SAND 2011-4586C
SAND2011-4586C

Chemical Supply Chain and Resilience Project
FY 2010 Workshop

The N-ABLE™ Simulation Environment

Brian S. Jones
Sandia National Laboratories
bsjone@sandia.gov

July 12, 2011

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Takeaways

Flexible, High-Performance, Repeatable Simulation Platform

Computer Scientists work side-by-side with Domain Experts

Methodological Best Practices

Simulation Software Contributors

Brian Jones

Eric Eidson

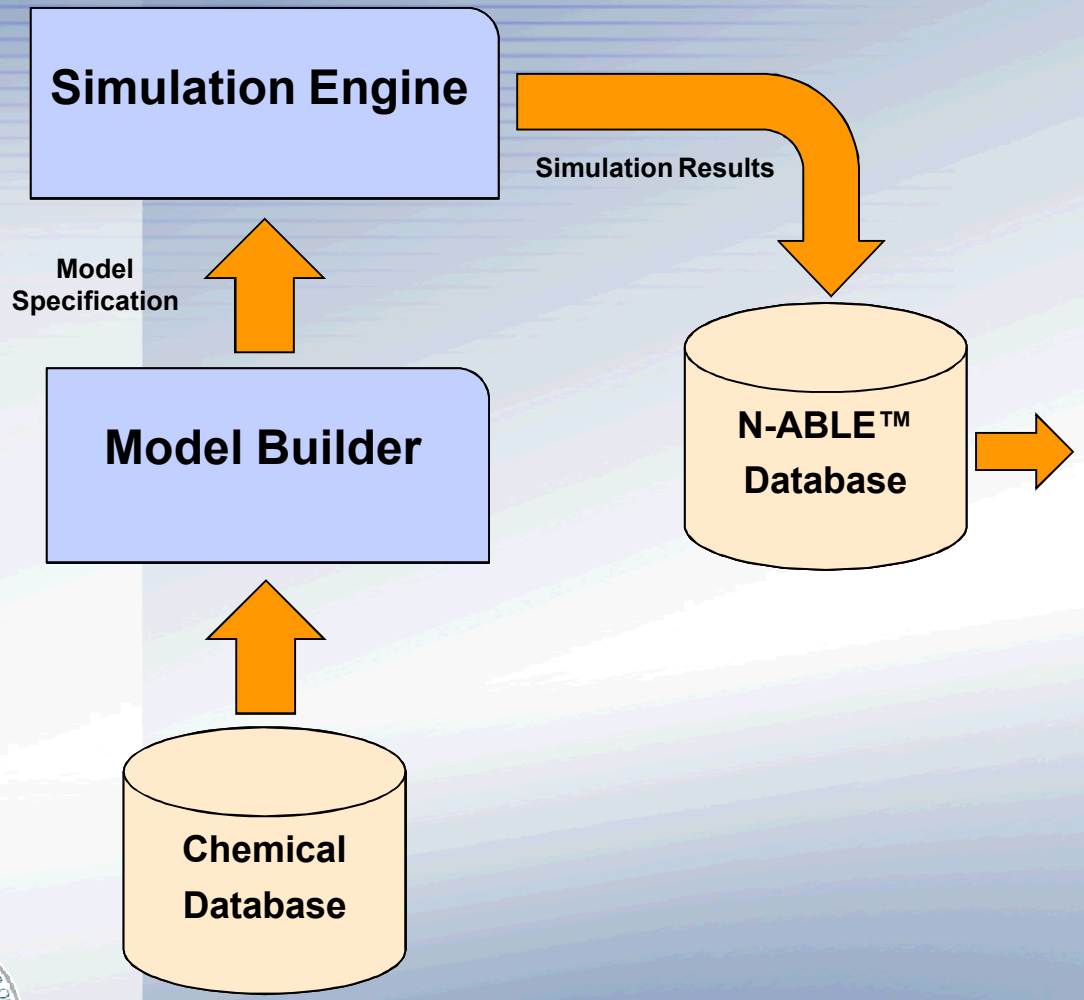
Greg Mackey

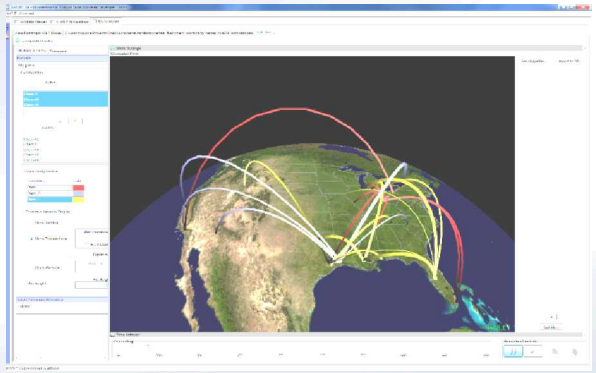
John Masciantoni

Roger Mitchell

Mark Ehlen

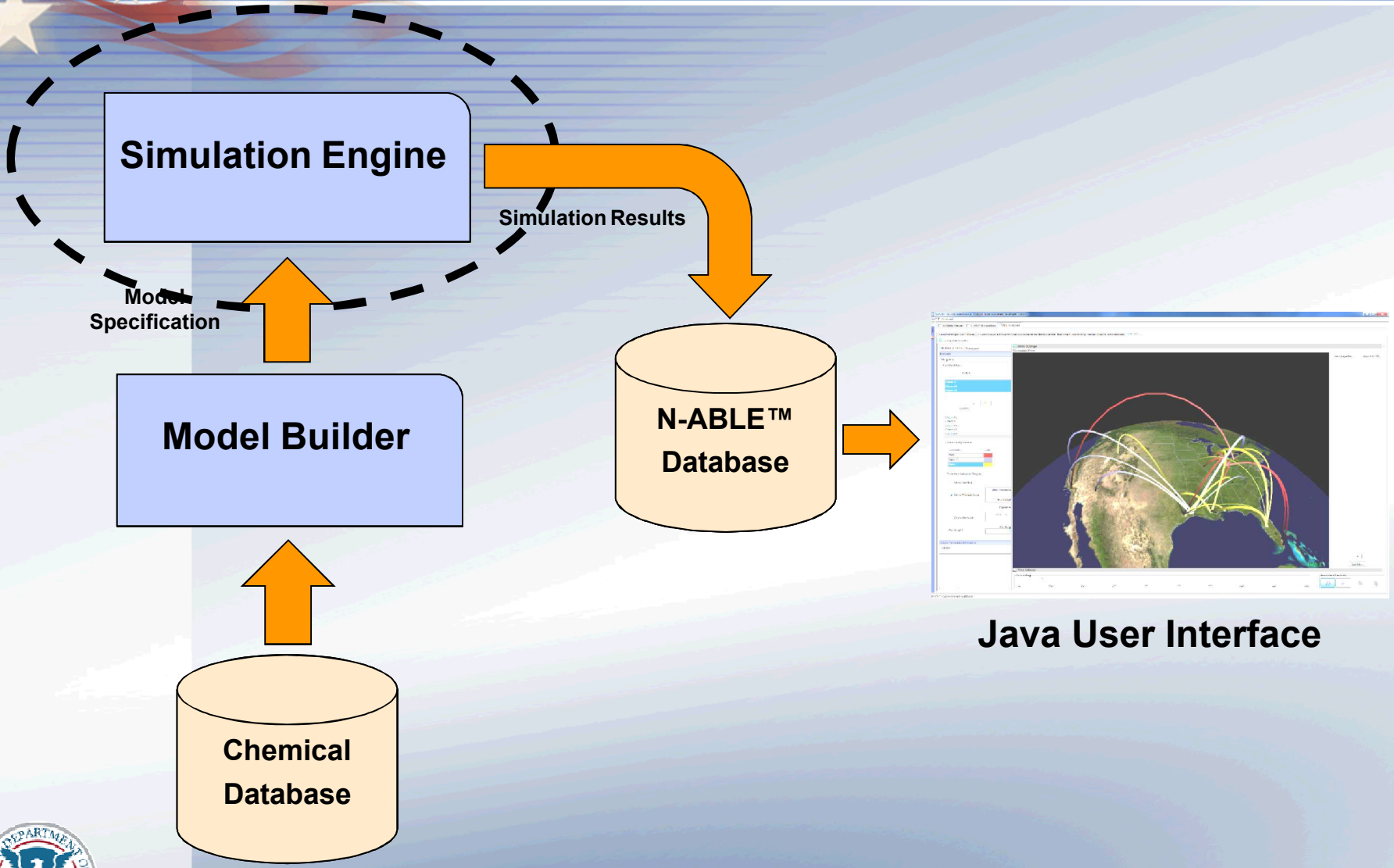
Architecture





Java User Interface

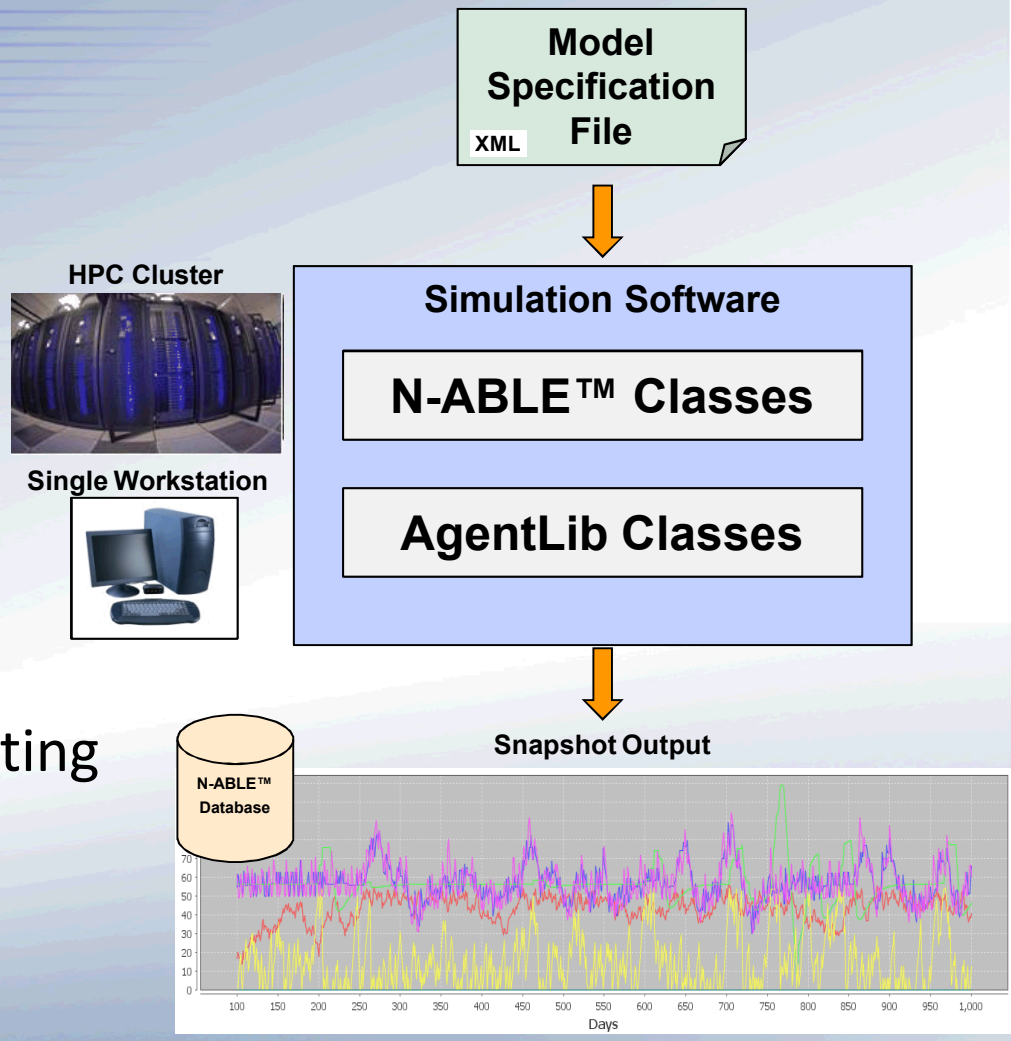
Section: Simulation Engine



Java User Interface

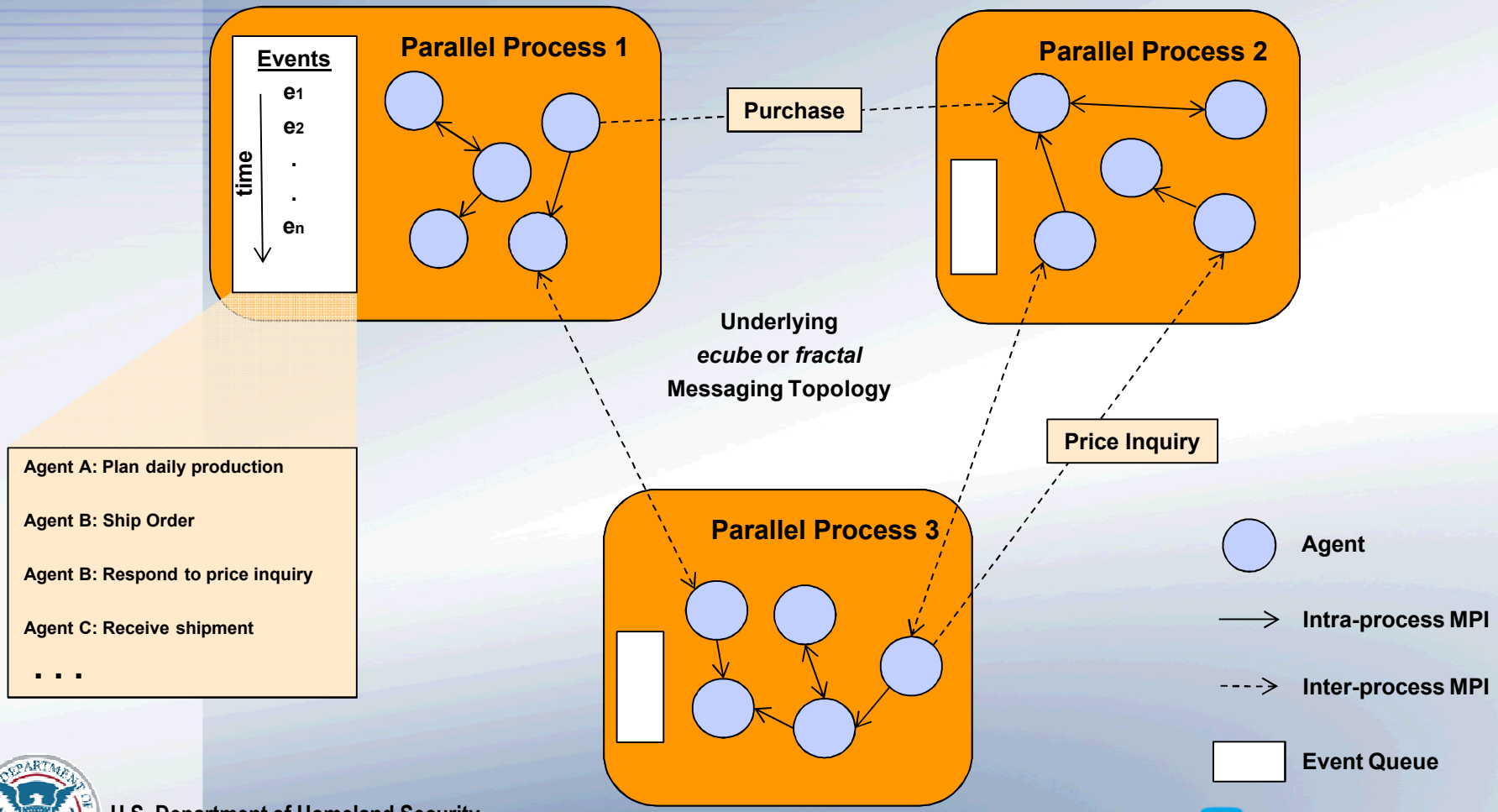
Simulation Engine

- **Object-Oriented Discrete Event Engine**
- **High Performance**
  - MPI Parallelism
  - Multi-core / HPC
- **Repeatable**
  - Stochastic aspects but reproducible given same starting seed
- **Portable**



