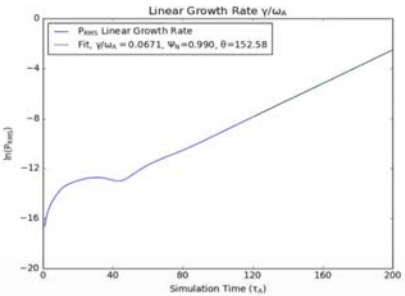
- OMFIT overview
- OMFIT setup options
- BOUT++ module overview
- Grid generation with Hypnotoad
- Running BOUT++
- Post-processing tools

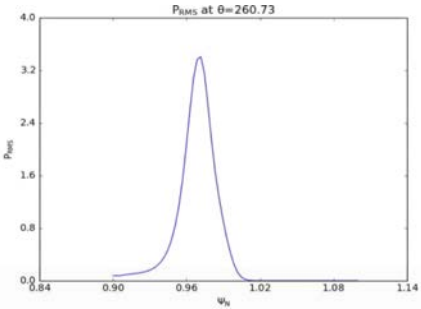
Several simple analysis tools available

- Visualization of evolved quantities (e.g. pressure perturbation)
- Growth rate calculation
- Mode structure visualization





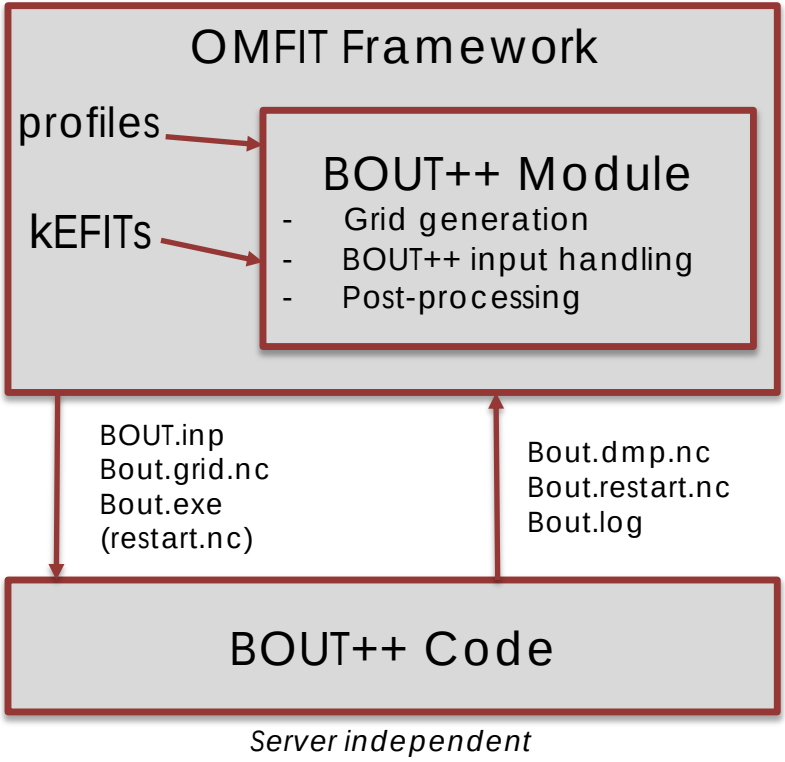




DIII-D NATIONAL FUSION FACILITY

OMFIT can be used to simplify the process of running BOUT++

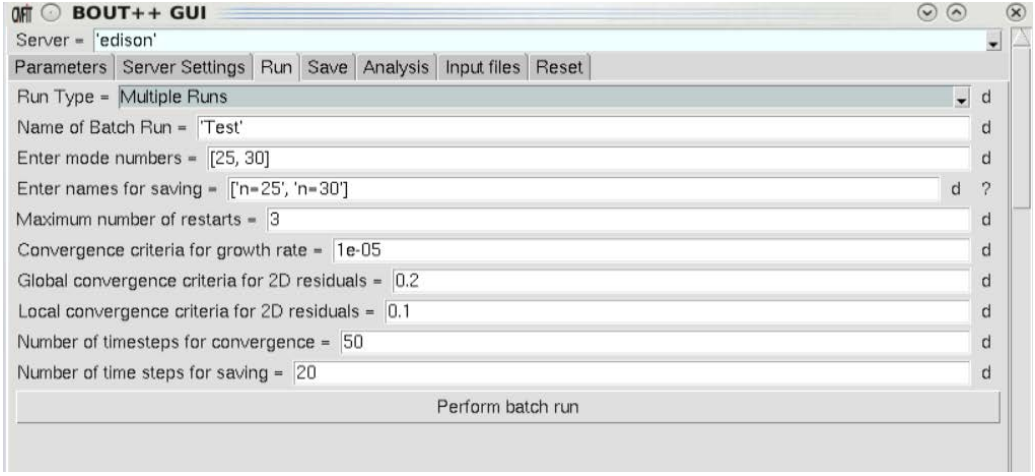
- Has potential to be coupled to entire pedestal workflow
  - Profile fitting
  - Kinetic EFITs
  - Varyped/ELITE/EPED
- GRIDGEN module has potential to be coupled with other codes
  - UEDGE
  - SOLPS



Always improving: BOUT++ module still under development

