OptiRisk SYSTEMS

XML representation of MP models for distributed optimisation applications

Patrick Valente

Gautam Mitra

CARISMA

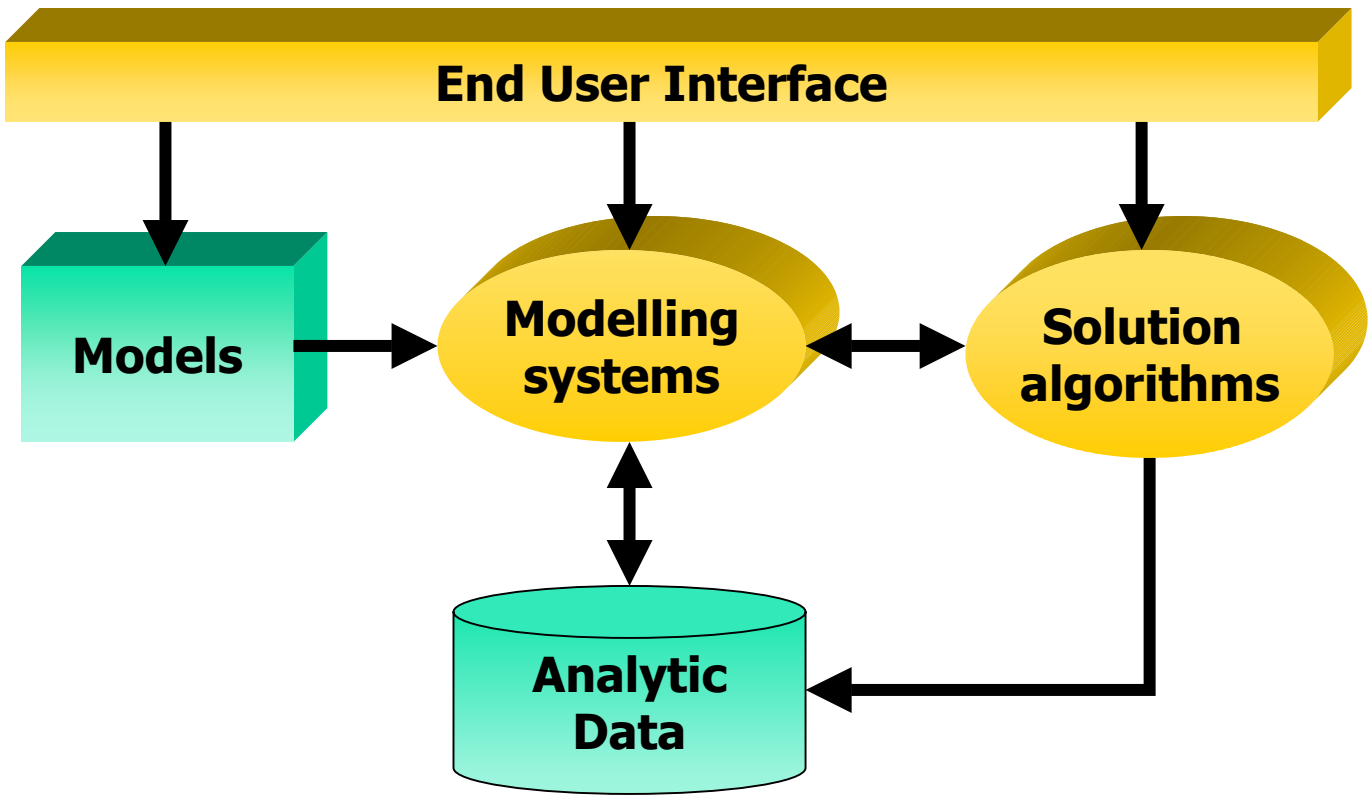
BRUNEL UNIVERSITY

Outline

- Optimisation tools their interaction
- Distributed Optimisation
- Examples of optimisation servers
- e-Services and optimisation
- Web services technology
- Representing MP using XML
- WebOpt and optimisation web services
- Conclusions