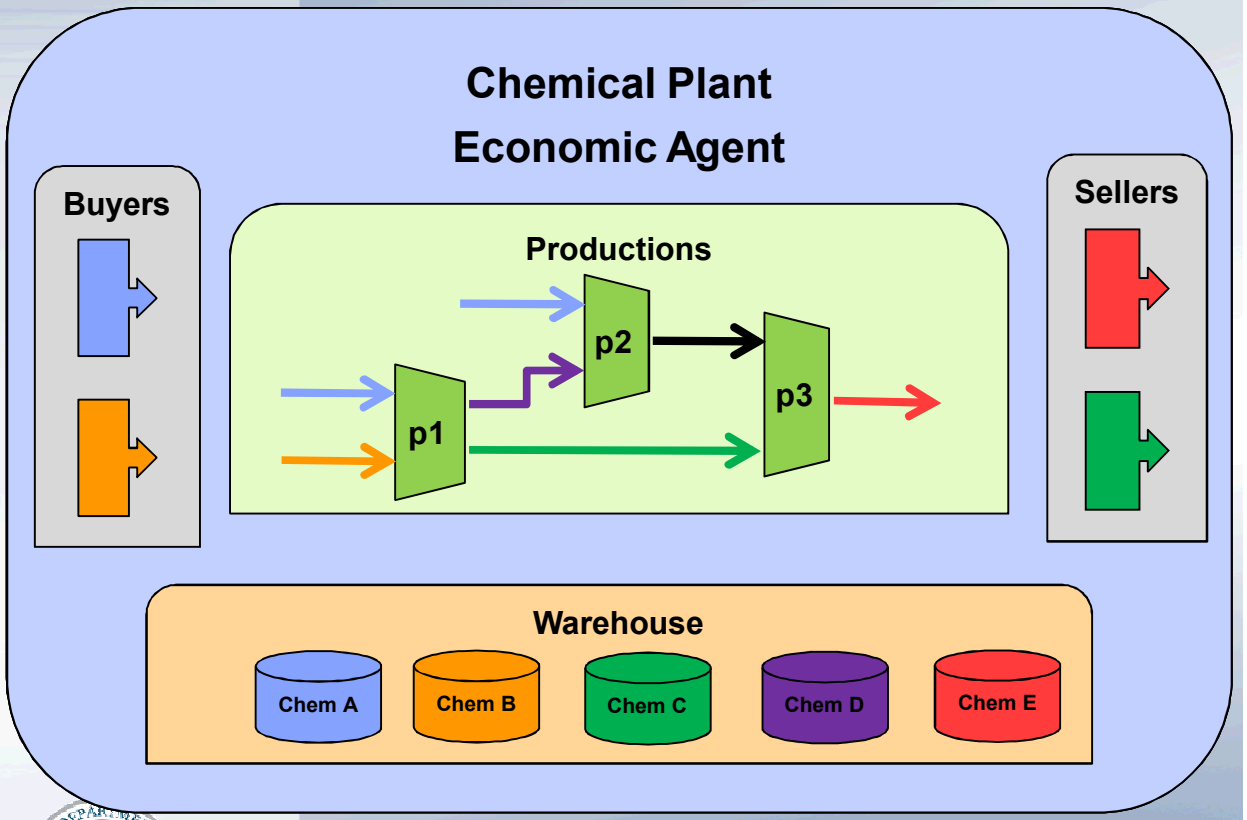
Parallelism

Parallel Processing allows N-ABLE™ to scale to meet problem needs



Object Structure / Define File

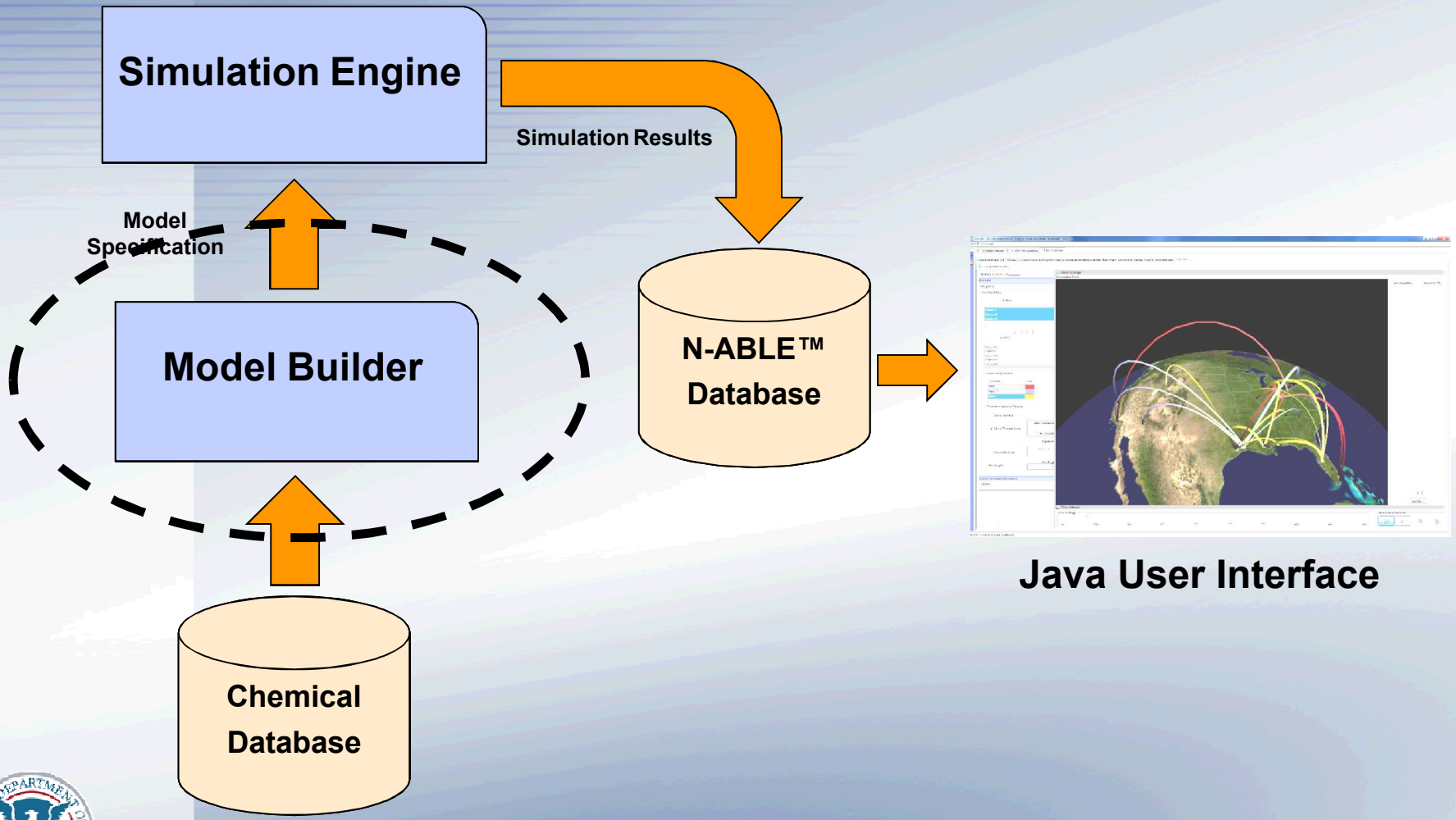
- Object structure mirrors the Model
- Modular design allows pluggable strategies



XML Model Specification, Simplified Snippet

```
<EconomicAgent name="ChemPlant_1">
    <ProductionSleepPeriods/>
    <Location longitude="-95" latitude="30" />
    <InvBasedProdManager>
        <Buyers>
            <FirmBuyer commodity="Chem A"/>
            <FirmBuyer commodity="Chem B"/>
        </Buyers>
        <Sellers>
            <FirmSeller commodity="Chem C"/>
            <FirmSeller commodity="Chem E"/>
        </Sellers>
        <Productions>
            <Producer maxDailyBatches="30" label="p1">
                <ProductionRecipe>
                    <Inputs>
                        <Input commodity="Chem A" amount="0.7"/>
                        <Input commodity="Chem B" amount="2"/>
                    </Inputs>
                    <Outputs>
                        <Output commodity="Chem C" amount="1.5"/>
                        <Output commodity="Chem D" amount="1.2"/>
                    </Outputs>
                </ProductionRecipe>
            </Producer>
            . . .
        </Productions>
        <ProductionChain>
            <Links>
                <Link to="p2" from="p1" commodity="Chem D"/>
                <Link to="p3" from="p1" commodity="Chem C"/>
                . . .
            </Links>
        </ProductionChain>
    </InvBasedProdManager>
    <InitialStock>
        <Stock minAcceptable="100" maxCapacity="3000" commodity="Chem A" amount="500"/>
        <Stock minAcceptable="300" maxCapacity="5000" commodity="Chem B" amount="1000"/>
        . . .
    </InitialStock>
</EconomicAgent>
```

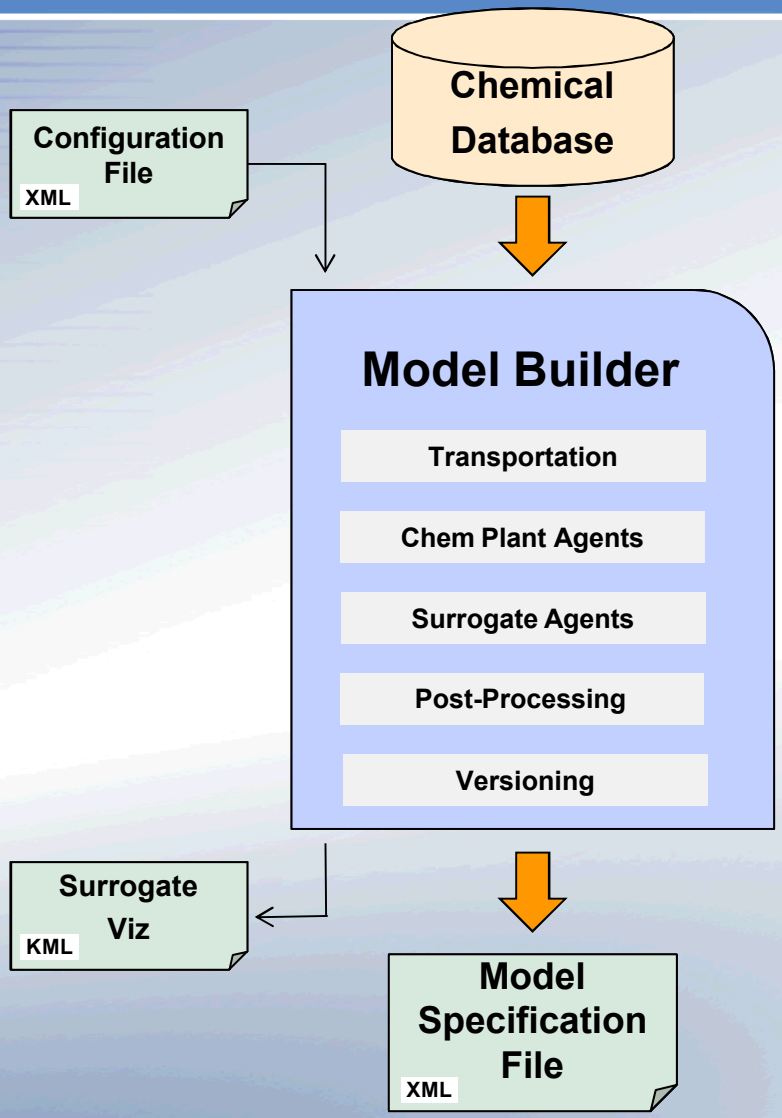
Section: Model Builder



Java User Interface

Model Builder

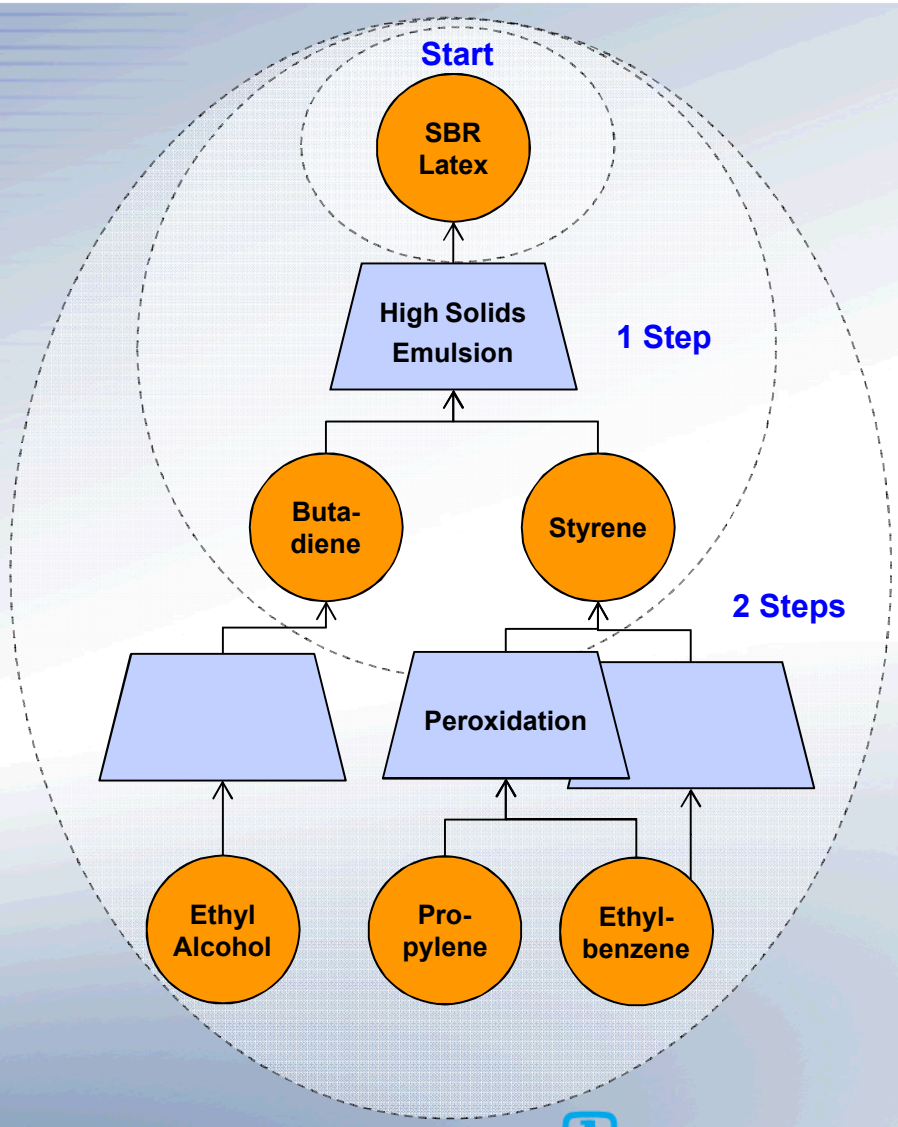
- Turns CDM data into N-ABLE™ model specification
- Modular Processing Pipeline
- File-based Configuration
- Captures Version Info
  - Software versions
  - Data versions



