Progress in the Development of an Integrated Modeling Tool to Support DIII-D and EAST

Q. Ren[1], L.L. Lao[2], M.S. Chu[2], R. Prater[2], H.St. John[2], Y.M. Jeon[5], G. Li[1], W. Guo[1], C. Pan[1],J.M. Park[3], A. Collier[2], G. Abla[2] , R. Srinivasan[4]

[1]Institute of Plasma Physics, Hefei, China
[2]General Atomics, San Diego, CA, U.S.A.
[3]Oak Ridge National Laboratory, U.S.A
[4]Institute for Plasma Research, Bhat, Gandhinagar, India
[5]National Fusion Research Institute, Dajeon, Korea

51st Annual APS Division of Plasma Physics Meeting
Atlanta, Georgia
November 2 - 6, 2007

OUTLINE

- **IMFIT is developed to provide an efficient integration platform to support tokamak research and operation**
- **Current version employs a PYTHON-based framework to manage tasks with a TASK-FLOW based execution model**
- **Analysis codes and tools are integrated into IMFIT through different PHYSICS and SERVICE MANAGERS**
- **Extensive GUI and web-based support are developed for ease of use**
- **IMFIT is being released for DIII-D and EAST**

IMFIT Is Designed To Efficiently Integrate Different Physics Modules To Support Multiple Devices For Data Analysis And Modeling

EQUILIBRIUM *EFIT…*

TRANSPORT *ONETWO…*

STABILITY *GATO…*

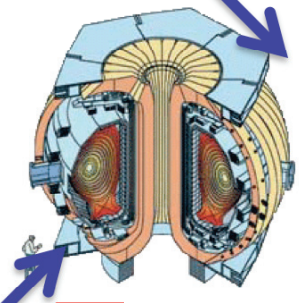
CURRENT DRIVE *NUBEAM…*

…

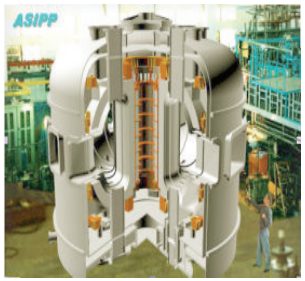
MDSplus netCDF

ITERDB PlasmaState

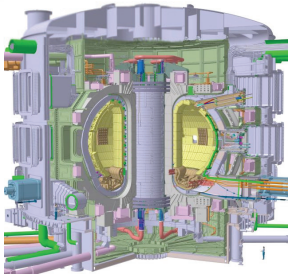
…



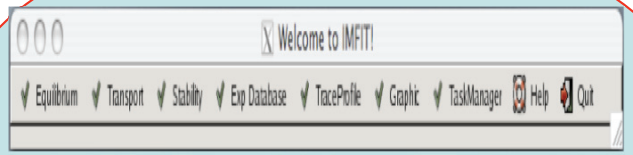
DIII-D



EAST



ITER

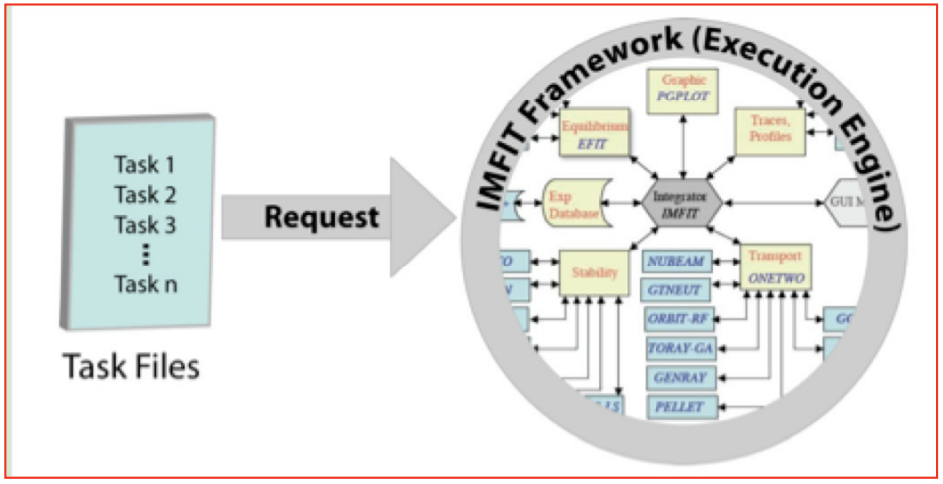


Web-based support

Efficient Integration Platform to Support Tokamak Research and Operation

IMFIT Manages Tasks Using PYTHON Task-Flow Based Framework and Task File

- **Framework is such an architecture that:**
  - provides standard logic for managing various physics codes
  - provides certain rules for development of components
- **Tasks are generally defined in terms of *Task Flow*, *Communication* in Task File**
  - *Compound Task*: combination of simple tasks
- **Framework reads in *Task File* and dynamically generate a sequence of actions**