Optimisation tools interaction

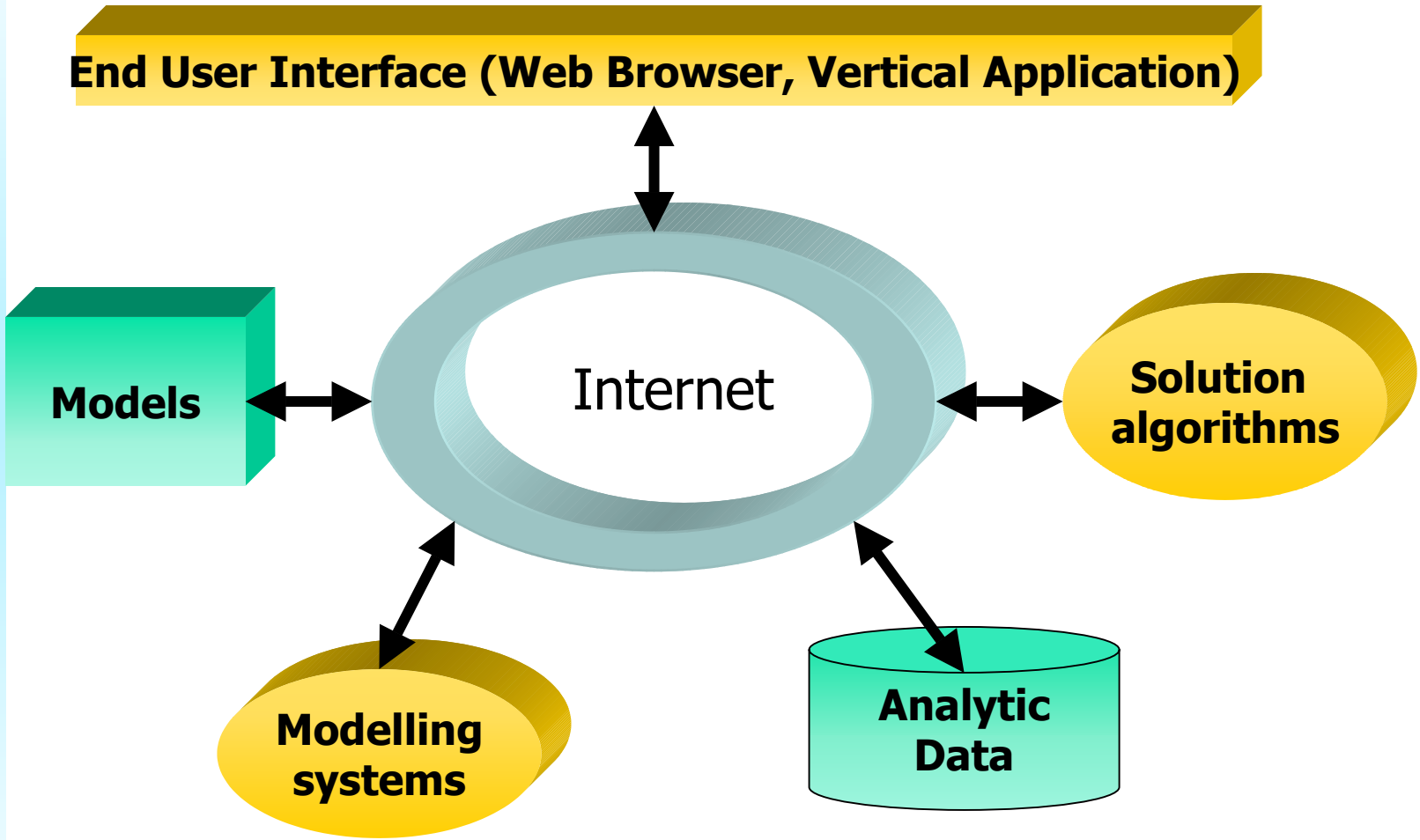
Traditionally, optimisation based applications comprise a set of models, analytical data and solvers connected via a modelling system.



