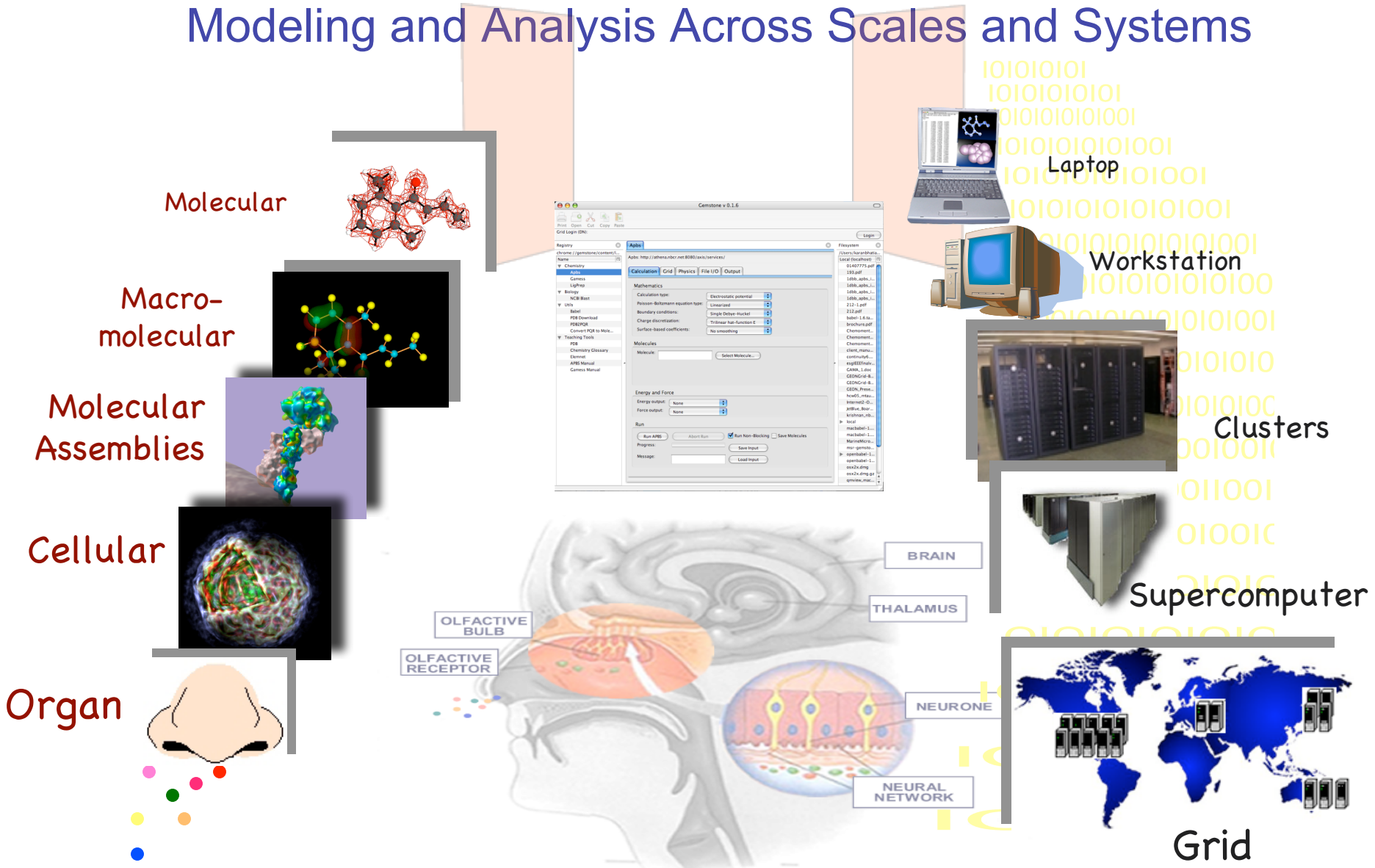
An End-to-End Web Services-based Infrastructure for Biomedical Applications

Sriram Krishnan*, Kim K. Baldridge, Jerry P. Greenberg,
Brent Stearn and Karan Bhatia

*sriram@sdsc.edu

Modeling and Analysis Across Scales and Systems



Goals

- Enabling integration across multi-scale biomedical applications
- Leveraging geographically distributed, disparate computational and data resources

Functional Requirements

- Making biomedical applications *Grid-aware*
  - Remote execution on Grid resources
    - Use of Grid-based schedulers
  - Support for multiple concurrent users
  - Access via disparate user interfaces
  - Use of standards-based security mechanisms
- Integration across multi-scale applications via the use of *Workflow* tools

