FACETS Report for Jan-May, 2007

1. Project summary and management

The FACETS project was initiated on January 15, 2007. In advance of receipt of funding, FACETS had an all hands meeting in Boulder during Nov. 30 through Dec. 1, 2006. This meeting allowed an exchange of ideas about coupling, codes to couple, standards and working procedures. In the interval since that meeting, FACETS has set up teams to address each of the critical areas. Those (and their leaders) are Framework (Cary), Core sources (McCune), Edge (Rognlien), Wall (Pigarov), Coupling research (Estep reporting with R. Cohen and Larson), Embedded turbulence calculations (our Scientific Application Partnership with Fahey as team leader), Performance (Kuehn), and Algorithms (McInnes), with the latter two being cross-cutting. In the interests of brevity, we have included the reporting of the cross-cutting activities within the other activities.

2. Framework

The framework team (Tech-X, LLNL, ParaTools) is examining the use of C++ framework to allow coupling of the multiple components that will comprise FACETS. Through a series of conference calls, an initial component interface that will allow implicit coupling has been defined. The design uses an abstract interface design to avoid explicit dependencies on particular implementations. For example, the I/O is done through an abstract API that is independent of the actual underlying library used. A concrete implementation using HDF5 has been implemented, but other mechanisms (e.g., netCDF) and/or data organizations could also be used. The use of C++ templates allows us to provide a uniform interface for a variety of data types and space dimensions and minimizes coding. All code adheres to a strict layering principle that prevents circular dependencies. A component/coupler system allows coupled multi-physics simulations. Each component can be initialized on a separate set of processing elements and communication between nodes is handled by parent components using a client-server model. This allows development and testing of each individual component separately, and then coupling the component into the system as described by an XML-like input file

3. Core sources

FACETS will model the power deposition due to neutral beam and fusion product fast ion sources, using PPPL's Monte-Carlo fast ion package NUBEAM. FACETS is leveraging an ongoing effort for MPI parallelization of NUBEAM and has added technical support for improvements to NUBEAM portability and build mechanisms.

MPI parallelization: dependence on file based I/O was removed; MPI broadcast is used to distribute core equilibrium and plasma profiles to child processes; MPI reduce is used to receive contributions to Monte Carlo sums from child processes. For an N processor MPI run, each processor is responsible for contributions from a fraction (1/N) of the Monte Carlo ions representing each species. Each processor's sublist is a random sample of the total list. Good load balance is achieved, as well as linear scaling for sufficient problem size; this has been verified up to 16 processors on various small Linux clusters.

NUBEAM is a “stateful” time dependent model; the slowing down and pitch angle scattering times for fast ions are long compared to time steps used. The code can resolve the transients associated e.g. with changes in injected beam power, or in response to macroscopic events such as sawteeth. In the MPI implementation, “distributed” states—particle sublists—are stored in portable NetCDF files locally for each process, between time steps.

To facilitate scaling studies, a Python code has been provided to repartition particle list files, allowing rerunning of the same calculation with a changed number of processor elements.

Portability and build systems: an autotools-based build system for NUBEAM has been provided, based on the PPPL NTCC version. The code has been ported to the Tech-X quad processor system quartic.txcorp.com, using the 64 bit version of Lahey-Fujitsu fortran (for the first time). Portability testing to SGI and IBM-based systems has also been initiated and issues are being addressed.

4. Edge

The initial activity of the edge model group has been the development of a utility that allows present/future versions of the 2D multi-species edge transport code UEDGE to be automatically converted to F90 source code with corresponding F90 modules to communicate variables between subroutines and python-script drivers. Standard UEDGE uses Basis-style variable descriptor files, Basis-script drivers, and MPPL source files. This utility now allows UEDGE to be run under Python on many different platforms using the Forthon software, or to fit into the FACETS framework as a F90 component, while allowing maintenance and upgrade of the Basis version for continued distribution to the existing user base. The resulting Python version of UEDGE has been built and tested on Linux and Mac platforms to date. While UEDGE is presently available for DOE programs, the necessary paperwork has been submitted for open-source release from LLNL in a form compatible with FACETS licensing plans. The development of a prototype wrapper file utilizing Babel that can provide communication of variables to couple to a core transport and wall models and to control UEDGE implicit time-advancement within the FACETS framework has begun and should be completed this summer. With distribution of the new Python version of UEDGE and test cases, we have begun joint work with the FACETS algorithm and performance groups. The algorithms work will initially focus on improving the preconditioning step of the UEDGE Newton-Krylov solver by comparing results between PETSc (including the link to Hypre’s algebraic multigrid) and the present UEDGE Newton-GMRES-ILUT. As the parallel version of UEDGE is revived (planned for the fall), preconditioning improvements for a domain-decomposed system will be added to the work. The performance group will provide information on bottlenecks for both the serial version, and when available, the parallel version. As the coupled core/edge system becomes available, the impact of the coupling on algorithms and performance will be added to this work.

5. Wall

A. Pigarov and S. Krasheninnikov (UCSD) have produced a simplified zero-dimensional wall model (WALLPSI) for computing wall temperature, trapped and mobile hydrogen concentrations as well as erosion rates. Alex Pletzer (Tech-X) has integrated WALLPSI into the FACETS repository, providing an interface callable from C and bringing the package into a cross-platform build system in order to prepare the ground for coupling to other physics modules (e.g. UEDGE). Current work on WALLPSI includes (i) code benchmarking by solving standard wall physics problems, (ii) incorporating data about the elementary processes describing hydrogen kinetics within the wall and on its surface, and (iii) coupling WALLPSI to UEDGE within the FACETS framework. In addition, there is an ongoing effort aimed at developing a one-dimensional model which will enable the study of plasma-wall interactions and their effect on plasma stability.