Model Builder

Process Graph Traversal

- **Input:**
  - Starting chemicals
  - # of Steps
- **Output:**
  - Chemicals and Processes connected to starting set
  - List of plants that utilize the processes



* Graph simplified for clarity

Model Builder

Builder Configuration File

```
<?xml version="1.0" encoding="UTF-8"?>

<modelbuilder-ccnfig>

  <!-- ========================
       DATA AND SIM SETTINGS
       ======================== -->
```

Database settings
Chemical graph traversal settings
Simulation duration, snapshot settings, etc.

```
  <!-- ========================
       AGENT CONFIG
       ======================== -->
```

Buyer/Seller exceptions
Plant Agent behavioral settings

```
  <!-- ========================
       SURROGATE CONFIG
       ======================== -->
```

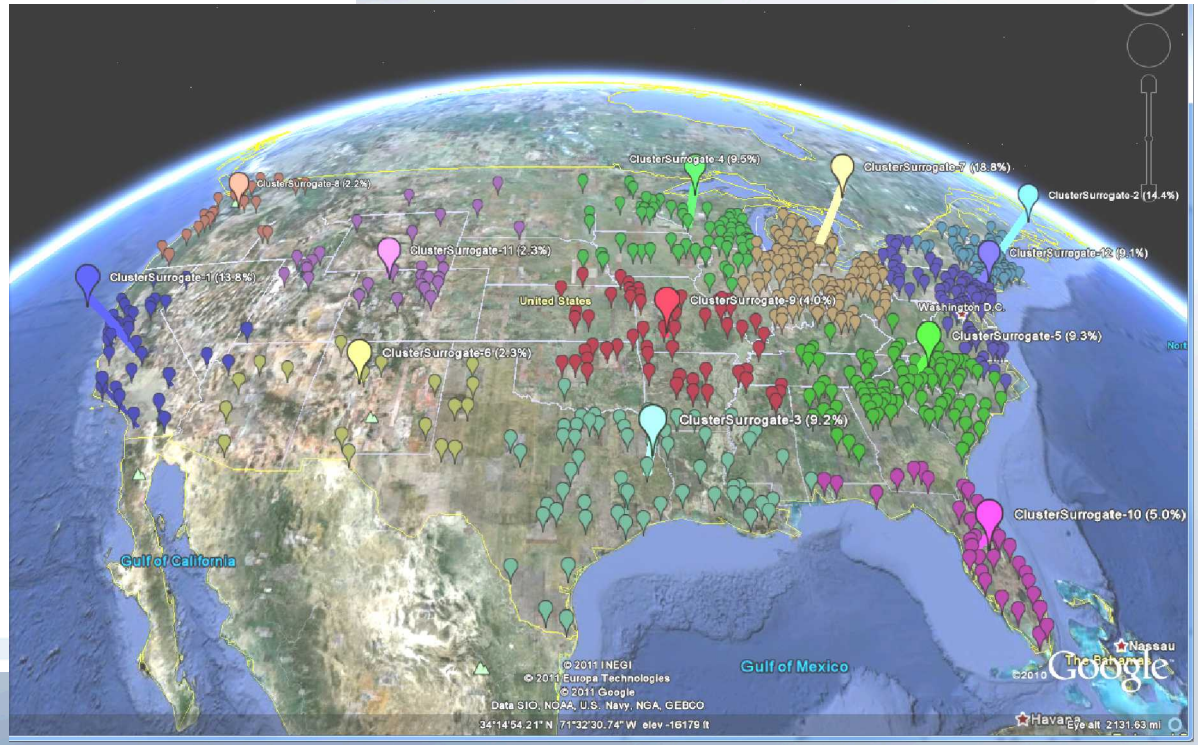
Supply / Demand augmentation
Surrogate Agent behavioral settings

```
  <!-- ========================
       TRANSPORTATION
       ======================== -->
```

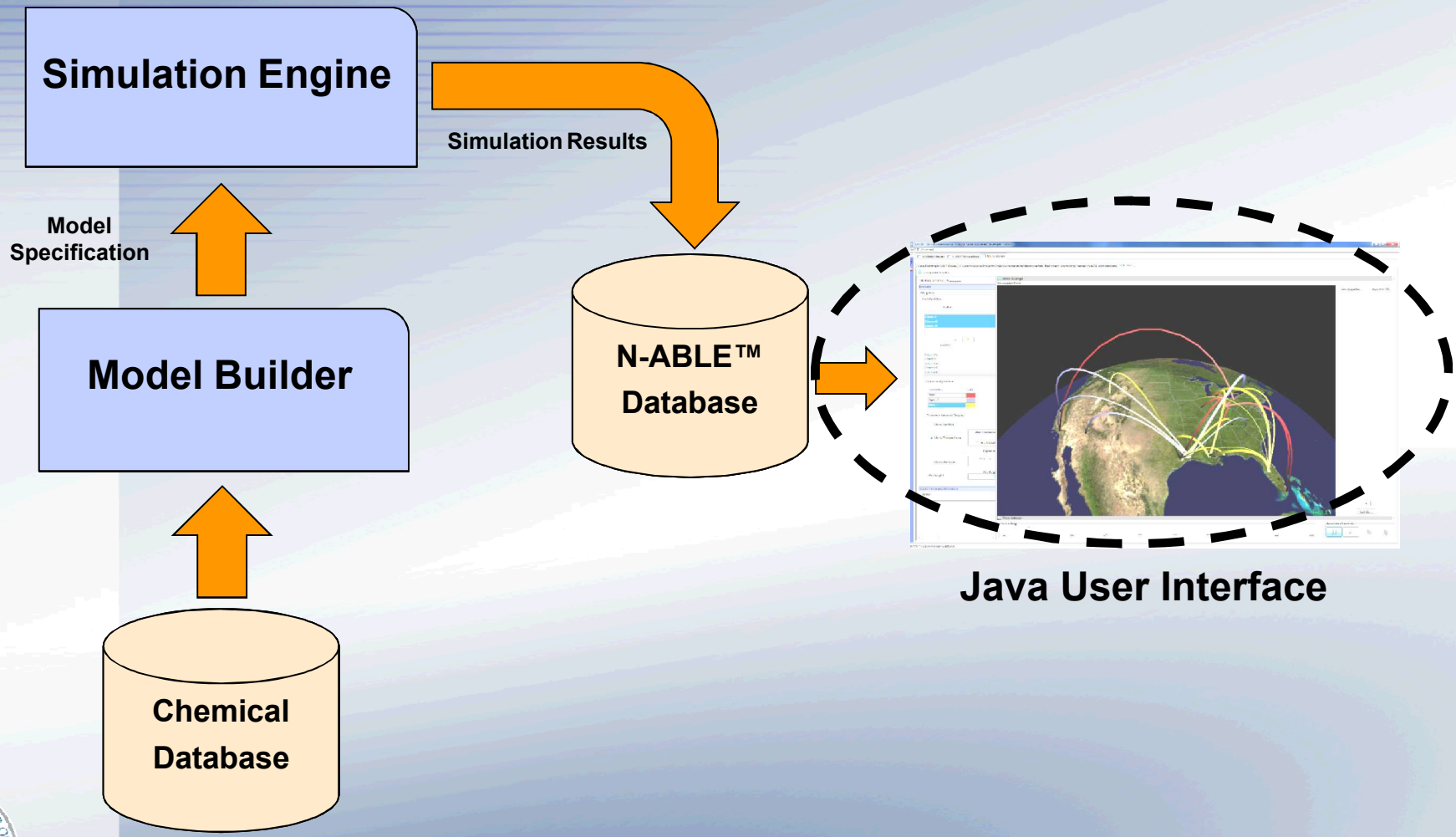
Network or simpler 'straight line' shipping
Network settings (rail, truck, water, etc.)

```
</modelbuilder-config>
```