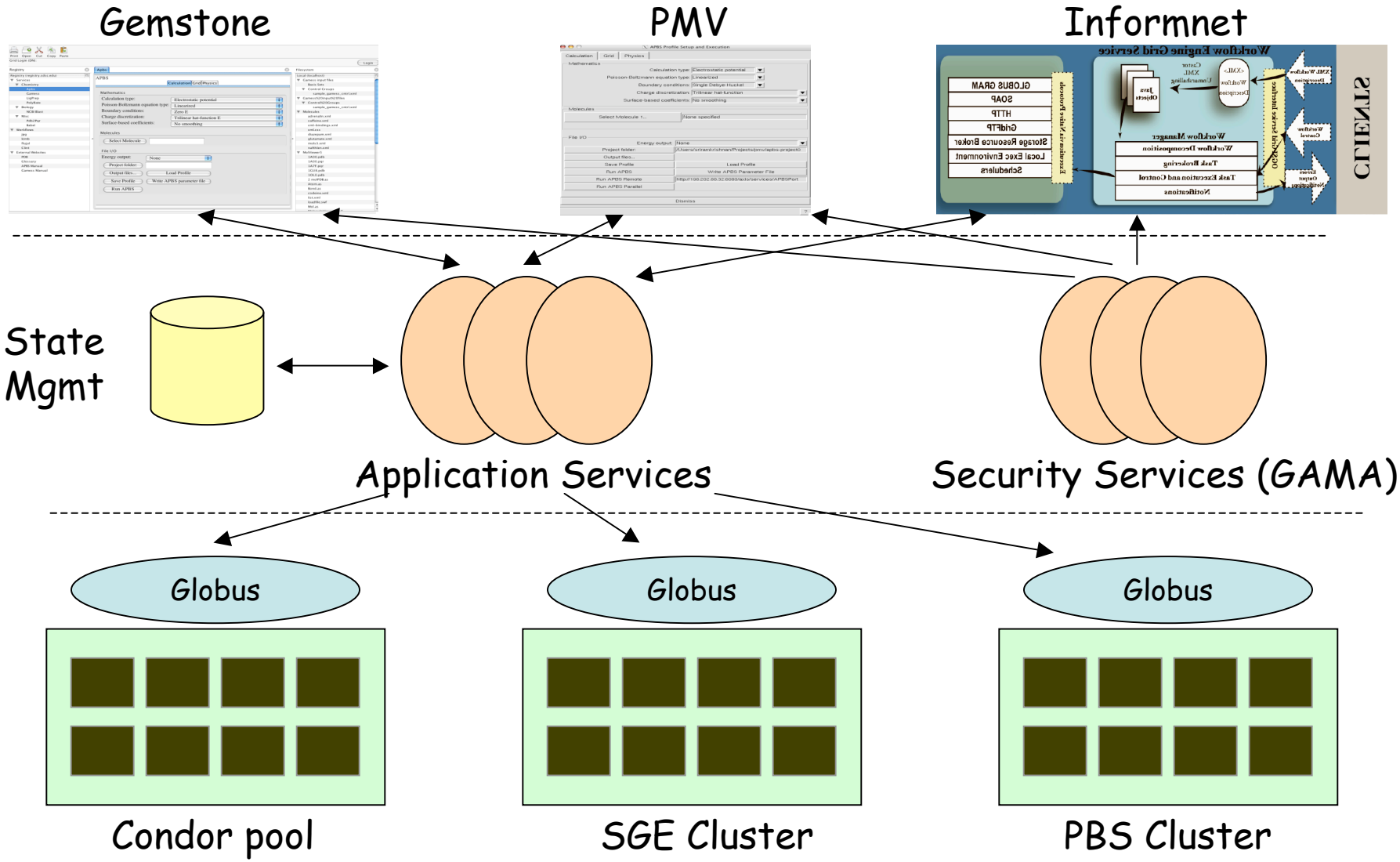
Towards a Services Oriented Architecture

- Applications are wrapped as services
  - Provide transparent execution on Grid resources
  - Users are free to use clients of their choice
  - Multiple standards-based security alternatives to choose from
- Services exchange strongly typed data defined using XML schemas
  - Aids in the creation of complex workflows

Talk Outline

- Motivation for a Services Oriented Architecture
- Overall end-to-end architecture
- Technical Details and Challenges
- Sample User Interfaces
- Status and Evaluation
- Conclusions

Architecture Overview



Technical Details and Challenges

- Application Services
  - Data Typing and Operations
- State Management
- Scheduling
- Security

