THE INTEGRATED PLASMA SIMULATOR—A TOOL FOR MULTIPHYSICS AND ENGINEERING SIMULATIONS

Lee A. Berry

SWIM team—Wael Elwasif, David Bernholdt and Samantha Foley (ORNL/CSM), Don Batchelor (UTK), and collaborators at PPPL, MIT, GA, ….

Oak Ridge TN

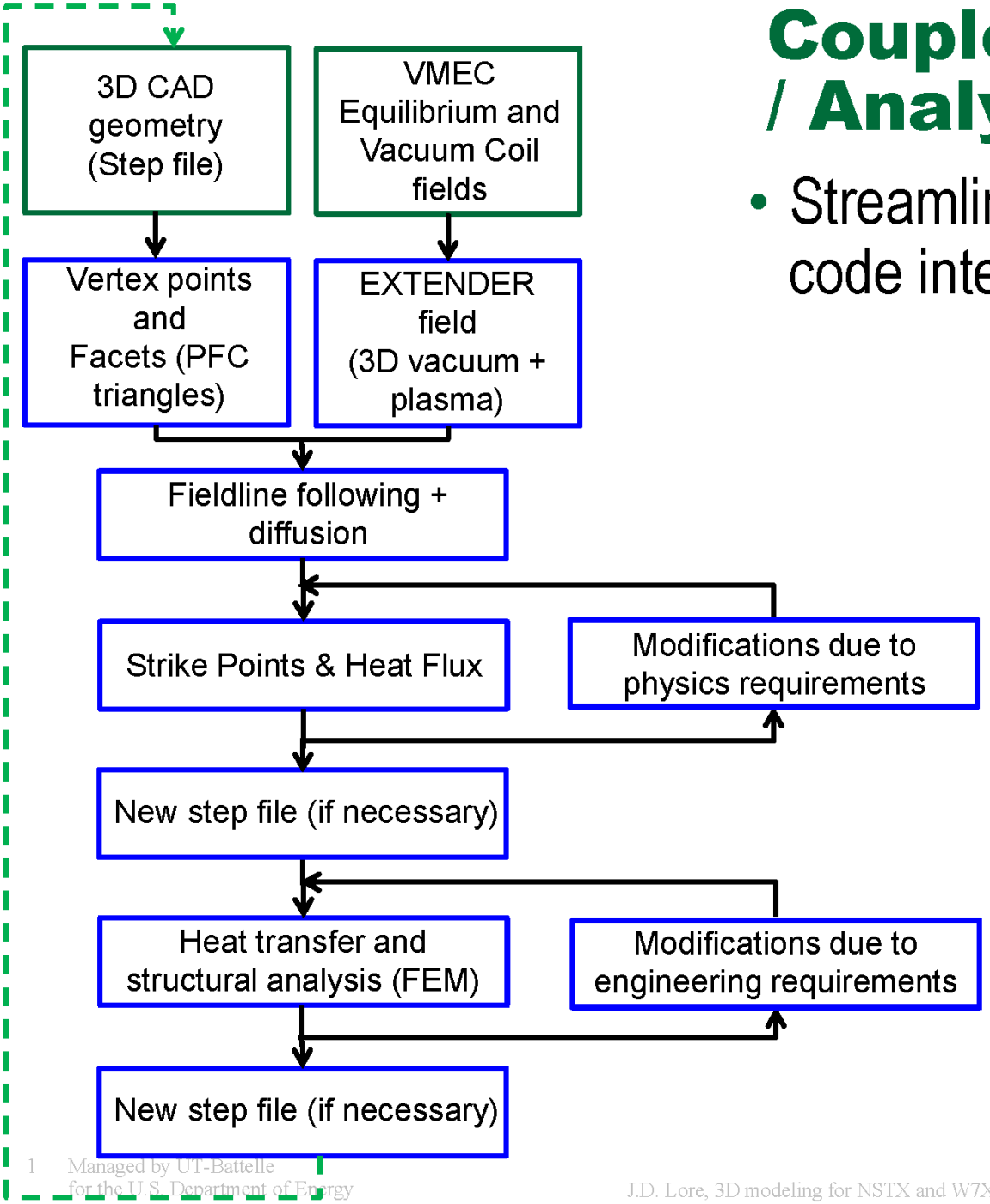
February 2, 2012

Topics/Outline

- **Motivation for this talk**
- **The SWIM project**
- **ITER simulations**
- **Workflows and the IPS**
- **The IPS, services, events, data, …**
- **Components**
- **Tools--portal**
- **Simple demo—'hello world'**

Coupled Optimization / Analysis In Progress

- Streamlining CAD-physics code interface is critical



- Max Divertor Flux
- Max Scraper Flux
- Pitch angle
- …

- Pressure drop
- Fluid temp rise
- Max fluid temp
- Max CFC temp
- Stresses
- …

Codes

- VMEC
  - 3D equilibrium (nested closed surfaces)
- EXTENDER
  - Uses virtual casing principle to calculate fields due to plasma currents outside of the VMEC domain
- ANSYS-CFX
  - Fluid, heat transfer, and structural FEM
- Pro/E and Catia
  - CAD software
- EMC3-Eirene
  - Coupled 3D fluid plasma + 3D kinetic neutral and PSI
- Misc
  - Fieldline following and part intersection code
  - Various output parsing/input file generation routines
  - Transport codes (PENTA, DKES, etc)

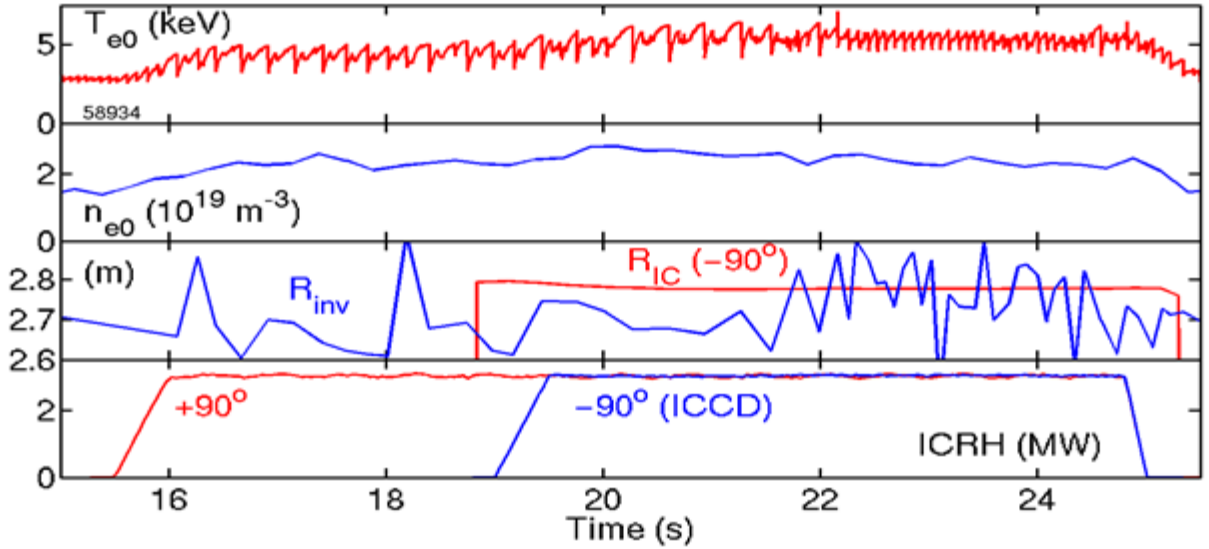
Simulation of Wave Interactions with MHD → SWIM

The scientific motivation for SWIM is based on two simple experimental observations:

Macroscopic (MHD) instabilities can limit plasma performance in fusion devices

RF waves can mitigate and control instabilities – sometimes producing instability, sometimes reducing or eliminating instability

Example – Sawtooth control on JET with Minority Current Drive on JET



- **ICRF minority current drive can either increase or decrease period and amplitude**
- **Important because sawteeth can degrade confinement or trigger other deleterious instabilities – neoclassical tearing modes, disruptions**

Project goals

Address basic science and practical operational questions :