Chemical Demand, Spatial Clustering

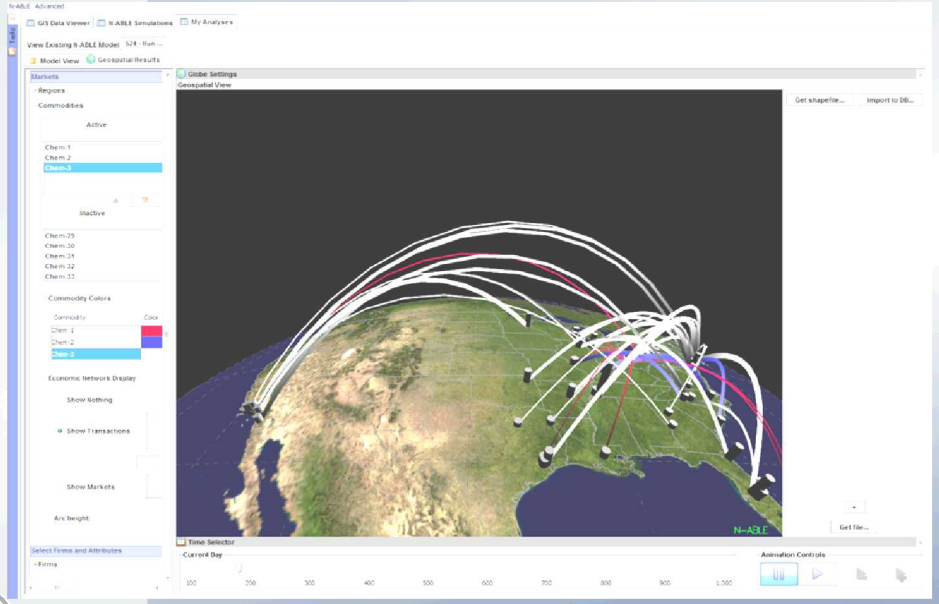


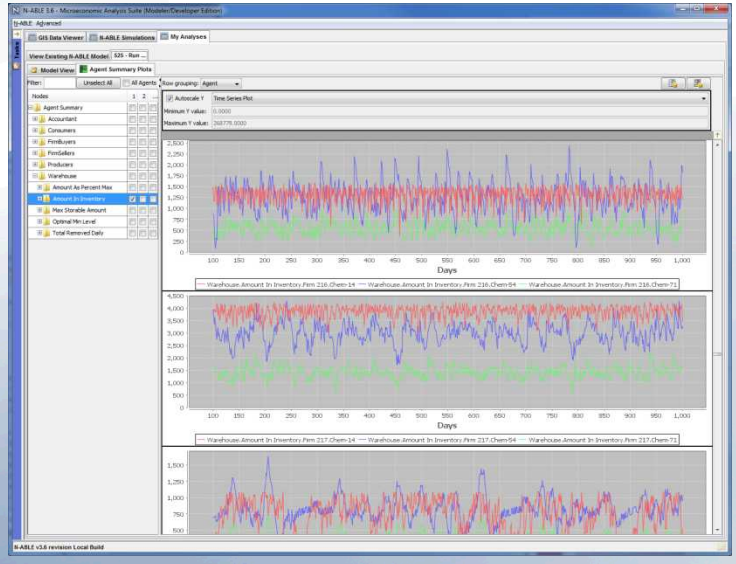
Section: User Interface



User Interface

- **Supports three main activities:**
  - Pre-sim Model Building and Analysis
  - Post-sim Viz and Analysis
  - Simulation Archiving, including Baseline/Disruption management
- **Portable Java code run via Webstart**
- **Supports both 2D and 3D visualization via OpenGL**