Evolution of the optimisation tools

- Optimisation increasingly used as an inference engine
- Modelling systems and solvers are becoming available in **component form**
- Optimisation tools now allow more MP paradigms (QMIP and SP)
- Internet technology has facilitated the creation of distributed applications

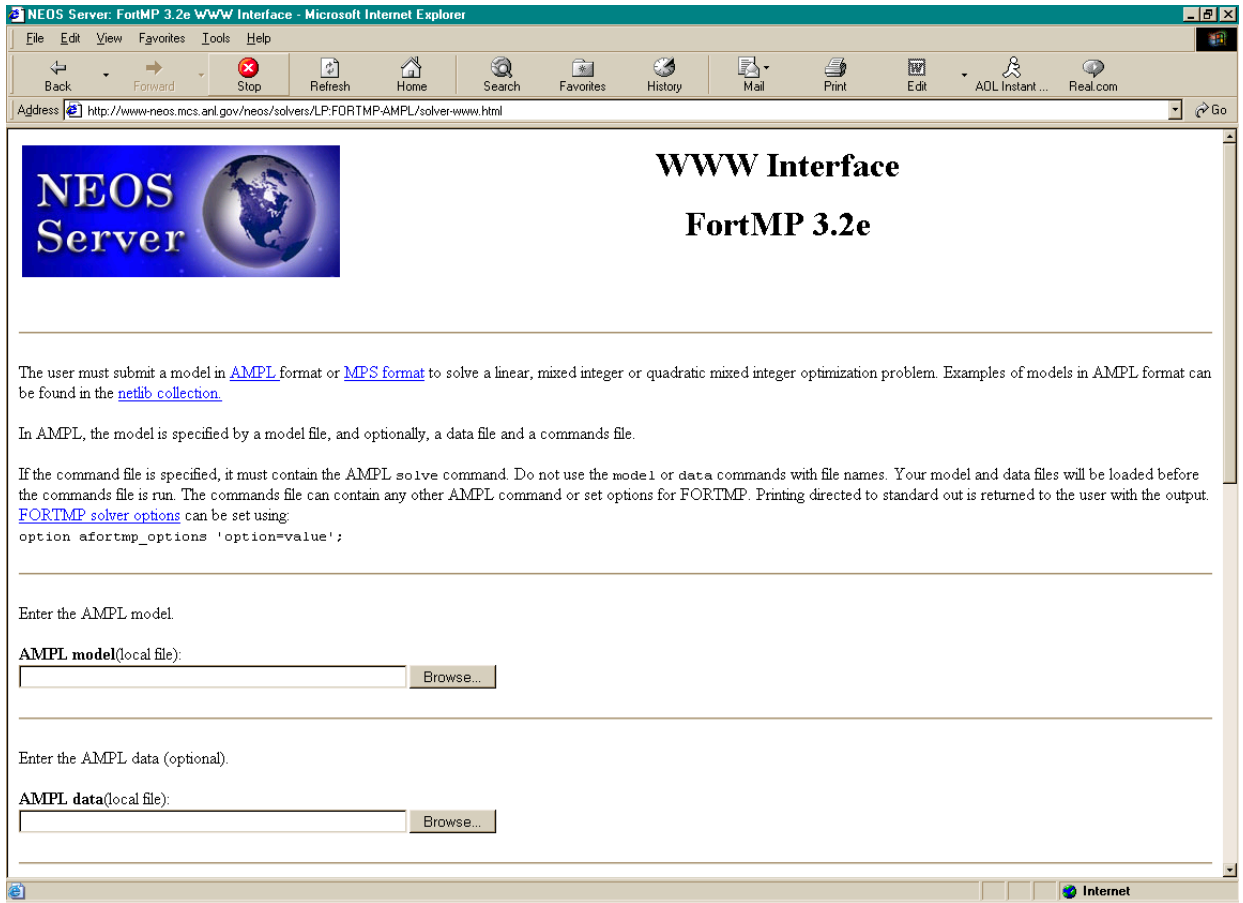
Distributed optimisation



NEOS

- Provides access to several classes of solvers
  - Linear Programming
  - Mixed Integer Programming
  - Mixed Integer Nonlinearly Constrained Optimisation
  - Nonlinearly Constrained Optimisation
  - Stochastic Programming
  - Complementarity Problems
  - Others
- Can be used as remote solver using the kestrel client for AMPL
- Website: http://www-neos.mcs.anl.gov/neos