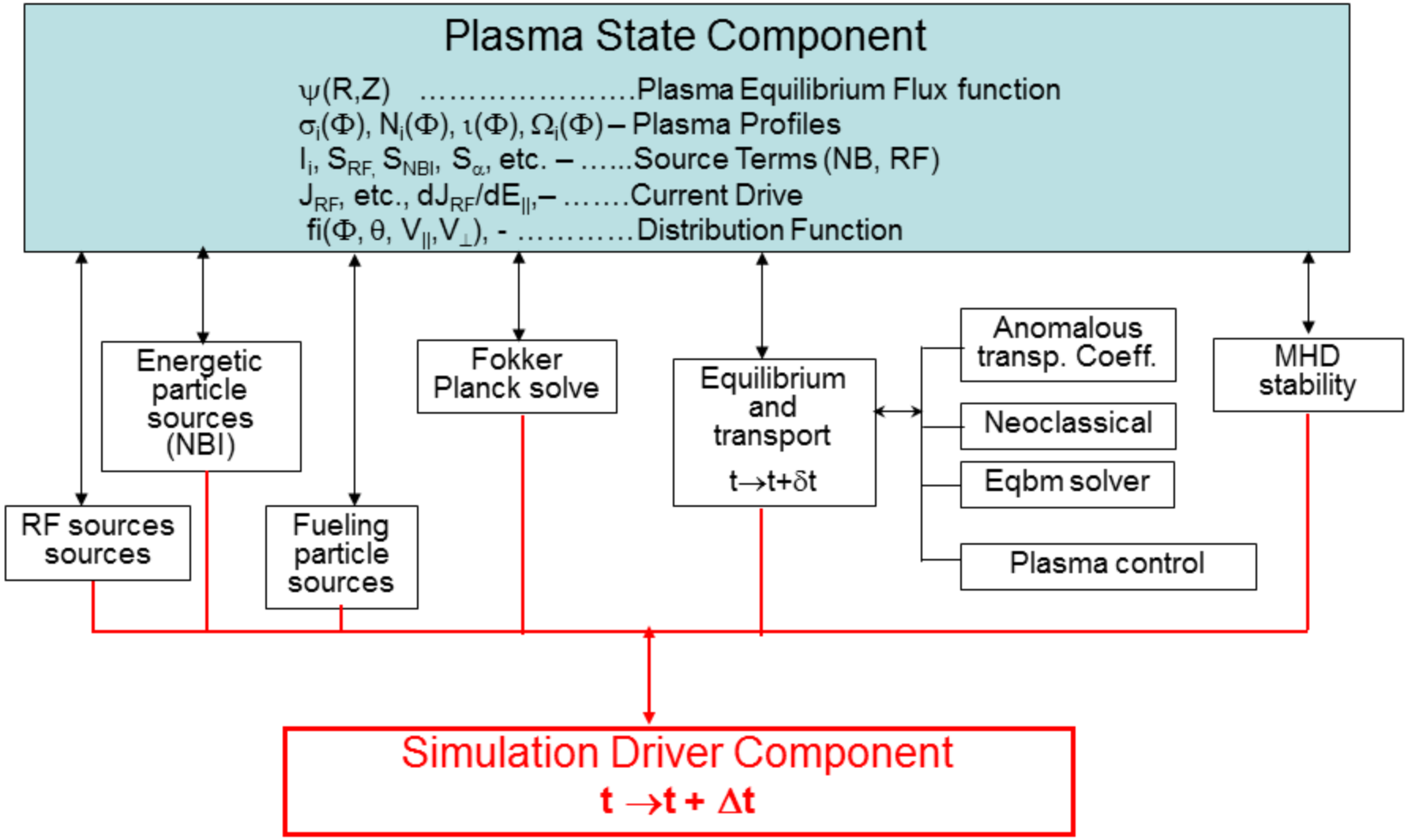
- **How does RF control sawtooth instability behavior? → Can the ITER ICRF system control sawteeth?**
- **How does electron cyclotron current drive control Neoclassical Tearing Modes? → How much power will it take on ITER?**

and:

- **Provide a base of experience with framework/component architecture applied to integrated fusion simulation that can be factored into the design of a larger-scale fusion simulation project.**
- **Develop IPS so that it provides a computational environment that is useful for a broad range of plasma simulation applications**

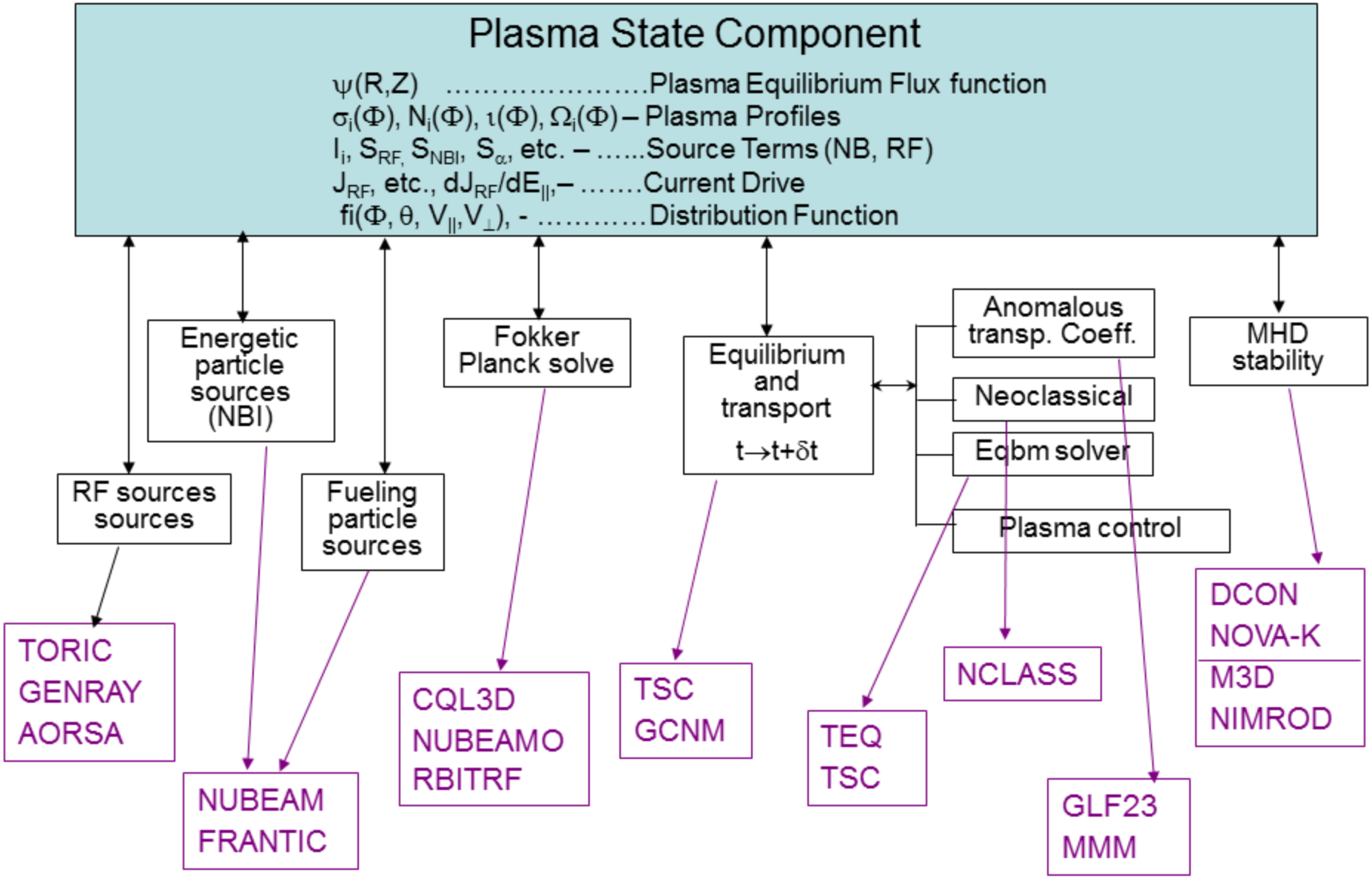
⇒ Is the tool of choice for those performing tokamak simulations