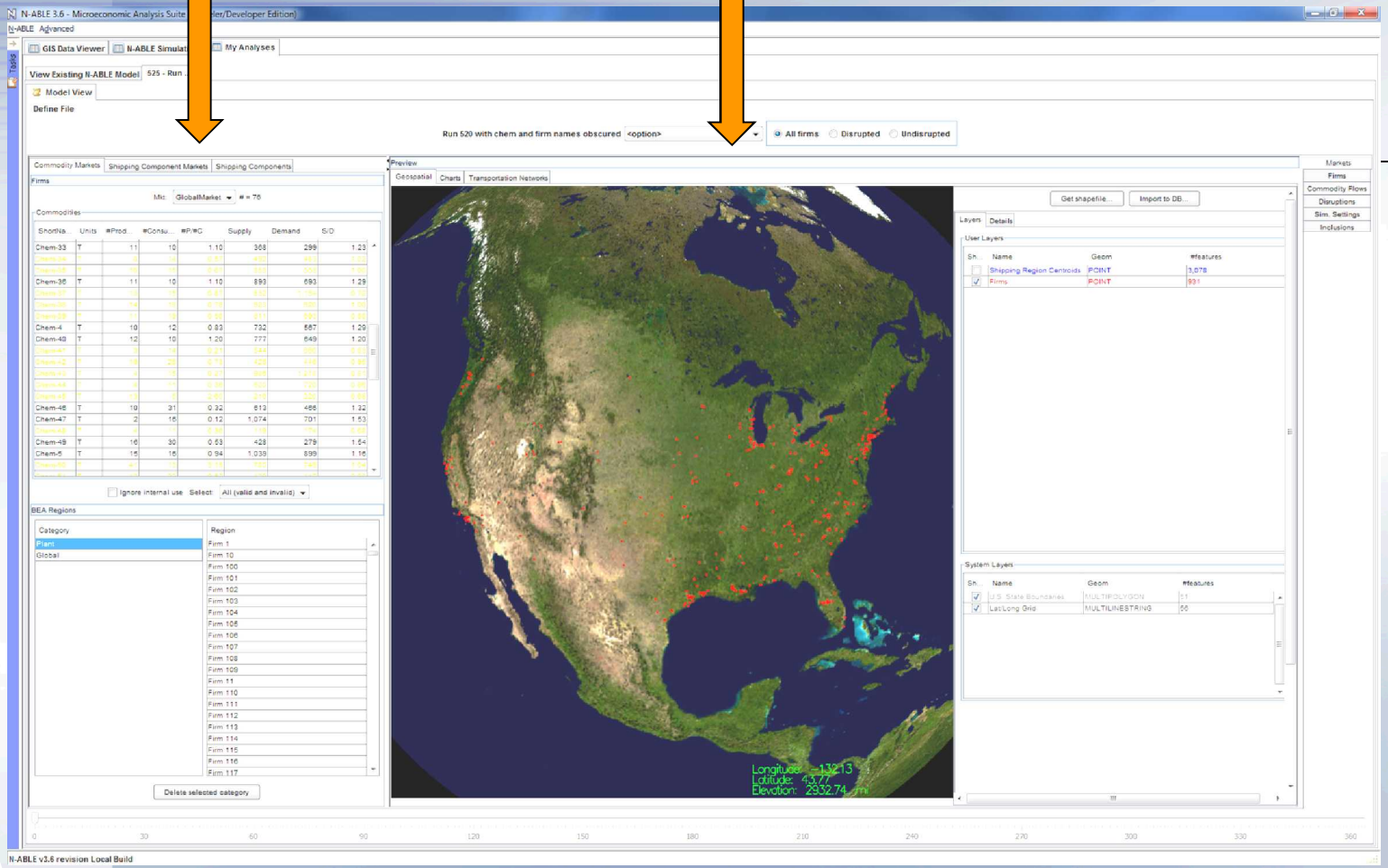


User Interface: Pre-Simulation

Overall Market Statistics

Geospatial Agent View

Markets Tab

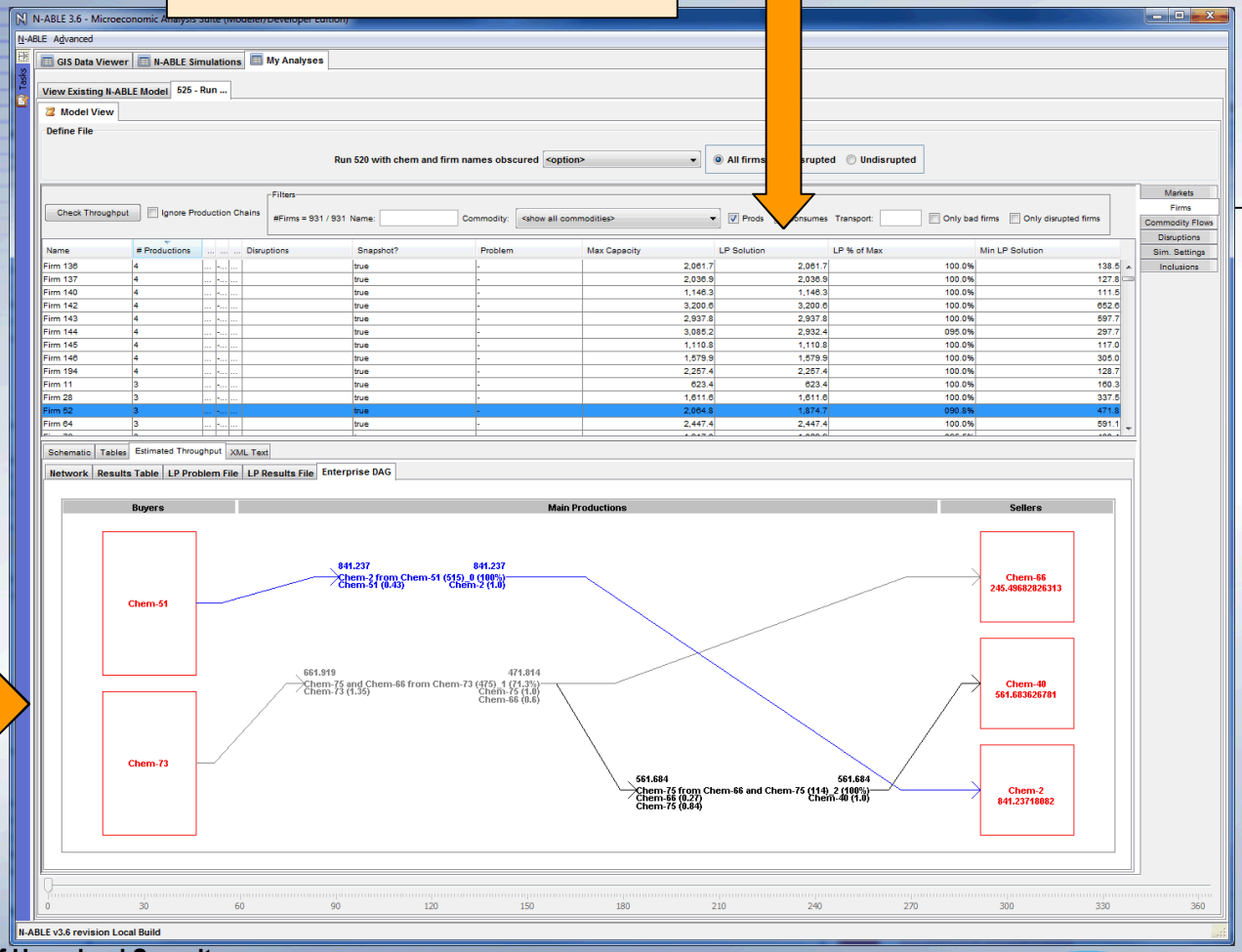


User Interface: Pre-Simulation

Plant Production Process Checks

Firms Tab

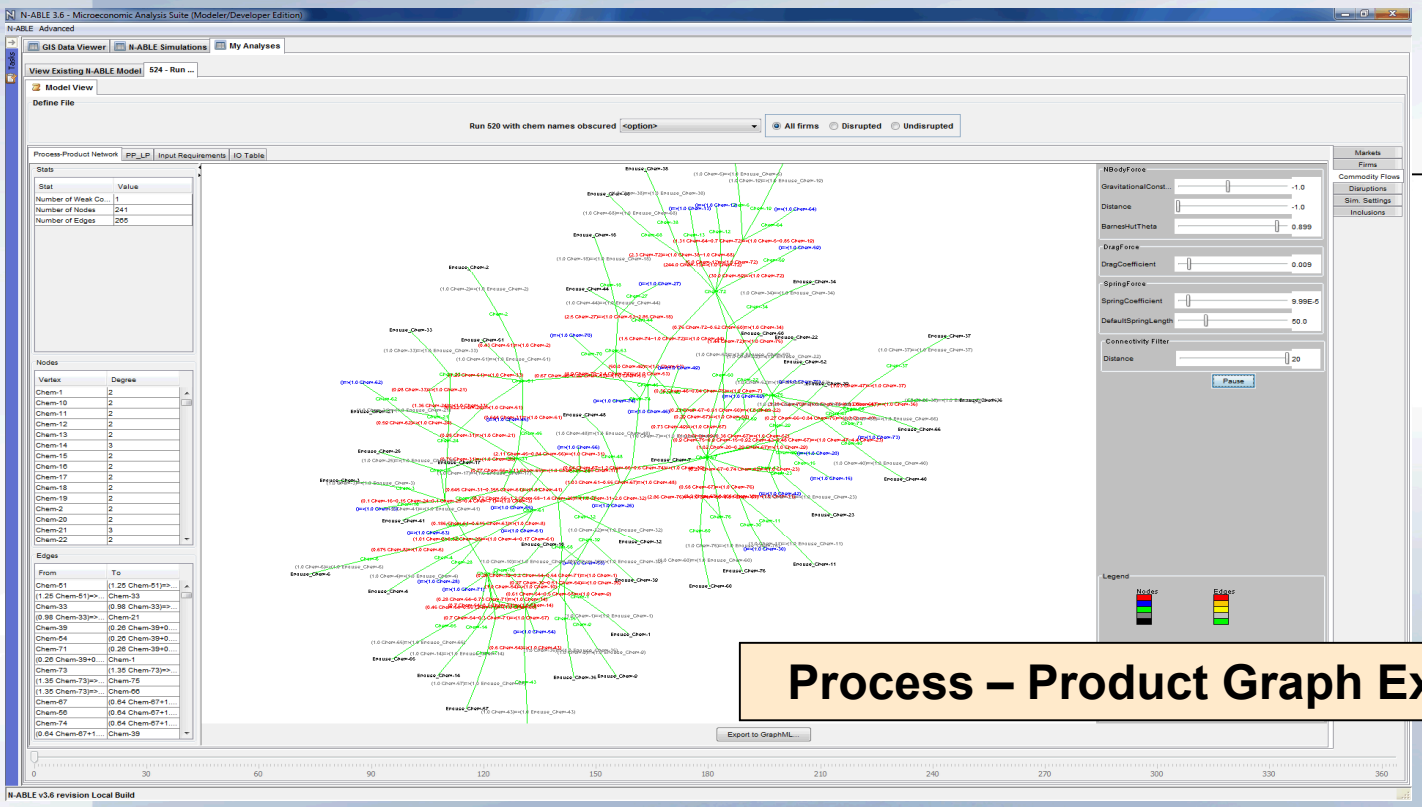
Production Diagrams



User Interface: Pre-Simulation

End-to-end throughput checks

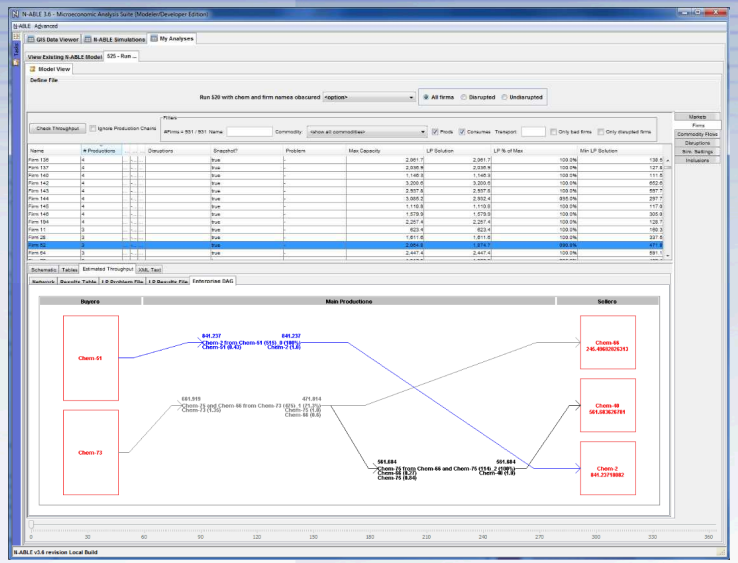
Commodity Flows Tab



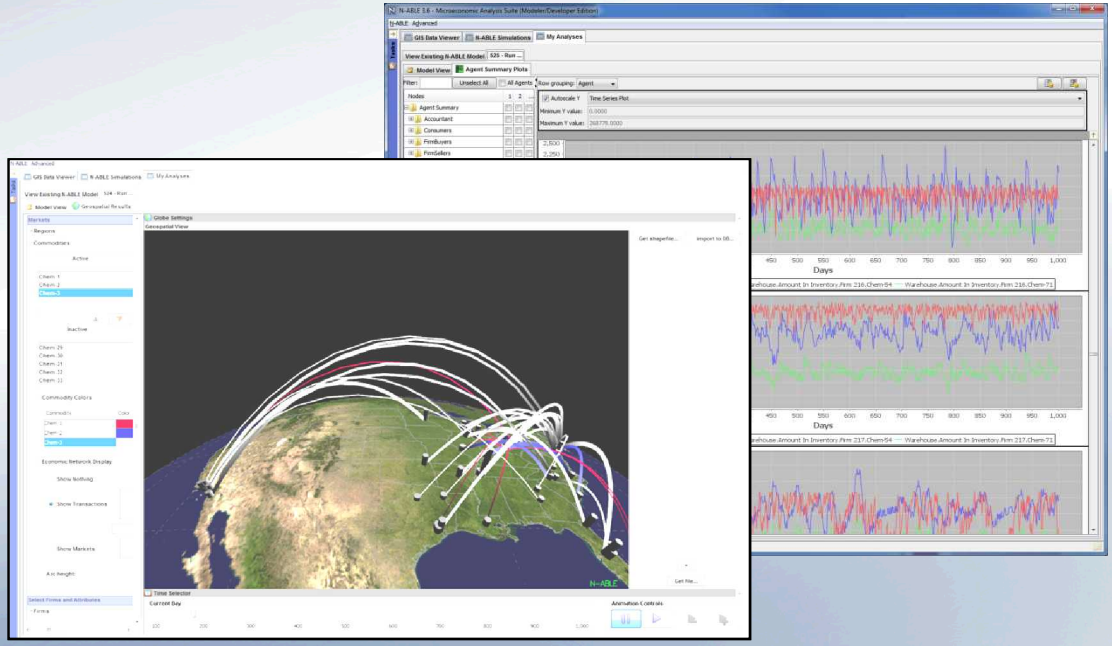
Process – Product Graph Exploration

User Interface

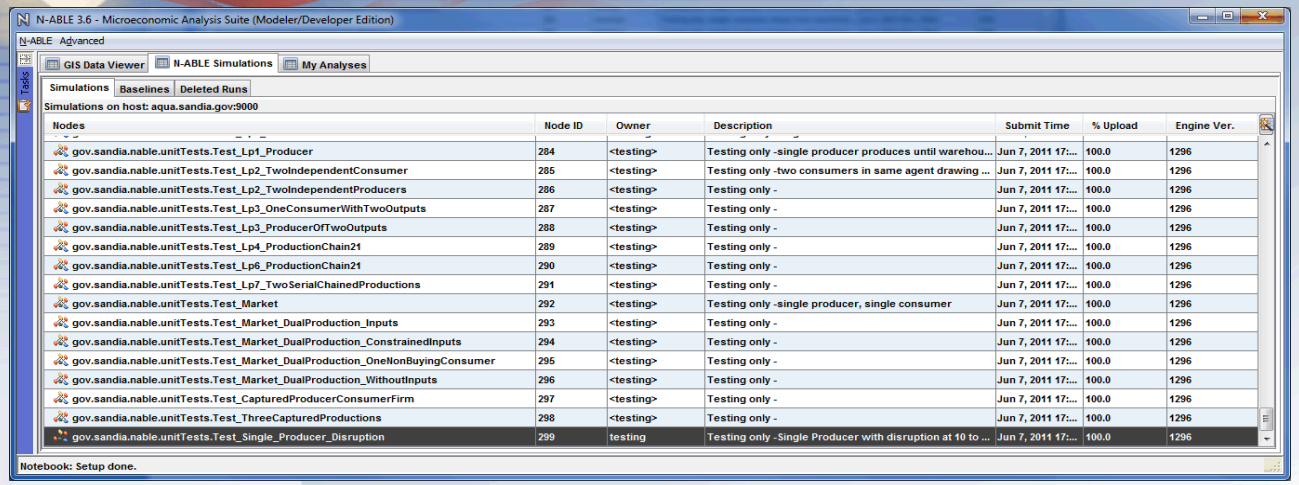
Pre-Simulation Analysis



Post-Simulation Analysis