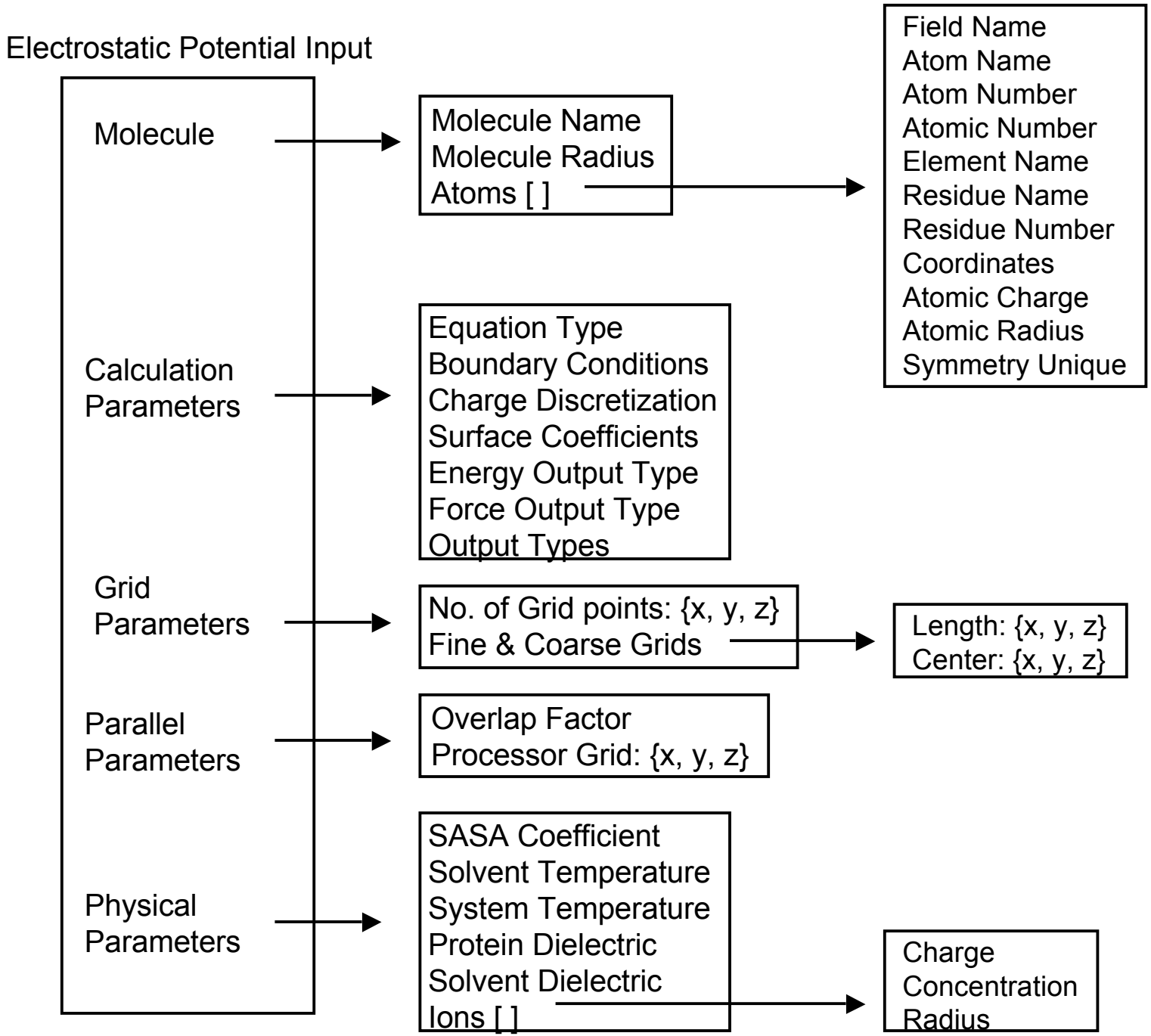
Data Typing Issues

```
ATOM      1  CA  ALA   403     302.952 172.086  61.234  0.070 2.275
ATOM      2  CA  PHE   404     300.425 169.247  61.082  0.070 2.275
ATOM      3  CA  VAL   405     303.060 166.741  60.099  0.070 2.275
ATOM      4  CA  HSD   406     302.667 164.496  63.103  0.070 2.275
ATOM      5  CA  TRP   407     299.201 163.824  61.775  0.070 2.275
ATOM      6  CA  VAL   409     303.586 160.385  60.021  0.070 2.275
ATOM      7  CA  GLY   410     301.476 158.708  62.646 -0.020 2.275
ATOM      8  CA  GLU     3     280.442 157.436  68.520  0.070 2.275
ATOM      9  CA  ILE     4     281.745 159.036  71.747  0.070 2.275
ATOM     10  CA  VAL     5     281.352 158.451  75.502  0.070 2.275
ATOM     11  CA  HSD     6     281.014 161.480  77.762  0.070 2.275
ATOM     12  CA  ILE     7     282.779 161.559  81.125  0.070 2.275
ATOM     13  CA  GLN     8     281.483 163.609  84.125  0.070 2.275
ATOM     14  CA  ALA     9     284.564 163.755  86.377  0.070 2.275
ATOM     15  CA  GLY    10     284.153 165.418  89.764 -0.020 2.275
ATOM     16  CA  GLN    11     281.233 167.582  90.892  0.070 2.275
ATOM     17  CA  CYS    12     281.680 170.431  88.465  0.070 2.275
ATOM     18  CA  GLY    13     282.299 168.195  85.512 -0.020 2.275
ATOM     19  CA  ASN    14     279.242 166.471  86.869  0.070 2.275
ATOM     20  CA  GLN    15     277.035 169.516  87.517  0.070 2.275
ATOM     21  CA  ILE    16     277.798 170.436  83.911  0.070 2.275
ATOM     22  CA  GLY    17     276.950 166.924  82.798 -0.020 2.275
ATOM     23  CA  ALA    18     273.419 167.698  83.872  0.070 2.275
ATOM     24  CA  LYS    19     273.486 170.809  81.708  0.070 2.275
ATOM     25  CA  PHE    20     275.293 169.251  78.771  0.070 2.275
ATOM     26  CA  TRP    21     272.734 166.498  78.760  0.070 2.275
ATOM     27  CA  TYR    52     275.692 159.625  72.557  0.070 2.275
ATOM     28  CA  PRO    63     273.156 158.690  77.534  0.020 2.275
ATOM     29  CA  ARG    64     276.471 156.904  77.238  0.070 2.275
ATOM     30  CA  ALA    65     278.499 158.910  79.729  0.070 2.275
ATOM     31  CA  ILE    66     280.286 158.199  82.986  0.070 2.275
```

