

# Assembling a SWIM IPS Simulation

---

**D. B. Batchelor**

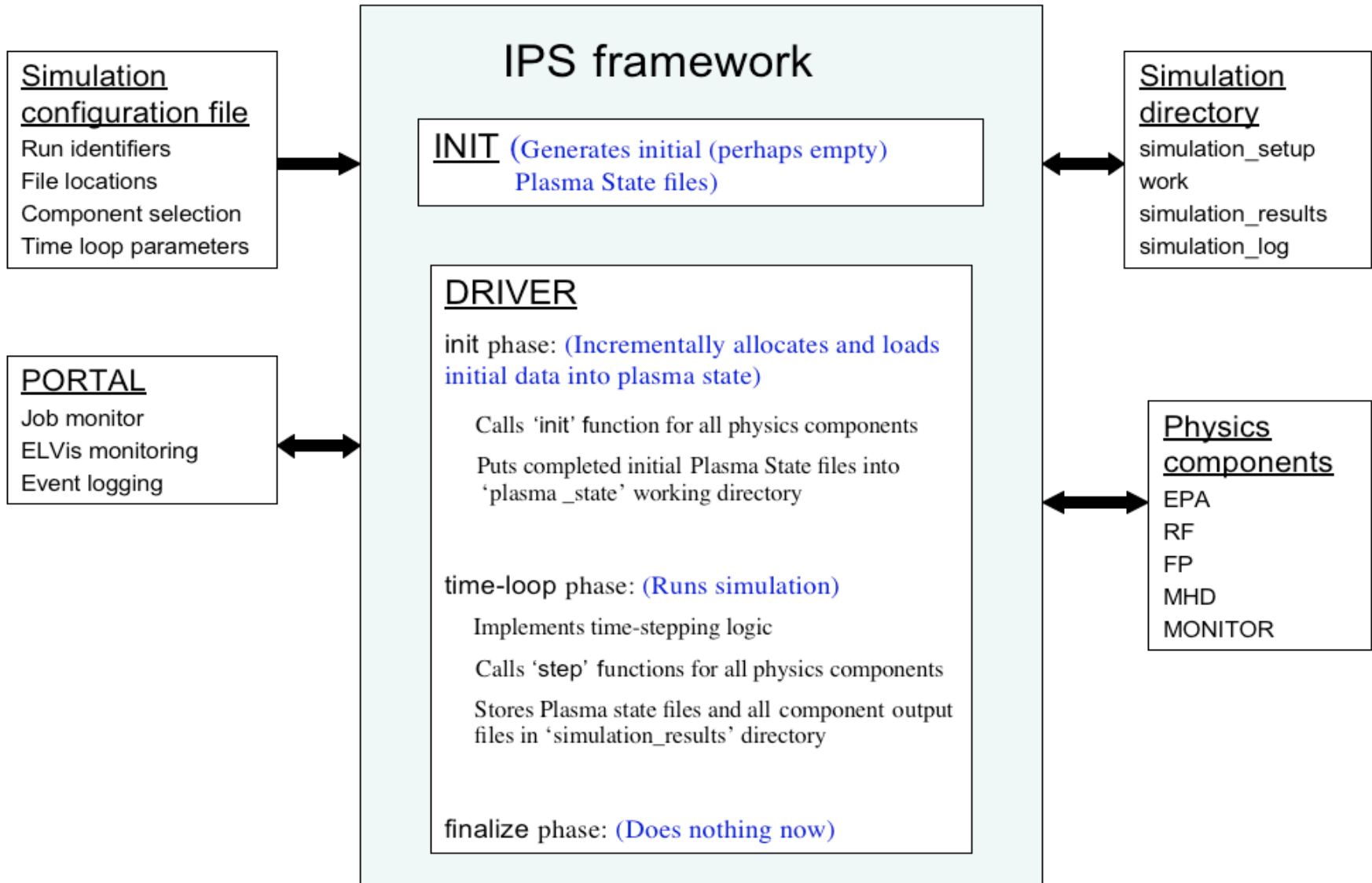
**US-EU Workshop on Integrated  
Simulation**