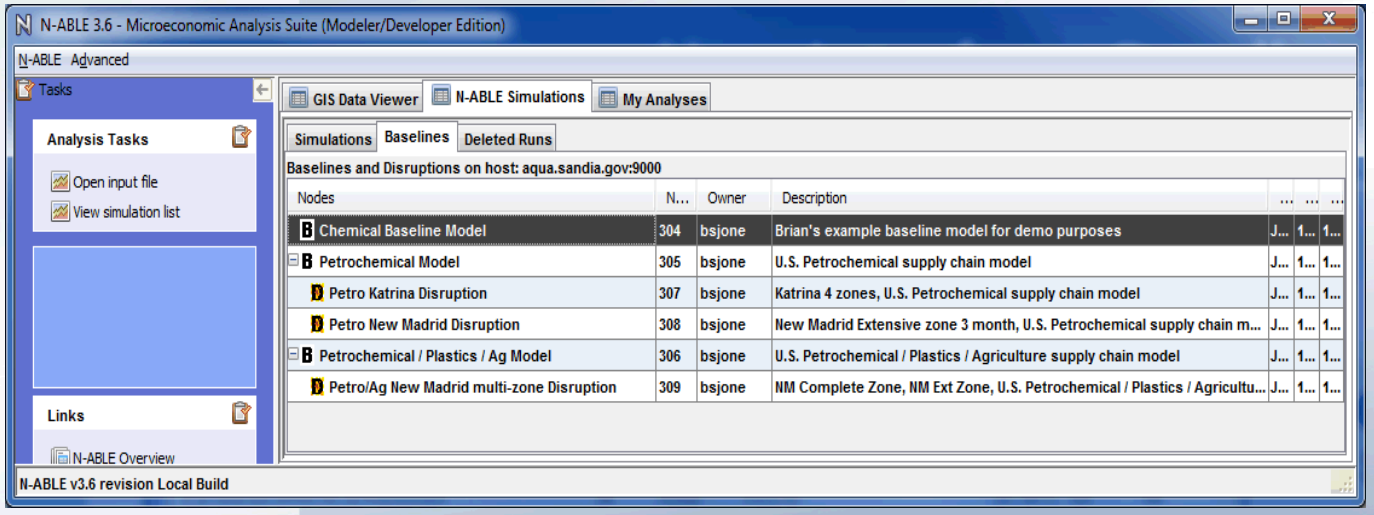


User Interface: Simulation Archive



Centralized Archive of all Simulations in Database

All inputs are versioned and recorded – traceable and repeatable

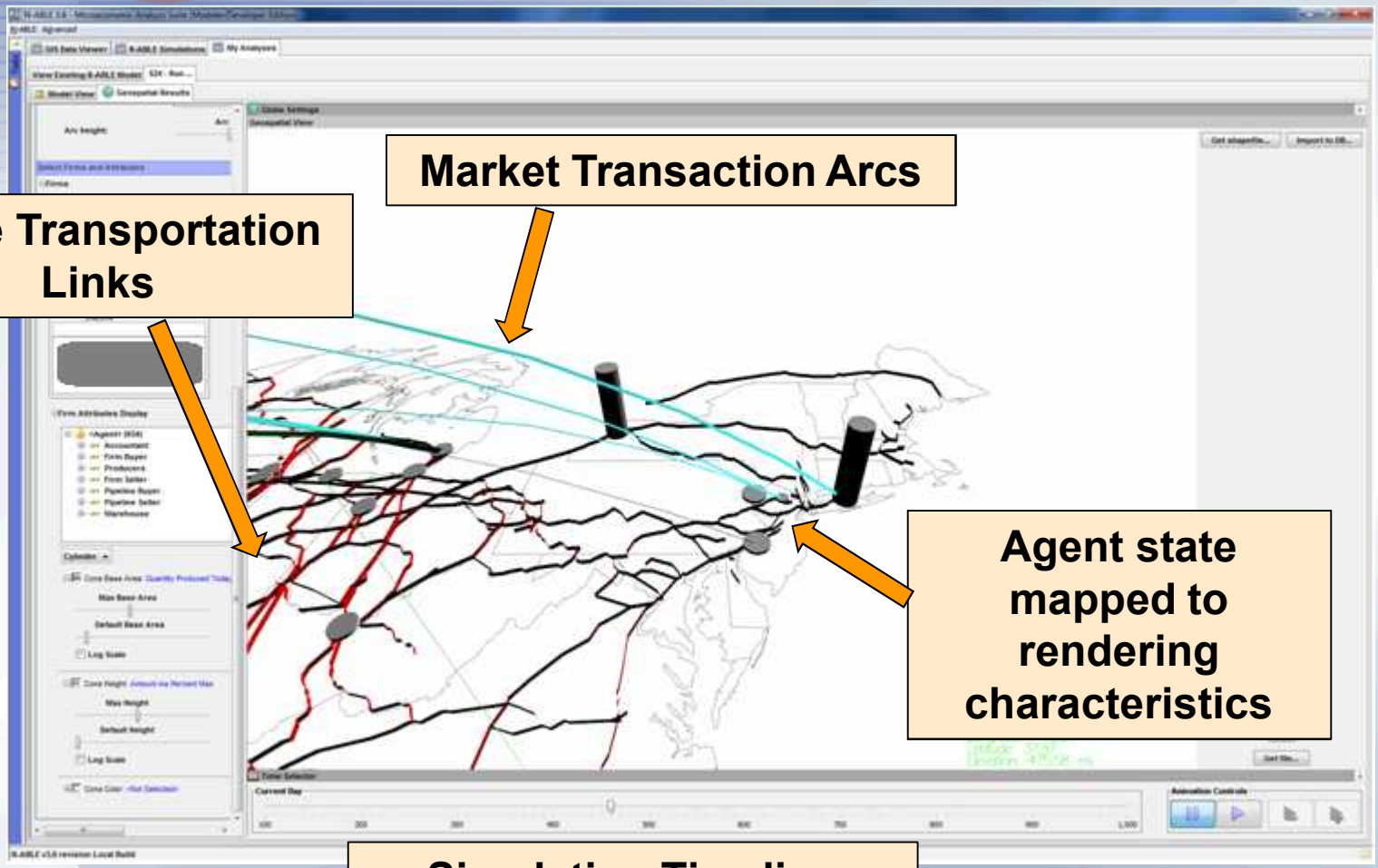


Simulations can be promoted to “Baseline” status

Baseline simulations can have disruptions applied

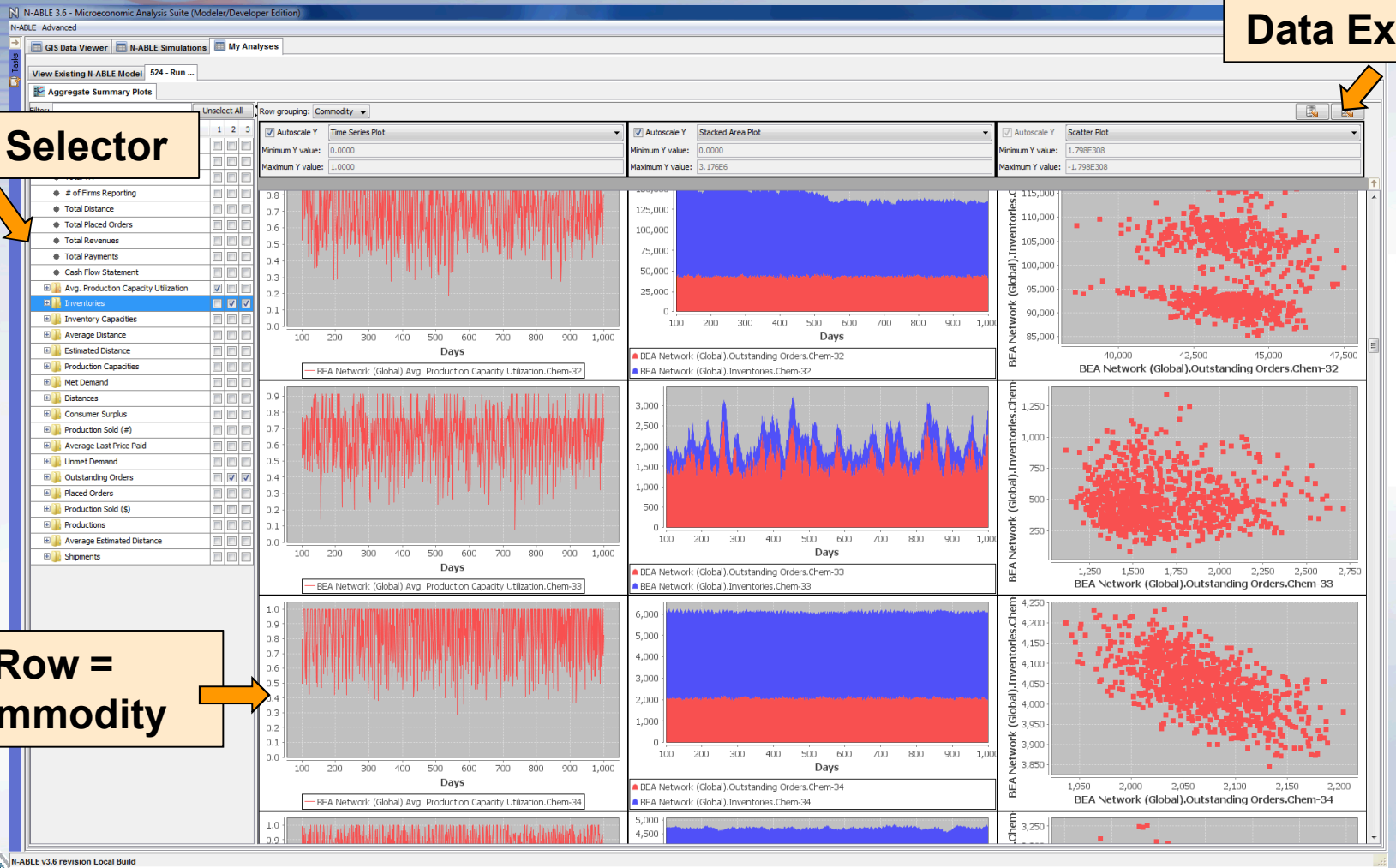
User Interface: Post-Simulation

Geospatial Simulation Animation



User Interface: Post-Simulation

Time Series Visualization - Aggregates



User Interface: Post-Simulation

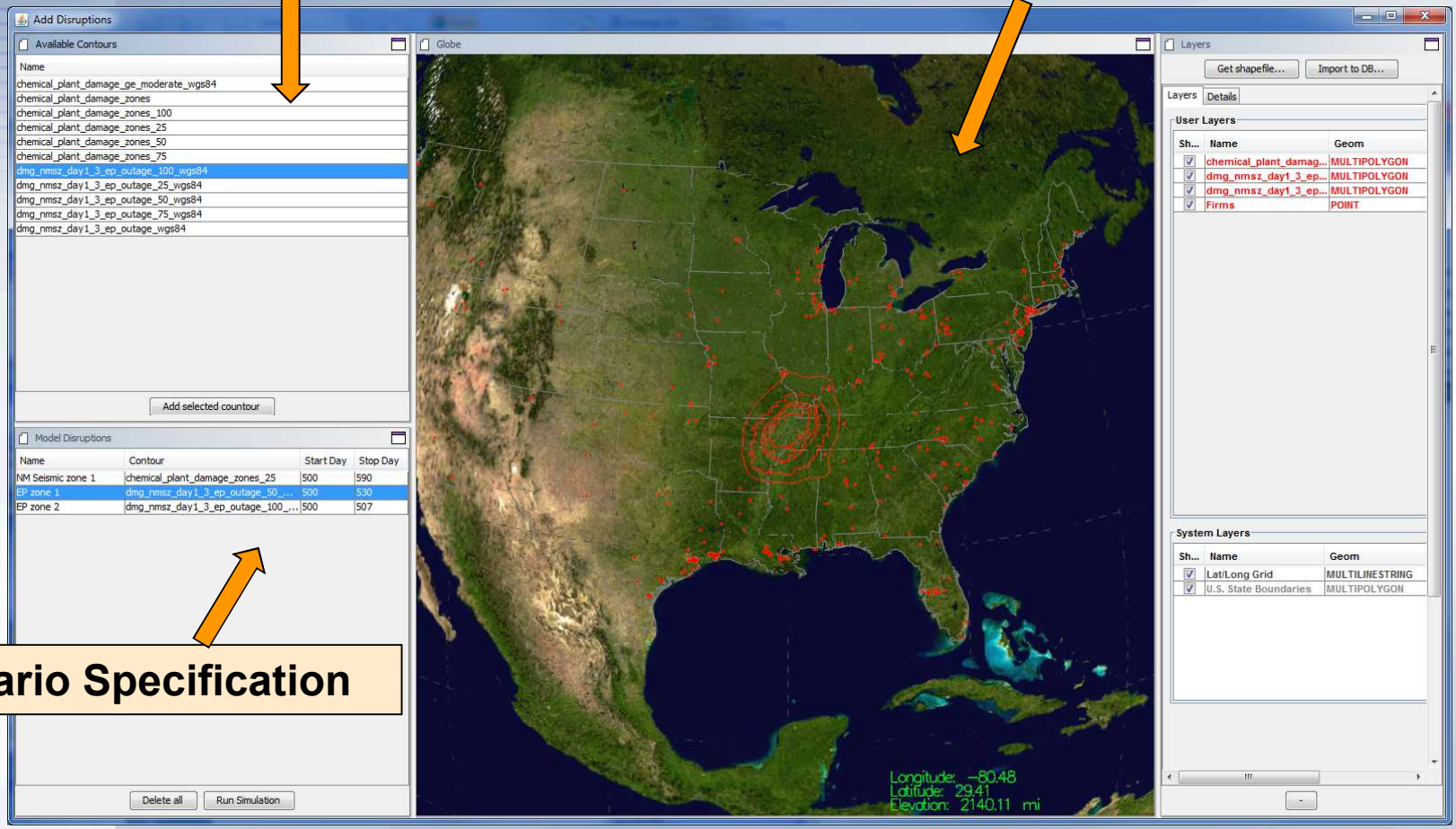
Time Series Visualization – Agent Level