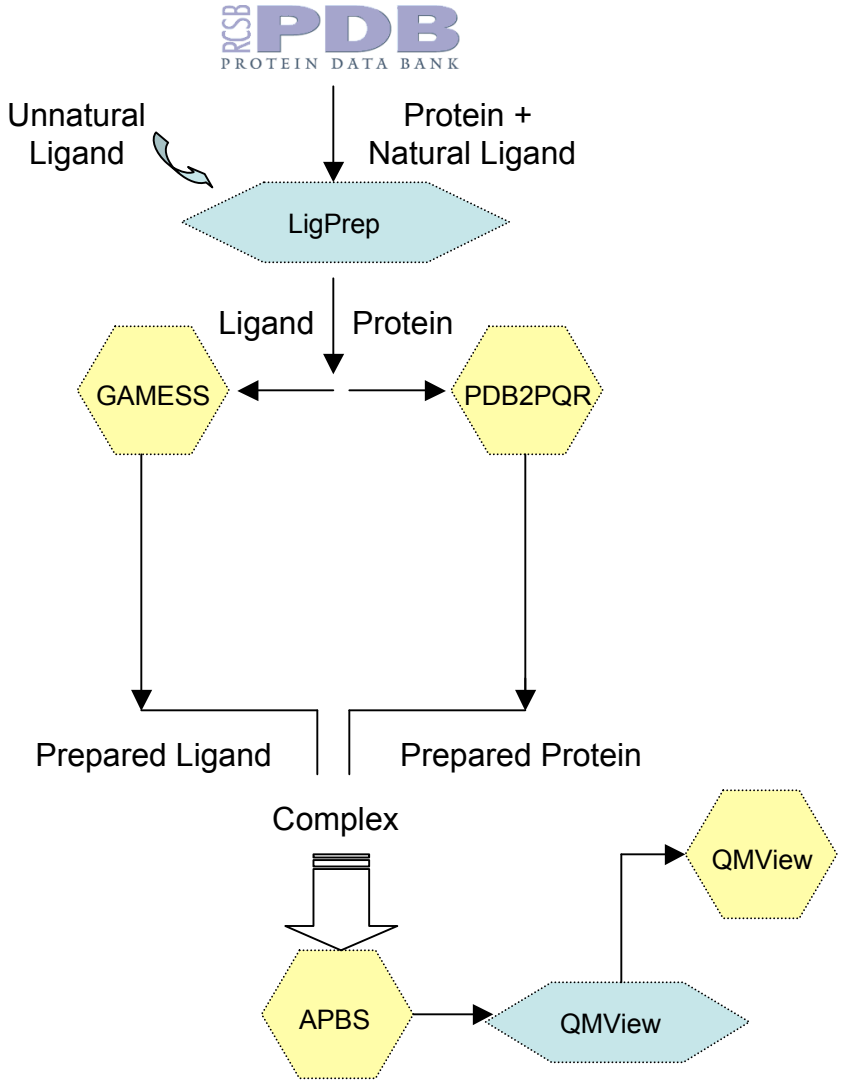
- Lack of common strongly defined data types
  - All applications use custom file-based formats
- Application integration via third party tools extremely difficult to perform
  - Need application specific transformation scripts

Application Services

- Requests and responses are strongly typed
  - Use of XML Schemas to define data structures passed around
- Application functionality exposed as science oriented WSDL operations
  - Available services: APBS, GAMESS, QMView, LigPrep
- Implementation details
  - Services *wrap* scientific codes - no (or minimal) modification required to these codes
  - Software tools used - Apache Axis, Jakarta Tomcat


Electrostatic Potential Input
Molecule
Calculation Parameters
Grid Parameters
Parallel Parameters
Physical Parameters
Molecule Name
Molecule Radius
Atoms [ ]
Field Name
Atom Name
Atom Number
Atomic Number
Element Name
Residue Name
Residue Number
Coordinates
Atomic Charge
Atomic Radius
Symmetry Unique
Equation Type
Boundary Conditions
Charge Discretization
Surface Coefficients
Energy Output Type
Force Output Type
Output Types
No. of Grid points: {x, y, z}
Fine & Coarse Grids
Length: {x, y, z}
Center: {x, y, z}
Overlap Factor
Processor Grid: {x, y, z}
SASA Coefficient
Solvent Temperature
System Temperature
Protein Dielectric
Solvent Dielectric
Ions [ ]
Charge
Concentration
Radius

Workflows and Strong Data Typing



Ligand-Protein Interaction

- Baldridge, Greenberg, Amoreira, Kondric
- GAMESS Service
  - More accurate Ligand Information via GAMESS-XML
- LigPrep Service
  - Generation of Conformational Spaces
- PDB2PQR Service
  - Protein preparation
- APBS Service
  - Generation of electrostatic information
- QMView Service
  - Visualization of electrostatic potential file
- Applications:
  - Electrostatics and docking
  - High-throughput processing of ligand-protein interaction studies
  - Use of small molecules (ligands) to turn on or off a protein function

