Automated Reconstruction and Experimental Integrated Modeling and Data Analysis in DIII-D

Presented by

L.L. Lao for the *DIII-D* and the *IMFIT* Teams

ITER Integrated Modeling Technology Workshop
Cadarache, France
June 8-10, 2011

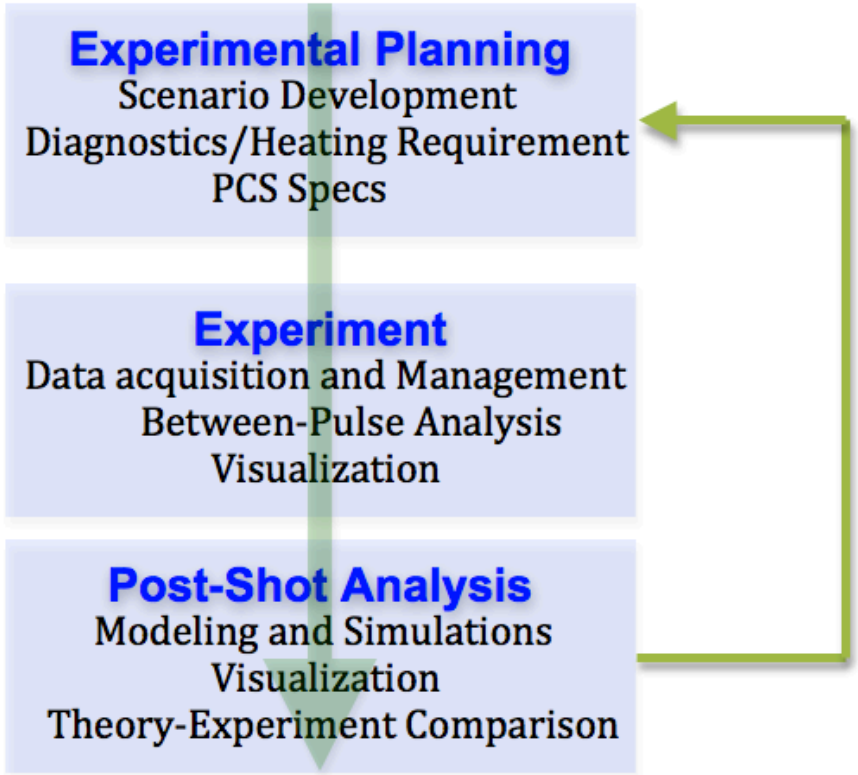
***DIII-D* and *IMFIT* Teams:**
L.L. Lao,[1] G. Abla,[1] M.S. Chu,[1] A. Collier,[1] R.J. Groebner,[1] S. Flanagan,[1] W. Guo,[2] N. Kim,[1] X. Lee,[1] G. Li,[2] T.H.. Osborne,[1] C. Pan,[2] J.M. Park,[3] R. Prater,[1] J. Qian,[2] Q. Ren,[2] H.E. St. John,[1] D.P. Schissel,[1] S. Smith,[1] N. Bisai,[4] R. Srinivasan,[4] Y. Liu,[5] A. Sun,[6] M. Worrall,[7] M. Murakami,[3] V.S. Chan,[1] B. Wan,[2] and J. Li[2]

[1]General Atomics, San Diego, CA U.S.A.
[2]ASIPP, Hefei, Anhui, China
[3]Oak Ridge National Laboratory, Oak Ridge, TN USA
[4]Institute for Plasma Research, Bhat, Gandhinagar, India
[5]Dalian University of Technology, Dalian, Liaoning, China
[6]Southwestern Institute of Physics, Chengdu, Sichuan, China
[7]Colorado School of Mines, Golden, CO USA

Both Fast Reconstruction and Full Experimental Analysis Are Necessary to Support Tokamak Operation and Research

- **Between-shot automated reconstruction requires fast analysis to guide operation**
- **Full experimental analysis necessary for accurate interpretation of experimental results and planning**
  - Detailed equilibrium, transport, and stability modeling
- **First-principle HPC integrated modeling**
- **PCS, plasma simulator**

DIII-D automated reconstruction and development of new *modeling* tool *IMFIT* to support integration



Outline

- **DIII-D automated between-shot and overnight analysis is based on a parallel multi-node Linux system to provide crucial equilibrium and transport information to guide plasma operation**
- **Extensive after-shot analysis is supported with a variety of Fortran, Python, and IDL based analysis tools for detailed physics studies and experimental planning**
- **A new modeling and analysis tool *IMFIT* is being developed to provide a central platform for integration of experimental analysis** *Operation and Research*
  - First version installed in DIII-D and recently in EAST and SST-1

The Goal of DIII-D Automated Between-Shot Analysis is to Provide Immediate Understanding of Previous Discharges

- **Evaluate plasma shape, beam heating and torque, RF heating and current drive, magnitude of bootstrap current, confinement factors, ….**
- **Find changes needed in parameters for upcoming discharge**
  - For example, how to adjust EC launch angles to improve alignment of ECCD with a resonant surface
- **Archive analysis for general use**
  - All results are stored in MDSplus for general access

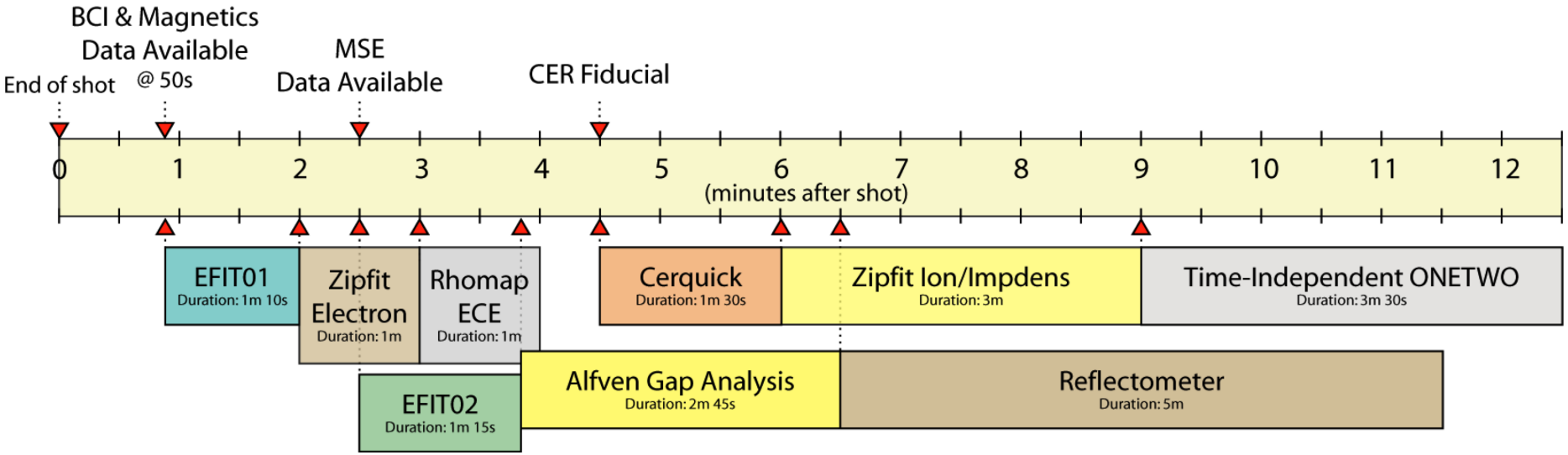
Specific framework requirements

General Overview of Between-Shot Processes

- **Codes (*IDL, Python, etc*) are launched onto a Linux parallet cluster after data dependencies are satisfied**
- **Data required to run code can be read from files or specific DIII-D databases such as *PTDATA* or *MDSplus***
- **When code finishes, results are typically stored into *MDSplus* using a loader code**

DIII-D Automated Between-Shot Analysis Consists of Equlibrium, Profile, and Transport Analyses

- **Entire between-shot analysis cycle takes about 12 minutes on a 48-node distributed Linux system**
  - Six 2 quad-core AMD Opteron 2.0 GHz



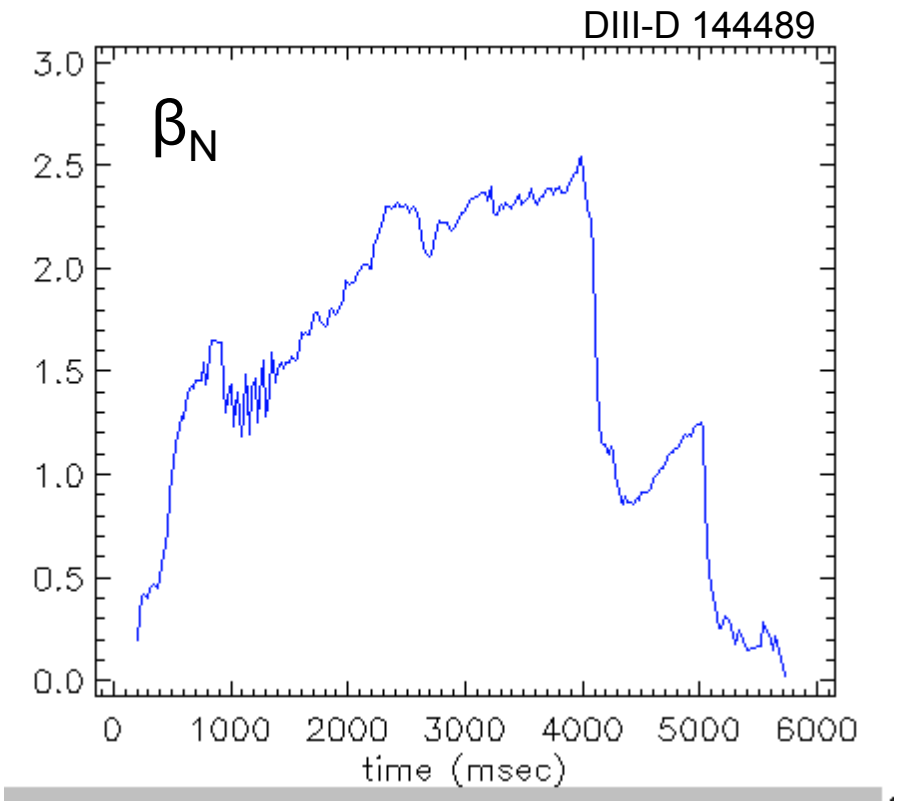
Equilibria with and without MSE Are Routinely Reconstructed for all Discharges with *EFIT*

- **200-250 time slices are typically analyzed**
  - Real-time *EFIT's* from PCS also available post-shot
- **Many physical quantities crucial to the interpretation of experiments computed directly from the equilibria**
  - Magnetic configuration, divertor geometry, stored energy, gaps, internal inductance, ….
  - Time derivative of a sequence of equilibria: Surface loop voltage, Ohmic heating power, energy confinement, …

Developing between-shot kinetic EFIT reconstruction
Essential for pressure and stability control

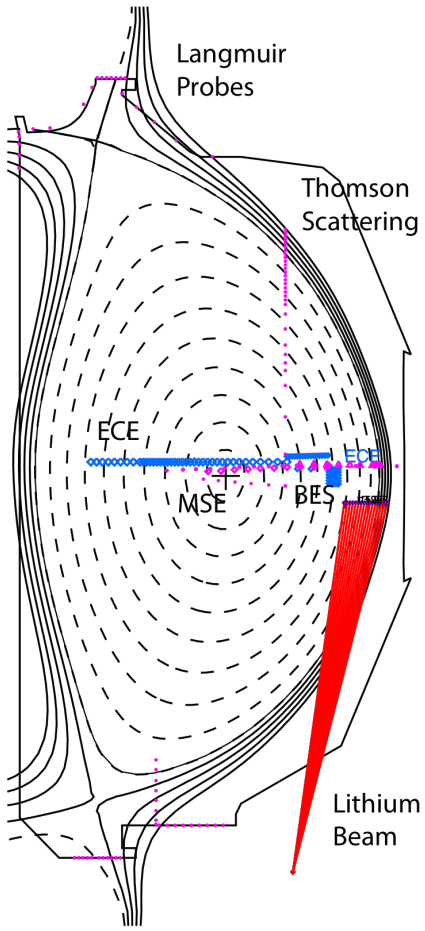
EFITViewer Provides a Convenient Tool to Interpret and Plan Subsequent Discharge *Display and Visualization*