Integrated Plasma Simulator design – Physicists view

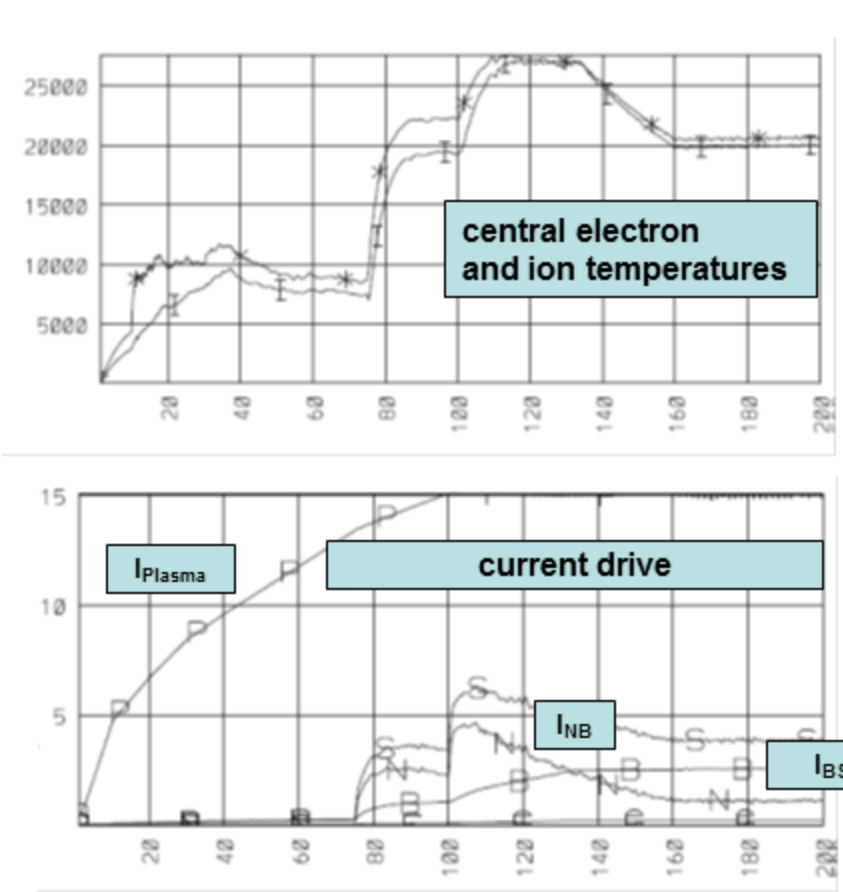


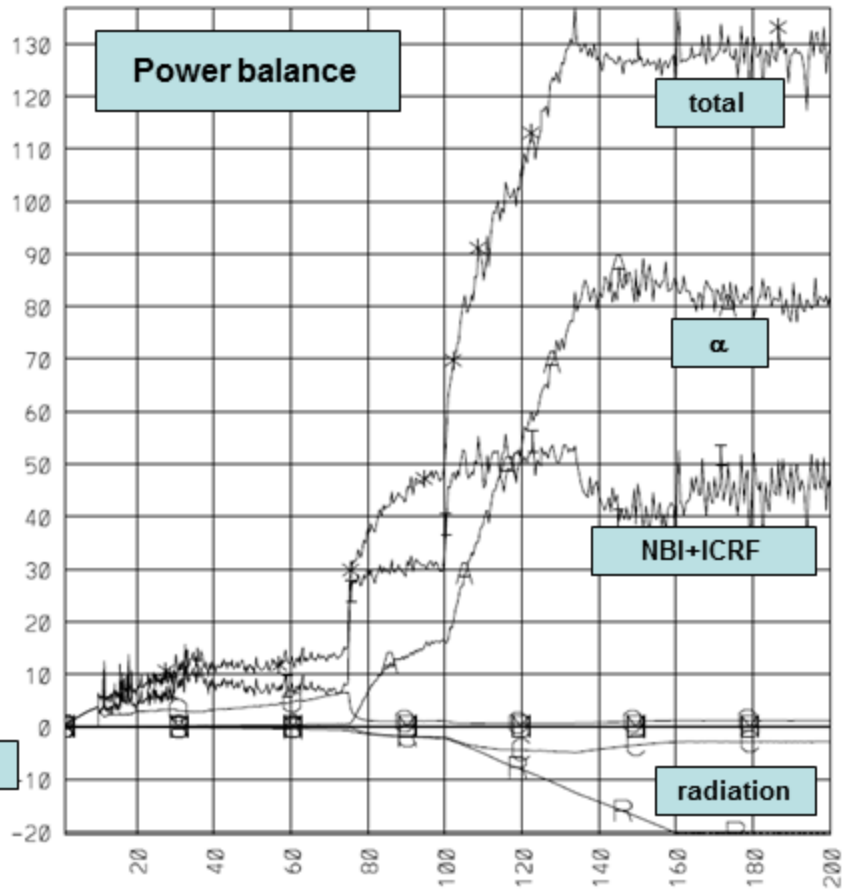
Component architecture, organized by abstract physics functionality

Integrated Plasma Simulator design – Components are implemented by mature, well-validated codes



When we started, a single long-time-scale ITER simulation, such as below, with minimal level of physics detail took up to 6 weeks with serial processor technology





TSC (Free-Boundary Equilibrium and Profile Advance)

TORIC (RF Ion Cyclotron) – 32 poloidal Fourier modes (poorly converged)

NUBEAM (neutral beam injection) – 1,000 Monte Carlo particles (poor statistics)

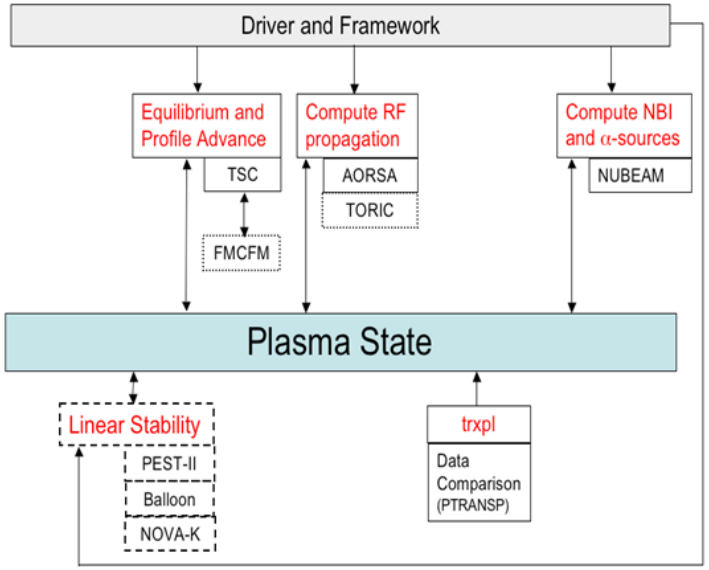
Goal: demonstrate the use of massively-parallel computers to accelerate ITER simulations, while improving the level of physics fidelity of the simulations.

Coupling of TSC (Free-Boundary Equilibrium and Profile Advance)

AORSA (massively parallel RF Ion Cyclotron solver) - 256×256 poloidal Fourier modes

TORIC (semi-spectral ICRF solver) – 147 poloidal modes, 409 radial nodes

NUBEAM (parallel neutral beam injection) – 1,000,000 Monte Carlo particles



These simulations benefit from component level concurrency to minimize time in (near) serial operations

Summary of ITER simulations with IPS

Simulations at very high resolutions to show capability of massive parallelism

- **TSC +AORSA + NUBEAM (1,000,000 particles/species)**
- **TSC + TORIC (255 poloidal modes) + NUBEAM (1,000,000 particles/species)**
- ***running times ~ 30 hr on 1600 cores***

Simulations at resolutions more typical of present practice for comparison

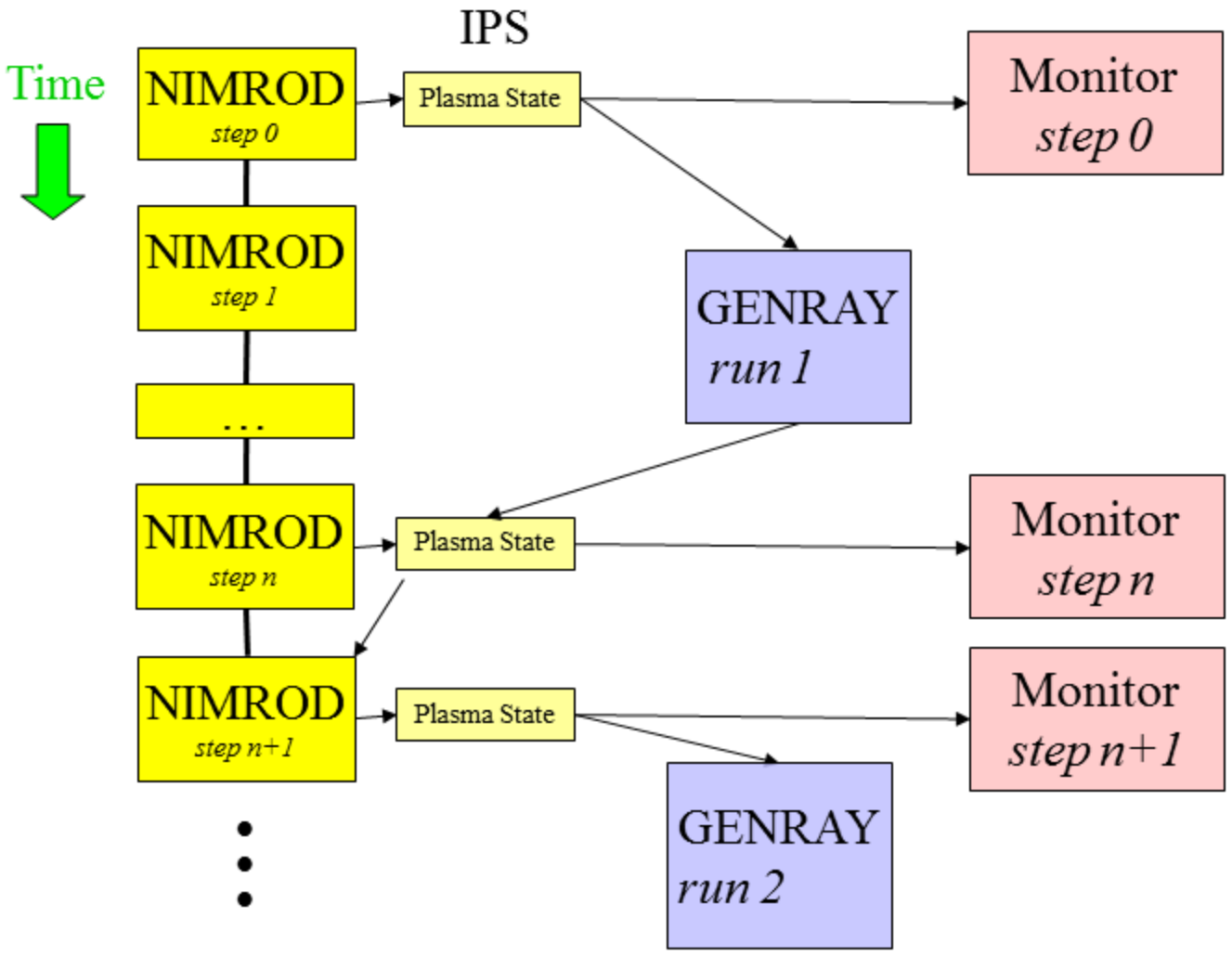
- **ITER hybrid scenario**
- **TSC (1 core), TORIC (31 poloidal modes, 4 cores), NUBEAM (5,000 particles/species, 16 cores)**
- **Typically ramp-up from 1.5 sec into flattop 550 sec**