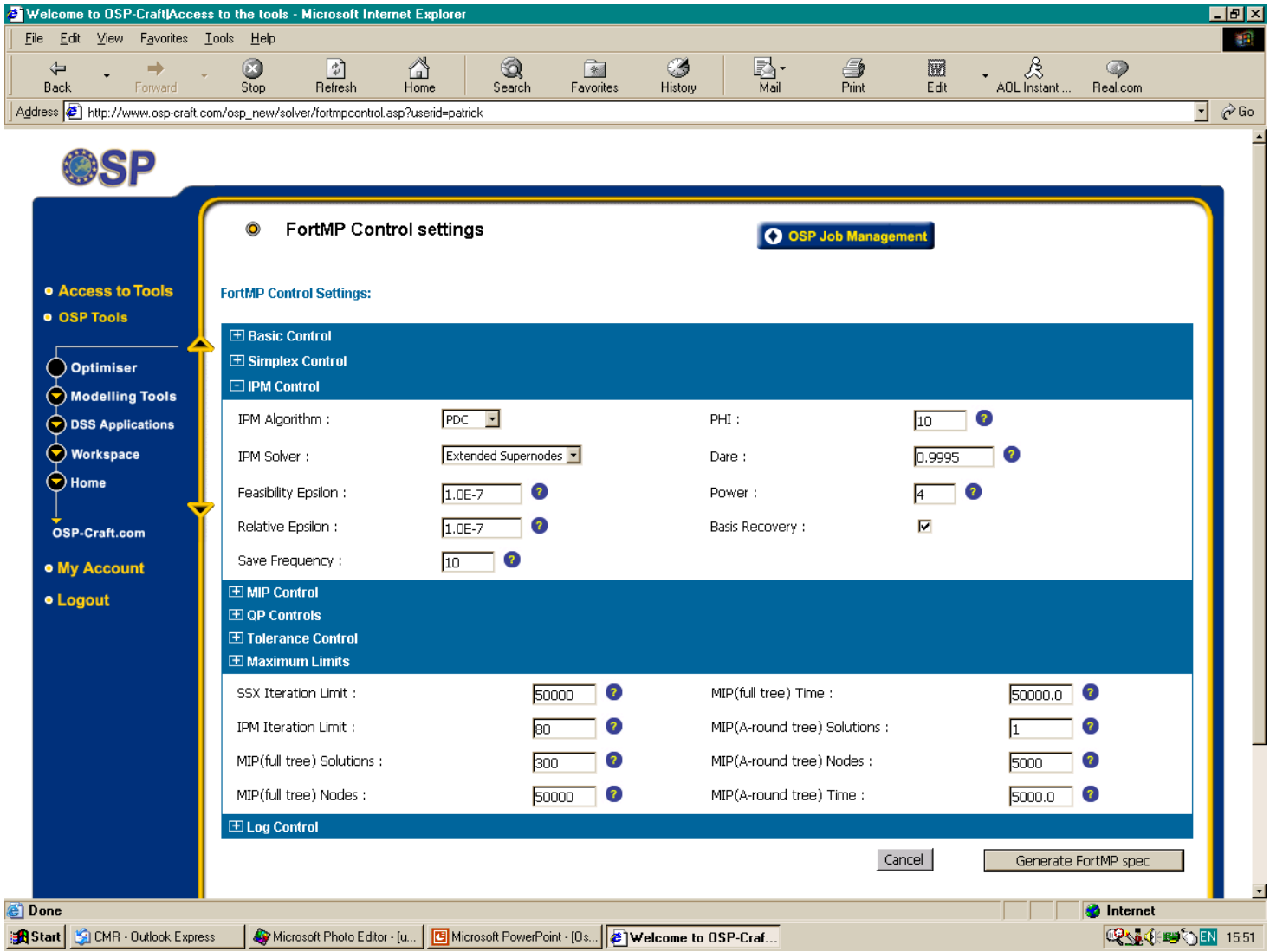
NEOS


NEOS Server: FortMP 3.2e WWW Interface - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit AOL Instant ... Real.com
Address http://www-neos.mcs.anl.gov/neos/solvers/LP:FORTMP-AMPL/solver-www.html
Go
NEOS Server
WWW Interface
FortMP 3.2e
The user must submit a model in AMPL format or MPS format to solve a linear, mixed integer or quadratic mixed integer optimization problem. Examples of models in AMPL format can be found in the netlib collection.
In AMPL, the model is specified by a model file, and optionally, a data file and a commands file.
If the command file is specified, it must contain the AMPL solve command. Do not use the model or data commands with file names. Your model and data files will be loaded before the commands file is run. The commands file can contain any other AMPL command or set options for FORTMP. Printing directed to standard out is returned to the user with the output. FORTMP solver options can be set using:
option afortmp_options 'option=value';
Enter the AMPL model.
AMPL model(local file):
Browse...
Enter the AMPL data (optional).
AMPL data(local file):
Browse...
Internet