Time Evolution of Equilibrium Data



Visualization of Diagnostic Points

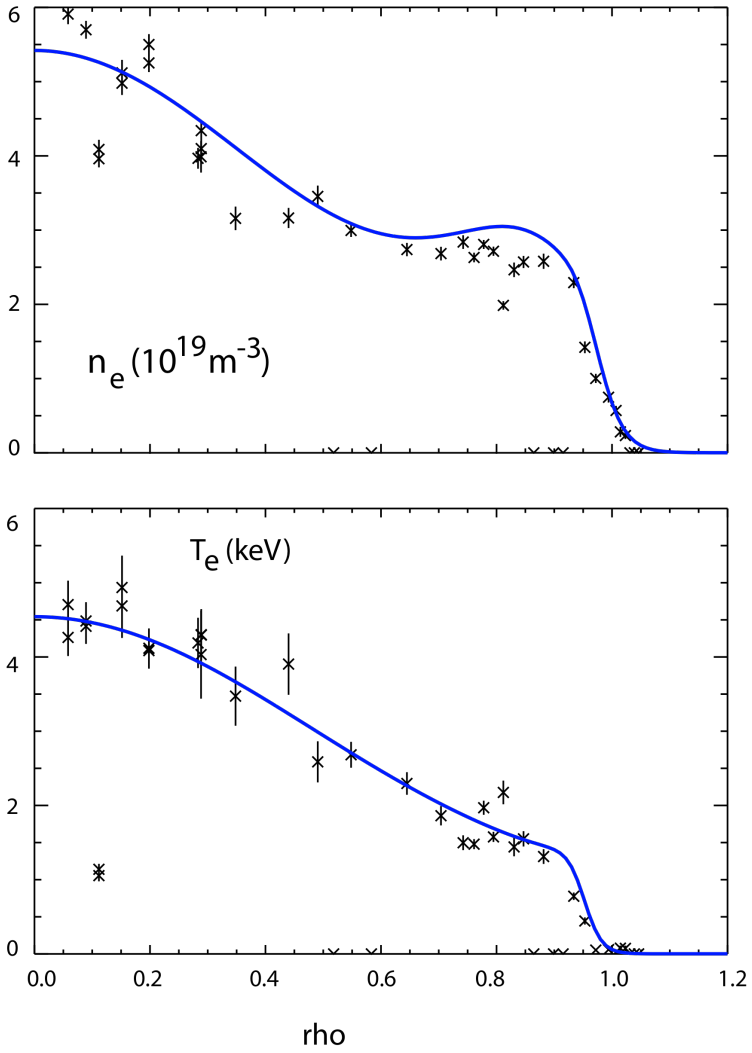
shot	133221
time	3775.00
chi**2	83.184
Rout(m)	1.668
Zout(m)	0.030
a(m)	0.599
elong	1.840
utri	0.642
ltri	0.407
indent	0.000
V (m**3)	19.252
A (m**2)	1.884
W (MJ)	0.712
betaT(%)	2.081
betaP	0.808
betaN	1.801
In	1.155
Li	0.908
Li3	0.714
error(e-4)	0.952
q1	6.931
q95	4.218
dsep(m)	0.053
Rm(m)	1.733
Zm(m)	-0.034
Rc(m)	1.700
Zc(m)	-0.023
betaPd	0.803
betaTd	2.069
Wdia(MJ)	0.707
Ipmeas(MA)	1.201
BT(0)(T)	-1.697
Ipfit(MA)	1.194
Rmidin(m)	1.070
Rmidout(m)	2.265
gapin(m)	0.053
gapout(m)	0.087
gaptop(m)	0.103
gapbot(m)	0.177
Zts(m)	0.739
Rvsin(m)	1.108
Zvsin(m)	1.166
Rvsout(m)	1.350
Zvsout(m)	1.348
Rsep1(m)	1.277
Zsep1(m)	-1.170
Rsep2(m)	1.284
Zsep2(m)	1.132
psib(Vs/R)	0.120
elongm	1.357
qm	1.003
nev1(e19)	3.351
nev2(e19)	3.480
nev3(e19)	3.203
ner0(e19)	4.028
n/nc	-0.555
dRsep	0.010
qmin	1.003
rhoqmin	0.000



Data Available for Portal Access

ZIPFITs Provide Equilibrium Mapped Profiles for Kinetic Quantities

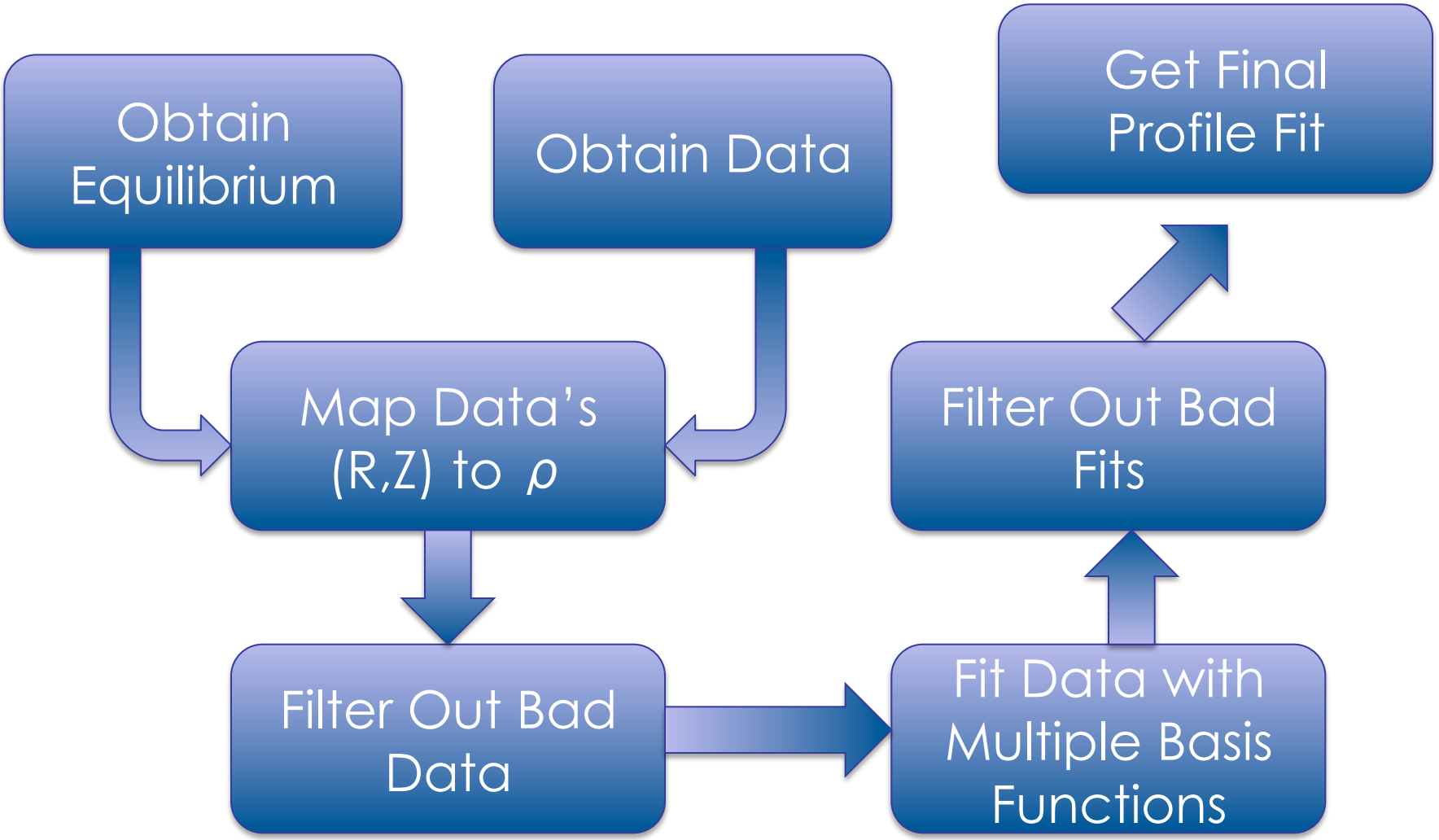
- Electron Density
- Electron Temperature
- Ion Temperature
- Toroidal Rotation
- Impurity Density



Between-Shot ZIPFITs Start after Equilibrium and Data Available

- **Equilibria come from between shot EFIT's**
  - Currently based only on magnetics (EFIT01)
- **Kinetic data come from a variety of diagnostics mapped into common space**
  - Thomson Scattering
    - Electron Density
    - Electron Temperature
  - Electron Cyclotron Emission
    - Electron Temperature
  - Charge Exchange Recombination
    - Ion Temperature
    - Rotation
    - Impurity Density

