

# SWIM Project Workshop

---

**D. B. Batchelor – ORNL**

**Sept. 25-27, 2006**

**Oak Ridge National Laboratory**

**Swim project goals**

**Project status**

**Workshop goals**

# Scientific objectives (from the proposal)

---

- 1 To improve our understanding of interactions that both RF wave and particle sources have on extended-MHD phenomena, and to substantially improve our capability for predicting and optimizing the performance of burning plasmas

*Three physics goals:*

- *Integrated Plasma Simulator for comprehensive long-time simulation of tokamak discharges*
- *Sawtooth instability stabilization and destabilization*
- *Neoclassical tearing mode stabilization*

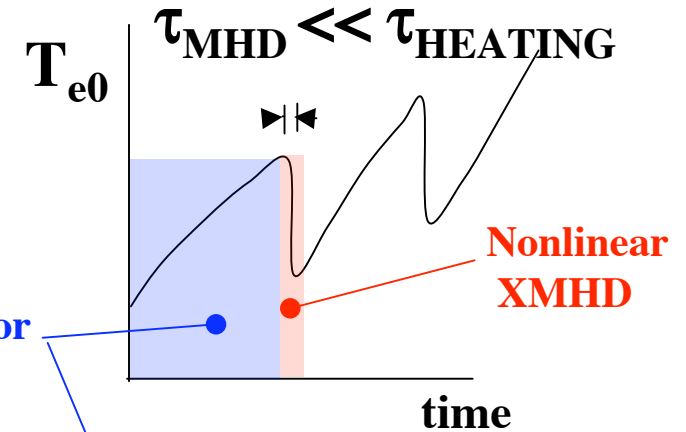
- 2 To develop an integrated computational system for treating multi-physics phenomena with the **required flexibility and extensibility to serve as a prototype for the FSP**, address mathematics issues related to the multi-scale, coupled physics of RF waves and extended MHD, and to optimize the integrated system on high performance computers.

*Our objective is that the community adopt the SWIM architecture as the backbone for computational fusion research*

# We propose two physics campaigns distinguished by the time scale of unstable MHD motion

## Fast MHD phenomena – separation of time scales

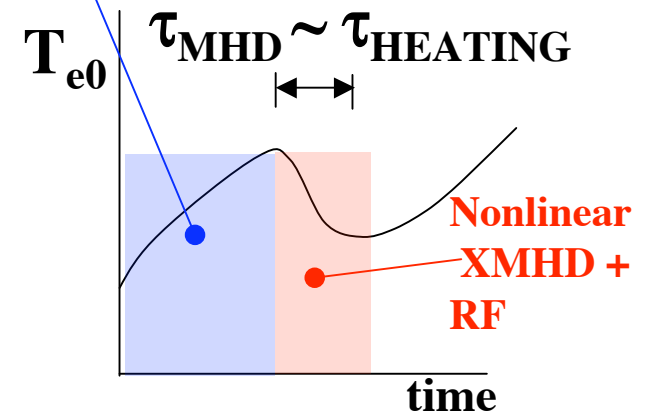
- Response of plasma to RF much slower than fast MHD motion
- RF (mainly ICRH) drives slow plasma evolution, sets initial conditions for fast MHD event
- Example: sawtooth crash



Integrated plasma simulator

## Slow MHD phenomena – no separation of time scales

- RF affects dynamics of MHD events  $\Leftrightarrow$  MHD modifications affect RF drive plasma evolution
- Deals with multi-scale issue of parallel kinetic closure including RF (mainly ECRH)
- Example: Neoclassical Tearing Mode



