A Web-based Collaboratory for Supporting Environmental Science Research

Xiaorong Xiang      Yingping Huang      Greg Madey
Department of Computer Science and Engineering
University of Notre Dame

Steve Cabaniss
Department of Chemistry
University of New Mexico

WSS‘03: WI/IAT 2003 Workshop on Applications, Products of Web-based Support Systems
October 13, 2003, Halifax
This research was supported in part by NSF ITR Grant No. 0112820
Introduction

Combination of words “collaboration” and “laboratory” first coined by William Wulf (1996):


- **BioCoRE**: [http://ks.uiuc.edu/Research/biocore](http://ks.uiuc.edu/Research/biocore)


**An example of E-Science**


- R. M. Jakobovits, J. F. Brinkley, C. Rosse, and E. Weinberger (1998): Enabling clinicians, researchers, and educators to build custom Web-based biomedical information system
The NOM Collaboratory

- Interdisciplinary project
  - Supporting research on Natural Organic Matter (NOM)
  - Understanding NOM behavior is an important environmental research area
  - Simulations of NOM in the soil and groundwater

- Objectives
  - Information and model sharing
  - Data repository and analysis support tools
  - Electronic communication tools

- No installation or maintenance of computation resources needed by the environmental scientists
NOM Collaboratory Components

- NOM simulators
- Search engine
- NOML upload
- Molecule editor
- Molecule validation
- Simulation finder
- Automatic restart
- Completion predictor

- Data Analysis
  - Graphical reports
  - XML reports
  - Data mining

- Communication Tools
  - Discussion board
  - Chat room
  - File sharing
NOM Collabatory

NOM Simulator
Provide an intelligent interface to facilitate using the NOM simulator

Search Engine
Provide an interface to search simulation information

Discussion Board
Provide a threaded discussion board

Chat Room
Provide a real time chat box

Molecule Editor
Provide an Interface to define new molecule type

Molecule Validator
Provide an administration role to validate the newly added molecule for public usage

NOML
Provide a XML-based Markup Language definition

File Upload
Provide an Interface to upload publications
Web Technologies

- Sun Java 2 Enterprise Edition (J2EE)
  - Java Servlets, Java Server Pages (JSP)
  - Enterprise Java Beans (EJB)
  - Java Transaction Service/API (JTS/JTA)

- RDBMS (Oracle):
  - Java DataBase Connectivity (JDBC)
  - Data analysis packages
  - Data mining technologies

- XML
  - NOML
  - XSQL
Web-based Interface

- Sign up
- Login
- Submitted simulations
- Simulation reports
- Dynamic running time prediction
- Terminate
- Find similar simulations
- Restarter
- Parameter input
- Start a new one
- Invoke simulation
- Static running time prediction
- Email notification

Database

Simulation engines
Web-based Interface Logic

- Input the simulation parameters
- Invoke the simulation
- Stop the simulation
- View the real-time simulation results
Web Interface Implementation
Example of Interface

Welcome to NOM Research Group

<table>
<thead>
<tr>
<th>Molecule Name</th>
<th>Make it available to public</th>
</tr>
</thead>
<tbody>
<tr>
<td>3位CFA</td>
<td></td>
</tr>
</tbody>
</table>

Atoms of the molecule:
- Each field must be a non-negative integer.
- Default value is 0.

<table>
<thead>
<tr>
<th>Atom</th>
<th>C</th>
<th>H</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>66</td>
<td>64</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Atom</th>
<th>O</th>
<th>S</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>38</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Functional groups of the molecule:
- Each field must be a non-negative integer. Default value is 0.

<table>
<thead>
<tr>
<th>Functional Group</th>
<th>DoubleBond</th>
<th>Rings</th>
<th>Phenyli</th>
<th>Alcohols</th>
<th>Phenols</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ethers</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Aryllics</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Thiols</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Phosphoesters</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Create New Molecule Type
Interface Features

- Email notification
- Running time prediction
  - Static
    - number of molecules
    - number of time steps
  - Dynamic
    - current time step
    - current wall clock time
Interface Features (cont)

- **Find similar simulations**
  - Environment parameters
  - Molecule types and distributions
  - Retrieve the data sets from database
  - Points on a high dimension space
  - Euclidean distance
  - Ordered list
  - Review the simulation results or restart

- **Automatic restarter**
  - Save the state of each objects in the system to database every check point
  - Load the state to the core simulation engine
XML-based NOM Markup Language

- **NOML:**
  - Standard data format
  - Environment.dtd, Molecules.dtd, Setup.dtd
  - Environment.xml, Molecules.xml, Setup.xml
- Facilitates communication
  - User ==== User
  - Application ==== Application
  - User ==== Application
- Extensions planned
NOML Uploader

Client Side

Server Side

upload

NOML
file format
(.xml)

Display
ID

Servlet

Multipart
Parser

XML
FileInput
Stream

Call
JavaBean

Parse
Molecule

Parse
Environment

Parse
Setup

JavaBean

Database
Data Input Options

- SCM format
- XML format
- Web interface

Input

Database

Core Simulation Engine
Other Tools

- Molecule editor
  - Define new molecule type
- Molecule validator
  - Authorized persons (Chemists) to validate data
  - Share the molecule type
- Search engine
  - Ad-hoc query
  - View results of the completed simulations
  - Restart some simulations
Architecture
NOM Simulation Engine

- Design
  - Agent-based
  - Stochastic
  - Discrete event

- Packages
  - Swarm
  - RePast

- Languages
  - Java
  - SQL & PL/SQL

Input

Databases

Simulation Engine

Output

XML

Forms
NOM Simulation Engine

- Read simulation parameter from the database (JDBC)
  - Environmental parameters (pH, temperature, light intensity, and so on)
  - Molecule types and distributions
- User defined time has been separated to a large number of equal size time steps
- Write relevant data into the database every time step (JDBC)
  - Trace the dynamic properties of individuals and the system over time
Data Analysis

- Analysis
  - SQL
  - Statistics
  - Data mining

- Presentation
  - Oracle Reports
  - XML/XSLT
  - XSQL
Report Example

Please provide the following information.
Current Time Step is: 1969  Refresh

- Time Step: 1969
- Status: adsorbed
- Start Y: 0
- End Y: 299

Generate Reports
Generate XML file

Weight Distribution

<table>
<thead>
<tr>
<th>Count</th>
<th>361</th>
<th>695</th>
<th>1390</th>
<th>2780</th>
<th>5560</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Conclusion

- Web-based Collaboratory
- A Web-based simulation architecture (multiple simulation servers, database servers, and data mining technologies)
- A Web-based configuration interface
- NOML data upload
Future Work

- Collaboratory
  - More communication tools
    - Audio
    - Video
- More simulation models for NOM study
- NOML extensions
- JDBC performance
  - AutoCommit issues
  - Batch inserts
  - Use of sqlloader
- User testing
Thank You!

Questions?