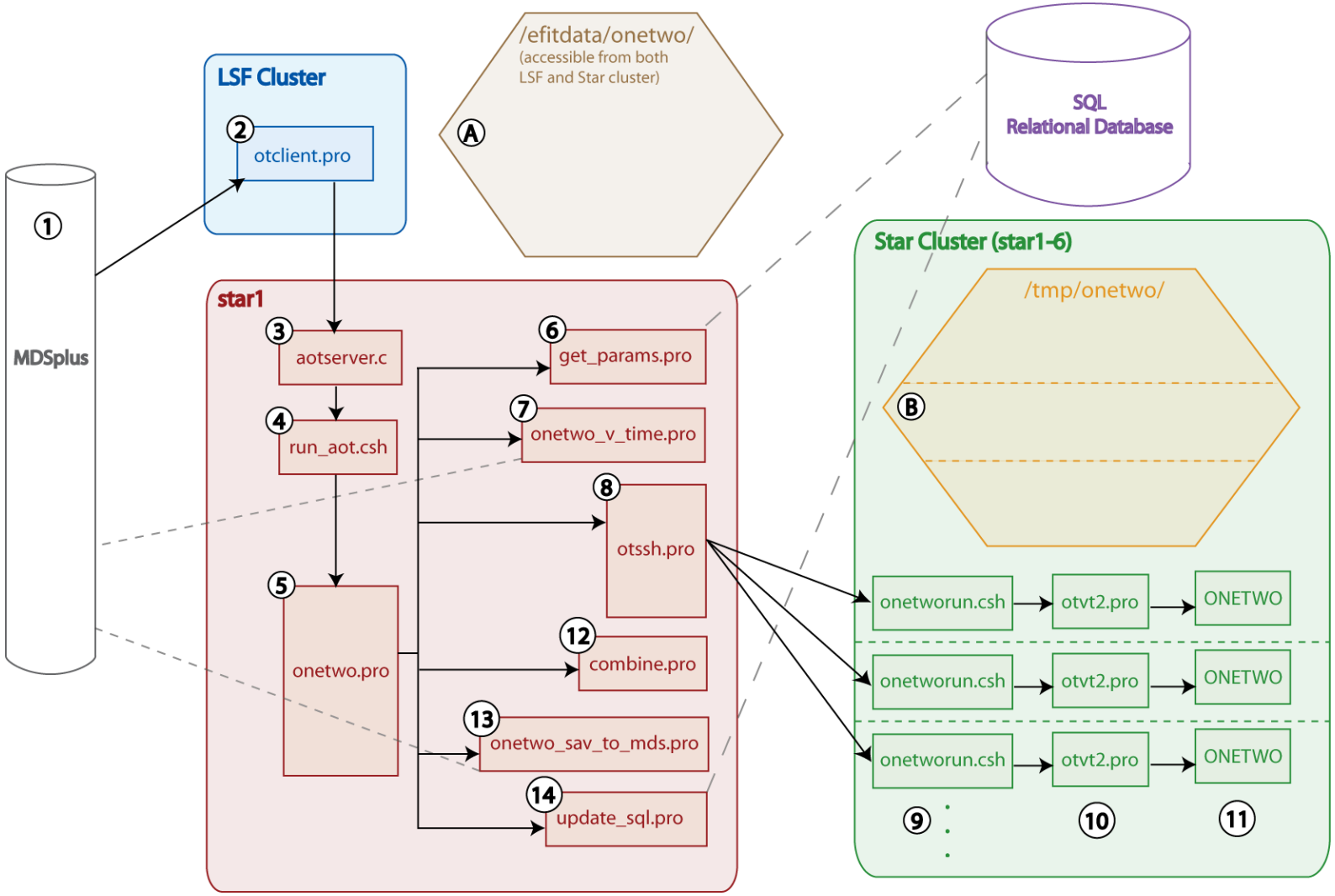
ZIPFIT Flow Chart



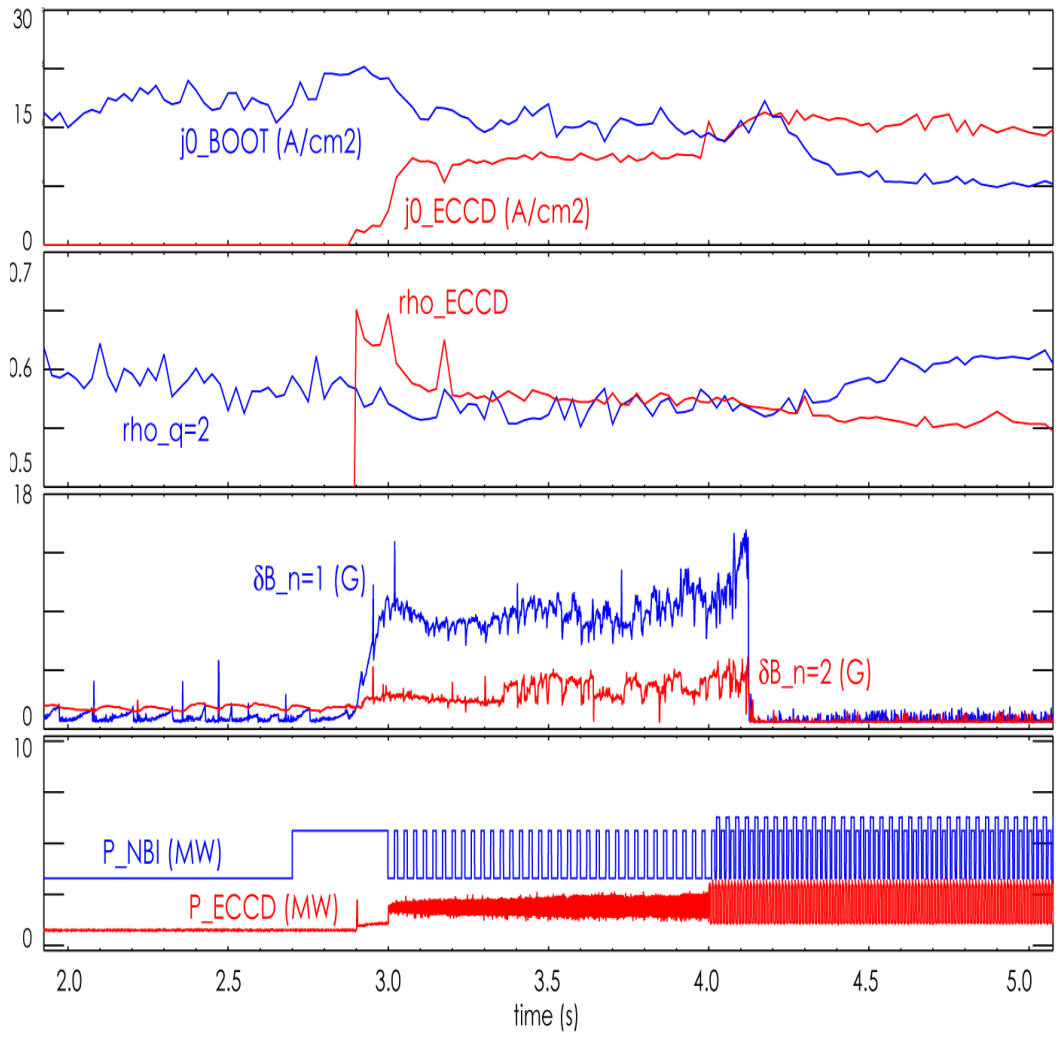
Automated Runs of *ONETWO* Follow Equilibria Reconstructions and Profile Fits

- ***AutoONETWO* uses MSE *EFIT* and *ZIPFIT* fits**
- **For NBI it uses the *NFREYA* Monte Carlo "reduced model," for completion in 4 minutes (on 24 Linux nodes)**
  - *NUBEAM* model is available for after-shot analysis
- **For ECH uses *TORAY-GA* with 30 EC rays**
- ***ONETWO* uses 201 radial grid points, as needed for the narrow ECH deposition profiles**
- **All data are stored in MDSplus for plotting and use in analysis**
- **Currently only analysis mode**

Between-Shot *AutoONETWO* Flow Diagram



AutoONETWO shows when ECCD is right place and size for suppressing *n* = 2 NTM



- *AutoONETWO* shows how the ECCD compares to the local bootstrap current
- *AutoONETWO* shows how well the ECCD location is tracking the q=2 surface
- Improved tracking and control of EC power and location can follow from this information

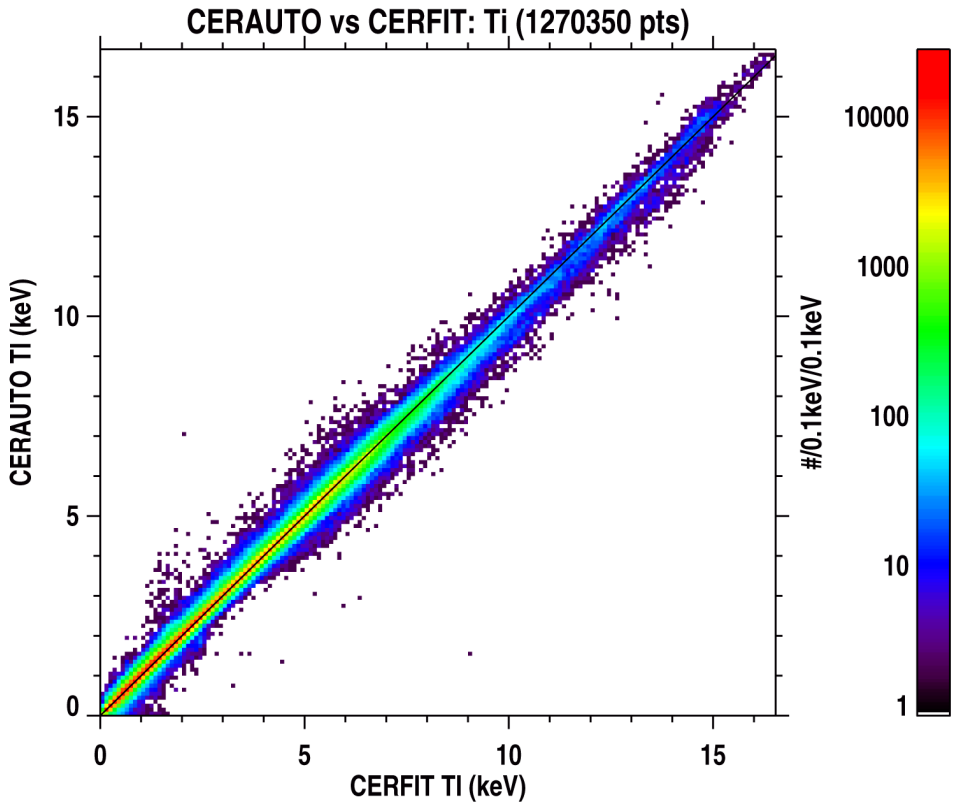
Strengths and Limitations in *AutoONETWO*

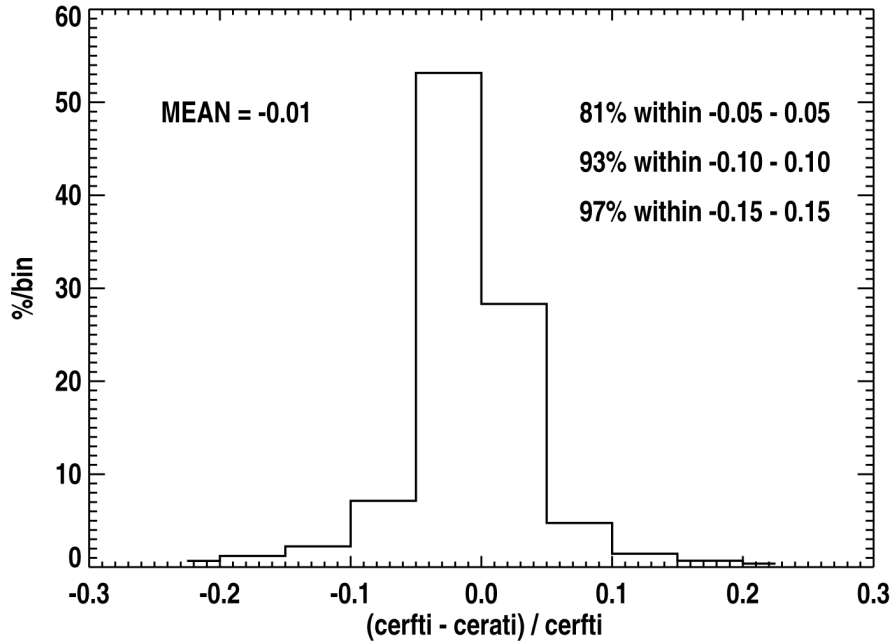
- **MSE *EFIT's* are usually sufficiently accurate**
- **Automated fits to kinetic quantities sometime have issues related to "bad" data points**
- **At present, time derivatives are ignored except for Ohmic heating power, so in dynamic situations confinement and diffusivities are not well described**
- ***AutoONETWO* is redone overnight using improved CER analysis (*CERAUTO*, by R. Groebner)**
- **Presently only analysis mode, would be useful to do predictive runs between-discharge**
  - GLF23, TGLF transport models computationally intensive

CERAUTO Is an Automated Fitting Process for CER Data

- **Goal of *CERAUTO* process is to provide automatic high quality fits to Charge Exchange Recombination (CER) Spectroscopy data that can be used with high confidence**
- **Several ingredients are required to make *CERAUTO* successful**
  - Use *CERFIT*, the standard non-linear least squares fitting routine for DIII-D CER data
  - Use time-tested fitting models (number and initial location of peaks, background treatment, etc.) to fit the data
  - Analyze only C VI spectra
- **Use two major tests to reject questionable results**
  - Reject results for which reduced χ^2 exceeds a specified threshold
  - Reject results for which ratio of amplitudes of passive and active C VI lines exceeds a specified threshold

CERAUTO Fits for T_i for Tangential Chords Compare Well to More Complete *CERFIT* Analysis

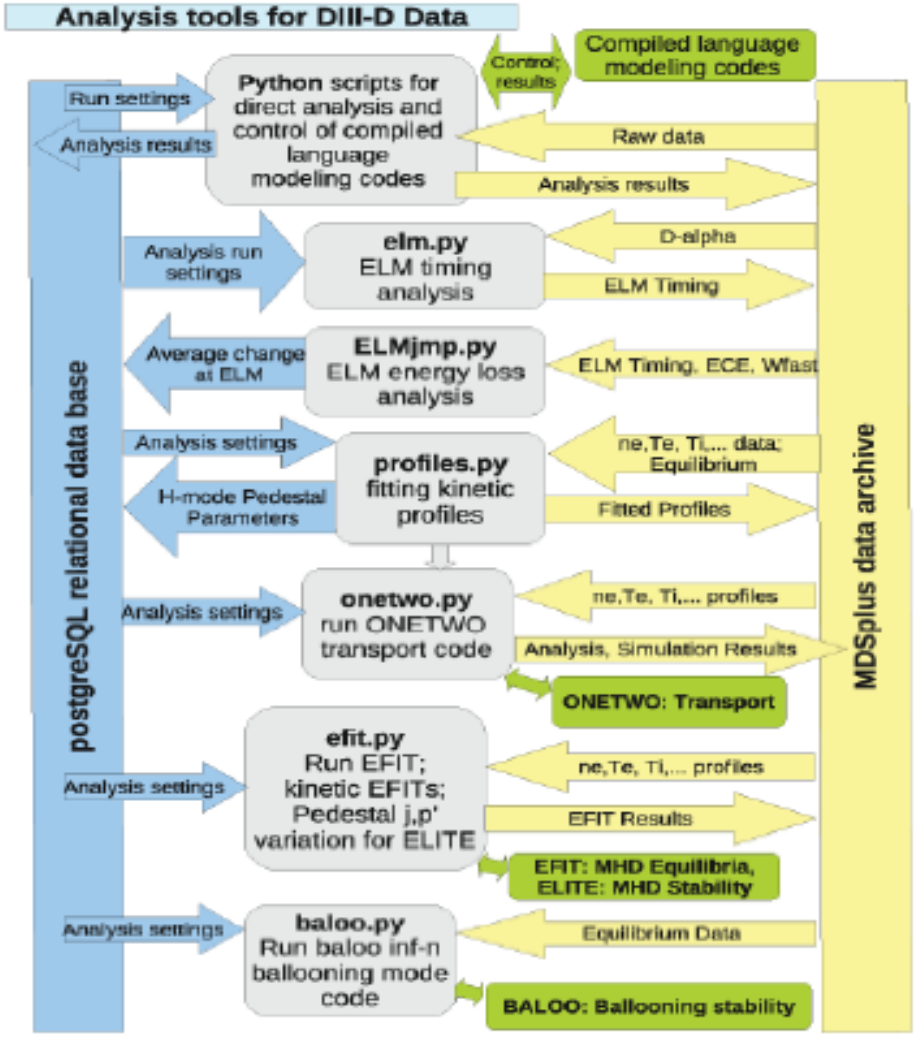




PyD3D Provides a Comprehensive set of Python Tool Kit for Analysis of ELM and Pedestal Physics Data