Simple Task Description

Inward Communication INPUT

↓

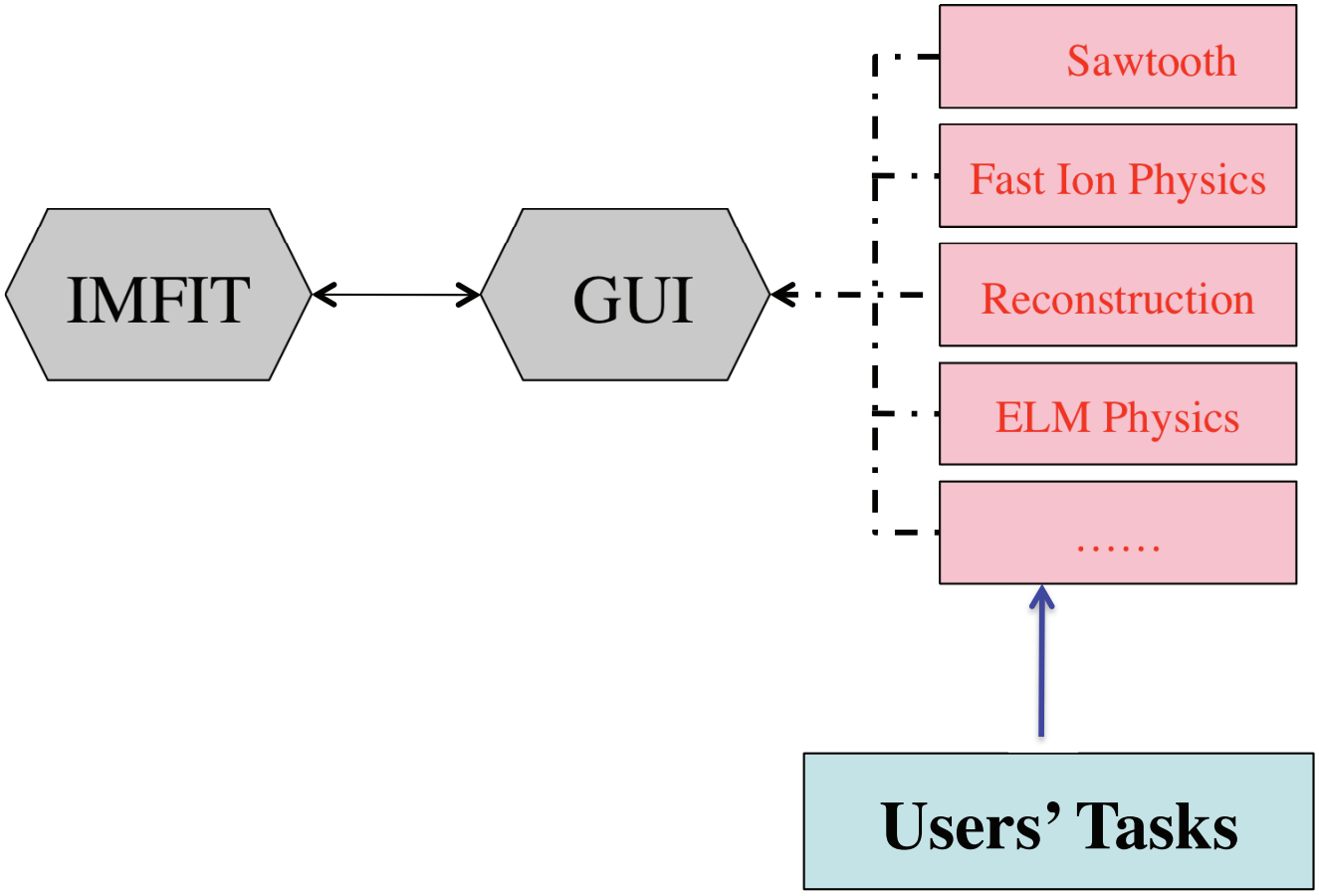
Task Flow
Action 1
Action 2
…..

