Optimization Services
OS Server
and
OS Libraries

Robert Fourer
Jun Ma
Northwestern University
Kipp Martin
University of Chicago

Jun Ma
maj@northwestern.edu
Industrial Engineering and Management Sciences, Northwestern University
11/15/2005
Outline

• Motivation
• OS Framework
• OS Library
• OS Server
• Conclusion/User Experience
Motivation
Future of Computing

GRID
Workstation
Workstation
Workstation

Future of Computing

WSDL
Web Page and Service Server

SOAP/XML

UDDI
IDLE COMPUTER
NEOS

intelligent Agent

intelligent Agent

Computing Socket

USER

Laptop
OS Library

- **OSCommon**
  - representationParser
    - OSiL Reader/Writer
    - OSrL result
    - OSoL option
    - Etc.
  - util
    - data structure
    - io
    - xml
    - etc
  - communicationInterface
    - OShL (hook up to solvers/analyzers: solve, send, retrieve)
    - OScL (call to simulations)
    - OSdL (discover in registries)
- **localInterface**
  - OSInstance
    - etc.
  - nonlinear: defines all the nonlinear operator/operands/functions

```java
OSiLReader reader = new OSiLReader();
reader.read(example.osil);
reader.getLinearConstraintCoefficients();
reader.calculateNonlinearFunction(5, x); // x is double[]
```
OS Library

- **OSAgent**
  - Solver agent
  - Simulation agent
  - Solver agent
- **OSSolver**
  - Utility and implementation of os-compatible solvers
- **OSSimulation**
  - Utility and implementation of os-compatible simulation.
- **OSRegistry**
  - Allows os developers to register their services
  - Lets os users discover os services
  - Let os users/developers validate instances
- **OSAnalyzer**
  - Utility and implementation of os-compatible analyzers.
- **OSScheduler**
  - Schedules optimization jobs over the distributed system
  - Takes care of all the non-optimization related chores.

```java
OSSolverAgent agent = new OSSolverAgent();
agent.solverAddress = "http://1.2.3.6/impactSolverService";
String osrResult = agent.solve(osiInstance, osolOption);
```
OS Framework
Optimization Services Protocol (OSP)

GET /xt/services/ColorRequest HTTP/1.0
Content Length: 442
Host: localhost
Content-type: text/xml; charset=utf-8
SOAPAction: "/getColor"

<soap:Envelope>
  <soap:Body>
    <soap:Body>
      </soap:Body>
      OSP – specifies soap content
      Communication Interface Representation
      e.g. hook ("<OSiL> ... </OSiL>"

      <soap:Body>
      </soap:Body>
      </soap:Envelope>
OS Server

- Networking Protocols: HTTP, SOAP, OSP
  (OS server: Tomcat, Axis, OS library)

OSServer =

- http parser
- soap parser
- osp handler
- impactSolverService
Conclusion/User Experience

• Open Environment
• Convenience just like Using Utility Services
• No High Computing Power Needed
• No Knowledge in Optimization Algorithms and Software (solvers, options, etc.)
• Better and More Choices of Modeling Languages
• More Solver Choices
• Solve More Types of Problems
• Automatic Optimization Services Discovery
• Decentralized Optimization Services Development and Registration
• More Types of Optimization Services Components Integrated (Analyzers/Preprocessors, Problem Providers, Bench Markers)
• Smooth Flow and Coordination of Various Optimization Services Components.
• A Universal, Scalable and Standard Infrastructure that promotes Collaboration and Other Related Researches
• Concentration on Good Modeling
Interested?

- **MC43 – Standards for Optimization Problem Representation**
  - OSiL (Fourer, Ma, Martin)
  - OSiL stochastic extension (Gassmann, Fourer, Ma, Martin)
  - Panel on standards
  - etc.

- **TC44 – Optimization Tools and Modeling Languages**
  - OSmL (Ma, Martin)
  - Impact Solver Services (Huanyuan Sheng, Ma, Mehrotra)
  - etc.

- **TD43 – Distributed Optimization Systems**
  - Optimization Services Framework (Fourer, Ma, Martin)
  - etc.