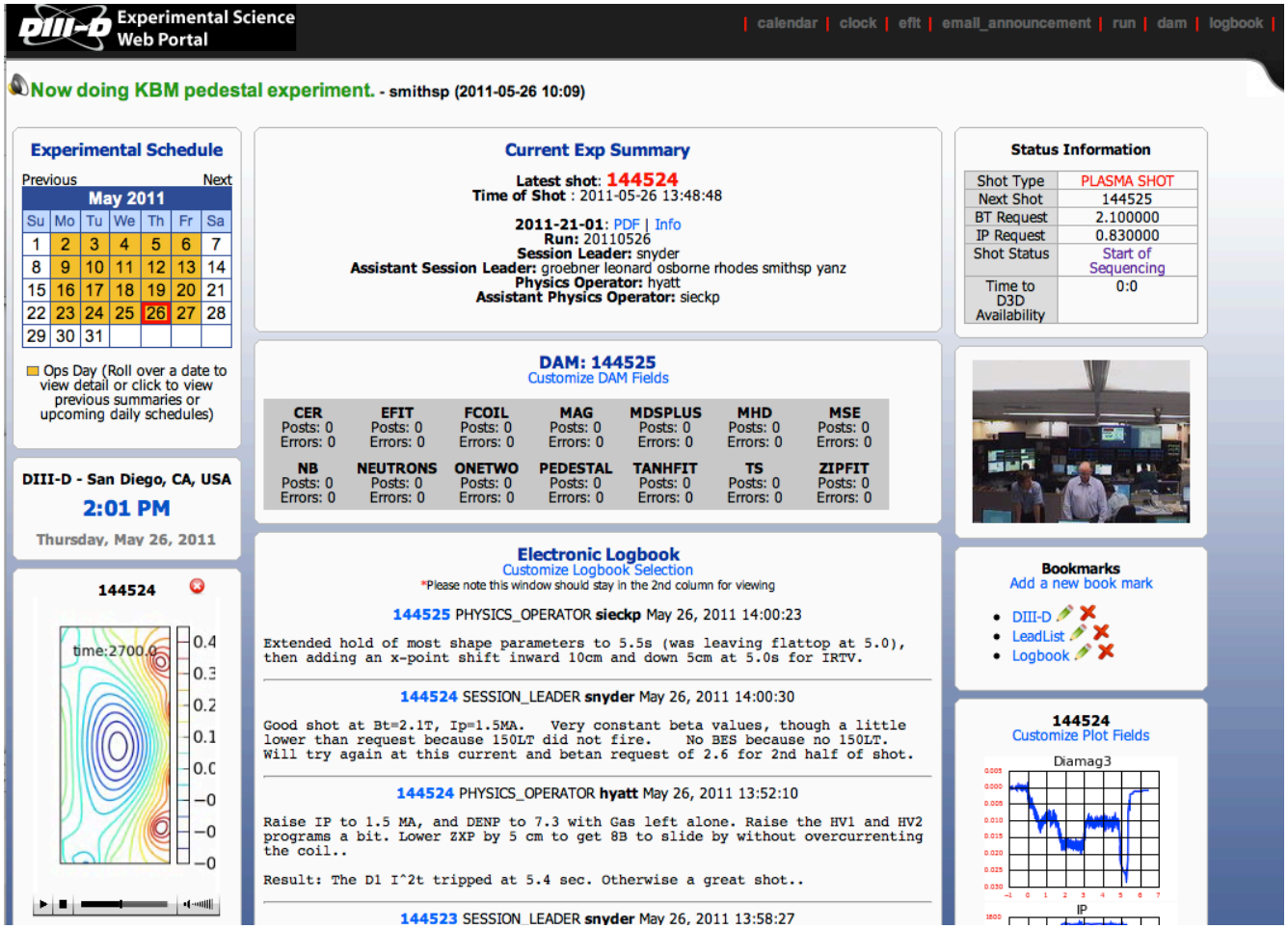
- Python based data analysis tool kit developed by T. Osborne *https://diii-d.gat.com/~osborne/python/CLICKME.html*
- Determine ELM timing and compute ELM energy loss from fast EFIT analysis
- Full electron and ion profile fittings with good edge resolution
- Interfaces to run kinetic *EFIT* and *ONETWO*
- Interfaces to MDS+ and ITER Pedestal Profile Databases

Kinetic *EFIT* Reconstructions



D3DPortal Provides a Convenient Tool to Monitor Status of Experiments and Between-Shot Automated Analyses

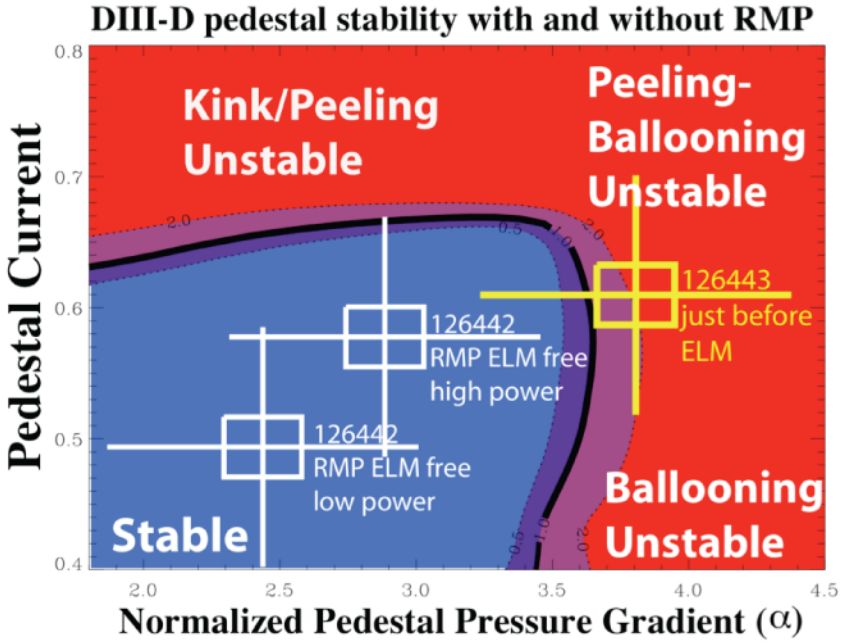
- *https://webportal.gat.com/apps/my/*



IMFIT Is Developed to Provide an Efficient Central Platform to Support Tokamak Experimental Research and Operation

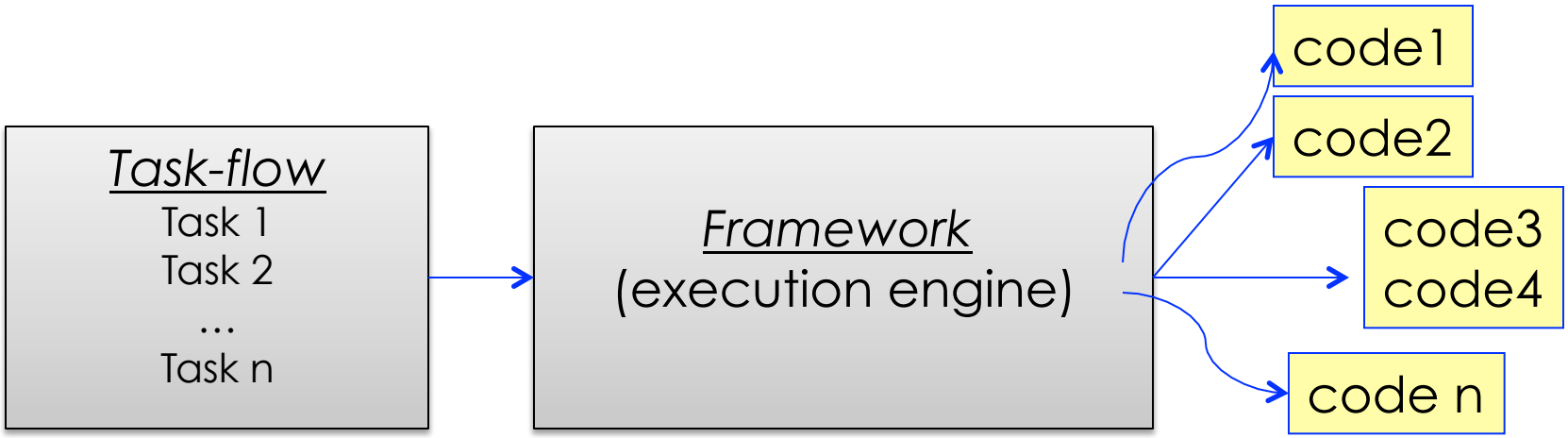
- **Primarily goal is to streamline and increase experimental productivity**
  - Equilibrium reconstruction, transport, and stability analyses
  - Theory-experiment comparison
  - Operation scenario development
- **Front-end experimental interface for SciDAC and FSP projects**
- **Training of students and scientists**
- **Currently only post experimental analysis, very limited between-shot or PCS support**

DIII-D EDGE STABILITY DIAGRAM



Snyder / Osborne, 2010 IAEA

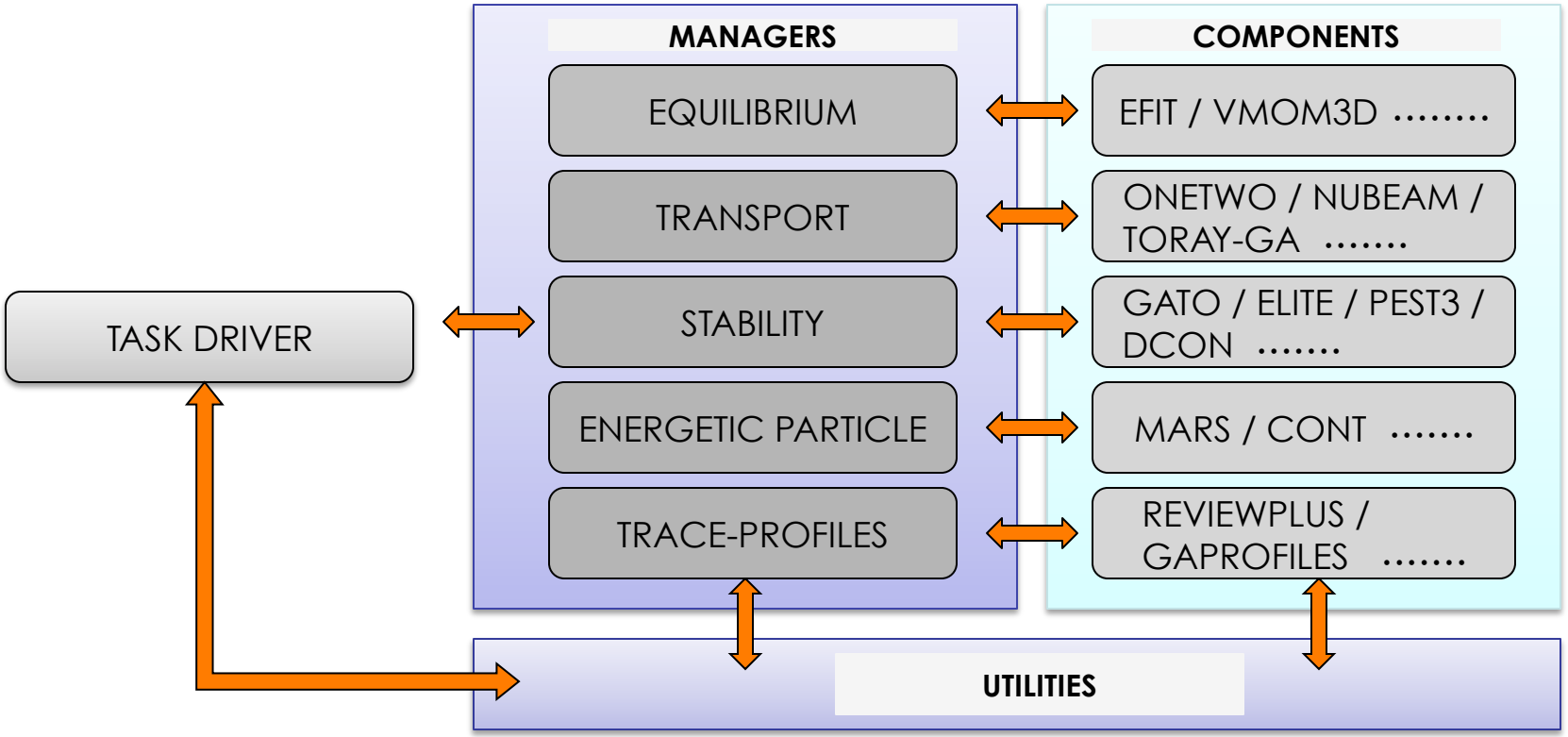
IMFIT Framework Based on a Modularized Task-Flow Architecture *Event Driven*