↓

Outward Communication OUTPUT

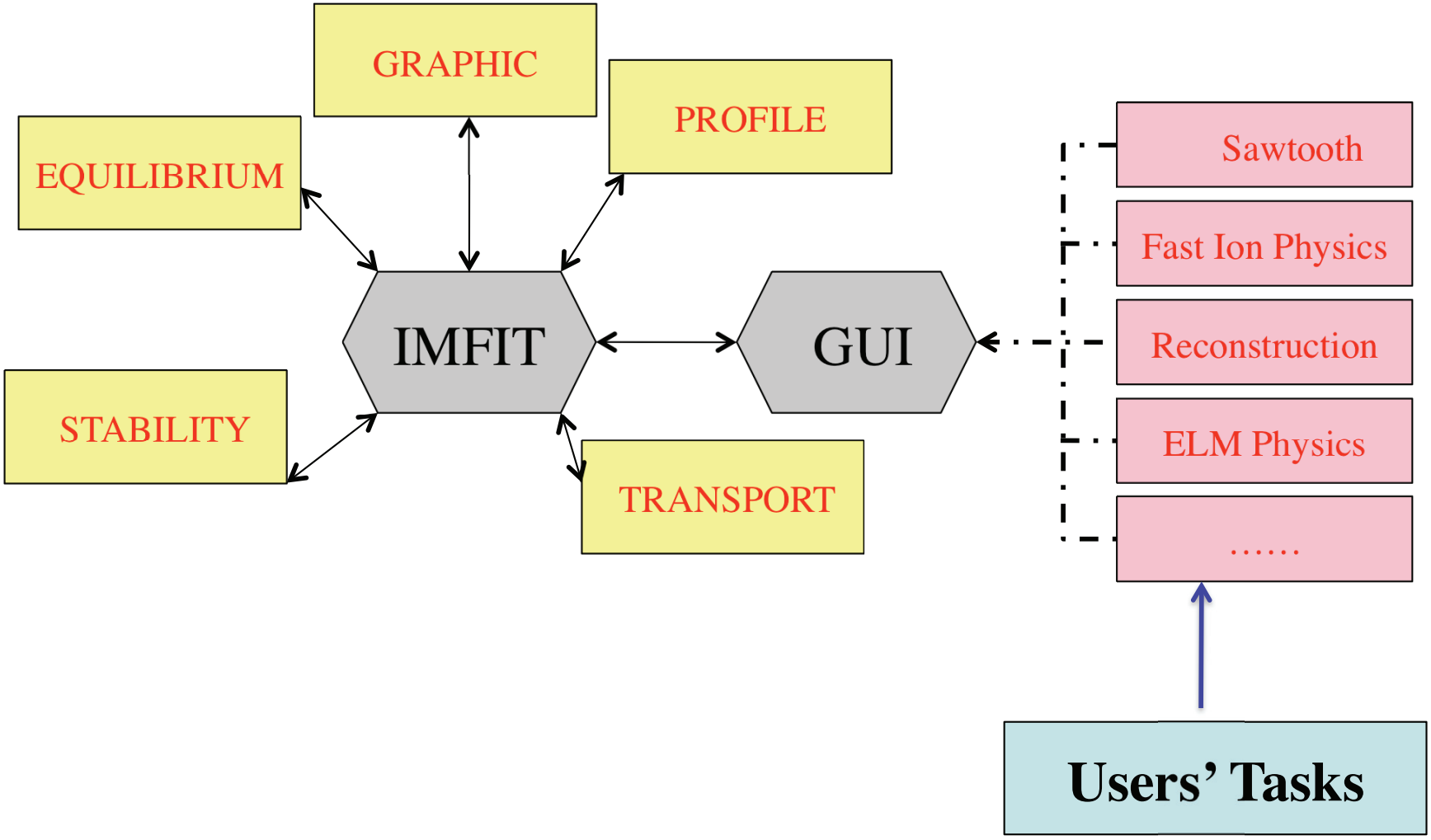
Users Interact With IMFIT Framework Through GUI Manager and Task File

- GUI manager receives the Task and its Task File information
- Task file information is sent to IMFIT Framework for processing and further action