**2010**

**Goteborg**

- **IPS from a users perspective – no development**

# The IPS from the inside



# Steps to running an IPS simulation

Step	Details
Get access to IPS	Check out from svn repository Do top level make
Figure out what physics components you need	Chase down binaries of physics executables – should be in <code>ips/&lt;components&gt;</code> or <code>physbin</code> Get appropriate specific input files for the physics codes you plan to use – talk to developer or previous user. These tend to live with component scripts in <code>ips/components/&lt;component_class&gt;/&lt;sub_class&gt;</code> of the ips directory
Generate a simulation configuration file	Specify overall simulation configuration: simulation/run identifiers, file naming conventions, what files constitute the plasma state data, what components to use Specify configuration for individual components: names of required input/output files, paths to binaries, number or processors required for executable, any other configuration data you choose
Launch job	<code>bin_path/ips --config= &lt;configuration_file&gt;</code> , or appropriate batch script
Sit back and watch on portal	<a href="http://swim.gat.com:8080/monitor">http://swim.gat.com:8080/monitor</a>

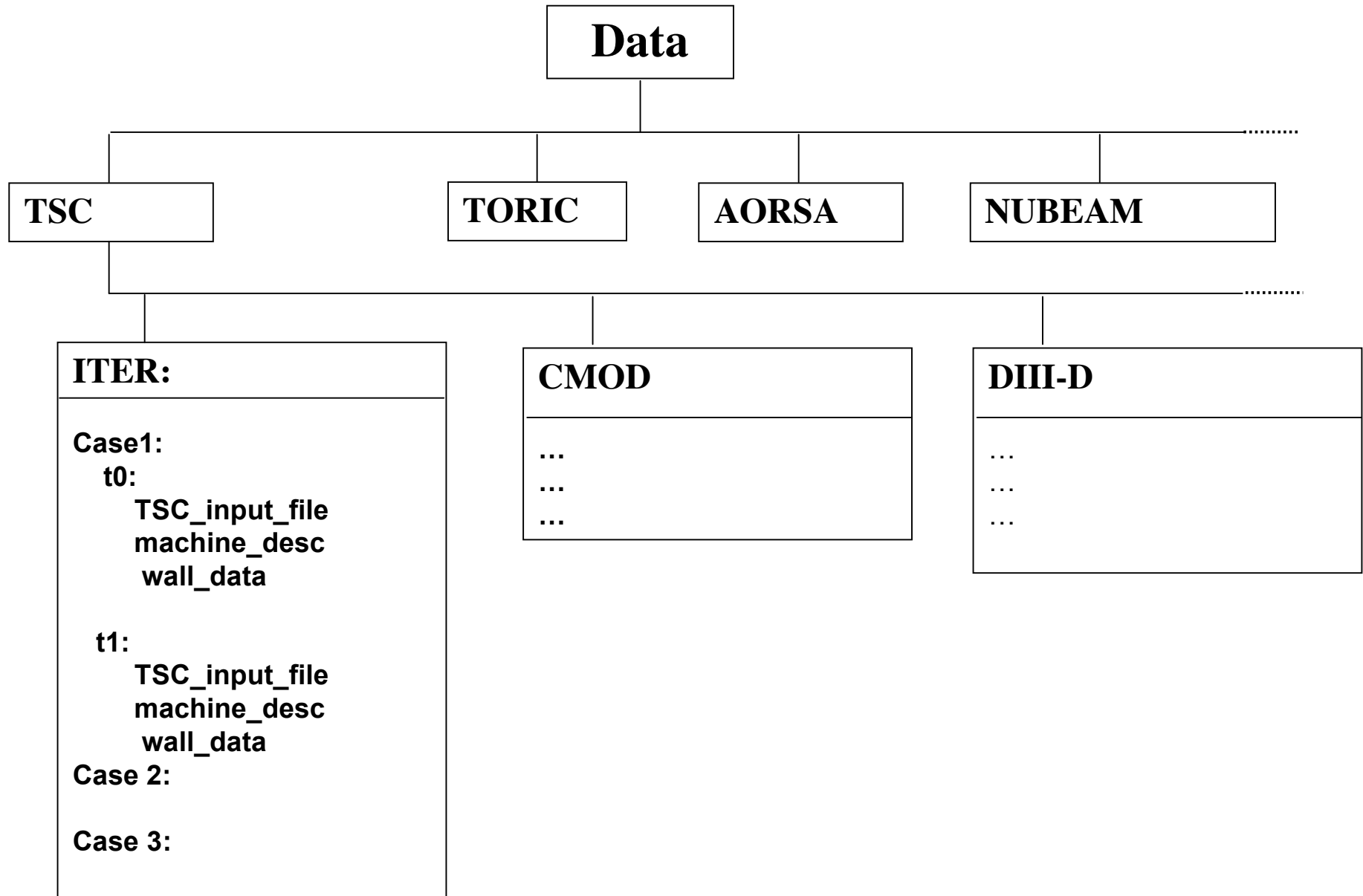
In principle it's not complicated. But there is a lot (*bewildering amount*) of flexibility. We are trying to assemble a set of best practices to streamline the process. We need more users