## These two regimes are related

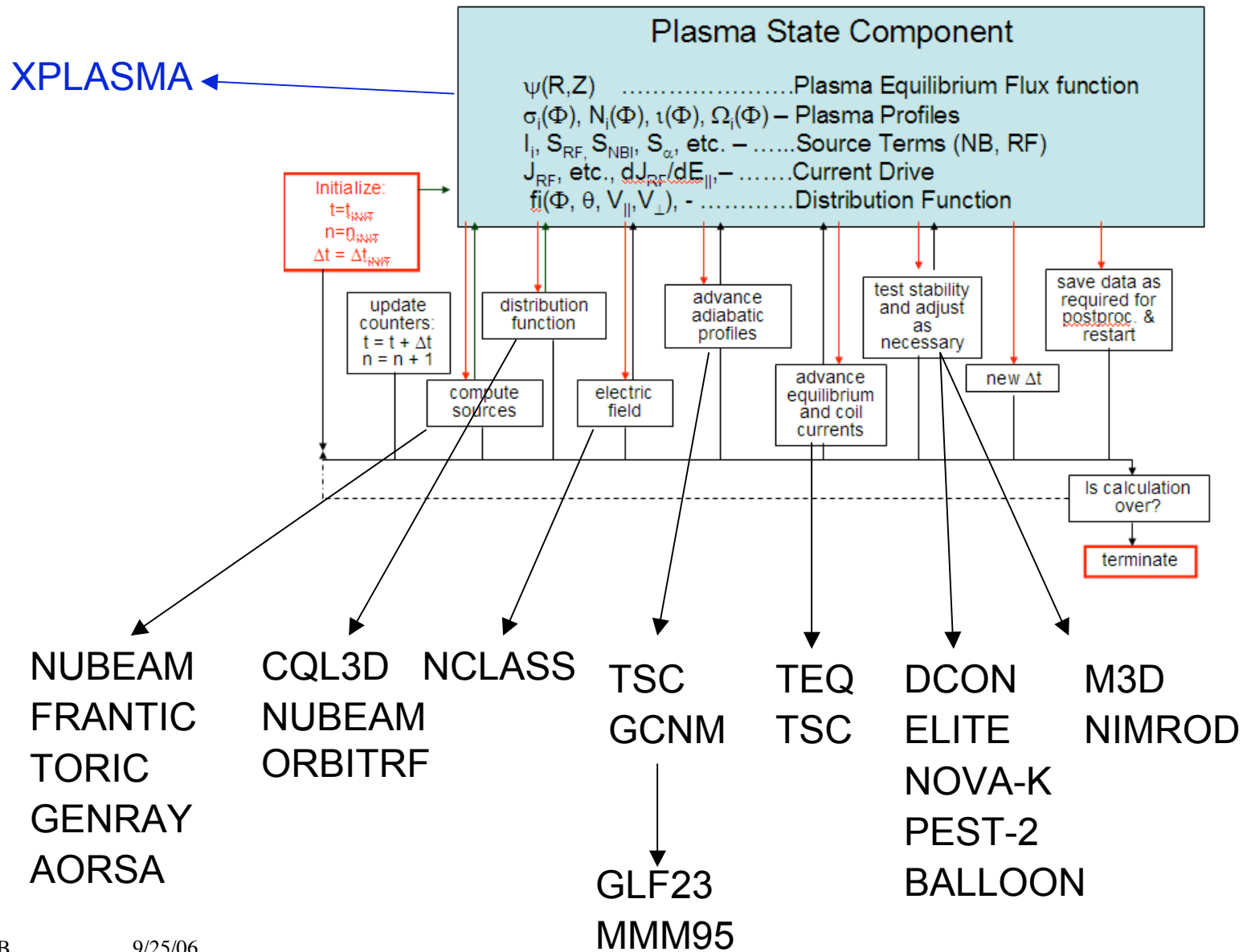
- Fast sawtooth crash can provide seed island for NTM growth
- Slow growth of NTM island can lead to fast disruption events
- Calculation of slow ramp of sawtooth, with incomplete reconnection or persistent islands, may actually require the same capabilities as NTM evolution

# Overall Project Strategy

---

- **Leverage previous investments to maximum possible extent**
  - Fusion SciDAC projects, predictive TRANSP, NTTC
  - Computer Science – SciDAC ISICs, Fusion Grid, user interaction system from LEAD weather prediction system, data management methods from Common Instrument Management Architecture CIMA, authentication technology from [myProxy](#)
- **Component-oriented architecture (*but not formal component system like CCA*)**
- **Design for flexibility and extensibility by abstracting interfaces at a high level – include *multiple codes* in each component from the beginning**
- **Start with a strictly *file-mediated exchange* of data between application components**
  - **No changes needed** to the application component code; SWIM overall interactions handled by scripting
  - Allows **isolating of any bugs** or problems
  - Eliminates issues of re-entrancy and overall system state consistency
- **Highest end computer systems handled from the start, not as an afterthought**
  - Porting codes to Cray XT3 at ORNL

# IPS design – Component based architecture, Plasma State component plays a central role, components implemented using existing fusion codes