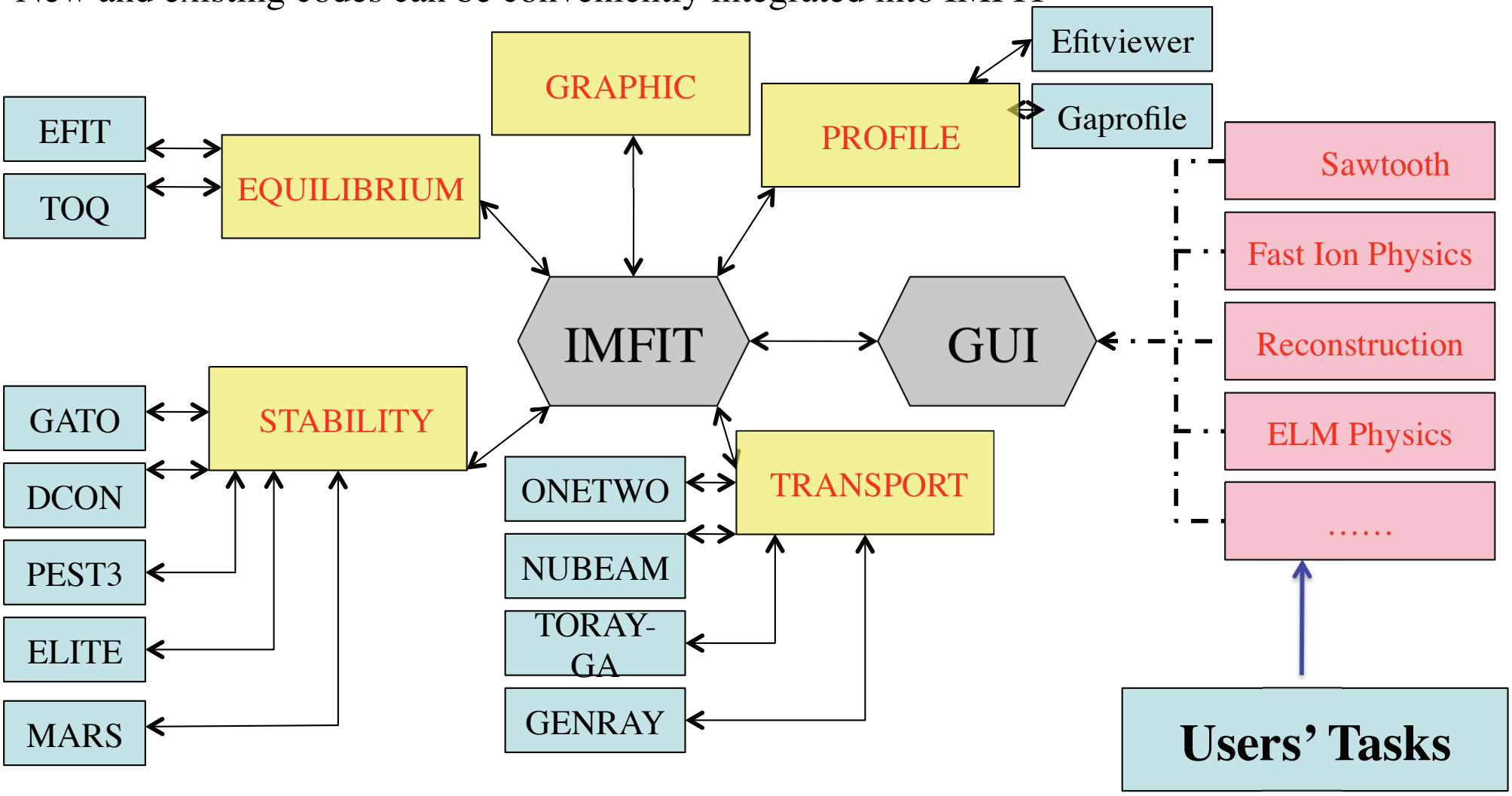
IMFIT Assigns Tasks To Responsible Managers Based on Task File Received