Jeon NFRI
Collier, Ren

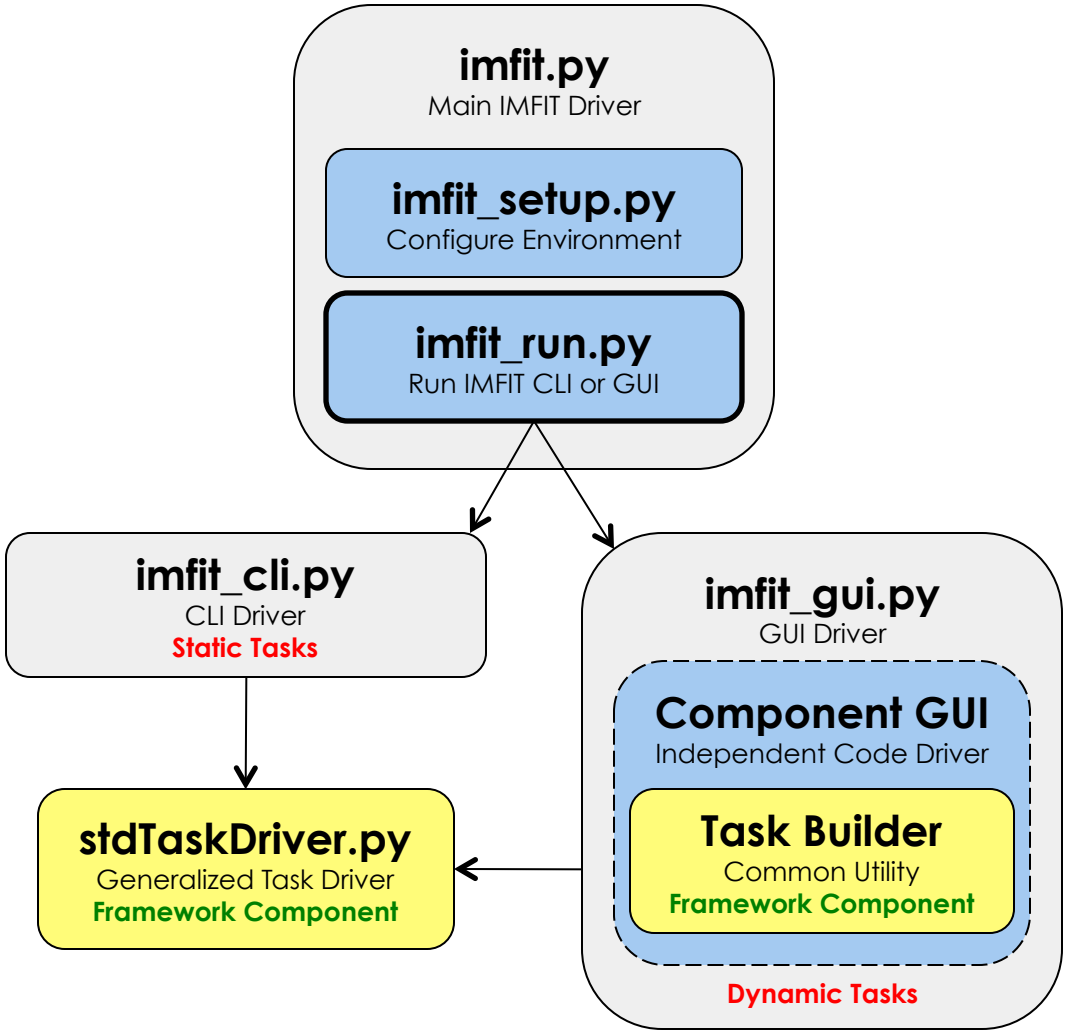
IMFIT Framework Consists of Managers, Components, and Utilities

- **Execution and communication managed by Task Driver with a NETCDF based state file**
- **Support both graphical user and command-line interfaces (*GUI*, CLI)**
- **Tools available for communication with other format state files**



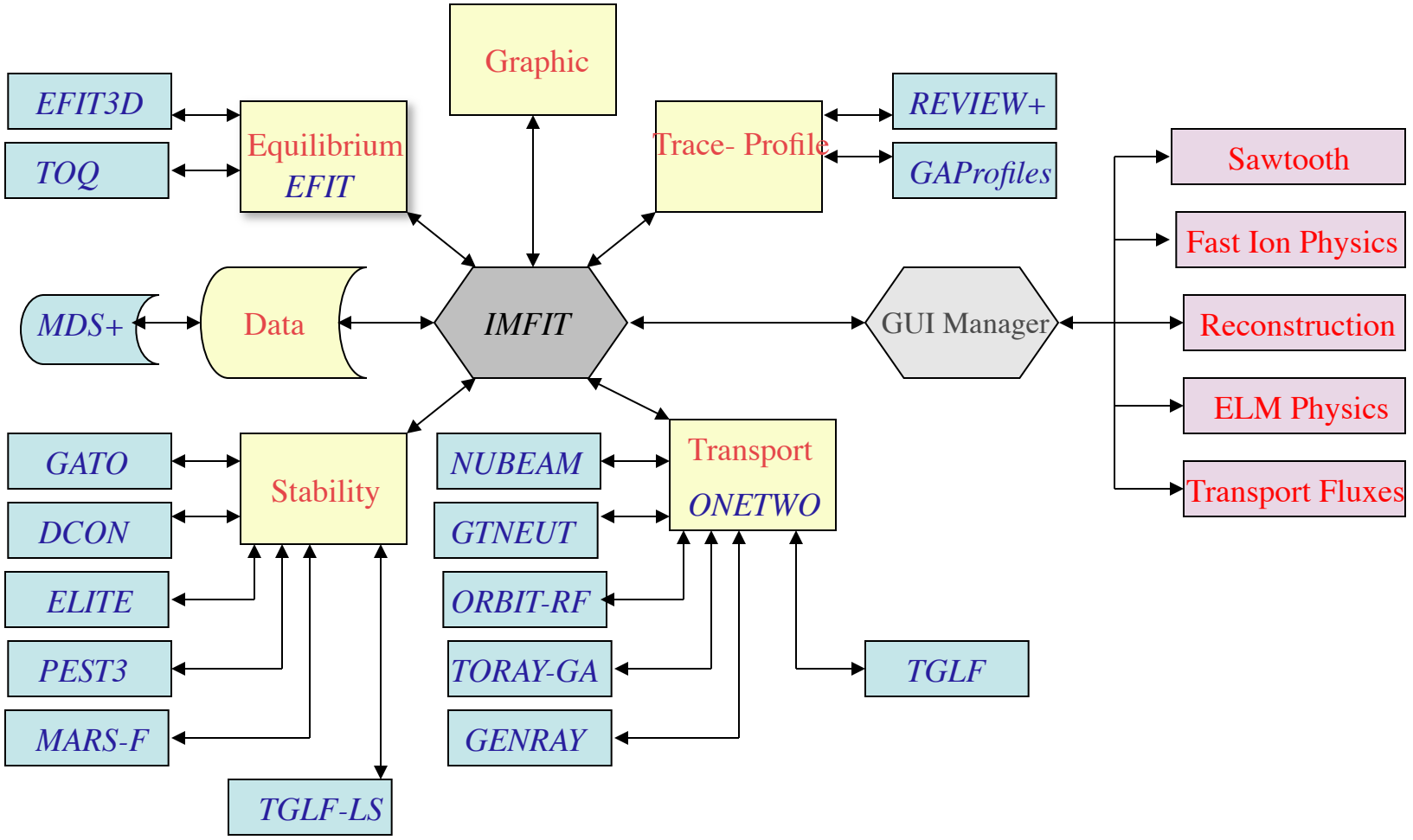
IMFIT Framework Written in Python with an Unified Execution Model

- **Component-based design**
- **Can use native Python language directly**

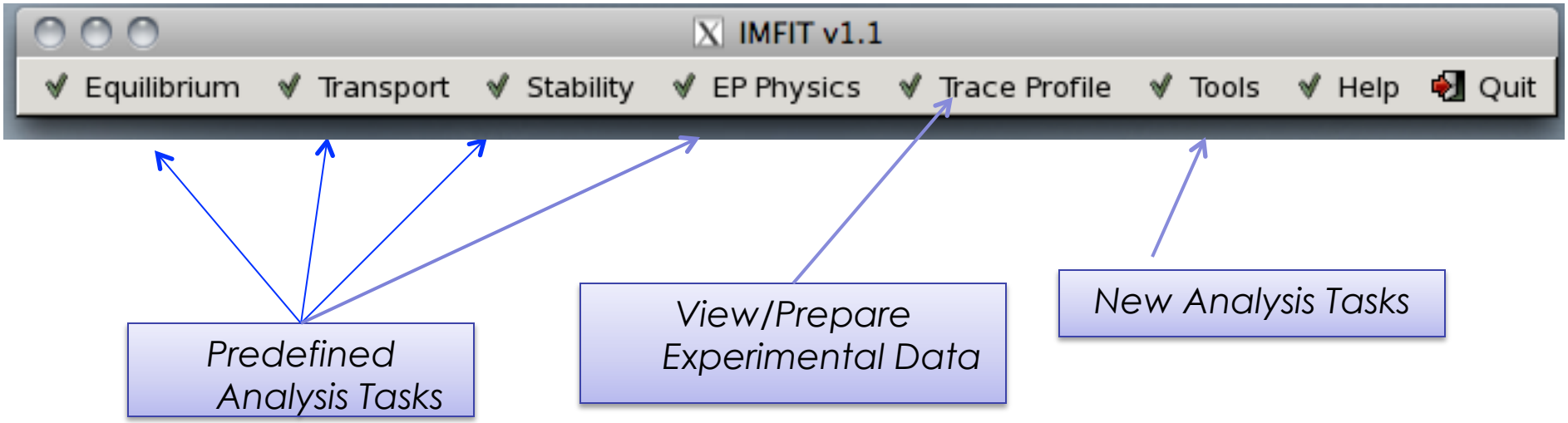


Users Interact with *IMFIT* through GUI or Configuration File

- **Analysis codes can be conveniently integrated into *IMFIT through the Component Manager***



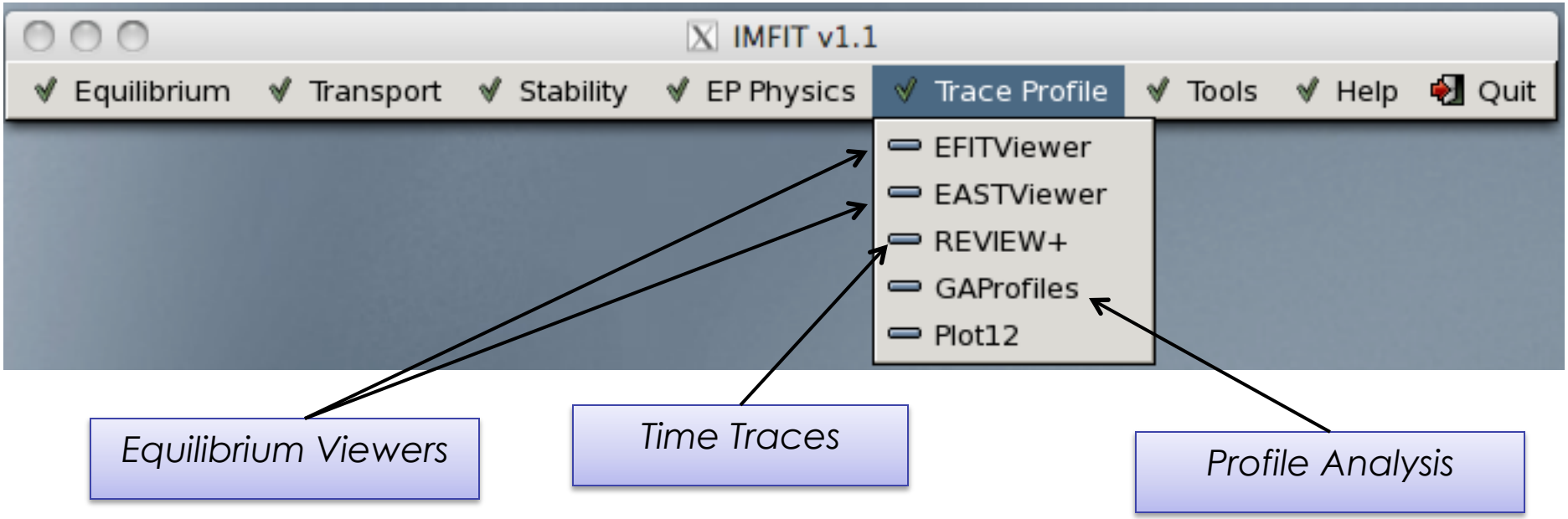
1st *IMFIT* Version to Integrate Equilibrium Reconstruction, Transport, and Stability Analyses Has Been Released

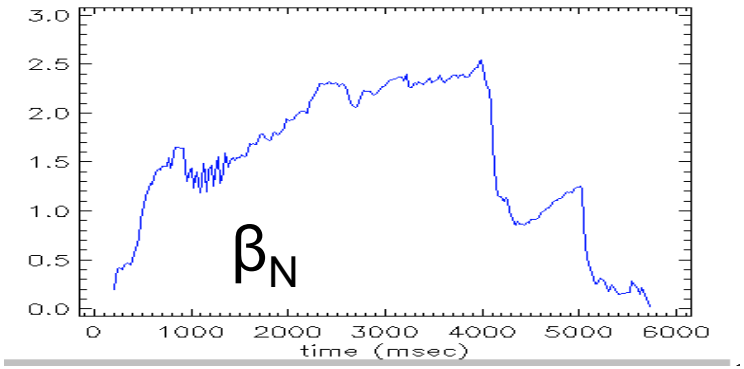


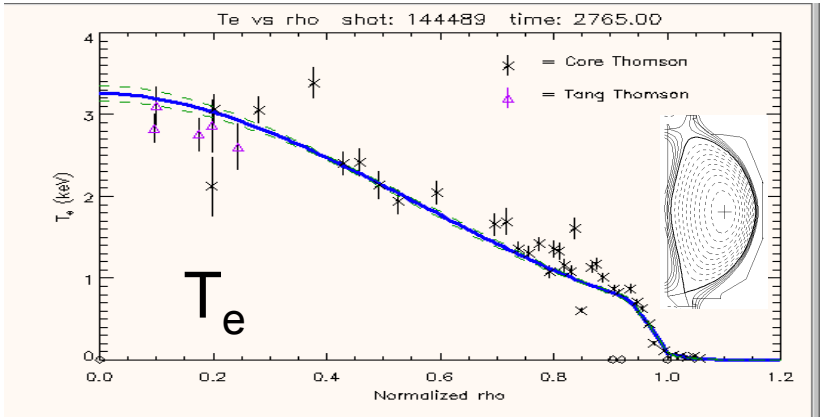
Choosing Menu options creates task-flows and submits them to the IMFIT server (local or remote)