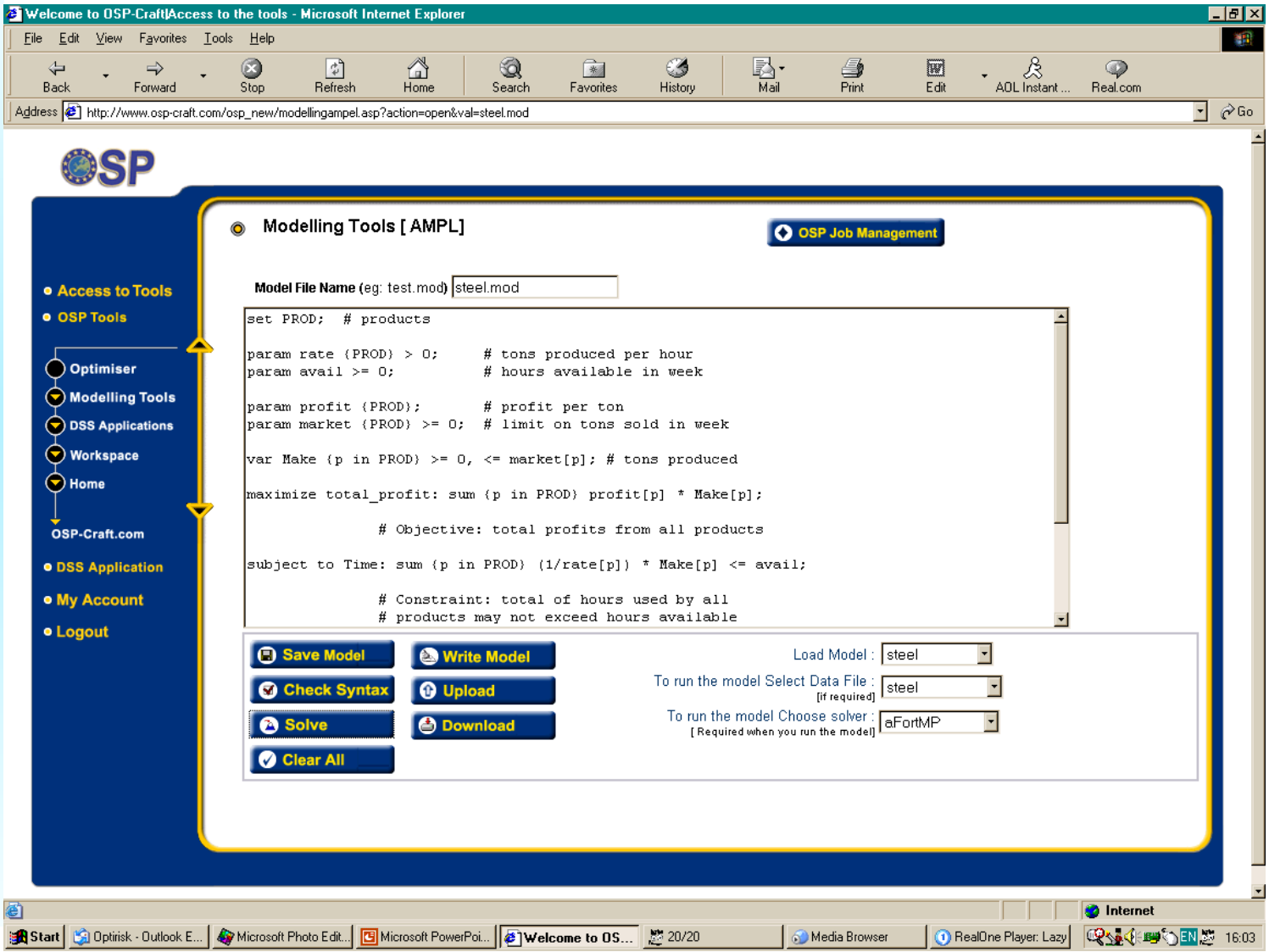
Optimisation Service Provider: OSP

- Solvers
  - CPLEX
  - OSL
  - FortMP
  - FortSP (SPInE's stochastic programming solver)
- Modelling systems
  - MPL
  - AMPL
- Vertical applications
  - Portfolio
  - Supply chain
- Website: www.osp-craft.com