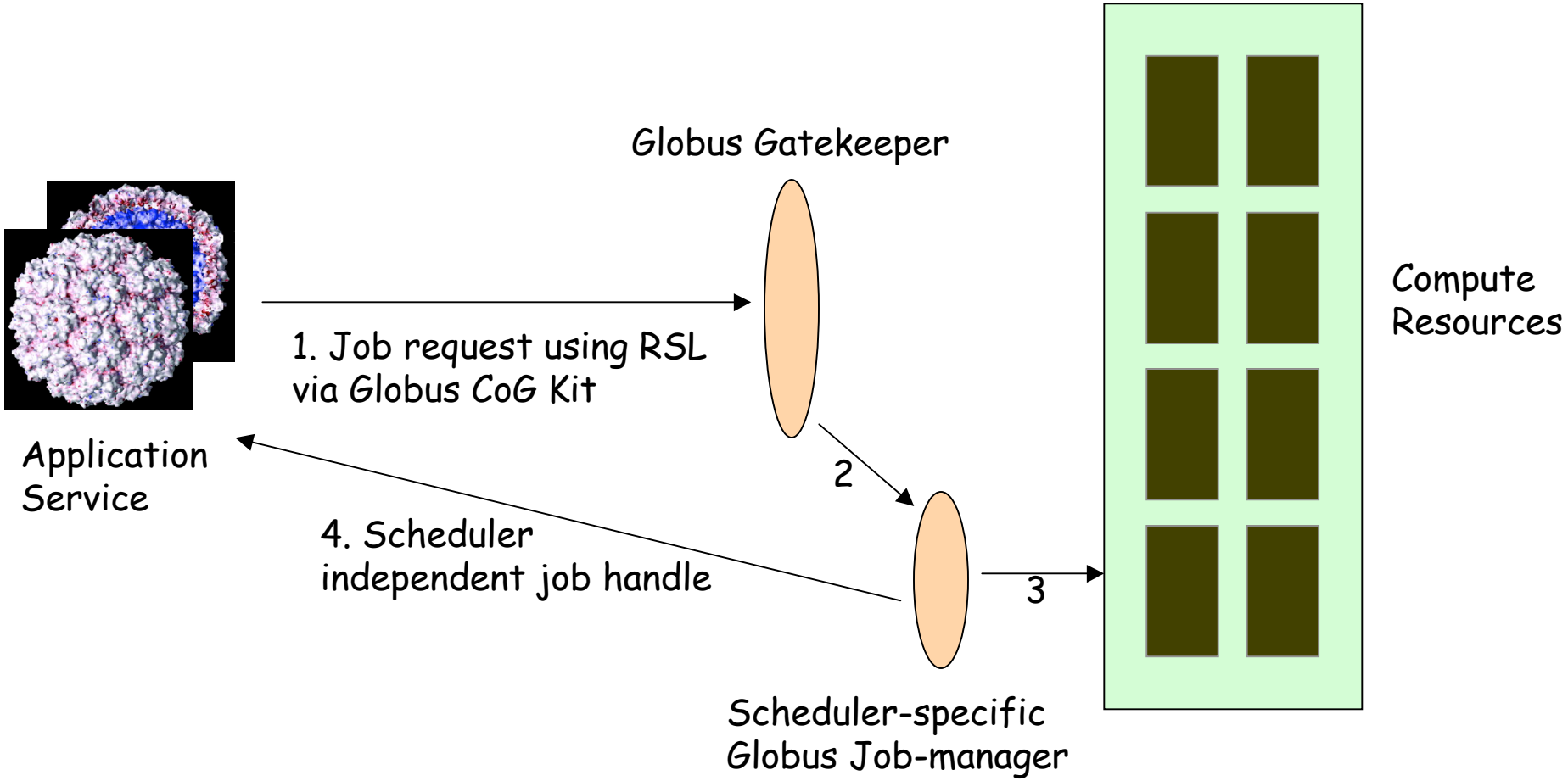
Service Operations

- Operations can be invoked synchronously, or asynchronously
- Synchronous Operations:
  - Block until the operation is finished
  - Outputs returned as a response to initial request
  - Suitable for short jobs
- Asynchronous operations:
  - Return immediately with a jobID
  - Can query for job status and outputs using the jobID
  - Suitable for long running jobs

State Management

- Application services are stateful
  - Metadata about job inputs and outputs
  - Job status for asynchronous jobs
  - Job history
- Use of a database for storing/retrieving service state
  - Access to PostgreSQL database via JDBC
- Future Work:
  - Web Service Resource Framework (WSRF) integration