- Each Manager maintains its own Task List and can interact with each other



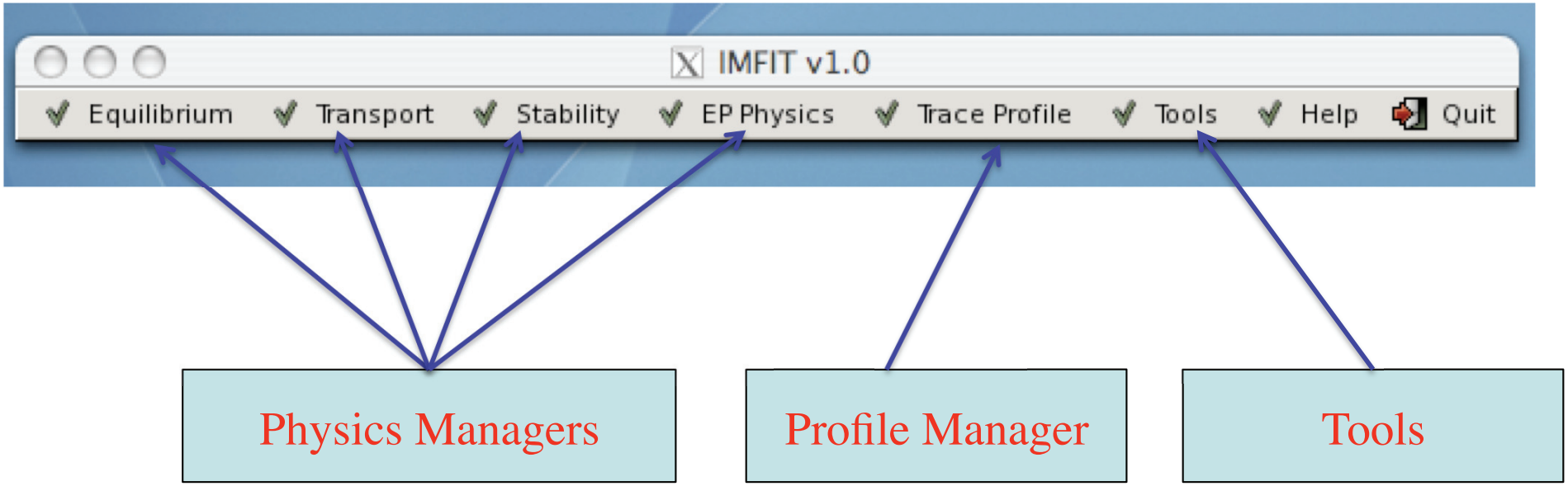
Managers Carry Out Assigned Tasks By Calling Corresponding Component