- **TSC alone – using TSC internal (analytic) models for NBI and ICRF**
  - ***No parallelism, 1 core, running time ~ 11 hr***
- **TORIC + NUBEAM + TSC – sequential execution of parallel components**
  - ***One level of parallelism, 16 cores, running time ~ 28 hr***
- **TORIC + NUBEAM + TSC – concurrent execution of parallel components**
  - ***Two levels of parallelism, 24 cores, running time ~ 12 hr***
- **Parameter study – pedestal location, pedestal height (chi pedestal)**
  - **Nine concurrent simulations run simultaneously**
  - ***Three levels of parallelism, 128 cores, running time ~ 16 hr***

The IPS Can Implement a Wide Range of Work Flows—Examples Include:

- **Tokamak simulations—time dependent evolution with ‘driver’ in control of time loop**
- **Slow MHD—time dependent simulations with XMHD in control**
- **Scenario optimization (Dakota, TechX)**
- **Parareal—iterative parallelization:**
  - **Classical—flow controlled by loops**
  - **Data driven—components run when data are ready (when dependencies are satisfied**

NIMROD/GENRAY coupling in IPS – NIMROD is run as a service, but controls time loop via simulation *event handling*



- **NIMROD exports magnetic geometry and n,T profiles to Plasma State**
- **GENRAY then calculates RF propagation and power deposition; exporting these quantities to the Plasma State**
- **NIMROD converts GENRAY data into momentum and energy source terms.**
- **Ultimately will include kinetic closure model**

Two levels of parallelism – parallel NIMROD run concurrently with GENRAY

Parareal: The Classic Algorithm

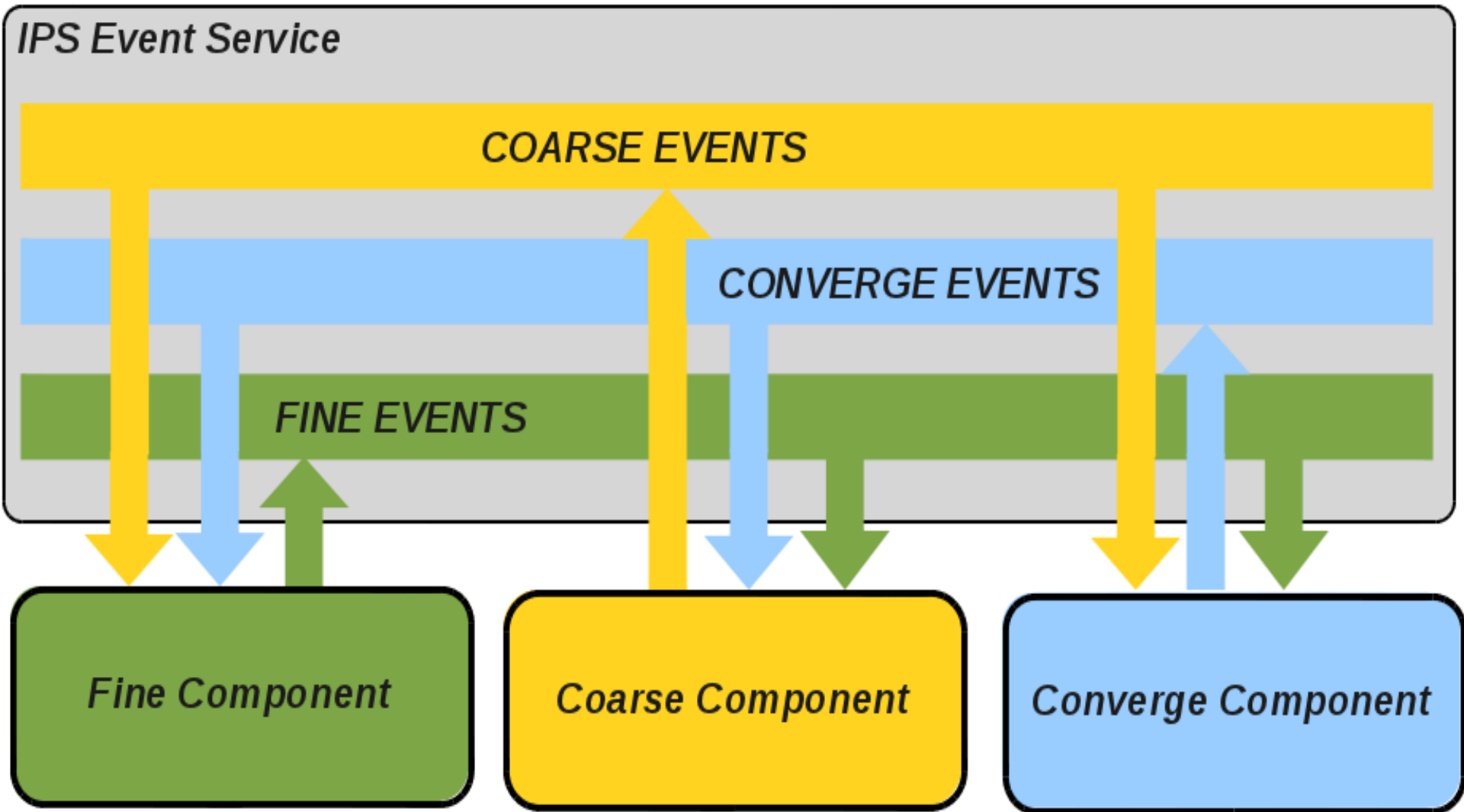
```
first_slice = 1
num_converged = 0
for iteration = 1, max_iterations
    for slice = first_slice..num_slices
       coarse_solve(iteration, slice)
    forall slice = first_slice..num_slices
       fine_solve(iteration, slice)
    for slice = first_slice..num_slices
       test_convergence(iteration, slice)
    num_converged +=
       first_non_converged_slice - first_slice
    if (num_converged == num_slices)
       end // SUCCESS
    else
       first_slice = first_non_converged_slice
end //Failed to converge in max_iteration
```