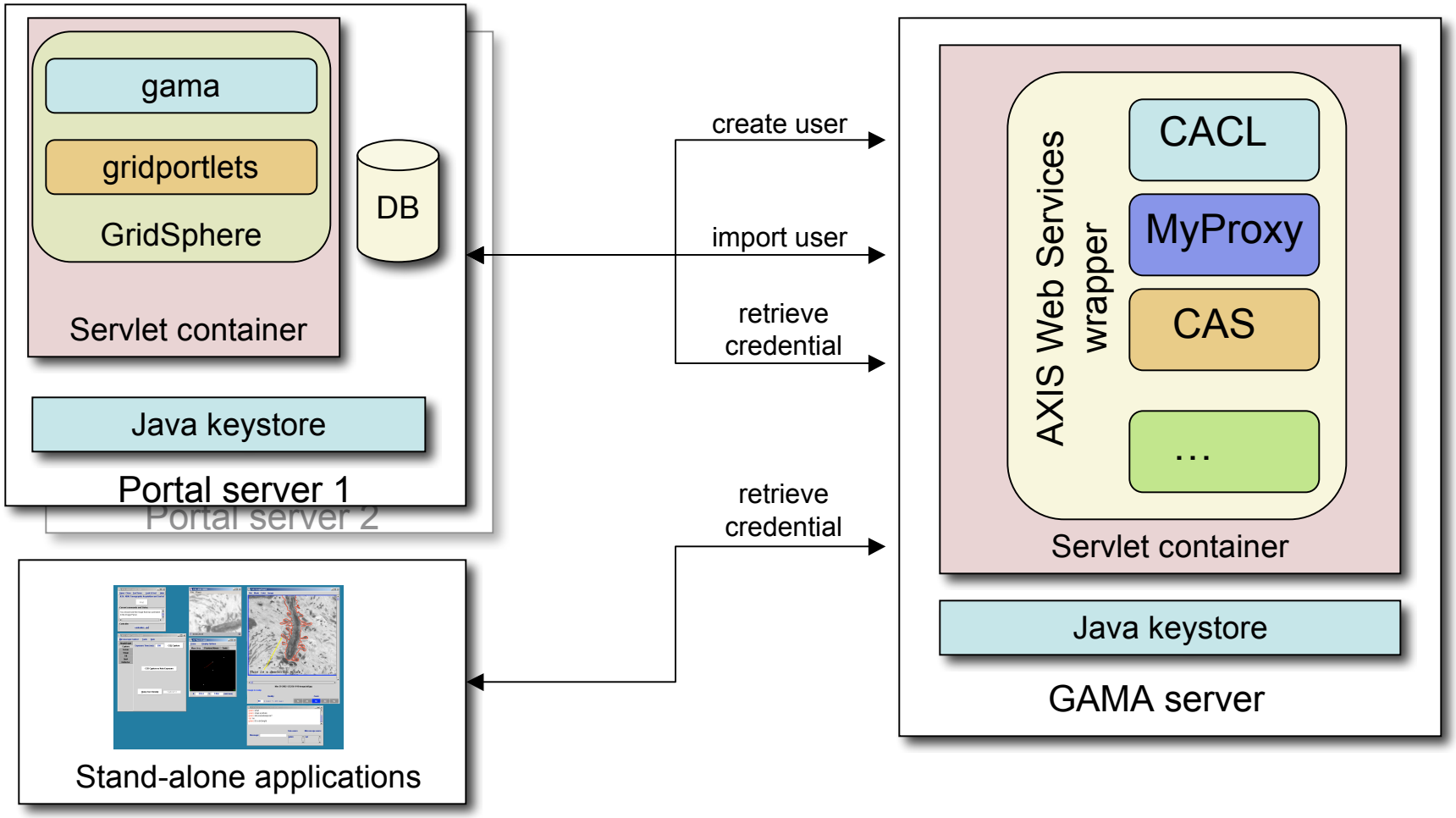
Scheduling


Globus Gatekeeper
Compute Resources
1. Job request using RSL via Globus CoG Kit
Application Service
2
4. Scheduler independent job handle
3
Scheduler-specific Globus Job-manager

Security

- GSI-based transport level (SSL) authentication
  - Use of Java CoG libraries and Tomcat to provide a secure socket connection
- Simple *grid-map* based authorization provided as an Axis Handler
  - Every Axis request passes through a chain of handlers before the target service is invoked
  - The grid-map Authorization Handler verifies if the client is authorized to access the service by looking up the grid-map using the Client's Distinguished Name (DN).
- Future Work:
  - SAML-based authorization techniques

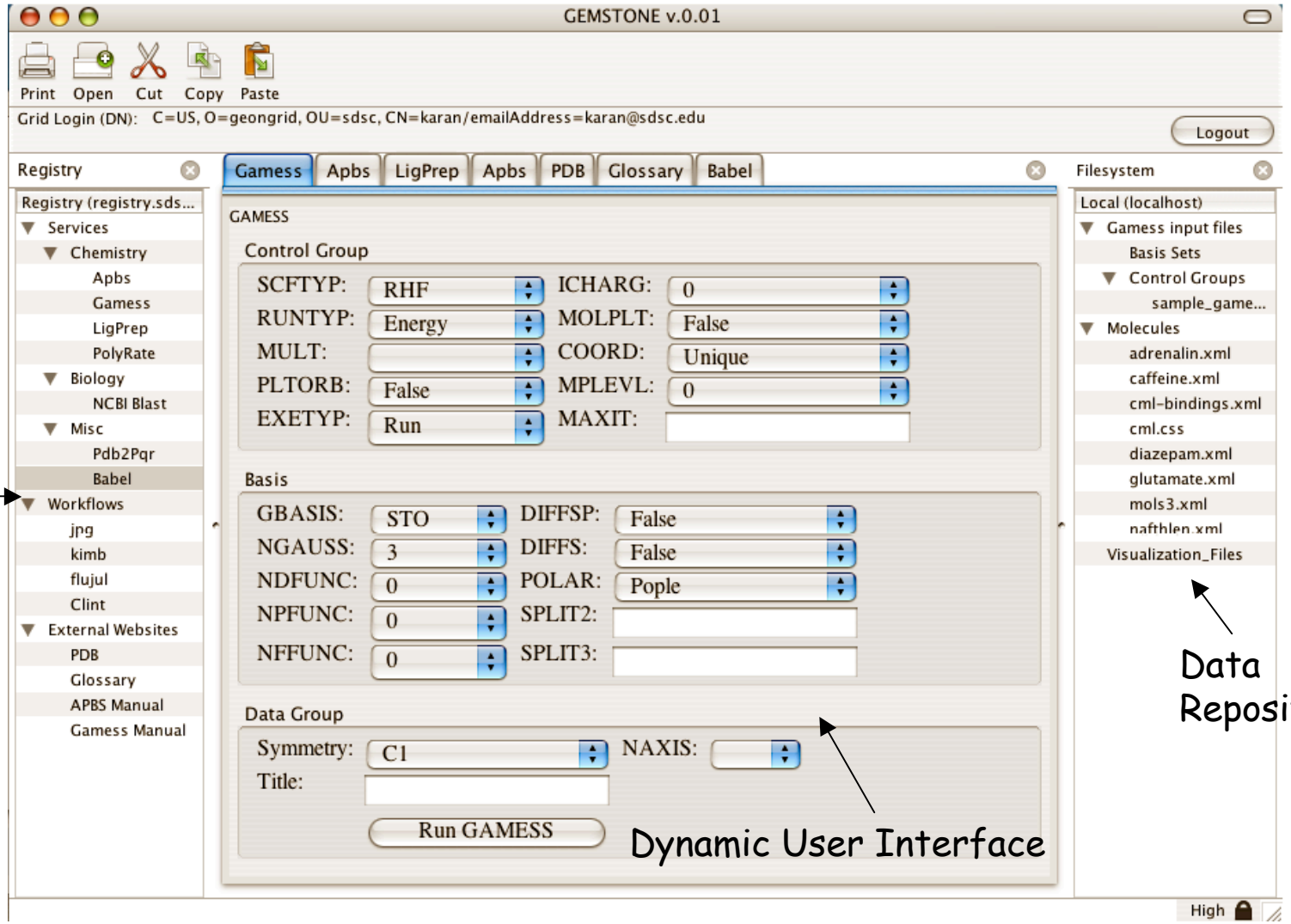
Certificate Management: GAMA



User Interfaces

- Web services are language and platform independent
  - Can be accessed via a multitude of clients
- Java
  - Gridsphere-based Web portals
  - Workflow tools: Kepler, Informnet, etc.
- Python
  - Python Molecular Viewer (PMV)
  - Workflow tools: Vision
- JavaScript
  - Gemstone: Mozilla-based Web services front-end