6. Coupling research

The activities of the coupling team (CSU, LLNL, ANL), including the physics algorithms, applied mathematics, and computer science groups, have so far been primarily organizational and information gathering. We have had a series of presentations covering the algorithms of the UEDGE 2D edge code, previous activity in coupling UEDGE to core transport, and mathematical tools for measuring the goodness of coupling algorithms. The physics algorithm group has also developed some simplified test models for coupling the core and the edge. The simplified test models -- one, a set of coupled non-linear ODEs; and two, a set of coupled PDEs, treating the edge with a two-temperature model and the core with a one-temperature model -- were developed as targets for algorithm development and instrumentation. The applied mathematics group has begun studying the simplest ODE test model behavior, including the stability properties of the model relevant to operator decomposition discretizations.

7. Embedded turbulence computations (SAP)

The goal of this project is to develop a prototypical steady state gyrokinetic transport (SSGKT) code that integrates micro-scale gyrokinetic turbulence simulations into a framework for practical multi-scale simulation of the International Thermonuclear Experimental Reactor (ITER). The prototype will have the capability to predict steady-state core temperature and density profiles given the H-mode pedestal boundary conditions. This addresses a key problem of critical scientific importance; namely predicting the performance of ITER given an edge boundary condition. A prototype code, TGYRO, has been implemented and tested on several platforms. This prototype couples multiple instances of GYRO together (which will give us the ability to scale to many thousands of processors) and implements a feedback loop between turbulence and transport – the transport part of the prototype is an overly simplified model, but will become more realistic in the following months.

Using standard fixed-profile GYRO gyrokinetic simulations we can now perform transport-timescale simulations in which the profiles are adjusted to achieve steady-state power balance. In short, the approach is to find the unique plasma profile for which losses due to core turbulence are balanced (macroscopic steady state) by sources (particles, momentum and energy.) We have done some early testing with the TGYRO prototype. Preliminary results (with four instances of GYRO) yield smooth profiles.

8. Team meetings

The FACETS team met in Boulder on Nov. 30-Dec. 1. The meeting schedule and talks are available on line at

https://www.facetsproject.org/facetsweb/meetings/30Nov06/FacetsAgenda30Nov06.html.

Plans are for an all-hands meeting in Boulder for Aug. 9-10, 2007.

9. Infrastructure

Standard groupware has been set up. This includes a mailman list and a wiki into which the portal for the website has been put. That website is at https://www.facetsproject.org/facets/. The website uses TRAC, so that it doubles as a bug/issue tracking system, and it allows one to browse the source. We are still evaluating this system, as the world of wikis is rapidly changing.

We have put into place a regression test system that checks out the FACETS code, builds, and runs unit tests on a nightly basis. Problems of code integrity, including lack of code documentation and cyclic dependencies, are reported. We will be extending this system to cover other components as they are added to the system.

10. Collaborations and synergistic activities

FACETS has active collaborations with PERI (through participation of Kuehn), TASCS (with Larson and McInnes, as well as through TASCS funding to Tech-X), TOPS (McInnes), and VACETS (through contacts with Bethel and Childs). It should also be noted that FACETS has provided access to its main email list to the other integration SciDACs, and that they have access to ours. Further, all of the integration SciDACs are inviting representatives to each others collaboration meetings.

As part of Tech-X TASCS funding, prototype Babel implementation for allowing C codes to call Fortran libraries has been created. This prototype implementation is consistent with the FACETS build system to enable quick implementation into the FACETS application once the Fortran libraries meet the FACETS API standards.

11. Presentations

J.R. Cary and the FACETS team: "FACETS: Framework Applications for Core-Edge Transport Simulations". Talk presented at the SWIM Collaboration Meeting, September 25 (2006).

J.R. Cary and the FACETS team: "FACETS: Framework Applications for Core-Edge Transport Simulations". Talk presented at the US-Japan Workshop on the Integrated Simulation of Fusion Plasmas, January 29 (2007).

S.E. Kruger et al.: "Overview of Framework Application for Core-Edge Transport Simulations (FACETS)", Poster presented at the 12th US-EU Transport Taskforce Workshop (2007).

M. Miah et al.: “Parallel Infrastructure and Components For Whole-Device Modeling”. Poster presented at the Sherwood Fusion Theory Conference (2007).

A. Pletzer et al.: “The FACETS project”, Poster presented at the Sherwood Fusion Theory Conference (2007).

J.R. Cary and the FACETS team: “FACETS: Framework Applications for Core-Edge Transport Simulations”. Poster presented at the US-Japan Workshop on the Integrated Simulation of Fusion Plasmas, SciDAC PIs meeting, February 5 (2007).

J.R. Cary and the FACETS team: “FACETS: Framework Applications for Core-Edge Transport Simulations”. Poster presented at the CPES Collaboration Meeting, April 30 (2007).

M.R. Fahey et al., "Status on the computational aspects of developing a fully gyrokinetic transport code," Poster presented at the 12th US-EU Transport Taskforce Workshop (2007).

J. Candy, "Progress on a Fully Gyrokinetic Transport Code," Poster presented at the 12th US-EU Transport Taskforce Workshop (2007).