Based on the public PyGTK graphical user interface toolkit for Python

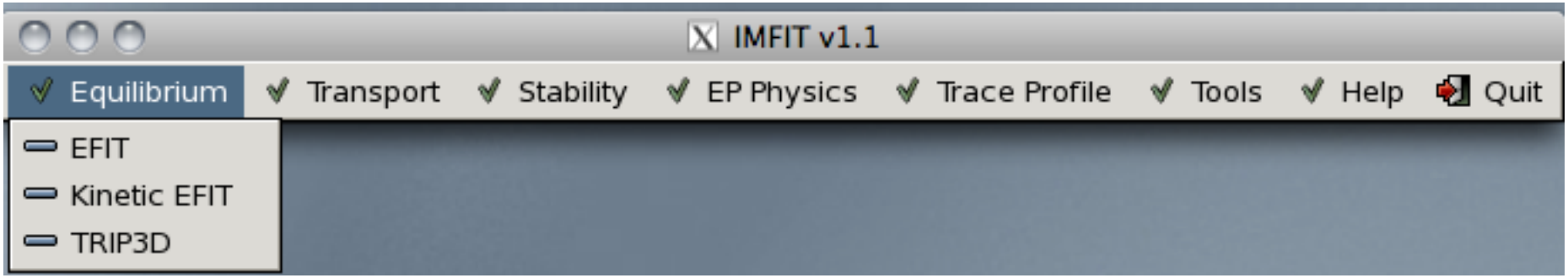
View and Prepare Experimental Data for Analysis



