Data Management and Simulation Support
Accelerating Carbon Capture through Computing

You-Wei Cheah, Joshua Boverhof, Abdelrahman Elbashandy, Deb Agarwal, Jim Leek, Tom Epperly, John Eslick, David Miller

IEEE 12th Intl Conf on eScience 2016
Carbon Capture Challenge

• The traditional pathway from discovery to commercialization of energy technologies is long\(^1\), i.e., \(~ 20-30\) years

• President’s plan\(^2\) requires that barriers to the widespread, safe, and cost-effective deployment of CCS be overcome within 10 years

• To help realize the President’s objectives, new approaches are needed for taking concepts from lab to power plant, quickly, at low cost and with minimal risk

• Carbon Capture Simulation Initiative (CCSI) designed to accelerate the development of CCS technology, from discovery through deployment, with the help of science-based simulations

---

Carbon Capture Simulation Initiative

- Identify promising concepts
- Reduce the time for design & troubleshooting
- Quantify the technical risk, to enable reaching larger scales, earlier
- Stabilize the cost during commercial deployment

National Labs
- Berkeley Lab
- Pacific Northwest National Laboratory
- NETL
- Los Alamos National Laboratory
- Lawrence Livermore National Laboratory

Academia
- Carnegie Mellon
- Virginia Tech
- Penn State
- Princeton University
- West Virginia University

Industry
- ADA
- Ameren
- B&W
- GE
- Alstom
- Fluor
- RAMGEN
- AEP
- Duke Energy
- EDISON
- Southern Company
- Chevron
- EASTMAN
- WorleyParsons
- URS

Essential for accelerating commercial deployment
CCSI Integrated Process Design Environment

Bench-scale Experiments

Particle-scale Simulations

Process Simulations

Small-scale Deployments

Uncertainty Quantification, Decision Support, Optimization, etc

Knowledge, Information, & Integrated User Environment

Decision Makers
CCSI Toolset

- Comprehensive, integrated suite of validated science-based computational models
- Modular design that leverages existing software components
- Simulation and data management support provided through CCSI Integration Framework
- Components:
  - Core capabilities for optimization, modeling and uncertainty quantification
  - Orchestration: FOQUS
  - Process simulation framework: Turbine, SimSinter, DMF
CCSI Toolset Architecture

**FOQUS**
Framework for Optimization Quantification of Uncertainty and Sensitivity

**Meta-flowsheet**: Links simulations, parallel execution, heat integration

**Turbine**
Parallel simulation execution management system
Desktop – Cloud – Cluster

**SimSinter**
Standardized interface for simulation software
Steady state & dynamic

**Simulation**
Aspen
gPROMS
Excel

**D-RM Builder**

**SimSinter Config GUI**

**Data Management Framework (DMF)**
FOQUS

• Framework for Optimization and Quantification of Uncertainty and Sensitivity
• Serves as the primary computational interface in the CCSI Toolset.
• Interface to simplify running complex modeling and UQ studies
• Modular design involving plugin system
• Flowsheet: Composite model, Meta-Flowsheet: Combination of flowsheets
• Provides GUI and platform for flowsheet analysis tools
• Developed in Python/PyQt/PySide
FOQUS: GUI
Turbine Science Gateway

• Scaling up experiments
  – Solving large scale simulations (particles, CFD)
    • Dense phase, reactive flows with complex submodels
  – Multiple simulation runs (optimization, UQ)
    • Multiple scales (Particle, Device, System)
• Batch system providing staging of input and output files
• Generic solution that can be extended to process modeling and simulation packages
• Integrated with FOQUS to schedule and scale-up simulation runs
Turbine Science Gateway: Components

- Designed to operate primarily in Windows
- Turbine Web application:
  - Windows service
  - RESTful, HTTP API
  - Five resources in API: Application, Simulation, Job, Consumer, Session
  - Python library for interfacing with other tools
- Turbine Client
  - Platform independent
- Turbine Database
  - SQLite
  - Stores state and results
- Turbine Server
  - Executes and manages simulation process through use of SimSinter through Turbine Workers
  - Multiple workers can be used to form Turbine Cluster
Turbine Server Experiences

• Framework can be used with single machines, clusters, Cloud computing resources
• Scale simulations to allow computations in thousands
• Successfully executed 400 instances of Aspen Plus simulations using Amazon EC2
• Harnesses Amazon EC2 spot instances vs owning a cluster of computers
• Parallelization increases application throughput and decreases time to solution
• Integrated Mass Transfer Model
  – Local optimization (single processor) 12 hours
  – Cloud optimization (4-6 consumers) 2.75 hours
Turbine Science Gateway: Use case

Plot of simulation runtime versus start time of simulation execution

NETL Optimization: Successes and Failures

runtime (minutes)

start time

Jan 22
Jan 23
Jan 24
Jan 25
SimSinter

• Provides extensible support with various commercial simulation tools
  – Aspen Custom Modeler, Aspen Plus, gProms, Microsoft Excel
• Standard Interface library for driving single-process Windows based process simulation software
• Based on .NET and Microsoft COM interface
• Connects Turbine Science Gateway with process simulation tools
• Sinter configuration files:
  – Created by model creators
  – Identify simulation input and output variables
  – JSON format
Simsinter Config GUI

- SimSinter Config GUI: Allow easier creation and editing of Sinter configuration files
DMF: Motivation

• Recognition that computational experiments are an important resource
• Financial decisions based on computational experiments
• Need for system to permanently store information about computational experiments:
  o Complete specification of the computational experiment (all the inputs)
  o Significant results files (outputs)
  o Metadata (who, when, what)
  o Dependencies of inputs and results (provenance)
DMF: Requirements and Impact

- Data management capabilities for CCSI data:
  - Browsing
  - Searching
  - Versioning
  - Metadata tracking
  - Dependency/Provenance tracking
  - Facilitate sharing

- Integration with other CCSI tools to provide better workflow
DMF: Components

- Developed in Python 2.7
  - Needs to run on both Windows & Linux platforms
- Two versions of the DMF:
- DMF Browser
  - GUI supporting both versions of DMF
  - Developed using PyQt / PySide
  - D3 for provenance visualization
- Command line tools
  - Basic Data uploader
  - Simulation uploader
DMF Browser: Provenance
Conclusions & Future Work

• Traditional end-to-end process for carbon capture takes decades
• The CCSI toolset integration framework is designed and deployed to scale simulations and facilitate the science for carbon capture simulation
• CCSI Phase I is completed
• Augmenting existing CCSI Toolset with tools to help
• Implementation of dashboard to present and integrate existing data in an effective manner
Disclaimer: This presentation was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.