Gemstone



Registry of Services

Data Repository

Dynamic User Interface

Initial Evaluation

- Commodity SOAP toolkits not the most ideal for transferring large inputs and outputs
  - XML representation of molecule data (originally in PQR format) approximately an order of magnitude larger
  - Larger transfer times
- Axis de-serialization very expensive for large inputs
  - Large memory footprint
  - Very time consuming
- However, Web service overhead is still at least an order of magnitude smaller than actual execution times

SOAP Performance: Alternatives

- Parsing techniques
  - Streaming
  - Pull-based
- Binary XML
  - More compact representation of data
  - More efficient data transport and parsing
  - Smaller memory footprint
- Data Format Description Language (DFDL)
  - Definition of structure of binary and character files
  - Files transferred in their native formats
    - Smaller sizes, hence faster transfer

Summary

- An end-to-end infrastructure for Grid-enabling biomedical applications that provides:
  - Remote execution on Grid resources
    - Access to schedulers
  - State management
  - Concurrent access via disparate interfaces
  - Standards-based security
- Ability to use workflow tools for coupling multi-scale biomedical applications

Status, Software and Demos

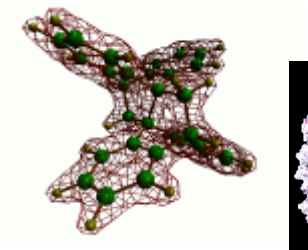
- Application services: http://nbcr.net/services
  - Alpha version of APBS service available for download and testing
  - Opal, a toolkit for wrapping legacy scientific applications available for alpha testing
  - GAMESS, QMView, LigPrep services available soon
- Gemstone: http://grid-devel.sdsc.edu/gemstone
- GAMA: http://grid-devel.sdsc.edu/gama
- Demos at SC2005
  - NCRR Booth (656): 11/14 7-8PM,
    11/15 12-2PM, 11/17 12-2PM
  - SDSC Booth (1838): 11/15 5-6PM

Appendix

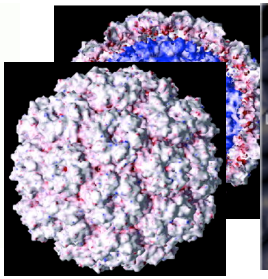
Computational Infrastructure for Multiscale Modeling