Solver Engines: FortMP

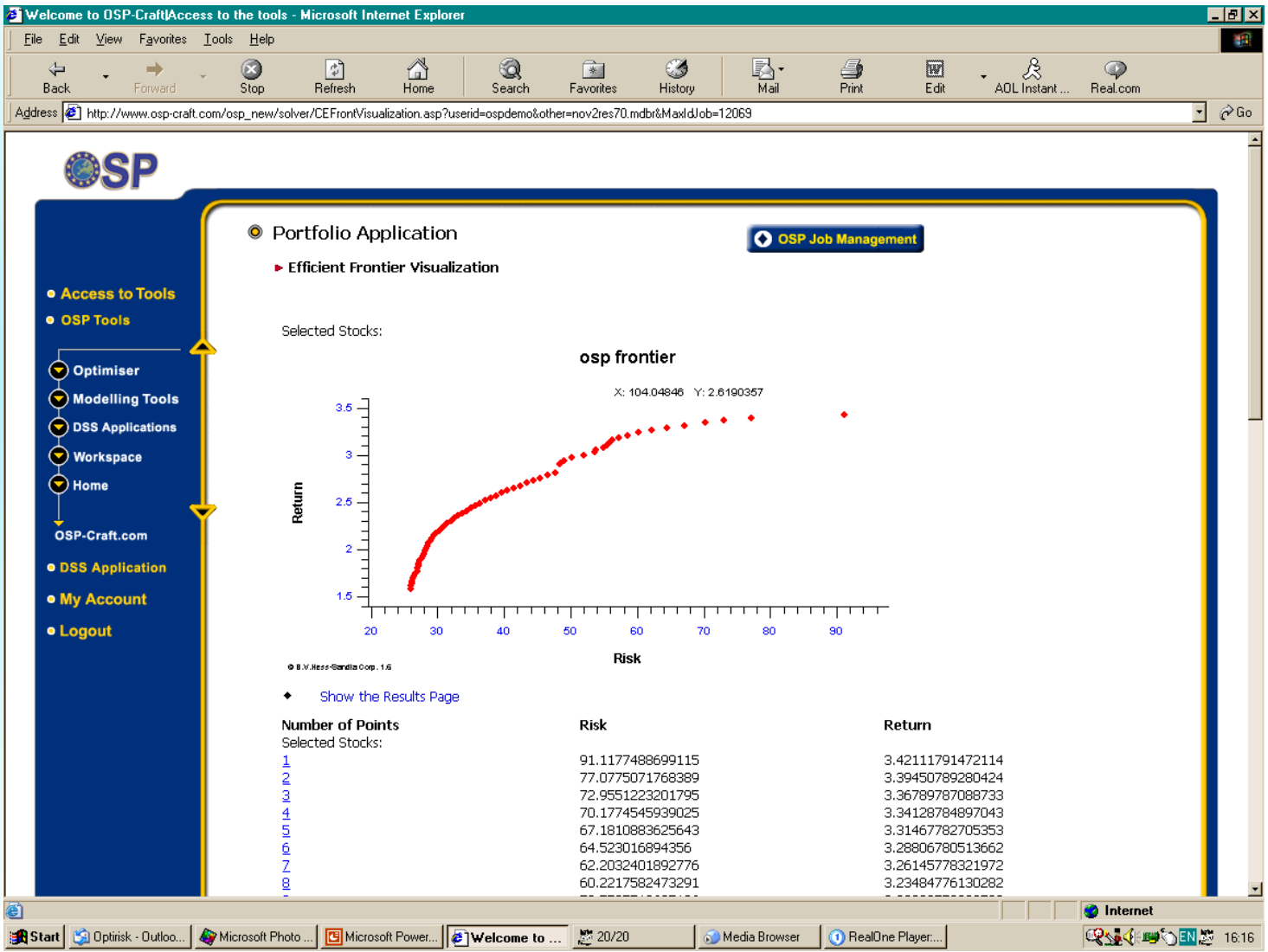


Modelling systems: AMPL



Vertical applications: Portfolio

(Discrete Efficient Frontier Visualisation)