# Elements of the integrated Plasma Simulator

## Simulation directory

**/IPS\_run\_XYZ/**

---

**/Simulation\_setup/**  
Simulation definition  
Initial plasma state  
Code specific initialization

**/work/**  
**Current plasma state data**  
<Component>\_/work/  
Configuration files  
Scripts and executables  
Input files  
Code scratch files  
Output files  
  
<Other components...>

**/Simulation\_results/**  
Plasma state history  
Current component restart  
Component history

## Simulation Driver

### IPS Driver Script

Computer environment initialization

Plasma state initialization

Initialization of all components

**Advance Sequence: [t → t+Δt]**

Evaluate Simulation Control Logic

Step all components (e.g. RF\_step())

Evaluate results of time step  
(terminate, or adjust and restart current step, or accept current time step)

Commit data from time step to /Simulation\_results directory

Finalize simulation

## Typical component

### <Component> script

<Component>\_initialize

<Component>\_step  
prepare <code> input  
run <code>  
process <code> output

<Component>\_finalize

### Framework

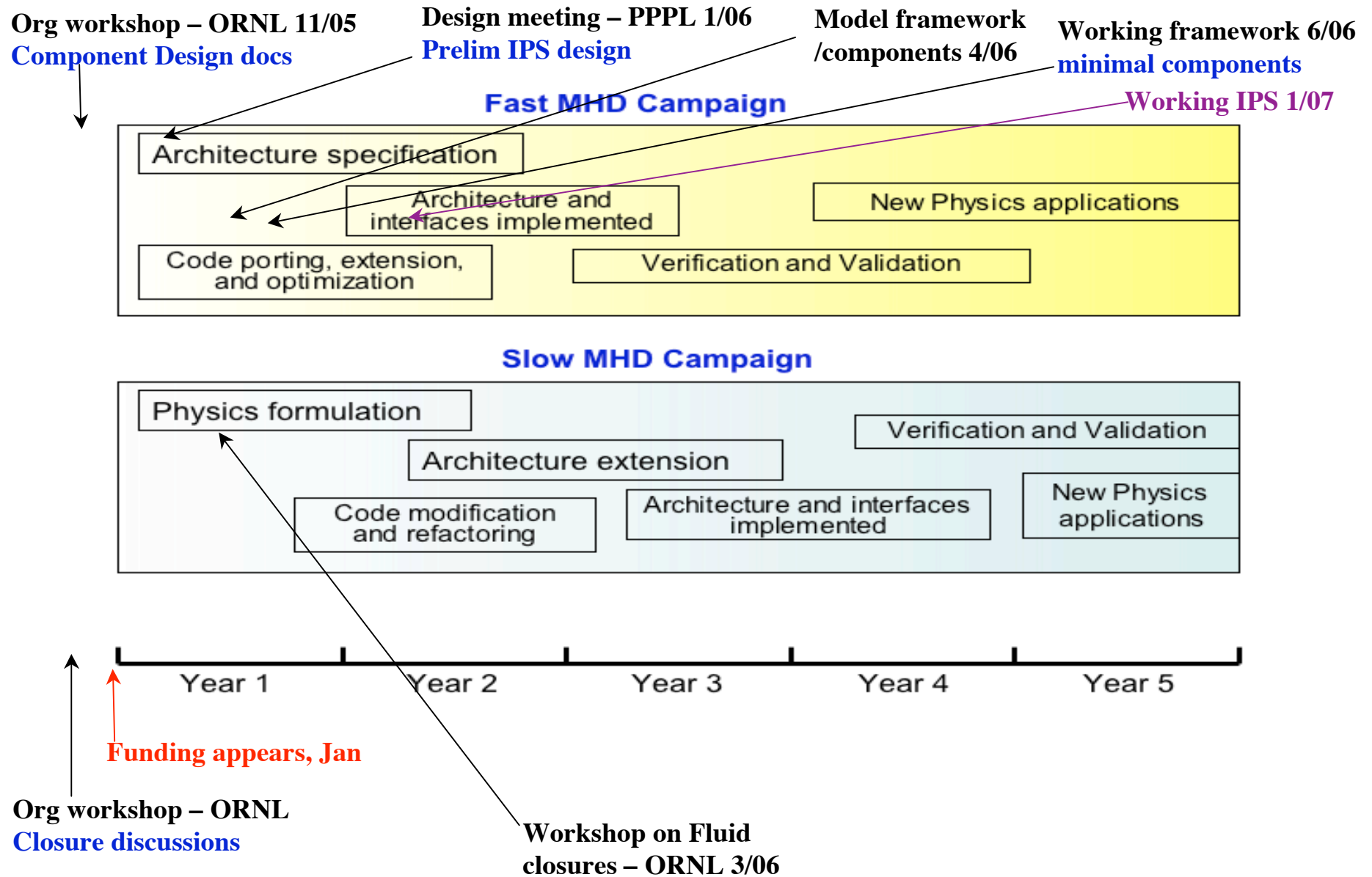
CS services  
Portal  
Job manager  
Data manager  
Event channel