Sequential Phase

Parallel Phase

Computationally Cheap

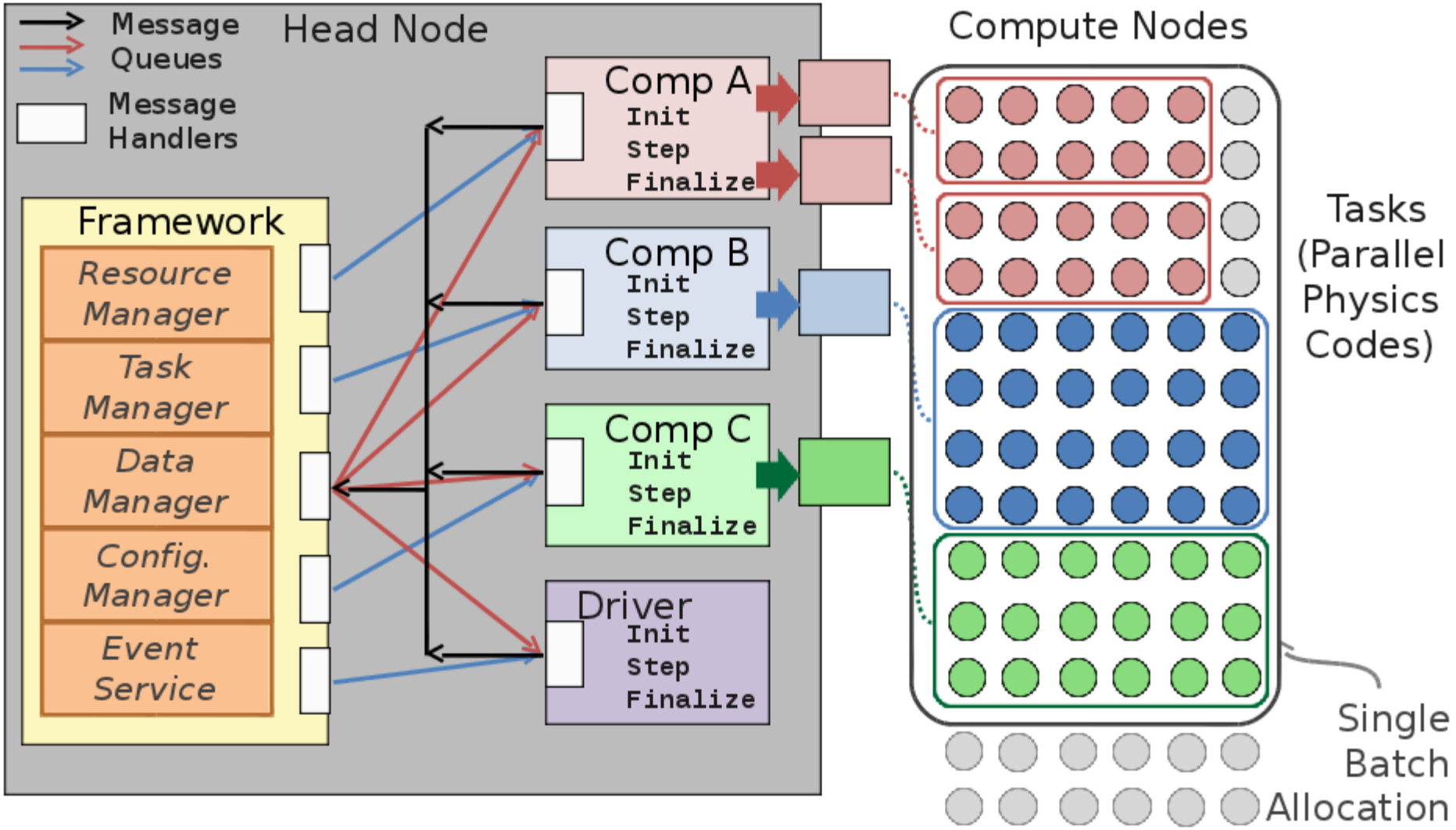
Dependency Propagation Using the IPS Event Service



The Underlying Framework
The Integrated Plasma Simulator (IPS)

- Component-based Python framework for loosely coupled simulations
- Originally designed for time-stepped fusion simulations
  - Flexibility allows use in other domains, and other control-flow paradigms
- Major features:
  - Thin component layer in Python that wraps stand-alone executables
  - Inter-component data exchange using the file system
  - Framework services used to assemble a simulation
    - **Resource management** - **Task management**
    - **Data management** - **Asynchronous event services**
  - Simulations execute within a single batch allocation

IPS Architecture & Execution Model



A Component Is

- **The subject of formal definitions**
- **In SWIM (typically) :**
  - **A physics application (TSC, AORSA, GENRAY, etc.)**
    - **This application is invoked by a Python module that includes**
      - **Init**
      - **Step**
        - **Prepare input**
        - **Run physics code**
        - **Process output**
      - **Finalize**
      - **Restart**
- **But it can be anything that Python can do or invoke**

The Hello Component

berryla@berryhplap:~/ips/components/drivers/hello

File Edit View Search Terminal Help

```
#! /usr/bin/env python

from  component import Component

class HelloWorker(Component):
    def __init__(self, services, config):
        Component.__init__(self, services, config)
        print 'Created %s' % (self.__class__)

    def init(self, timeStamp=0.0):
        return

    def step(self, timeStamp=0.0):
        print 'Hello from HelloWorker'
        return

    def finalize(self, timeStamp=0.0):
        return

~
~
~
~
~
~
~
~
~
~
"hello_worker.py" 19L, 432C                                   1,1           All
```

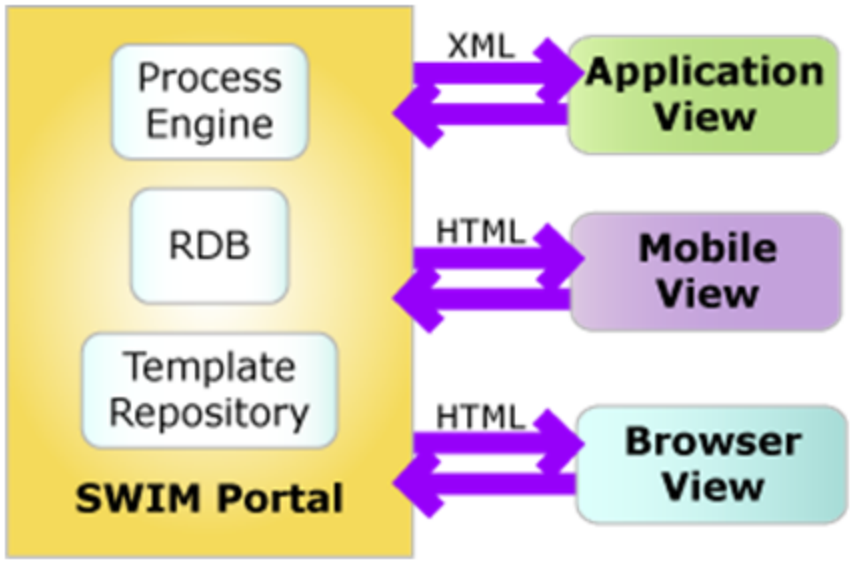
The Driver that Runs the Hello Component

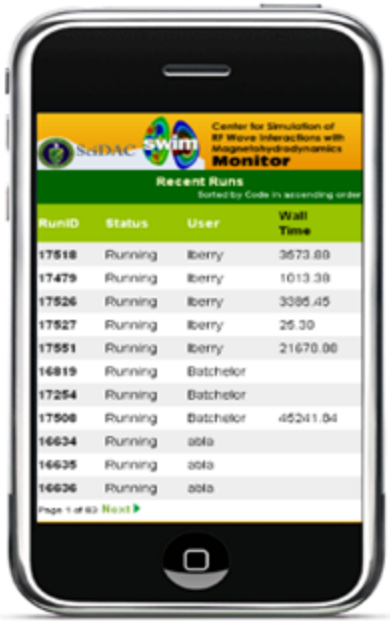
berryla@berryhplap:~/ips/components/drivers/hello

File Edit View Search Terminal Help

```
#! /usr/bin/env python

from  component import Component

class HelloDriver(Component):
    def __init__(self, services, config):
        Component.__init__(self, services, config)
        print 'Created %s' % (self.__class__)

    def init(self, timeStamp=0.0):
        return

    def step(self, timeStamp=0.0):
        print 'HelloDriver: beginning step call'
        try:
            worker_comp = self.services.get_port('WORKER')
        except Exception:
            self.services.exception('Error accessing worker component')
            raise
        self.services.call(worker_comp, 'step', 0.0)
        print 'HelloDriver: finished worker call'
        return

    def finalize(self, timeStamp=0.0):
        return

~
~
~
                                                    24,38         All
```

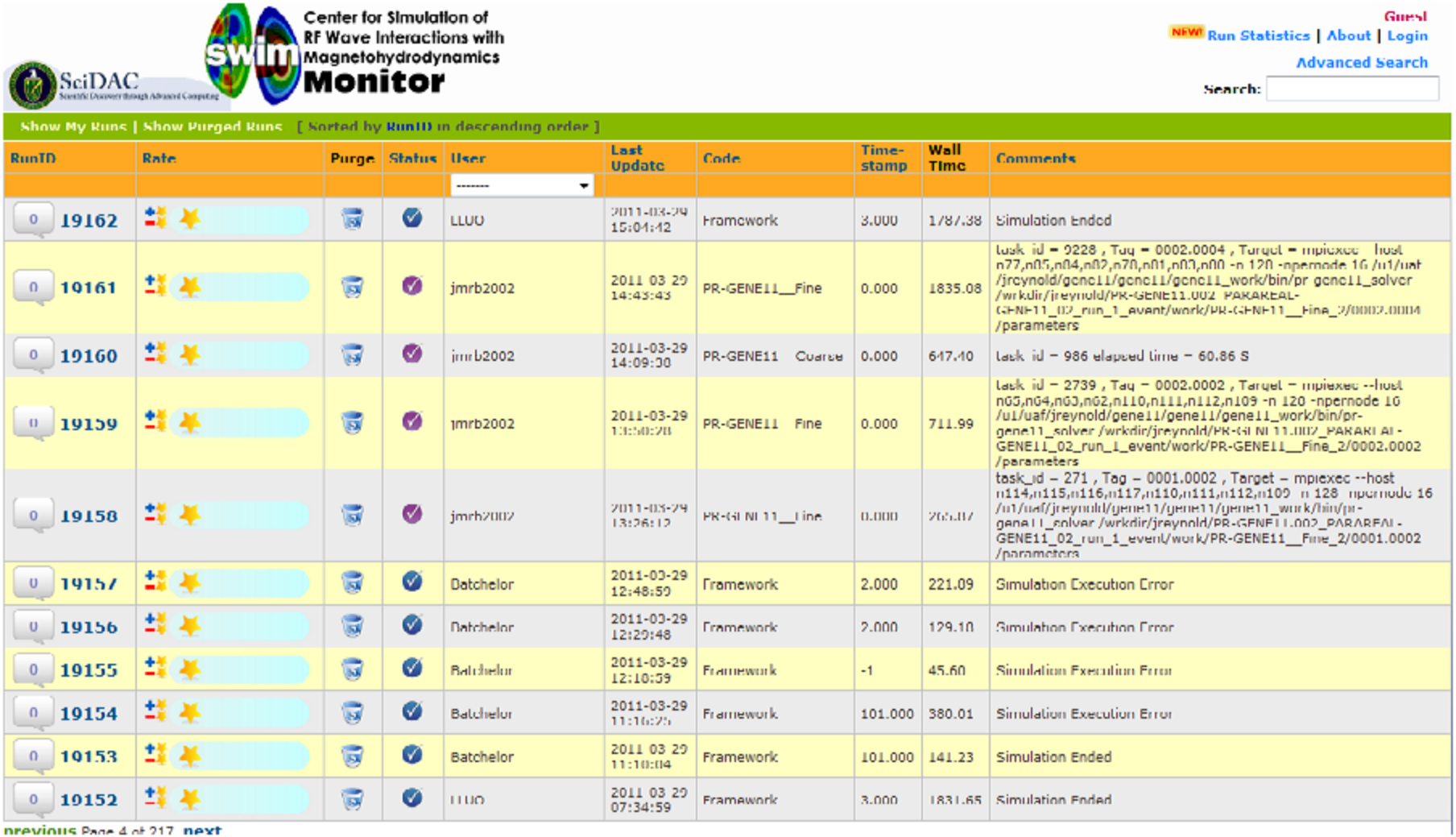
SWIM Portal Collects Information About SWIM Simulations and Serves Multiple Clients