- This summer, SULI student Cody Moynihan developed
  - Saving (large amounts of) data into easily accessible tree structures for analysis of different scanned parameters
  - Quantitative checks on convergence with scripts to rerun if criteria is not met

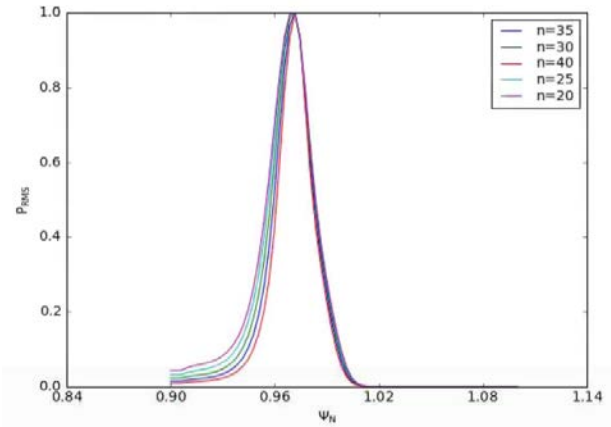
- Users (you guys) can add analysis tools for the OMFIT BOUT++ community!

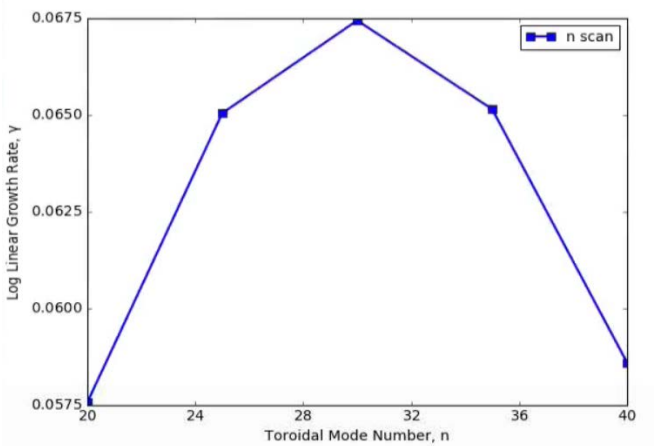


Always improving: BOUT++ module still under development

- This summer, SULI student Cody Moynihan developed
  - Saving (large amounts of) data into easily accessible tree structures for analysis of different scanned parameters
  - Quantitative checks on convergence with scripts to rerun if criteria is not met

- Users (you guys) can add analysis tools for the OMFIT BOUT++ community!





Questions/Comments