Physics services  
Re-gridding  
Time interpolation  
Mathematical libraries  
...

## Plasma State Component Functions

PS\_GET\_PLASMA\_STATE  
PS\_STORE\_PLASMA\_STATE  
PS\_COMMIT\_PLASMA\_STATE

# SWIM Work-plan and Timeline



# Agenda for SWIM workshop (today, 9/25)

---

**9:00 – 9:30**

**Review of project goals, project status, and workshop goals (Batchelor)**

**9:30 – 2:00 (lunch break 12:00 to 1:00)**

**Discussion of physics and mathematics/algorithmic issues – what physics don't we know? what gaps in mathematics or algorithms stand in the way of computing what we want to compute? (Jardin to moderate)**

**Physics/algorithm issues for all components, what are the computational requirements e.g. parallelization?**

**Objective: Develop list of needs/tasks – physics theory, mathematics, algorithms/libraries**

**2:00 – 5:30**

**Discussion of CS status (part I) (Bramley to organize) Repository and how we see it being used (30 min, Bernholdt)**

- **Codes, components, scripts, and portals: these are and how they interoperate (45 minutes, Bramley)**
- **The Plasma State Component and how it's to be used (45 minutes, Doug McCune)**
- **John Carey (30 min + 15 min) FACETS project**
- **Update on IPS design and IPS workflow (Batchelor 45 min)**

**Objective: familiarize team with IPS approach as presently conceived, expose issues, identify tasks, develop list of needs/tasks**

# Agenda for SWIM workshop (Tuesday, 9/ 27)

---

**9:00 – 11:00**

**Discussion of CS status (part II) (Bramley to organize)**

- RF component implementation (Berry)
- Data transfer for inter-component communication
- SWIM utilities and aids (build system, data management, ... (Ma)

**11:00 – 12:00**

**Slow MHD campaign and NTM roadmap (Kruger)**

**1:00 – 3:00**

**Progress on component scripts and implementing codes (component and code developers).**

**Are the codes ready? – ported, version controlled, documented, regression tests run, submitted and documented**

**Objective: identify roadblocks or gaps in component development**

**3:20 – 5:30**

**Discussion of unresolved simulator issues (Bramley to organize)**

**Short term issues (Elwasif)**

- *dividing tasks, coordination, and coherence between scripting levels*
- *Simulation driver logic and run setup, input/output file staging, Interaction of controller/driver and components with framework*
- *Control - handling of preprogrammed waveforms, feedback control, error recovery*

**Medium to longer term issues (Kohl)**

- *Authentication and job launch by software agents (portal, scripts, ...)*
- *Handling multiple time steps with single batch queue wait*
- *Shifting to in-memory transfers rather than file-based*

**Objective: resolve issues or develop plans to resolve them, list tasks**

# Agenda for SWIM workshop (Wednesday, 9/27)

---

**9:00 – 12:00**

**Develop a set of specific simulation targets**

**Discussion of issues and tasks.**

- **Physics issues and tasks (Jardin to compile and present)**
- **CS issues and tasks (Bernholdt)**

**Drafting of FY07 goals, work plan and schedule**

- **Draft FY07 goals and deliverables by time of PSACI PAC meeting. June 2007 (Batchelor)**
- **Informal presentation of individual's work plans**

**12:00 Adjourn**

## Things we have to do

---

- **Identify any outstanding physics or algorithmic issues and come up with a plan to resolve them (especially the Slow MHD campaign)**
- **Fix (i.e. make stationary) the interface to the Plasma State**
- **Fix the simulation directory structure**
- **Fix the driver script and define the interactions between the physics functions and the CS functions, i.e. framework**
- **Fix the simulation setup and operation procedures**
- **Confirm that the physics components are in hand and that people are prepared to implement them.**
- **Fix the specific simulation targets we will carry out**
- **Identify the deliverables for the year (and outline them for the 3 year project extent)**