Row = Agent

User Interface: Disruption Specification

Damage Contour Library

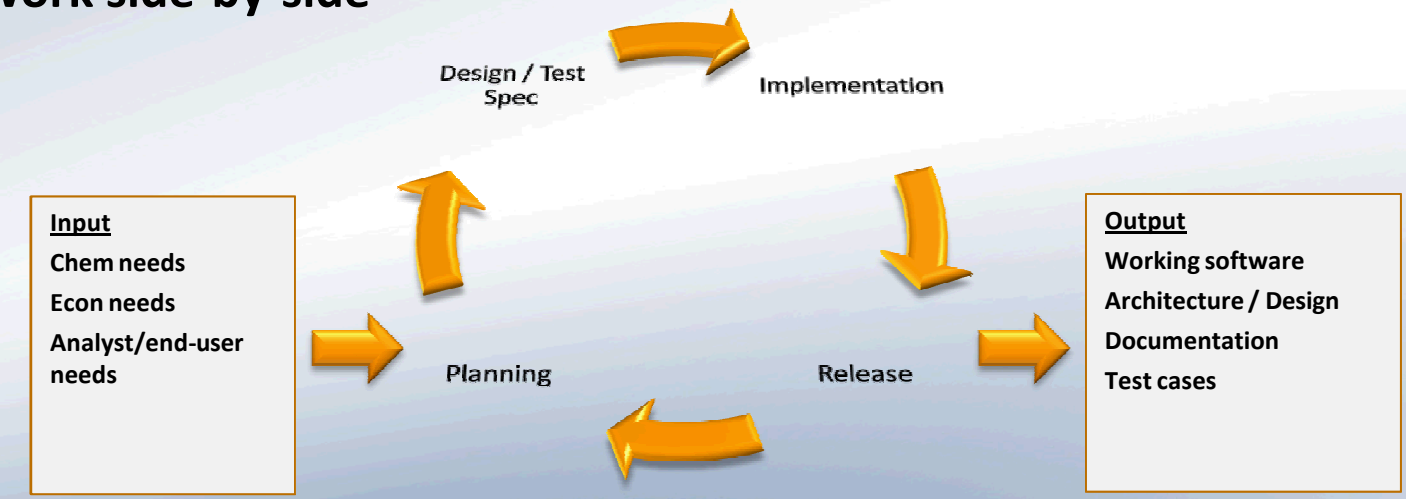
Contour + Agent Viz



Scenario Specification

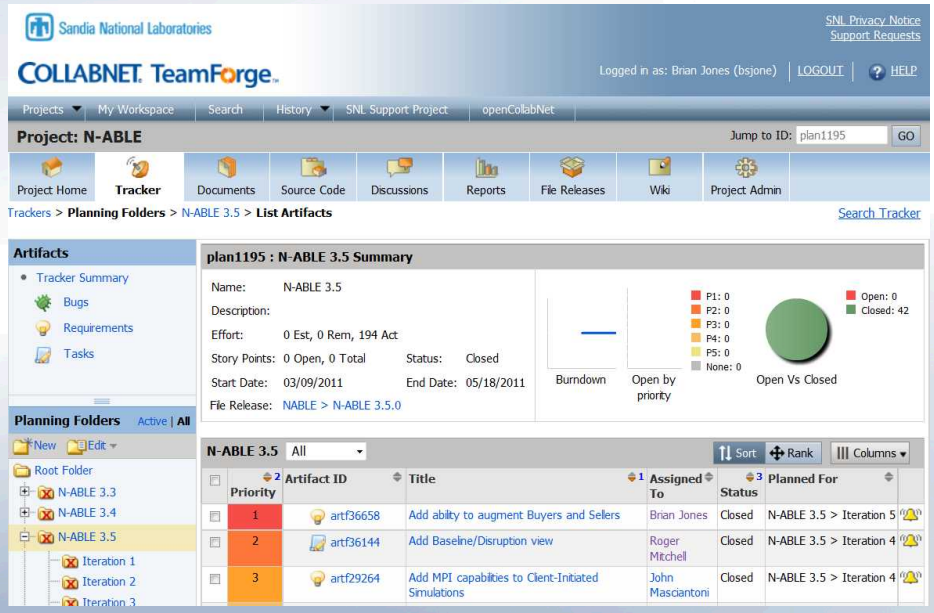
Software Methodology

- Iterative Development ("Agile")
- 2-week iterations to adapt as new ideas emerge
- Computer Scientists and Modelers work side-by-side
- Continuous, Automated Build & Test
- Track Defects, Requirements, Task assignments



Collaboration with TeamForge

- **Requirements / Task / Defect Tracking with Prioritization**
- **Coordination + Traceability**
- **Documents / Message Board**
- **Version Control**



Recap

Flexible, High-Performance, Repeatable Simulation Platform

Computer Scientists work side-by-side with Domain Experts

Methodological Best Practices

Backup Slides

Simulation Engine

- AgentLib
  - Parallel discrete event simulation engine (PDES)
  - General purpose – can be used for chemical supply chains, social networks, pandemic models, etc.
- N-ABLE™
  - Built on top of AgentLib
  - Special purpose – captures economic concepts and behaviors
- Shared characteristics
  - Portable C++ code
  - Tuned for performance
  - Flexible recording system to capture simulation state at each timestep