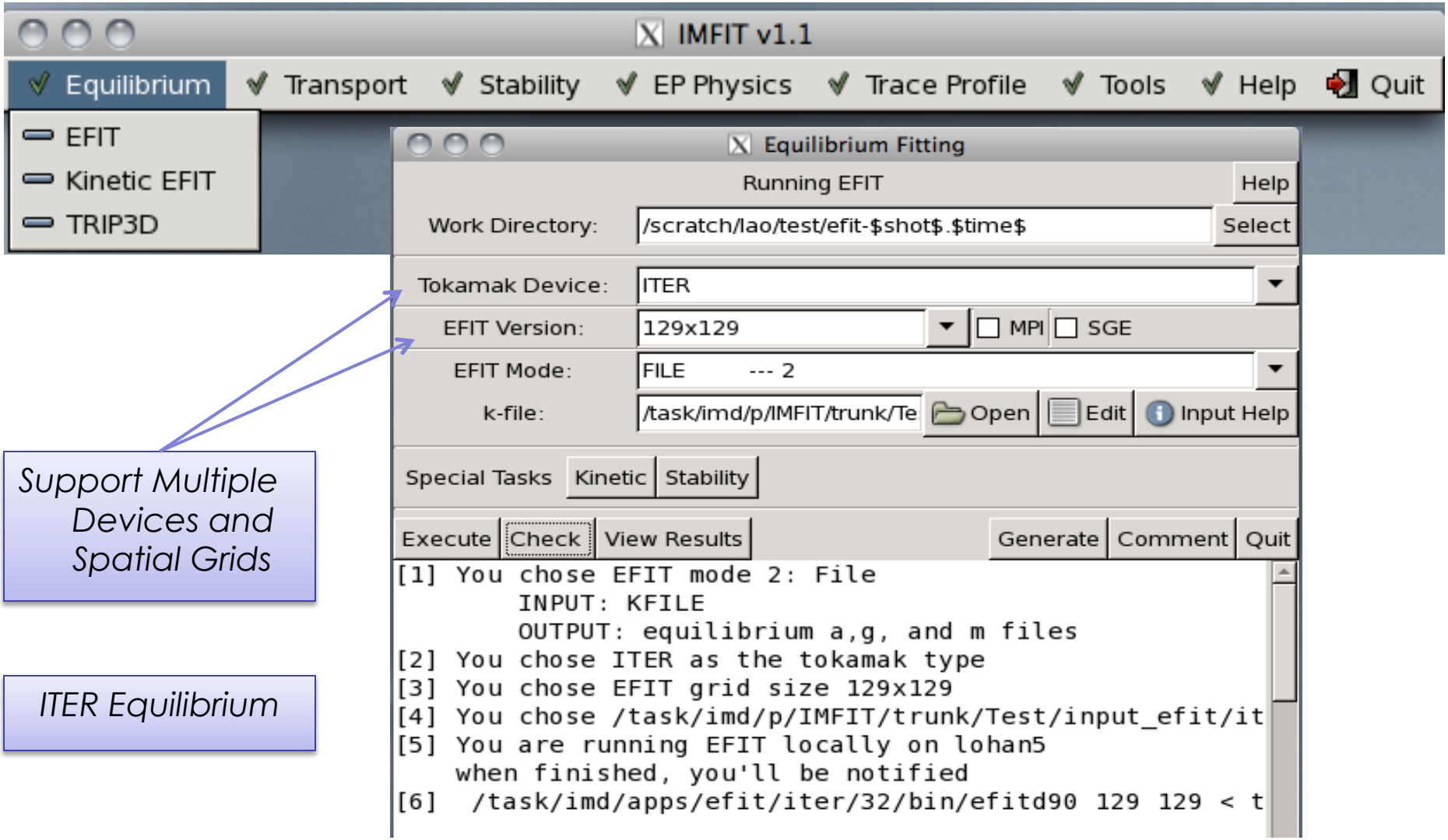




IMFIT Equilibrium Reconstruction



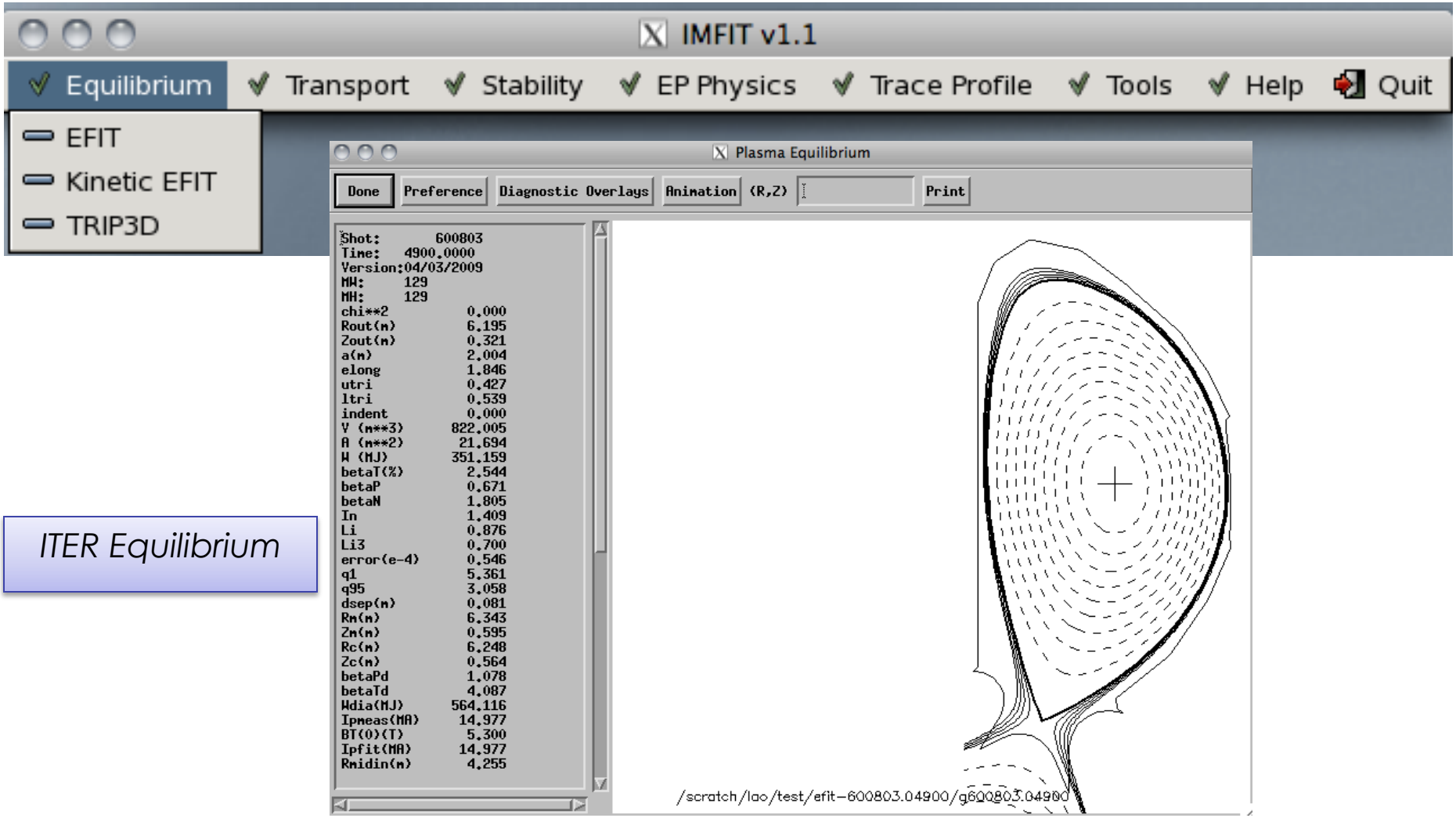
IMFIT Equilibrium Reconstruction



Support Multiple Devices and Spatial Grids

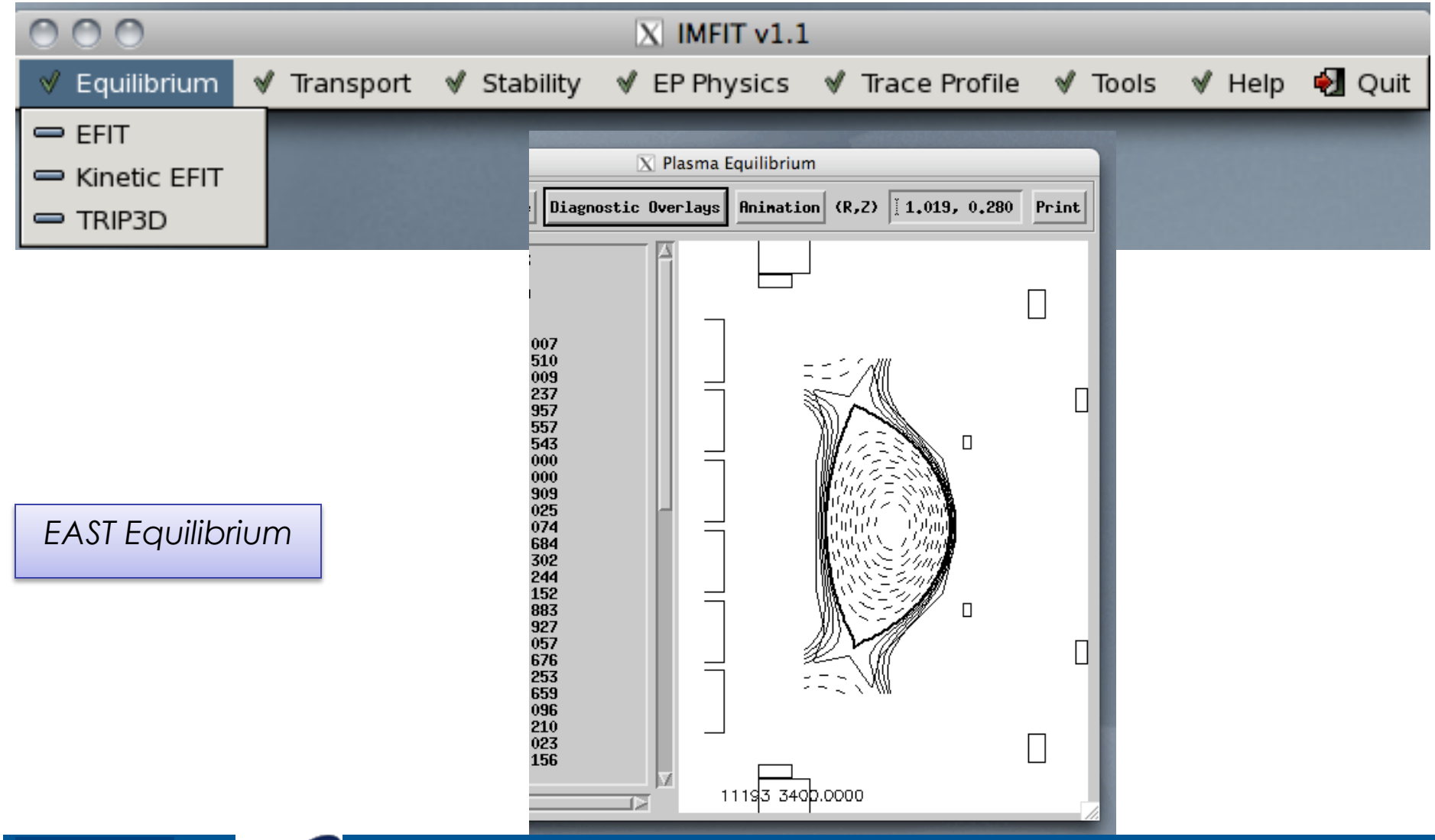
ITER Equilibrium

IMFIT Equilibrium Reconstruction



ITER Equilibrium

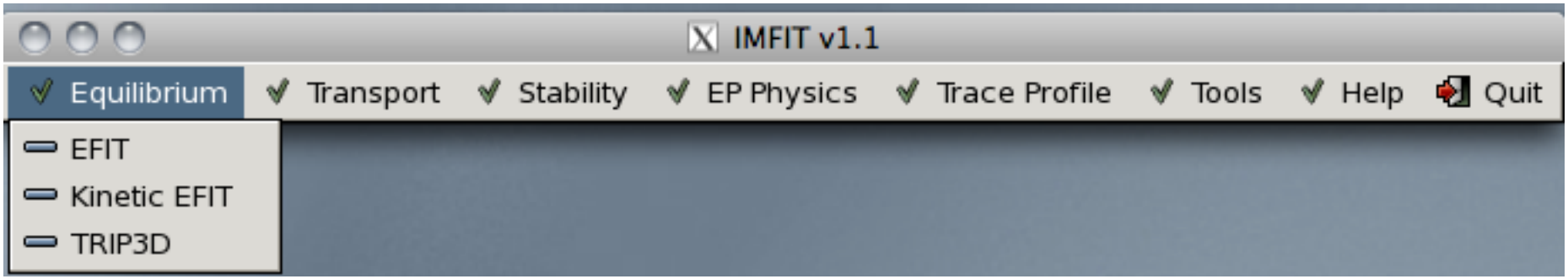
IMFIT Equilibrium Reconstruction



EAST Equilibrium

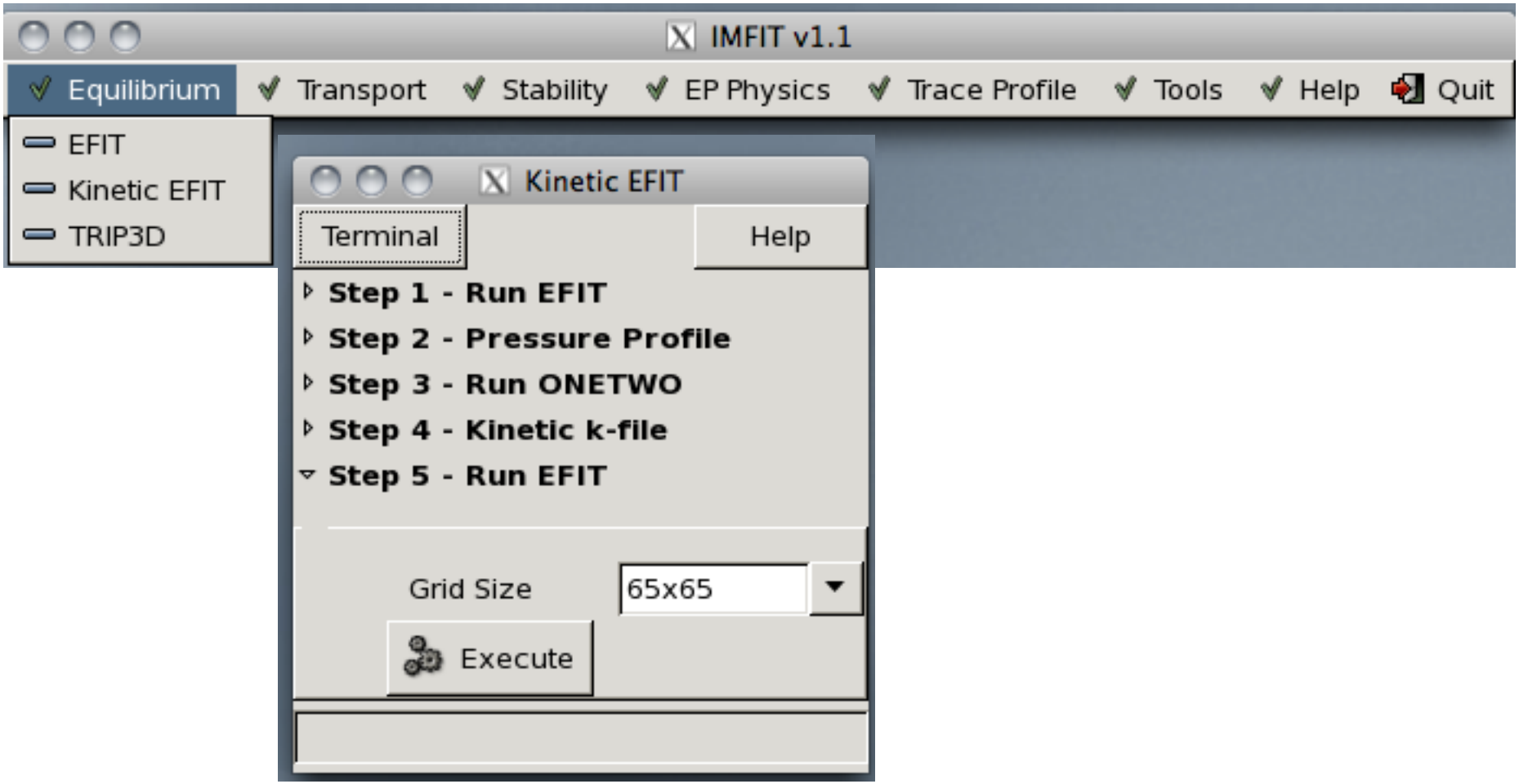
IMFIT Kinetic *EFIT* Reconstruction

Run managed by Equilibrium Manager



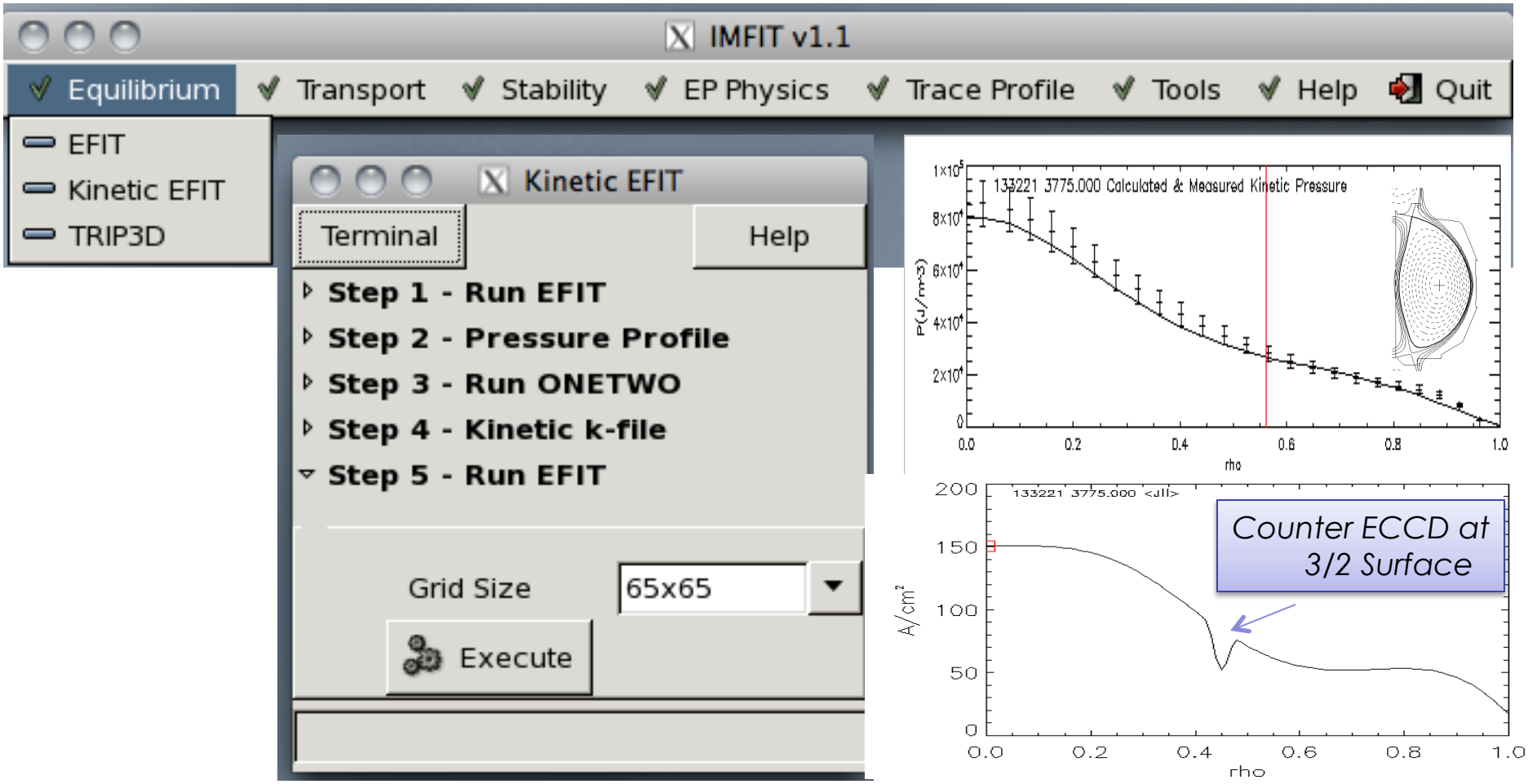
IMFIT Kinetic *EFIT* Reconstruction

Run managed by Equilibrium Manager

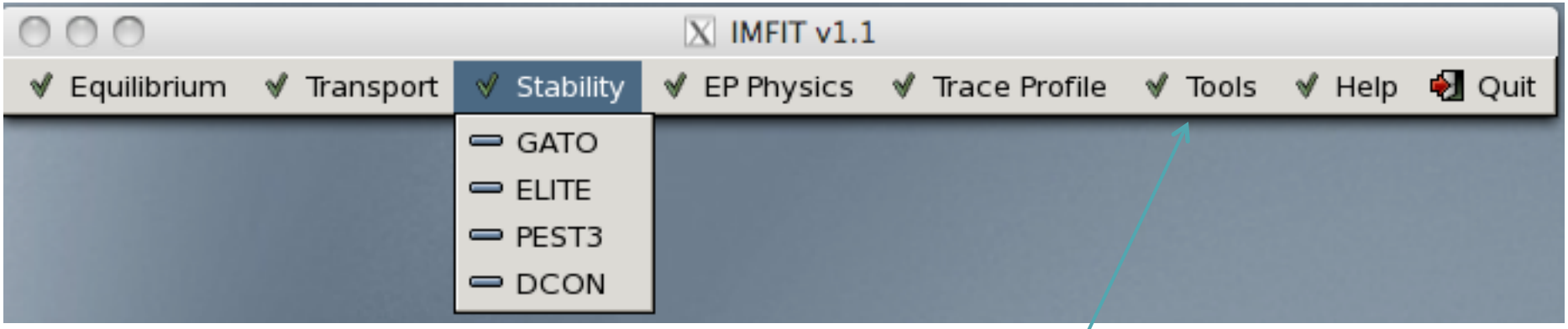


IMFIT Kinetic *EFIT* Reconstruction

Run managed by Equilibrium Manager

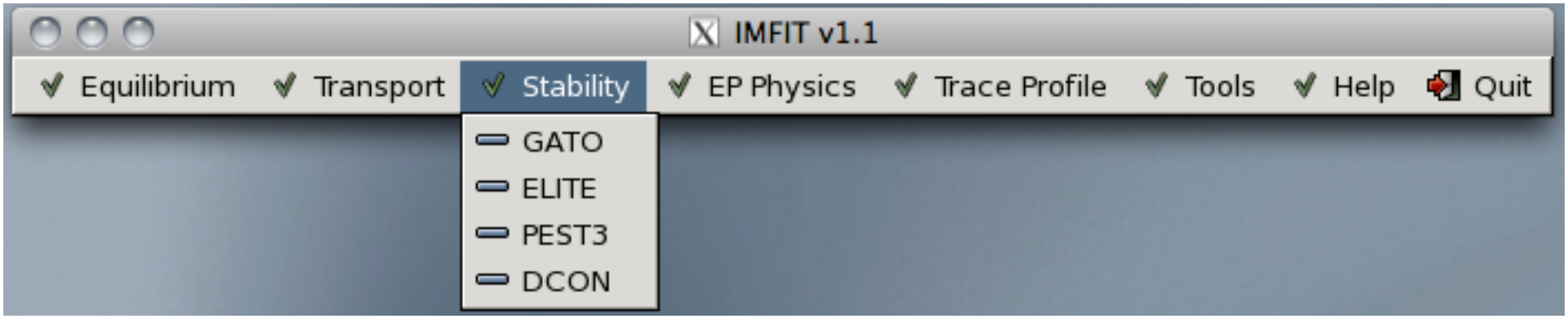


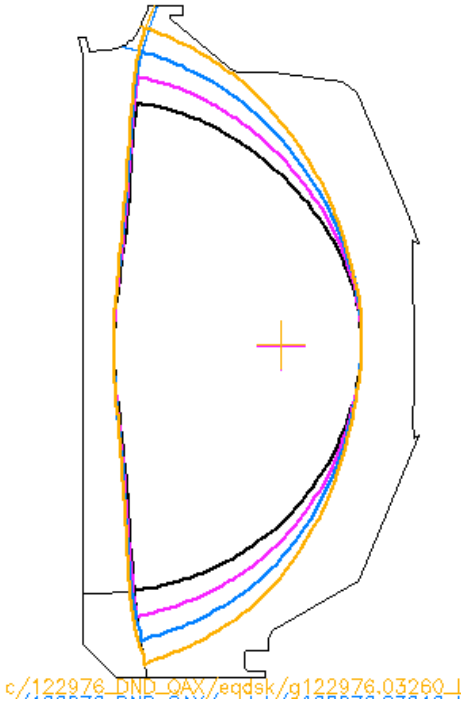
Stability Analyses Can Be Conveniently Carried Out by Interacting with Stability Manager