# Backup material

---

# IPS components and implementing codes

Component	Responsible person	Implementing code	Responsible person
1) Plasma state	McCune	XPLASMA	McCune
2) Energetic neutral source	McCune	NUBEAM	McCune
3) Neutral gas fueling source	McCune	FRANTIC	McCune
4) RF Field source	Batchelor	AORSA	Jaeger
		TORIC	Bonoli
		GENRAY	Harvey
5) Fokker Planck solution	Berry	CQL3D	Harvey
		NUBEAM	McCune
6) Neoclassical and bootstrap	Houlberg	NCLASS	Houlberg
7) Advance profiles	Jardin	TSC-solver	Ku
		GCNM	St. John
8) Advance equilibrium	Jardin	TSC-equilibrium	Ku
		TEQ	LoDestro
9) Anomalous transport coeff	Jardin	GLF23	Ku
		MMM95	Ku
10) Linear Stability	Kruger	DCON (low-n)	Kruger
		ELITE (edge)	Snyder
		NOVA-K (kinetic)	Gorelenkov
		PEST-2 (low-n)	Chance
		BALLOON (high-n)	Chance
11) Reduced Sawtooth model	Jardin	Porcelli	Bateman
12) 3D nonlinear MHD	Kruger	M3D	Breslau
		M3D/K	Fu
		NIMROD	Schnack
13) Waveform and feedback			

# Proposed teams to define the blocks and interfaces (WBS elements)

---

- RF: (AORSA, TORIC, GENRAY)
  - Bonoli (Bernholdt), Jaeger, Berry, Batchelor, Harvey
- NBI: (NUBEAM, ...)
  - McCune, , Houlberg,...
- Fueling: TBD
- Distribution function module (Monte Carlo (various), DKES, CQL3D)
  - Berry (Kohl), Batchelor, Hirshman, Spong, Harvey, Bonoli McCune, Fu, Gorelenkov, Schnack, Choi, Chan
- Evolve Profiles (GA Solver = GCNM, TSC):
  - Jardin (Bramley), St. John, Ku, Houlberg
- Equilibrium (incl TSC and TEQ)
  - Jardin (Bramley), Lodestro, McCune, Carlson, Kruger
- Linear stability (PEST, Nova-K, Linear NIMROD...)
  - Kruger (Schissel), Schnack, Gorelenkov, Spong, Sanchez, Jardin, Hegna, Chance, Fu
- Reduced stability models (sawtooth, NTM)
  - Hegna, Jardin, Bateman, Spong
- Nonlinear stability (NIMROD, M3D)
  - Schnack(Klasky), Fu, Breslau, Kruger, Sanchez, Strauss, Sugiyama, Spong
- Plasma state (XPLASMA, AJAX, LDRD architecture, ...)
  - McCune, Houlberg, St. John, Batchelor, Kruger, Kohl, Lodestro, Jardin, Berry

# Physics and mathematical analysis – Fokker-Planck, kinetics, and closure

---

## Fast MHD campaign

- **Formulation of fluid equations for advancing profiles**
  - Multiple plasma species
  - Non-Maxwellian populations
  - Plasma rotation
- **Rigorous calculation of  $j(E_{||})$  including RF, bootstrap, beams and ohmic**
- **Fokker Planck solutions**
  - Include radial transport in CQL3D
  - How to transform between orbit averaged  $f$  and  $f(x, v_{||}, v_{\perp})$
  - Representation of Monte Carlo orbit's in terms of distribution functions that RF and MHD codes understand

## Slow MHD campaign

- **Analyze consequences of Slow MHD campaign on design of EEM(/IPS) – e.g. 3D**
- **MHD closure with RF**
- **Defining equilibria with islands**
- **...**

**Examine literature to determine if good solutions already exist, outline research needs, agree on nearer-term compromises to get started**

**Team: Hirshman, Keyes, Ramos, Hegna, Fu, D'Azevedo, Harvey, Spong, Sanchez, Schnack, Batchelor ... (essentially all physicists)**

# What do these teams need to do? Answer the 7 questions

---

## 1) Specify the functionality for the component

- What physical processes does this component model?
- Mathematical description at a high level. Ideally the most general form of the equations we would ever envision for the component, not what any particular codes solves at present.
- Identify codes to be used initially and document approximations to the ideal