- New and existing codes can be conveniently integrated into IMFIT



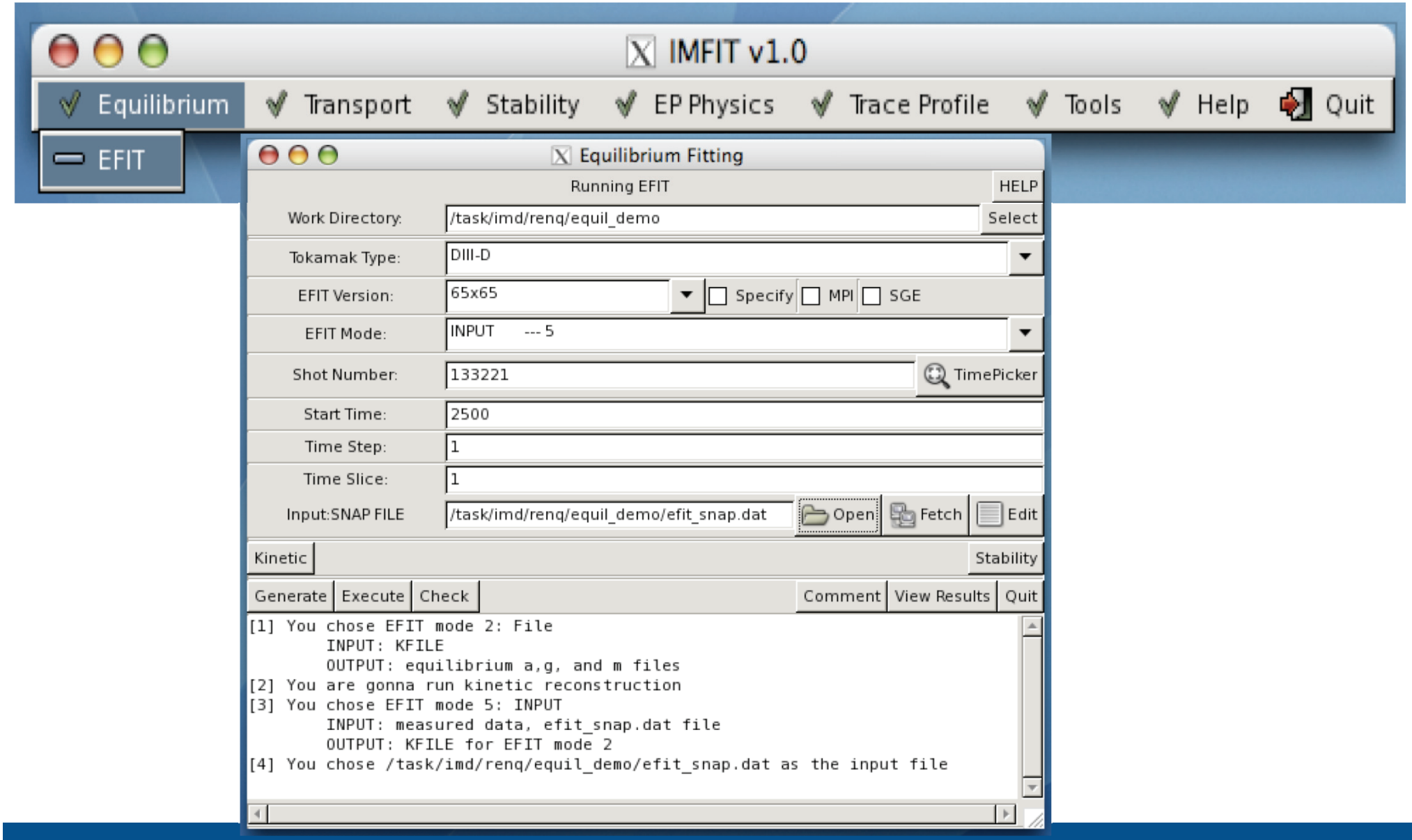
Extensive GUI Support Is Developed For Ease Of USE