Set of Biomedical Applications

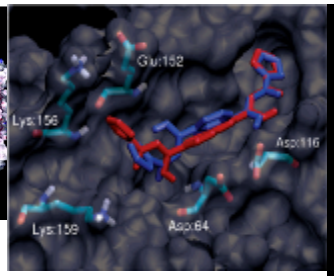
Infrastructure



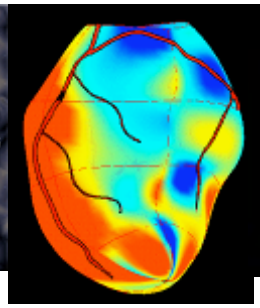
QMView
GAMESS



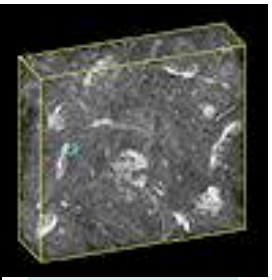
APBS



Autodock



Continuity



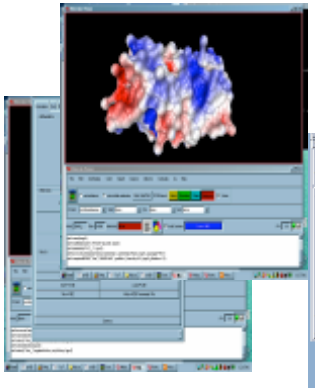
Gtomo2
TxBR

Computational Grid

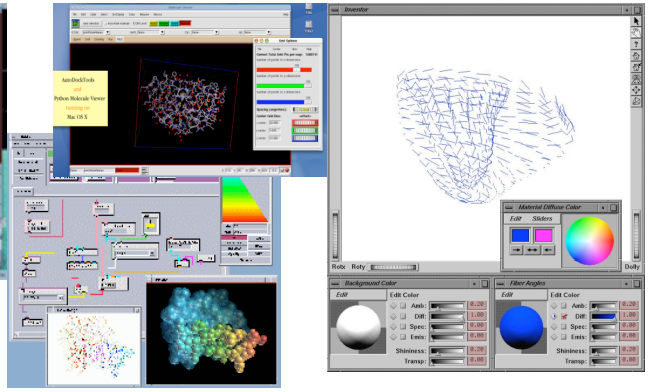
Rich Clients

Web Portals

Web Services

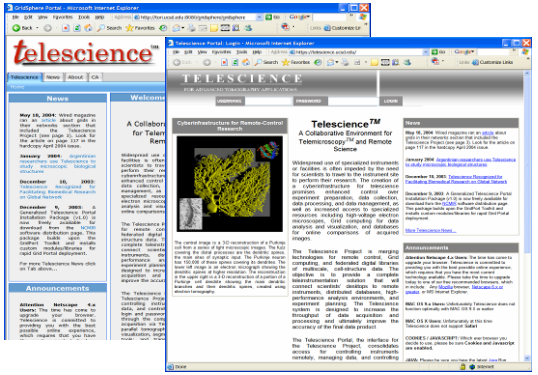


APBSCommand

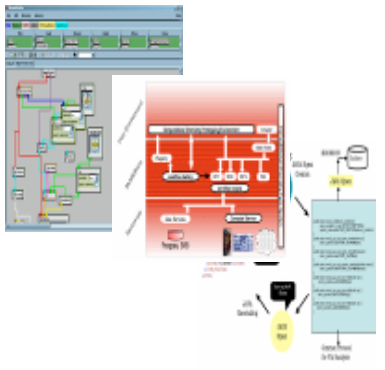


PMV ADT
Vision

Continuity



Telescience Portal

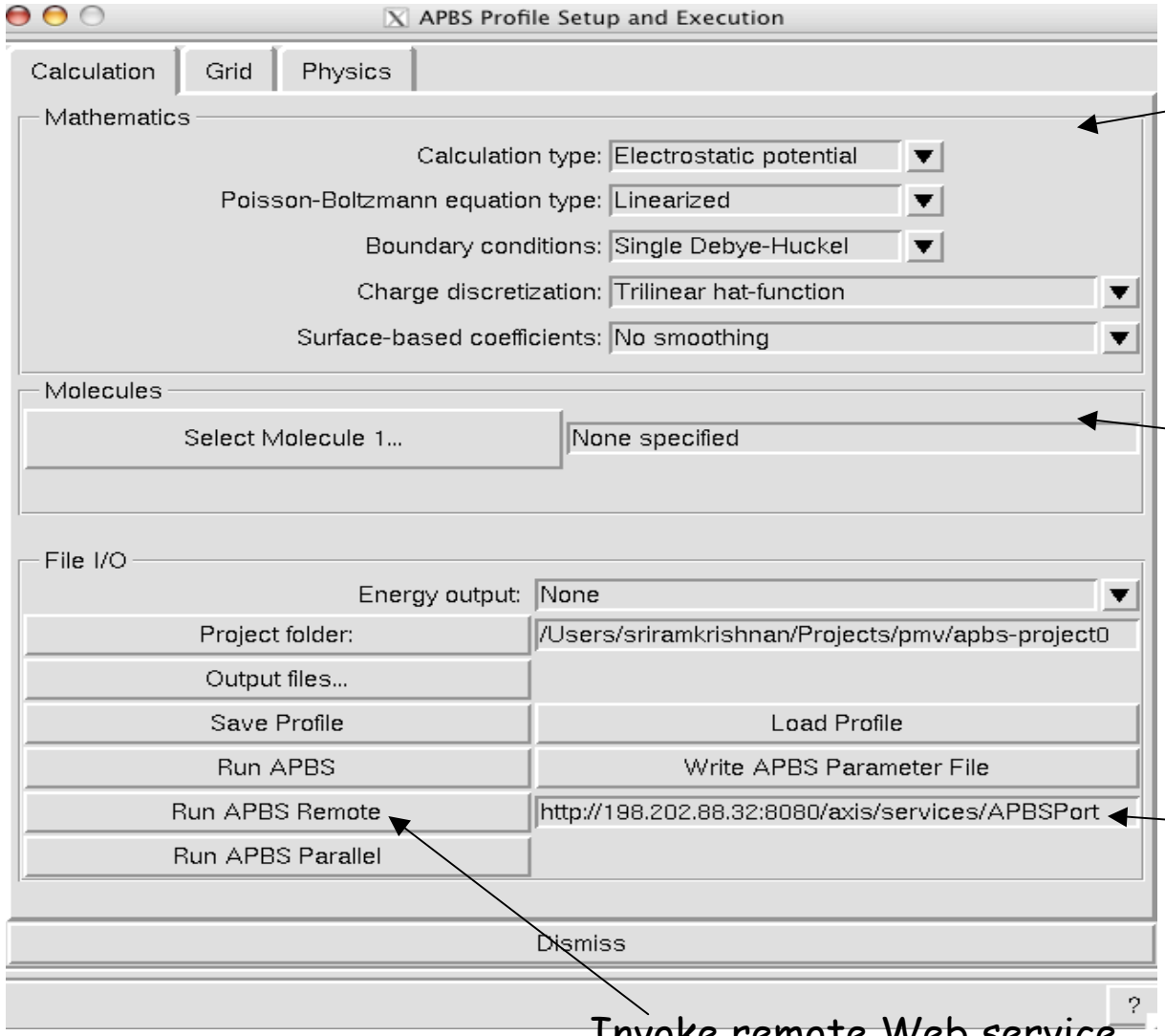


Workflow
Middleware

Sample Service: APBS

- Operations provided:
  - calculateBindingEnergy
  - calculateSolvationEnergy
  - calculateElectrostaticPotential
- Operations accept and return strongly typed parameters in XML format
  - Described by an XML Schema
  - Data binding provided by stub generators in various languages
    - WSDL2Java provided by Apache Axis
    - WSDL2PY provided by Python ZSI

PMV APBS Client: Michel Sanner, et al



Calculation Parameters

Molecule Data

URL of APBS Web service

Invoke remote Web service

(Incomplete) Acknowledgements

- Phil Papadopoulos
- Steve Mock
- Kurt Mueller
- Sandeep Chandra
- Nadya Williams
- Peter Arzberger
- Wilfred Li
- Robert Konecny
- Michel Sanner
- Wibke Sudholt
- APBS Team