## 2) Determine the interface to provide this functionality

- Generic form of the inputs. The information required to solve the general equations no matter what particular code provides the implementation.
- Specify the generic form of the outputs. The information this component will provide to other components
- Type of spatial discretization used for each field – dimensionality and geometry
- Interfaces provided – callable subroutines of the component, and interfaces used (subroutines called by the component)

## 3) Identify code specific data

- Specific inputs – grids, switch settings, parameters
- Identify code state data needed for restart
- Identify other code outputs (that might be useful for diagnostics or visualization for example) but that doesn't specifically enter into the simulation
- Determine where the required inputs (both generic and code specific) are going to come from – Plasma State or other components, external file, ...

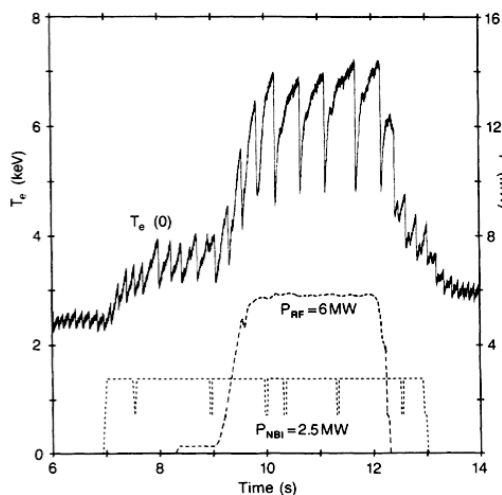
## More questions

---

- 4) Expectations about input data. e.g., that equilibrium has been reached, or that numerically conservation has been maintained.**
- 5) Shared infrastructure components used – such as global time-keeper, interpolators from one representation to another, flux surface average routines, storage of the component's state for restart.**
- 6) Physics or mathematical analysis/development needs**
  - opportunities for improvement
  - additional opportunities for parallelism that are not presently realized.
- 7) Computational needs this component will have when used for the fast MHD campaign.**
  - Including computations
  - Memory requirements
  - Number of calls or total time to carry out its designated task in a typical IPS simulation
  - Kind of parallelism (threaded or distributed memory or both).

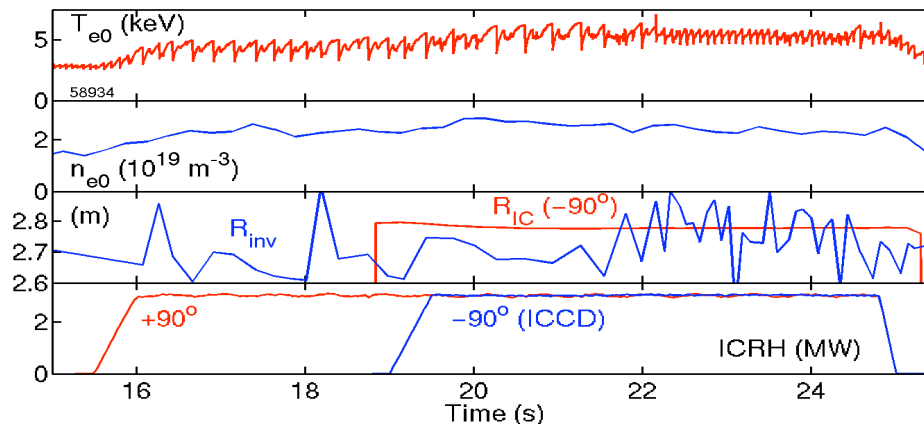
# There are several experimentally demonstrated mechanisms by which RF waves can control sawtooth behavior

## ICRF stabilization on JET



- ICRF heating can produce “monster” sawteeth – period and amplitude increased
- Likely stabilization mechanism – energetic particle production by RF