# What's in the Configuration File?

---

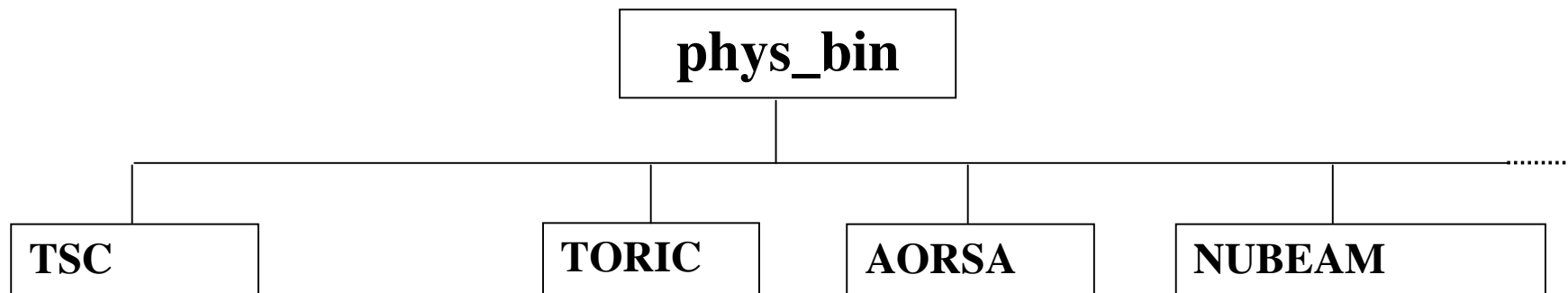
- **User specific data – paths to user directories**
- **Simulation specific data**
  - **Simulation identifiers – run ID, tokamak name, shot number**
  - **Run description text**
  - **Path to IPS binaries (not component code binaries)**
  - **Simulation mode – normal/restart (restart path, restart time)**
  - **List of files constituting Plasma State**
- **Component composition**
  - **List of PORTS – generic names for components**
  - **Implementation name for each port (eg AORSA or TORIC)**
- **Configuration for each component**
  - **Name of component script**
  - **Path to component executable – physics code binary**
  - **Path to component input data files**
  - **Component/implementation configuration data – exposed switch settings, other paths ...**
- **Checkpoint/restart schedule**
- **Simulation time loop**