Based on PYGTK, open source graphic user interface toolkit for PYTHON

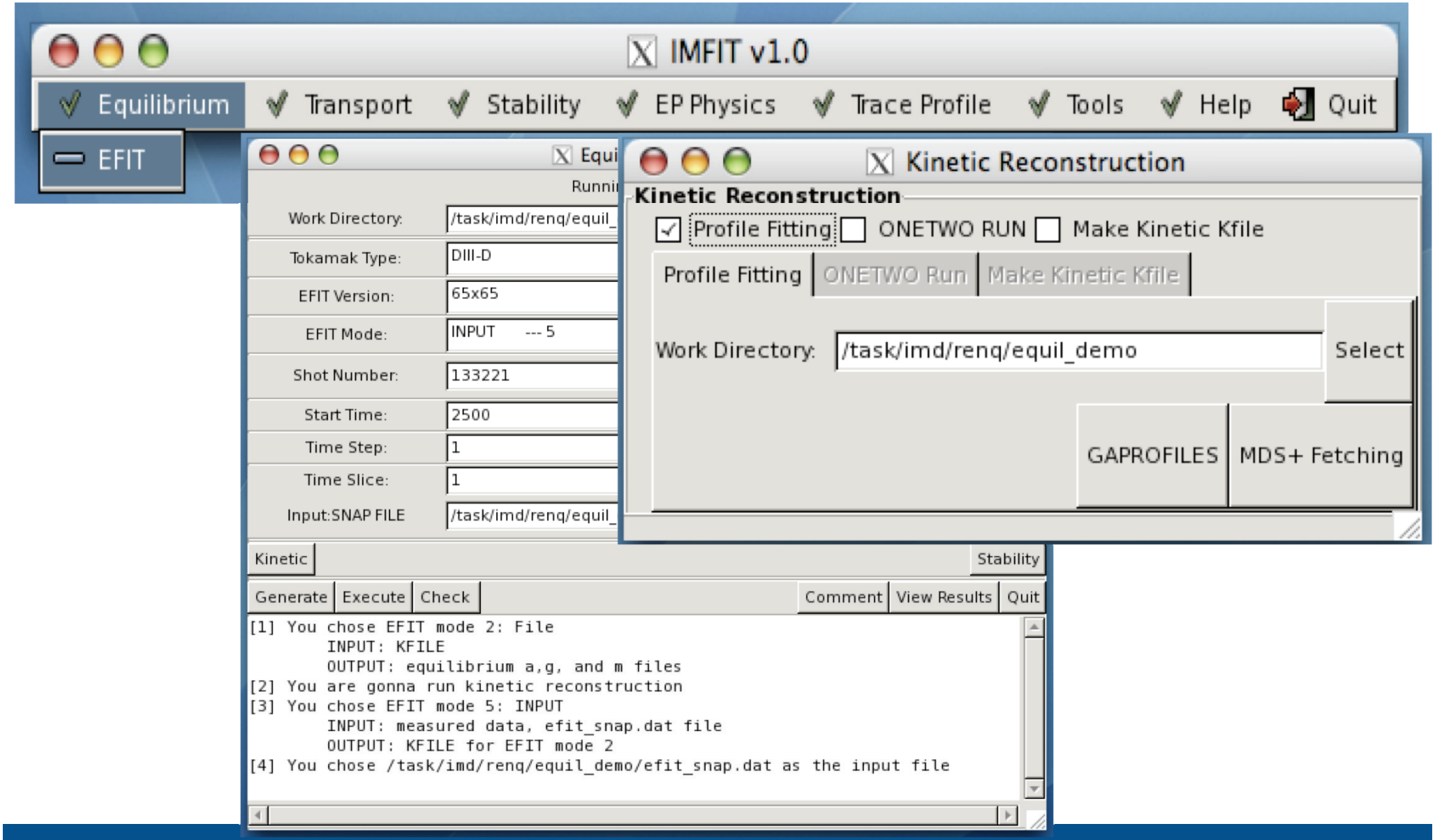


See IMFIT JP8.00086 Poster for more details On Tuesday Afternoon

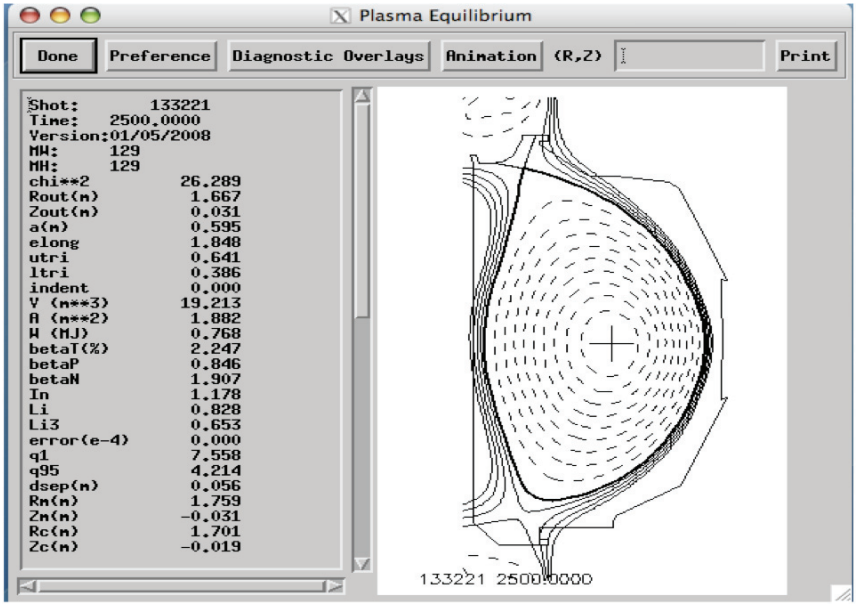
Extensive GUI Support: Kinetic Reconstruction Example