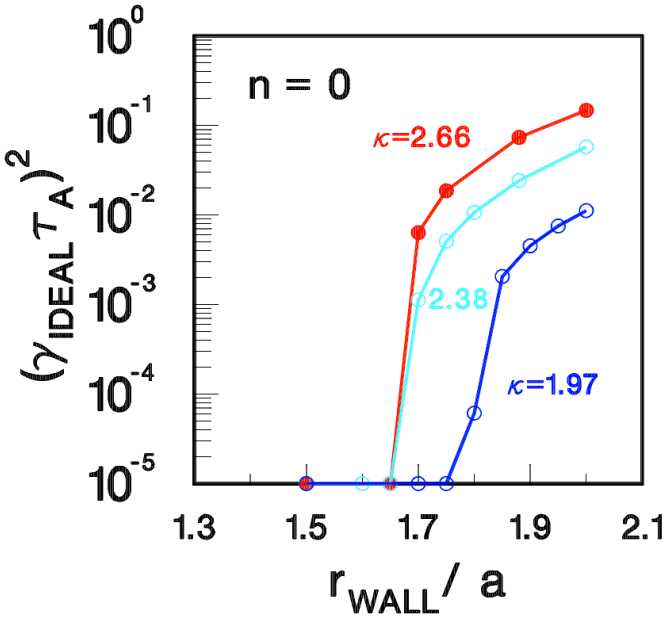


New stability analysis tasks can be conveniently defined using the Task Manager

IMFIT Experimental Stability Analysis



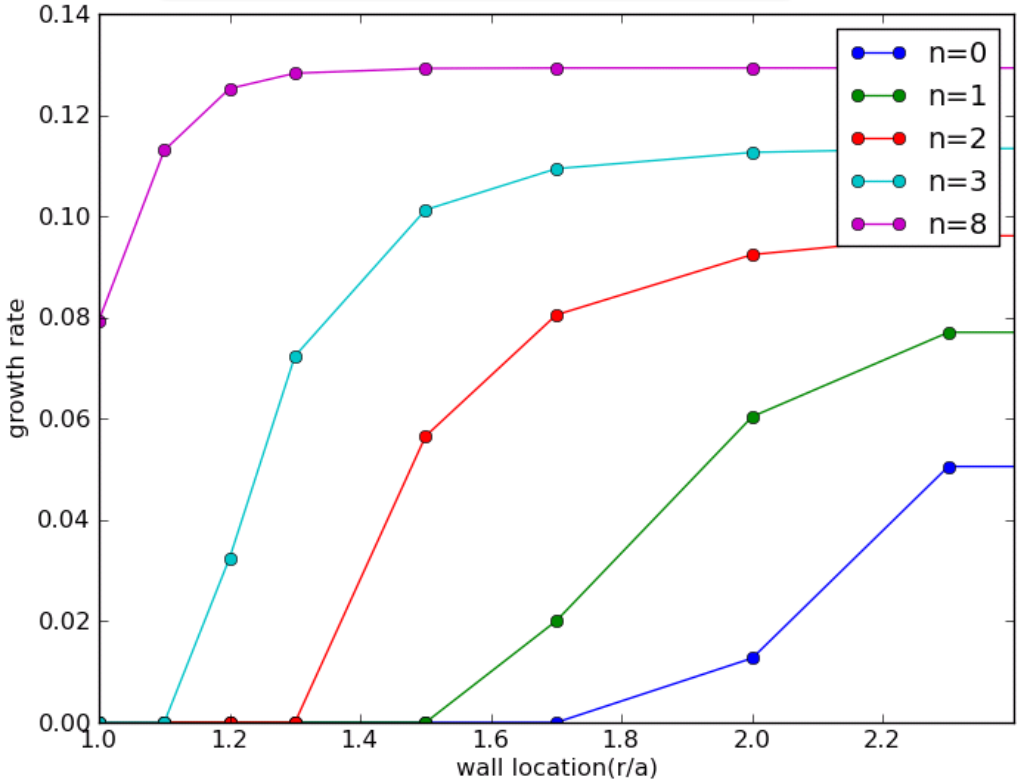




DIII-D Vertical Stability Analysis

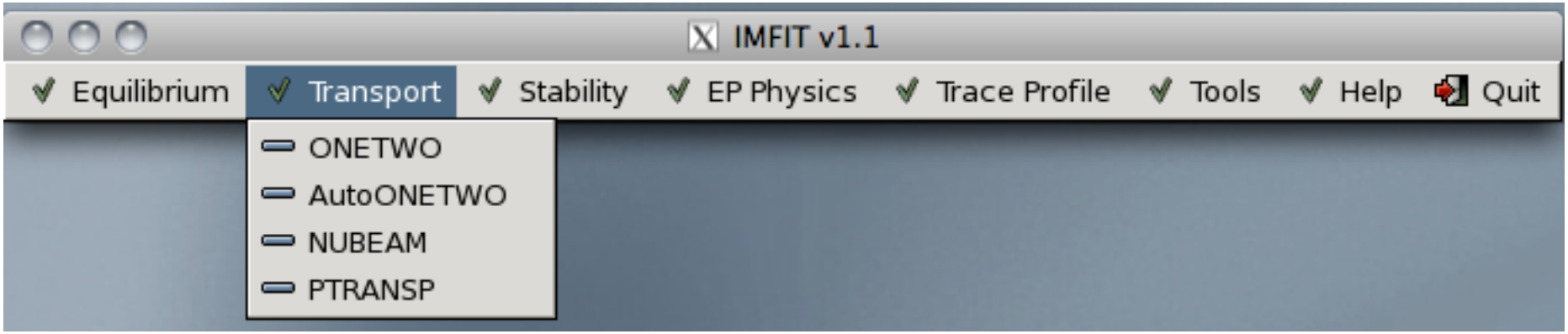
IMFIT Example: Ideal Stability Analysis

Ideal Stability Diagram
EAST AT Scenario

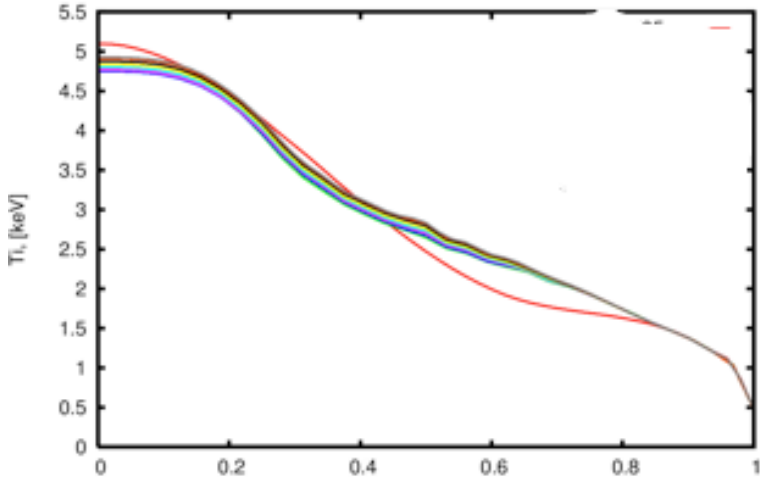


- **Ideal stability diagram tedious to generate**
- **IMFIT tool to allow more routine ideal stability analysis**
- ***IMFIT Task***
  - Kinetic equilibrium reconstruction or model equilibrium
  - GATO stability analysis for different *n* and conducting wall location
  - Visualize results

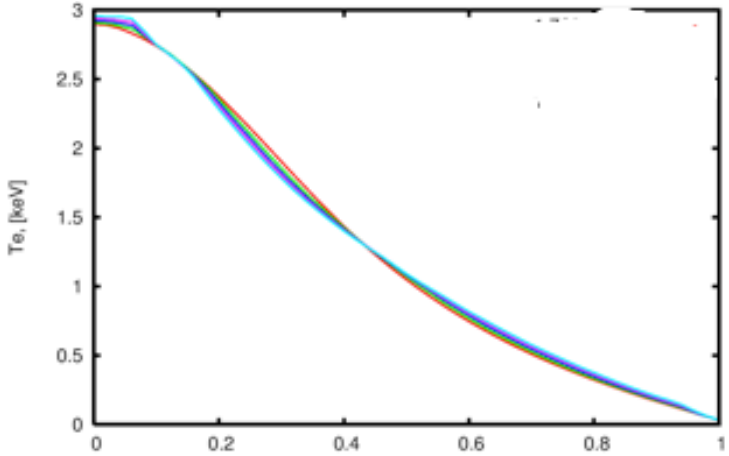
IMFIT Transport Development: *GLF23 / TGLF* Applications



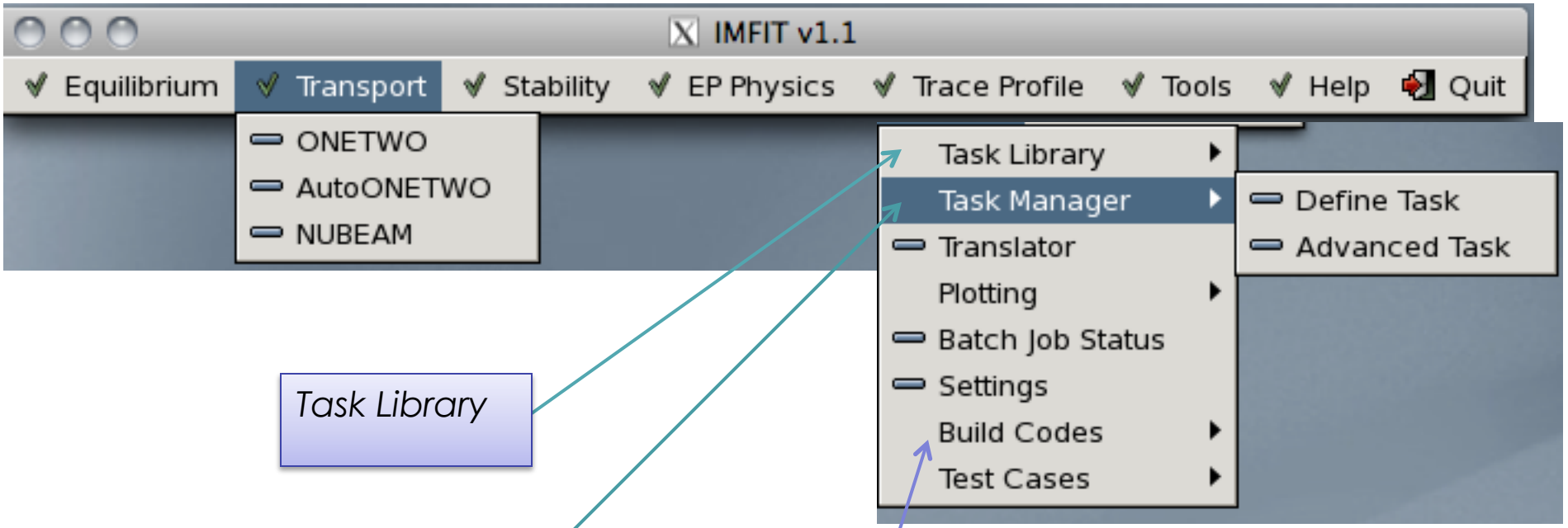
DIII-D H--Mode Simulation



EAST L-Mode Simulation



New Tasks Defined Using Task Manager



Task Library

New tasks can be conveniently defined using the Task Manager

Build and Manage Codes for Different Devices

Summary

- **Both fast automated reconstruction and full experimental analysis are crucial for DIII-D research and operation and are presently supported with a variety of *Fortran*, *Python*, and *IDL* based analysis tools**
- ***IMFIT* is being developed to provide a central platform for integration of experimental analysis**
  - 1st version released: magnetic reconstruction, transport and stability analysis (DIII-D, EAST, SST-1)
  - Ongoing developments: *TGLF*, energy and momentum transport, pedestal physics, EP physics, and 3D reconstruction, compound tasks (core/edge)
- **Experience gained from *IMFIT* useful for ITER Integrated Modeling Programme and FSP**