## Sawtooth control on JET with Minority Current Drive on JET

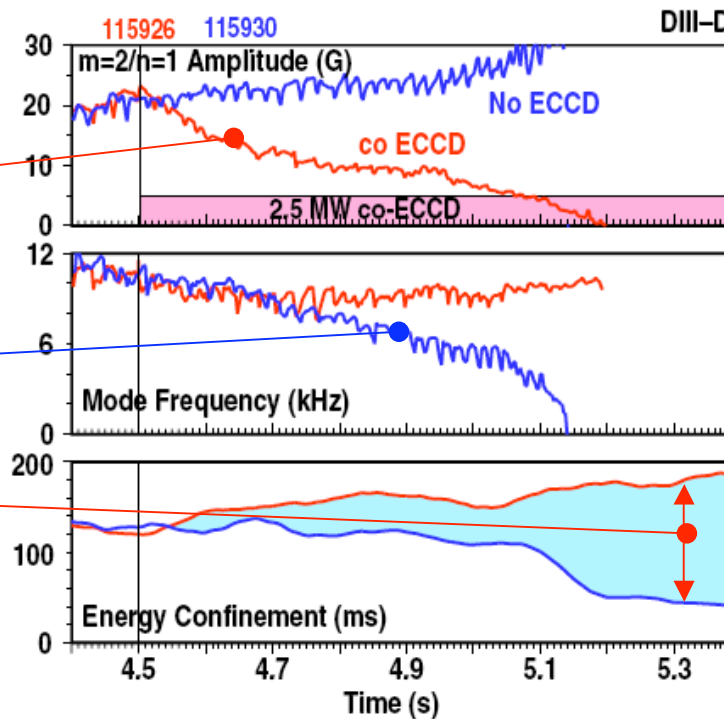


- ICRF minority current drive can either increase or decrease period and amplitude
- Likely stabilization/destabilization mechanism – RF modification of current profile

- Sawteeth can limit plasma performance themselves, or can trigger other instabilities – disruptions, neoclassical tearing modes
- Many physics processes interact – *qualitative* understanding exists but *quantitative* verification and *prediction* is lacking

# It has been demonstrated experimentally that suppression of NTM by RF leads to improvement in confinement

- Electron cyclotron current drive drives down mode amplitude
- keeps mode rotating (no drop in frequency)
- improves energy confinement



R. Prater  
APS 2003

- Empirical scaling of NTM pressure limits in ITER leave no margin in performance
- “Understanding the physics of neoclassical island modes and finding means for their avoidance or for limiting their impact on plasma performance are therefore important issues for reactor tokamaks and ITER” – ITER Physics Basis (1999)

# SWIM Physics Tasks (8-24-06)

---

## Component Developers

- Update Component Definition Documents – remember the seven deadly questions from the November meeting (see backup material for this call)
- Appendix 2 of IPS design description – add to or correct/edit table of component input and outputs (name, kind, providing component, units, definition (including standard math symbols) – Work with DBB
- Write GET\_PLASMA\_STATE\_<COMP> and PUT\_PLASMA\_STATE\_<COMP> subroutines
- Identify first specific code to implement the component
- Write component script – get help from CS and code developers

## Code developers

- Update the code specific parts of the Component Definition Documents. Make sure all input and output files are identified and described
- Port code to MHD cluster, place in Subversion repo along with build file and a collection of regression tests
- Write PREPARE\_<CODE>\_INPUT and PROCESS\_<CODE>\_OUTPUT codes
- Help write component script

## Theorists

- Work with Chris Hegna and Dalton Schnack to develop action plan for coupling RF+MHD and for NTM studies

# SWIM Computer Science tasks (8-24-06)

---

- Come to closure on layout of SWIM subversion repo (Bernholdt, Bramley) at a level that developers can start putting stuff in it – **we can always change it later**
- Improve or ratify the IPS design – **people are have started writing code around it**
  - **Help component developers write Python scripts.**
- Improve or ratify the Simulation Directory structure and the responsibilities for directory creation and file movement – **we can't write the Controller Script until this is nailed down**
- Write Controller (→ Driver) Script
  - Figure out how to encapsulate the simulation control logic (with physicists) – This affects both controller script and components. We need a way to allow users to tweak the logic but not break the system
  - How to communicate events, source wave-forms, feedback control adjustments
  - How to communicate error conditions, warnings, or information from the components .
  - How to keep track of and communicate information needed for restart or to change things in the middle of a step (like change of step size)
- Assess what the longer-term role for CS is in the SWIM project
  - If this is zero order design, sketch requirements/design for 1<sup>st</sup> order simulator implementation. Define what a truly ADVANCED simulation framework would be (one that the CS community would think was an advance)
  - Determine what studies have to be done to complete 1<sup>st</sup> order simulator
  - Define CS research or CS involvement in the SWIM project should be.