- A variety of meta-data is tracked and stored in Relational DB
  - User name, simulation name current status, code name, last time stamp, wall clock time, simulation comment, tokamak, host computer
  - Supports search and sort
- Each simulation is identified with unique RunID
- Using MDS+ for data storage
- Coming soon – access to experimental data archives, comparison with simulations

Summary Page Provides Snapshot of Current Status of SWIM Simulations (http://swim.gat.com:8080/)



Center for Simulation of RF Wave Interactions with Magnetohydrodynamics

SciDAC

SWIM Monitor

Guest

NEW! Run Statistics | About | Login

Advanced Search

Search:

Show My Runs | Show Purged Runs [Sorted by RunID in descending order]

RunID	Rate	Purge	Status	User	Last Update	Code	Time-stamp	Wall Time	Comments
19162				LLUO	2011-03-29 15:04:42	Framework	3.000	1787.38	Simulation Ended
19161				jmrb2002	2011 03 29 14:45:43	PR-GENE11__Fine	0.000	1835.08	task_id = 9228 , Tag = 0002.0004 , Target = mpiexec --host n77,n05,n04,n02,n70,n01,n00,n00 -n 120 -npernode 16 /u1/uaf/jreynold/gene11/gene11/gene11_work/bin/pr-gene11_solver /wrkdir/jreynold/PR-GENE11.002_PARAREAL-GENE11_02_run_1_event/work/PR-GENE11__Fine_2/0002.0004 /parameters
19160				jmrb2002	2011-03-29 14:09:30	PR-GENE11 Coarse	0.000	647.40	task_id = 986 elapsed time = 60.86 S
19159				jmrb2002	2011-03-29 13:50:28	PR-GENE11 Fine	0.000	711.99	task_id = 2739 , Tag = 0002.0002 , Target = mpiexec --host n03,n04,n03,n02,n110,n111,n112,n109 -n 120 -npernode 16 /u1/uaf/jreynold/gene11/gene11/gene11_work/bin/pr-gene11_solver /wrkdir/jreynold/PR-GENE11.002_PARAREAL-GENE11_02_run_1_event/work/PR-GENE11__Fine_2/0002.0002 /parameters
19158				jmrb2002	2011-03-29 13:26:13	PR-GENE11__Fine	0.000	266.07	task_id = 271 , Tag = 0001.0002 , Target = mpiexec --host n114,n115,n116,n117,n110,n111,n112,n109 -n 128 -npernode 16 /u1/uaf/jreynold/gene11/gene11/gene11_work/bin/pr-gene11_solver /wrkdir/jreynold/PR-GENE11.002_PARAREAL-GENE11_02_run_1_event/work/PR-GENE11__Fine_2/0001.0002 /parameters
19157				Batchelor	2011-03-29 12:48:50	Framework	2.000	221.09	Simulation Execution Error
19156				Batchelor	2011-03-29 12:29:48	Framework	2.000	129.10	Simulation Execution Error
19155				Batchelor	2011-03-29 12:10:39	Framework	-1	45.60	Simulation Execution Error
19154				Batchelor	2011-03-29 11:16:29	Framework	101.000	380.01	Simulation Execution Error
19153				Batchelor	2011 03 29 11:10:04	Framework	101.000	141.23	Simulation Ended
19152				LLUO	2011 03 29 07:34:59	Framework	3.000	1831.65	Simulation Ended

previous Page 4 of 217 next

Main SWIM portal page shows the most recent 12 runs. Clicking on any run-ID yields more detail. The search function allows rapid searching through the >2500 SWIM runs stored in the system