Other distributed systems for optimisation

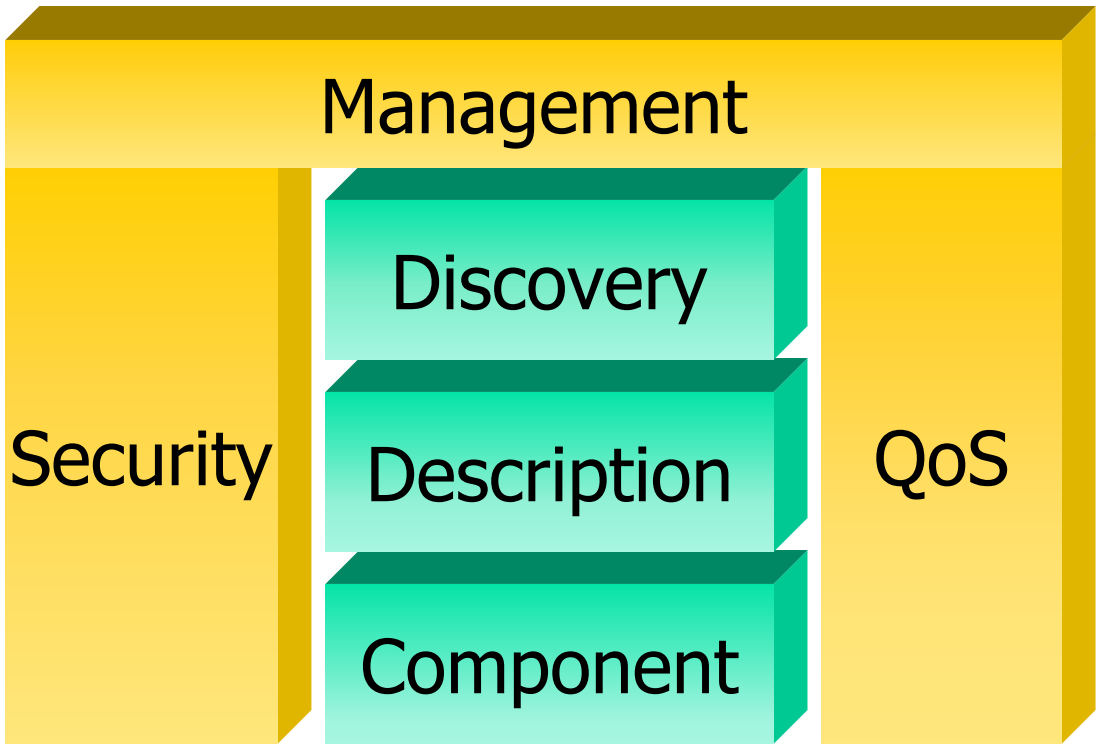
- AURORA / Financial GRID (Vienna Univ.)
- Open Optimization Framework (Imperial College)
- WebOpt (CARISMA et al.)