# SWIM data tree



# Physics executables are collected together SWIM phys\_bin – different on each computer supported

---



## Q. What do you get out of an IPS run?

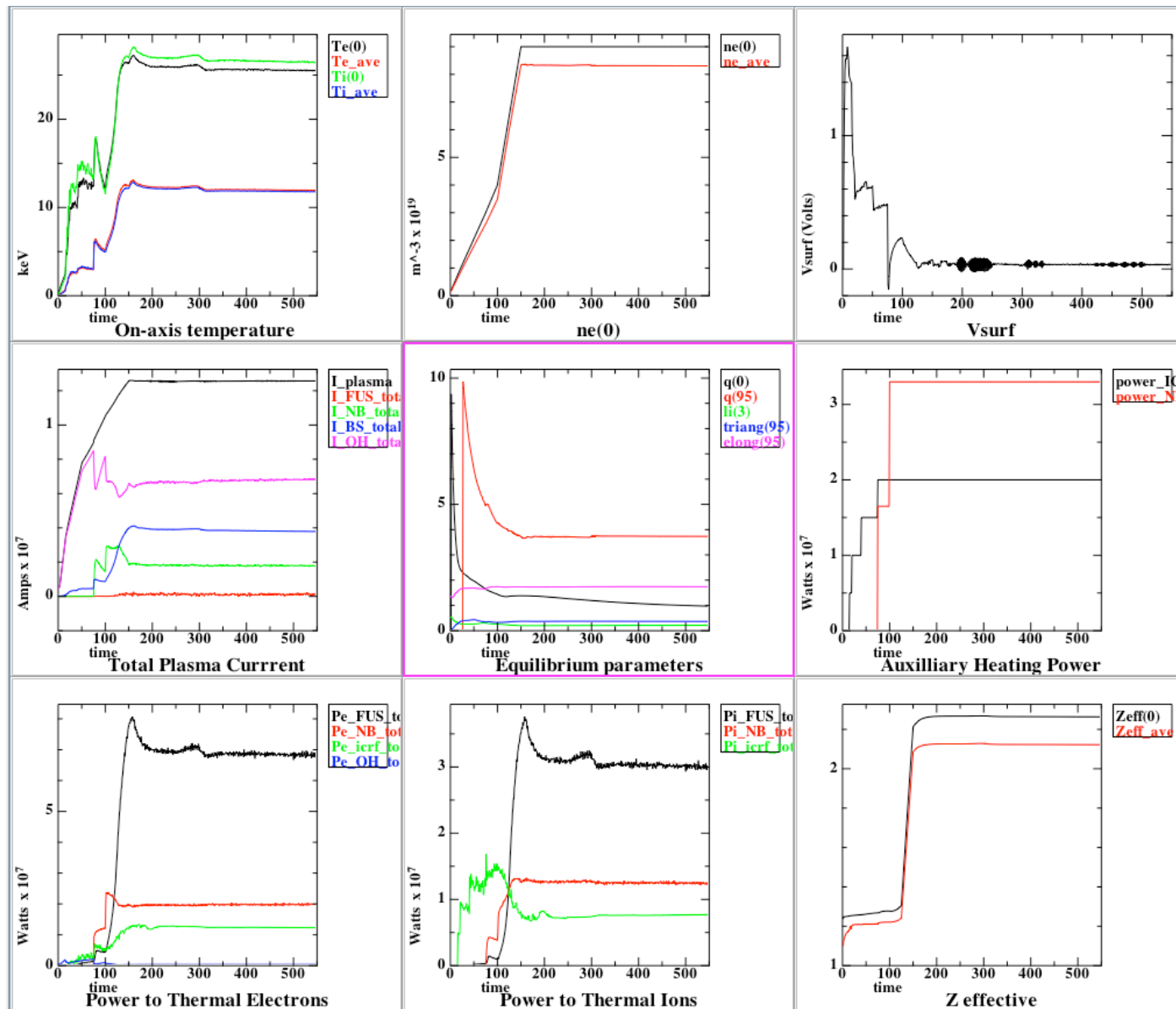
### A. The output directory + monitor file in W3\_dir

```
/scratch/scratchdirs/u2115/genray_testing/genray_test_1 $ ls
total 40
drwxrwxr-x  3 u2115 m876  4096 Nov 22 13:29 restart/
drwxrwxr-x  7 u2115 m876  4096 Nov 22 14:09 simulation_setup/
drwxrwxr-x 12 u2115 m876  4096 Nov 23 07:31 simulation_results/
drwxrwxr-x  9 u2115 m876  4096 Dec  1 04:36 work/
drwxrwxr-x  2 u2115 m876  4096 Dec  1 04:36 simulation_log/
-rw-rw----  1 u2115 m876 10357 Dec  1 04:36 genray_test.config
-rw-rw----  1 u2115 m876   311 Dec  1 04:36 franklin.conf
-rw-rw-r--  1 u2115 m876    30 Dec  1 04:36 PORTAL_RUNID
```

### Simulation results tree

```
drwxr-xr-x  3 u2115 u2115  4096 Nov 19 10:52 1.500/
drwxr-xr-x  2 u2115 u2115  4096 Nov 19 10:52 drivers_dbb_generic_c
drwxr-xr-x  3 u2115 u2115  4096 Nov 19 10:52 2.000/
drwxr-xr-x  3 u2115 u2115  4096 Nov 19 10:59 2.500/
drwxr-xr-x  2 u2115 u2115  4096 Nov 19 11:06 rf_ic_toric_5/
drwxr-xr-x  3 u2115 u2115  4096 Nov 19 11:06 3.000/
drwxr-xr-x  2 u2115 u2115  4096 Nov 19 11:06 nb_nubeam_6/
drwxr-xr-x  2 u2115 u2115  4096 Nov 19 11:12 monitor_monitor_3/
drwxr-xr-x  2 u2115 u2115  4096 Nov 19 11:12 epa_tsc_4/
drwxr-xr-x  2 u2115 u2115 12288 Nov 19 11:12 plasma_state/
```

# The monitor component extracts selected data from the plasma state file to generate summary time history





# You can look at the monitor data in many ways – e.g. profiles