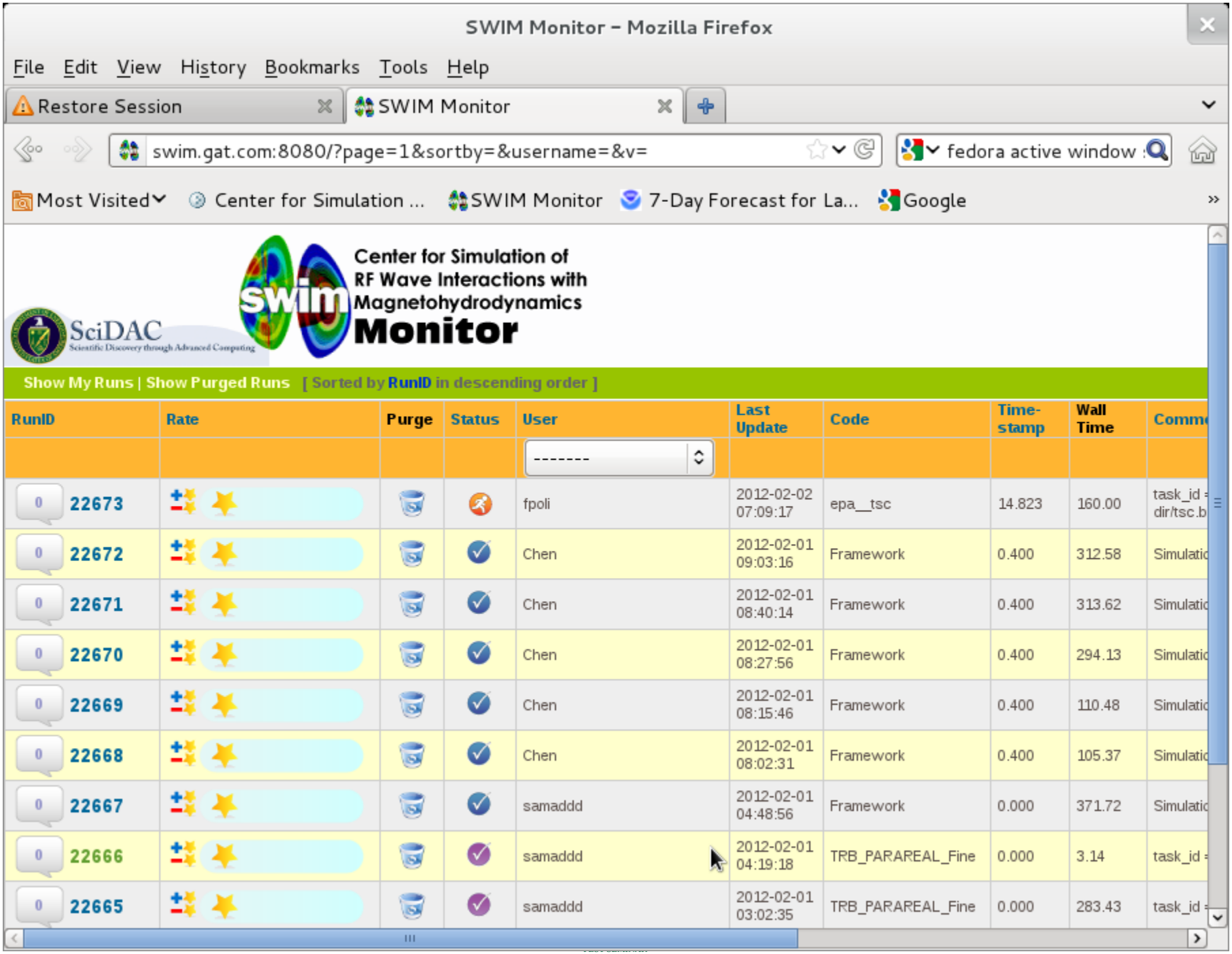
Live--Hopefully

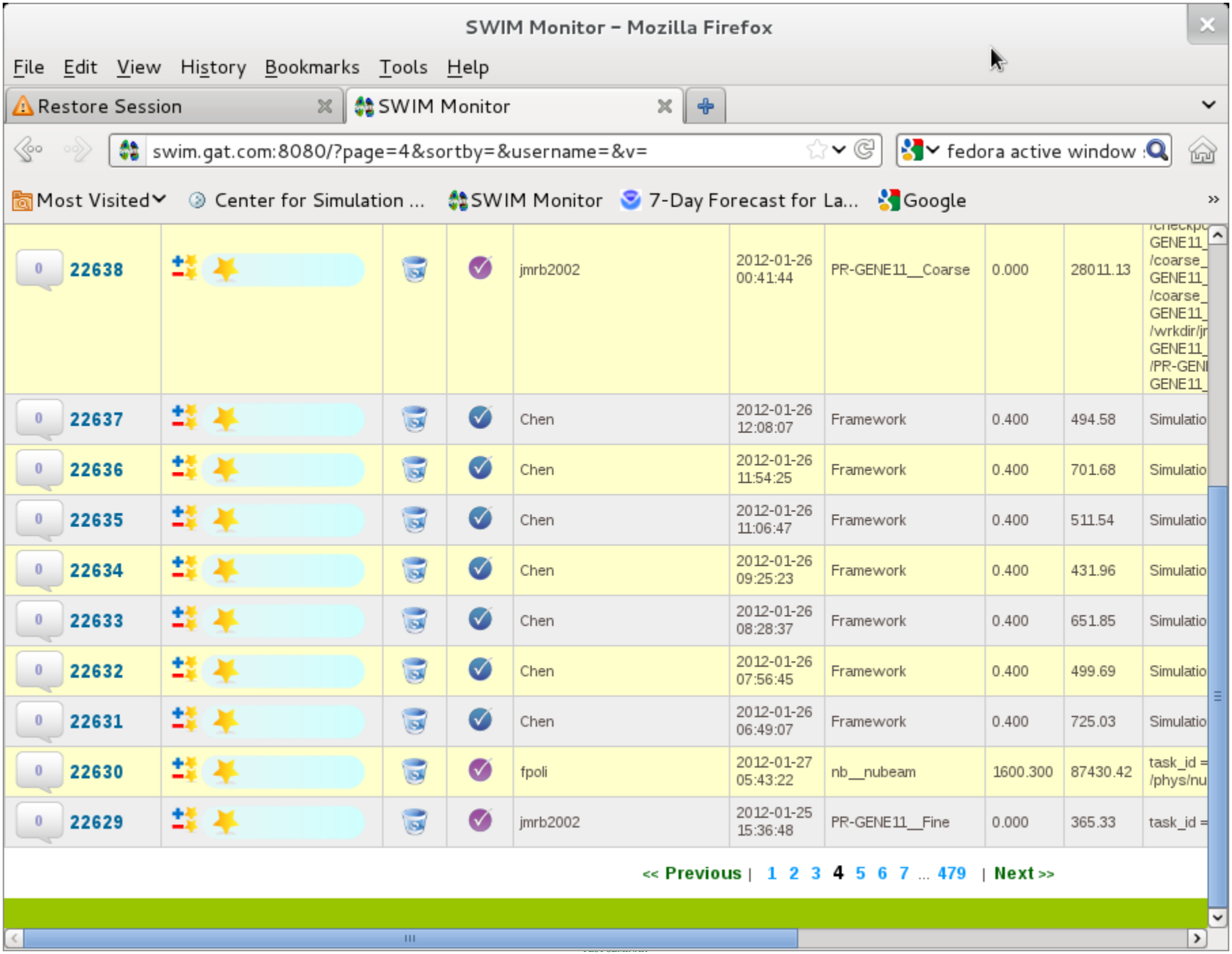
- **What's going in SWIM/IPS now**
- **Running the hello simulation**

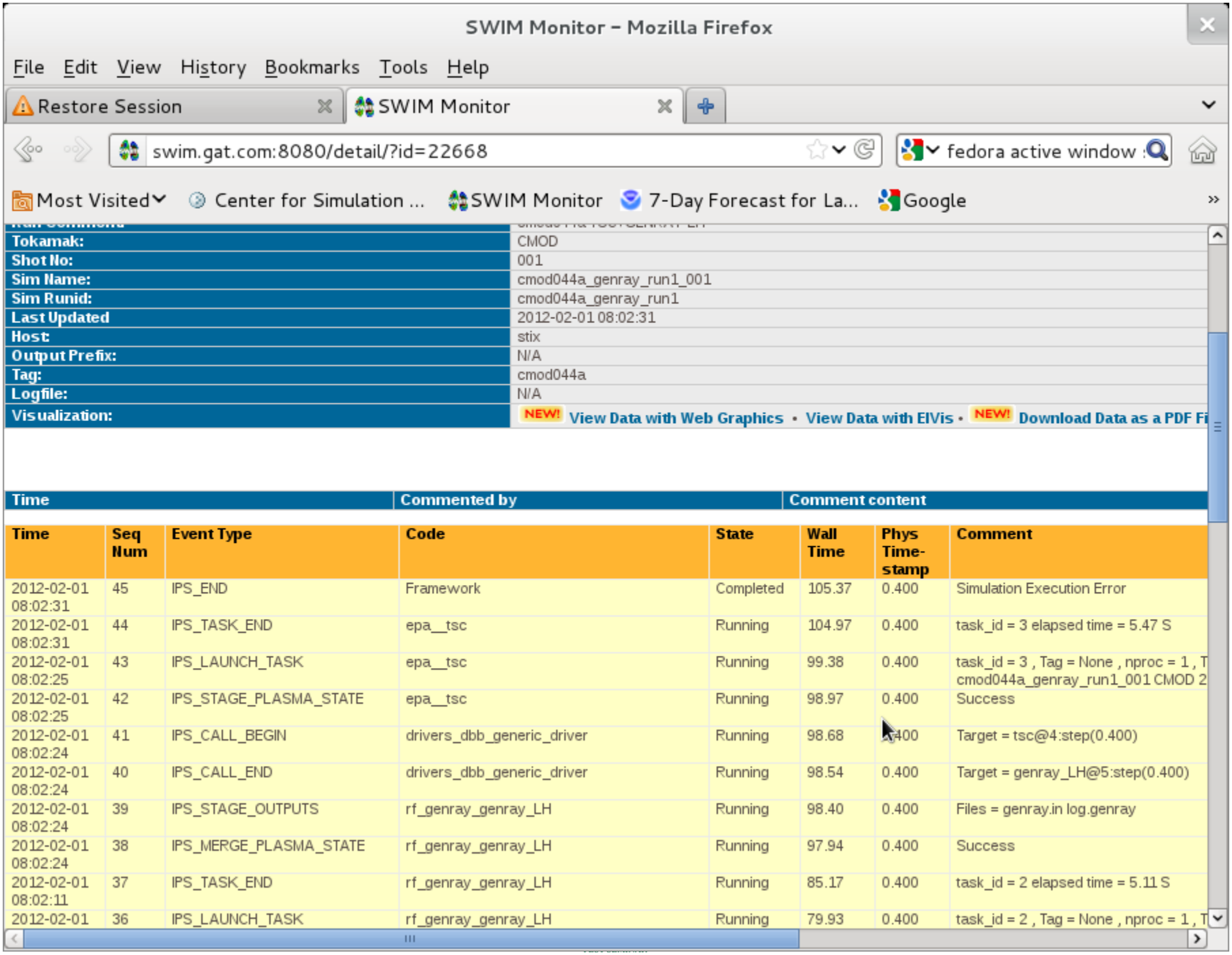
Center for Simulation of RF Wave Interactions with Magnetohydrodynamics Monitor

Show My Runs | Show Purged Runs [Sorted by RunID in descending order]

RunID	Rate	Purge	Status	User	Last Update	Code	Time-stamp	Wall Time	Comment
				-------					
0 22651				Chen	2012-01-30 07:12:06	Framework	-1	10.80	Simulation E
0 22650				Chen	2012-01-30 07:00:16	Framework	-1	17.30	Simulation E
0 22649				fpoli	2012-01-29 16:02:11	Framework	2000.000	120430.96	Simulation E
0 22648				fpoli	2012-01-29 01:05:05	Framework	2000.000	155662.40	Simulation E
0 22647				fpoli	2012-01-28 23:12:26	Framework	2000.000	149097.04	Simulation E
0 22646				fpoli	2012-01-28 06:28:37	nb__nubeam	907.487	88881.15	task_id = 2 /phys/nube
0 22645				Harvey	2012-01-26 18:31:21	Framework	-1	15.47	Simulation E
0 22644				Harvey	2012-01-26 17:48:07	Framework	-1	46.23	Simulation E
0 22643				Harvey	2012-01-26 17:36:34	Framework	-1	50.14	Simulation E