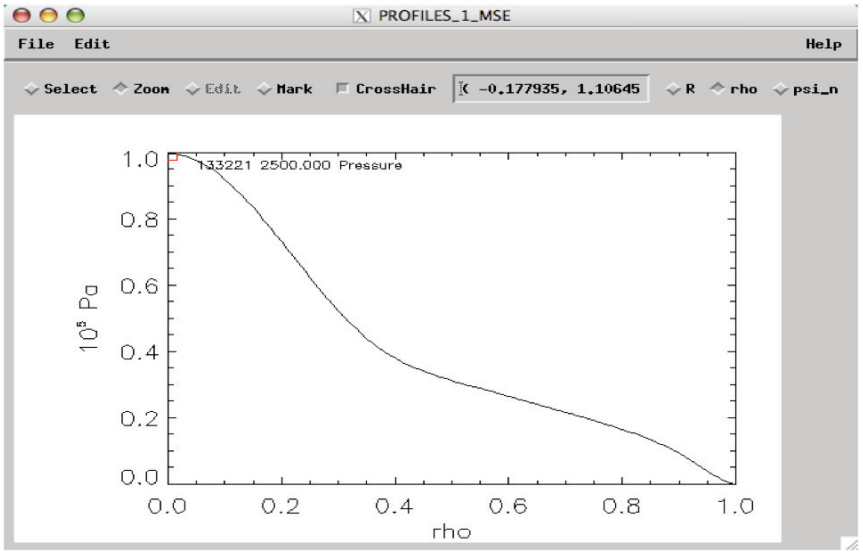


Extensive GUI Support: Kinetic Reconstruction Example



Extensive GUI Support: View Results Using Efitviewer





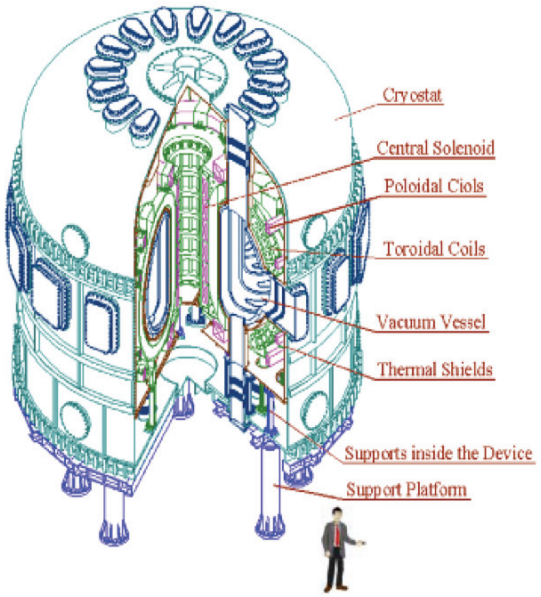
IMFIT Is Applied On EAST For Advanced Tokamak Scenario Development

Main Parameters of the EAST

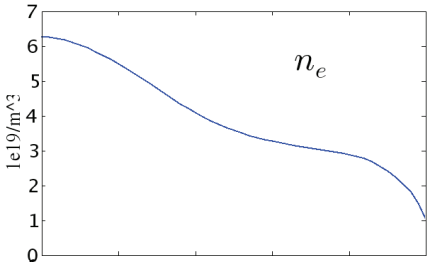
	Nominal	Upgrade
B_0	3.5 T	4.0 T
I_P	1 MA	1.5 MA
R_0	1.7 m	1.7 m
a	0.4 m	0.4 m
R/a	4.25	4.25
K_x	1.2-1.5	1.5-2
δ_x	0.2-0.3	0.3-0.5

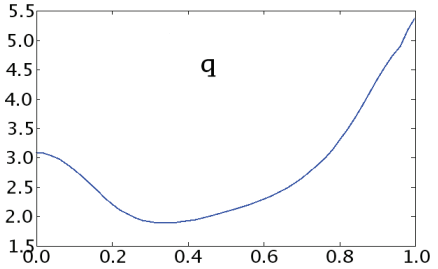
Pulse length 1000 s

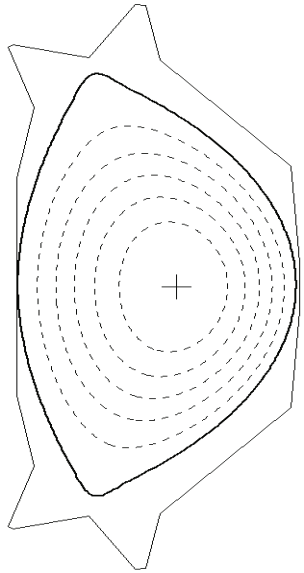
Configuration: Double-null divertor
Single-null divertor

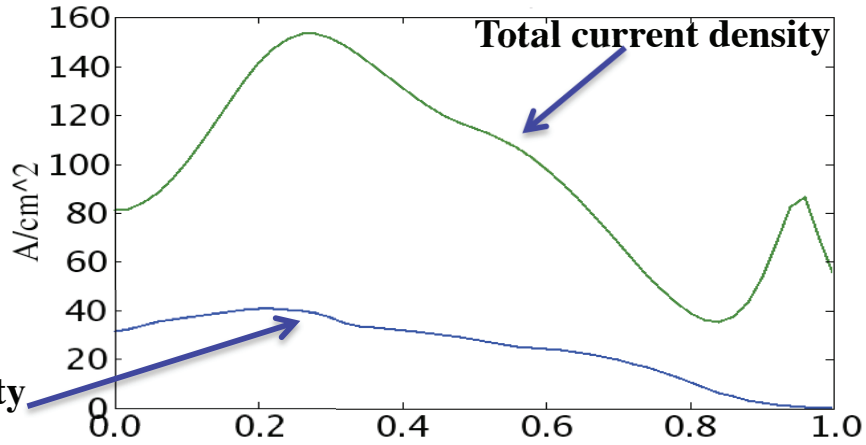


NBI Simulation Using IMFIT









- EAST is moving forward towards research phase
- See Professor Li's invited talk poster BI3.0003 Monday afternoon for more details
- IMFIT is being released for DIII-D and EAST

SUMMARY/FUTURE WORK

An Integrated Modeling and Fitting Tool *IMFIT* Based on Python Is Being Developed to Support DIII-D and EAST Research

Primary Goal:

- To increase experimental data analysis productivity and streamline analysis
- To allow new physics modules to be conveniently integrated to ease theory-experiment comparison

Future Work:

- Development of compound tasks
- Improvement of branching and error handling of Framework
- Improvement of physics codes as well as development of tools to facilitate analysis