SWIM Monitor – Mozilla Firefox
File Edit View History Bookmarks Tools Help
Restore Session
SWIM Monitor
swim.gat.com:8080/?page=1&sortby=&username=&v=
fedora active window
Most Visited
Center for Simulation ...
SWIM Monitor
7-Day Forecast for La...
Google
SciDAC
Scientific Discovery through Advanced Computing
Center for Simulation of
RF Wave Interactions with
Magnetohydrodynamics
Monitor
Show My Runs | Show Purged Runs [Sorted by RunID in descending order]
RunID
Rate
Purge
Status
User
Last Update
Code
Time-stamp
Wall Time
Comme
22673 fpoli 2012-02-02 07:09:17 epa__tsc 14.823 160.00 task_id = dir/tsc.b
22672 Chen 2012-02-01 09:03:16 Framework 0.400 312.58 Simulatic
22671 Chen 2012-02-01 08:40:14 Framework 0.400 313.62 Simulatic
22670 Chen 2012-02-01 08:27:56 Framework 0.400 294.13 Simulatic
22669 Chen 2012-02-01 08:15:46 Framework 0.400 110.48 Simulatic
22668 Chen 2012-02-01 08:02:31 Framework 0.400 105.37 Simulatic
22667 samaddd 2012-02-01 04:48:56 Framework 0.000 371.72 Simulatic
22666 samaddd 2012-02-01 04:19:18 TRB_PARAREAL_Fine 0.000 3.14 task_id =
22665 samaddd 2012-02-01 03:02:35 TRB_PARAREAL_Fine 0.000 283.43 task_id =


SWIM Monitor – Mozilla Firefox
File Edit View History Bookmarks Tools Help
Restore Session
SWIM Monitor
swim.gat.com:8080/?page=4&sortby=&username=&v=
fedora active window
Most Visited
Center for Simulation ...
SWIM Monitor
7-Day Forecast for La...
Google
0 22638 jmrb2002 2012-01-26 00:41:44 PR-GENE11__Coarse 0.000 28011.13 GENE11_ /coarse_ GENE11_ /coarse_ GENE11_ /wrkdir/jr GENE11_ /PR-GENI GENE11_
0 22637 Chen 2012-01-26 12:08:07 Framework 0.400 494.58 Simulatio
0 22636 Chen 2012-01-26 11:54:25 Framework 0.400 701.68 Simulatio
0 22635 Chen 2012-01-26 11:06:47 Framework 0.400 511.54 Simulatio
0 22634 Chen 2012-01-26 09:25:23 Framework 0.400 431.96 Simulatio
0 22633 Chen 2012-01-26 08:28:37 Framework 0.400 651.85 Simulatio
0 22632 Chen 2012-01-26 07:56:45 Framework 0.400 499.69 Simulatio
0 22631 Chen 2012-01-26 06:49:07 Framework 0.400 725.03 Simulatio
0 22630 fpoli 2012-01-27 05:43:22 nb__nubeam 1600.300 87430.42 task_id = /phys/nu
0 22629 jmrb2002 2012-01-25 15:36:48 PR-GENE11__Fine 0.000 365.33 task_id =
<< Previous | 1 2 3 4 5 6 7 ... 479 | Next >>



Tokamak:	CMOD
Shot No:	001
Sim Name:	cmod044a_genray_run1_001
Sim Runid:	cmod044a_genray_run1
Last Updated	2012-02-01 08:02:31
Host:	stix
Output Prefix:	N/A
Tag:	cmod044a
Logfile:	N/A
Visualization:	NEW! View Data with Web Graphics • View Data with ElVis • NEW! Download Data as a PDF Fi

Time	Commented by	Comment content

Time	Seq Num	Event Type	Code	State	Wall Time	Phys Time-stamp	Comment
2012-02-01 08:02:31	45	IPS_END	Framework	Completed	105.37	0.400	Simulation Execution Error
2012-02-01 08:02:31	44	IPS_TASK_END	epa__tsc	Running	104.97	0.400	task_id = 3 elapsed time = 5.47 S
2012-02-01 08:02:25	43	IPS_LAUNCH_TASK	epa__tsc	Running	99.38	0.400	task_id = 3 , Tag = None , nproc = 1 , T cmod044a_genray_run1_001 CMOD 2
2012-02-01 08:02:25	42	IPS_STAGE_PLASMA_STATE	epa__tsc	Running	98.97	0.400	Success
2012-02-01 08:02:24	41	IPS_CALL_BEGIN	drivers_dbb_generic_driver	Running	98.68	0.400	Target = tsc@4:step(0.400)
2012-02-01 08:02:24	40	IPS_CALL_END	drivers_dbb_generic_driver	Running	98.54	0.400	Target = genray_LH@5:step(0.400)
2012-02-01 08:02:24	39	IPS_STAGE_OUTPUTS	rf_genray_genray_LH	Running	98.40	0.400	Files = genray.in log.genray
2012-02-01 08:02:24	38	IPS_MERGE_PLASMA_STATE	rf_genray_genray_LH	Running	97.94	0.400	Success
2012-02-01 08:02:11	37	IPS_TASK_END	rf_genray_genray_LH	Running	85.17	0.400	task_id = 2 elapsed time = 5.11 S
2012-02-01	36	IPS_LAUNCH_TASK	rf_genray_genray_LH	Running	79.93	0.400	task_id = 2 , Tag = None , nproc = 1 , T

Command-line View of Hello

```
[berryla@berryhplap hello]$ pwd
/home/berryla/ips/components/drivers/hello
[berryla@berryhplap hello]$ ls
hello_driver.py            hello_world.config~     Makefile.include
hello_worker.py            hello_world.config_svn  workstation.conf
hello_worker_task_pool.py  Hello_world_sim.log
hello_world.config         Makefile
[berryla@berryhplap hello]$ /home/berryla/ips/bin/ips --config=hello_world.
config --platform=workstation.conf
Starting IPS
Created <class 'hello_driver.HelloDriver'>
Created <class 'hello_worker.HelloWorker'>
2012-02-02 10:05:17,890 FRAMEWORK       WARNING  Missing INIT specification
 in config file for simulation Hello_world_1
2012-02-02 10:05:18,070 FRAMEWORK       WARNING  RM: listOfNodes = [('dummy
node0', 2)]
2012-02-02 10:05:18,070 FRAMEWORK       WARNING  RM: max_ppn = 2
2012-02-02 10:05:18,070 FRAMEWORK       WARNING  Using user set procs per n
ode: 2
2012-02-02 10:05:18,070 FRAMEWORK       WARNING  RM: 1 nodes and 2 processo
rs per node
HelloDriver: beginning step call
Hello from HelloWorker
HelloDriver: finished worker call
[berryla@berryhplap hello]$
```

Exploring the Simulation Directories

```
[berryla@berryhplap hello]$ cd ~/scratch/
[berryla@berryhplap scratch]$ ls
aorsa2d_v2           resource_usage  simulation_setup  workstation.conf
hello_world.config   simulation_log  work
[berryla@berryhplap scratch]$ pwd
/home/berryla/scratch
[berryla@berryhplap scratch]$ ls
aorsa2d_v2           resource_usage  simulation_setup  workstation.conf
hello_world.config   simulation_log  work
[berryla@berryhplap scratch]$ ls simulation_setup/
hello_driver.py  hello_worker.py  portalBridge.py
[berryla@berryhplap scratch]$ ls work
DRIVERS_HELLO_HelloDriver_1  plasma_state
FWK_COMP_PortalBridge_3      WORKERS_HELLO_HelloWorker_2
[berryla@berryhplap scratch]$ ls simulation_log/
Hello_world_1-4388df5b-a464-481f-9097-6c81c6616d44.eventlog
Hello_world_1-5b7b198a-d2fb-49b1-9424-3962c5dcf1db.eventlog
Hello_world_1-918d87c4-e9bf-46ed-9d52-d142438c4a63.eventlog
Hello_world_1-cd0fa292-5a87-4ade-b121-7e267c2fd48b.eventlog
Hello_world_1-e4e8c806-8f64-490b-a50b-80abdddb7713.eventlog
[berryla@berryhplap scratch]$
```

Summary/Comments

- **SWIM/IPS result of a seven year $2M/year SciDAC effort**
- **This effort was preceeded by pre FSP work (DBB)—Physics of Today article/Dahlburg report/etc.**
- **LDRD on component based simulation (Elwasif/Kohl/Bernholdt/Berry)**
- **IPS/Swim widely used**
- **Hopefully will be maintained and used as part of new SciDAC proposal**
- **IPS is being used for a wide variety of fusion applications, and a few outside**
- **I'll get a reference list if folks are interested**