e-Services

- According to the definition:
  - The provision of **services** over an electronic network
- In practice:
  - Extends Application Service Provision (ASP) by enriching the underlying application with support infrastructure
- Underlying ideas:
  - Components and customer focus

e-Services

- Technology based on the concept of web-service



e-Services and optimisation

- The main ideas:
  - Provide each individual optimisation tool as an independent service
  - Enable remote interaction between the optimisation components
  - Take advantage of the e-Service paradigm for support and security

Web service

- Software system running on a server and providing a service by exposing a set of functions and methods.
- Concepts:
  - UDDI provides a worldwide registry for advertisment, discovery and integration of web services
  - WSDL describes the public interface of the service
  - SOAP is used to invoke methods provided by the web services
  - XML is used as the underlying representation format for the above and for the data sent to and from the web service

XML

- eXtensible Markup Language
- XML itself does not specify a "grammar"
- It is therefore truly extensible
- User defined tags can be combined and nested in any order
- XML data is stored in a tree structure.

XML example

```
<?xml version="1.0" ?>
<variable>
        <name>x1</name>
        <type>Integer</type>
        <upper>100</upper>
        <lower>5</lower>
</variable>
```

XML Schema and DTD

- Define grammars for XML documents
- DTD and XML Schema differ in that XML schema is itself defined in XML form.
- XML schema also supports *namespaces.*
- A number of XML Schemas exist for specific classes of contents (XSQL, MathML, ecc.)

XSD example

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<xsd:annotation>
      <xsd:documentation xml:lang="en">
            Schema for the definition of variables.
      </xsd:documentation>
</xsd:annotation>
<xsd:element name="variable" type="VariableType"/>
<xsd:complexType name="VariableType">
      <xsd:sequence>
            <xsd:element name="name" type="xsd:string"/>
            <xsd:element name="type" type="xsd:string"/>
            <xsd:element name="upper" type="xsd:double"/>
            <xsd:element name="lower" type="xsd:double"/>
      </xsd:sequence>
</xsd:complexType>
```

SOAP

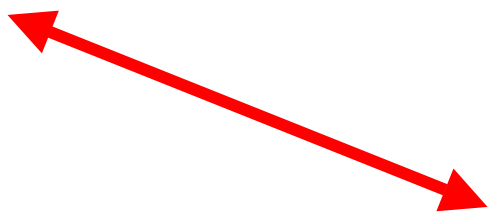
```
<POST /websolver HTTP/1.1
Host: www.webopt.org
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn
SOAPAction: "Some-URI"

<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  <SOAP-ENV:Body>
       <m:loadModel xmlns:m="Some-URI">
           <linearProgram>
                …XML representation of an LP…
            </linearProgram>
       </m:loadModel>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

MathML

```
<apply>
    <leq/>
    <apply>
   <plus/>
        <apply>
          <times/>
            <cn>3</cn>
            <ci>x</ci>
         </apply>
        <apply>
          <times/>
            <cn>5</cn>
            <ci>y</ci>
         </apply>
    <cn>1<cn>
</apply>
```

- Developed by W3C
- Provides a "Presentation language" and a "Content language"
- Lacks of sparse data representation mechanism

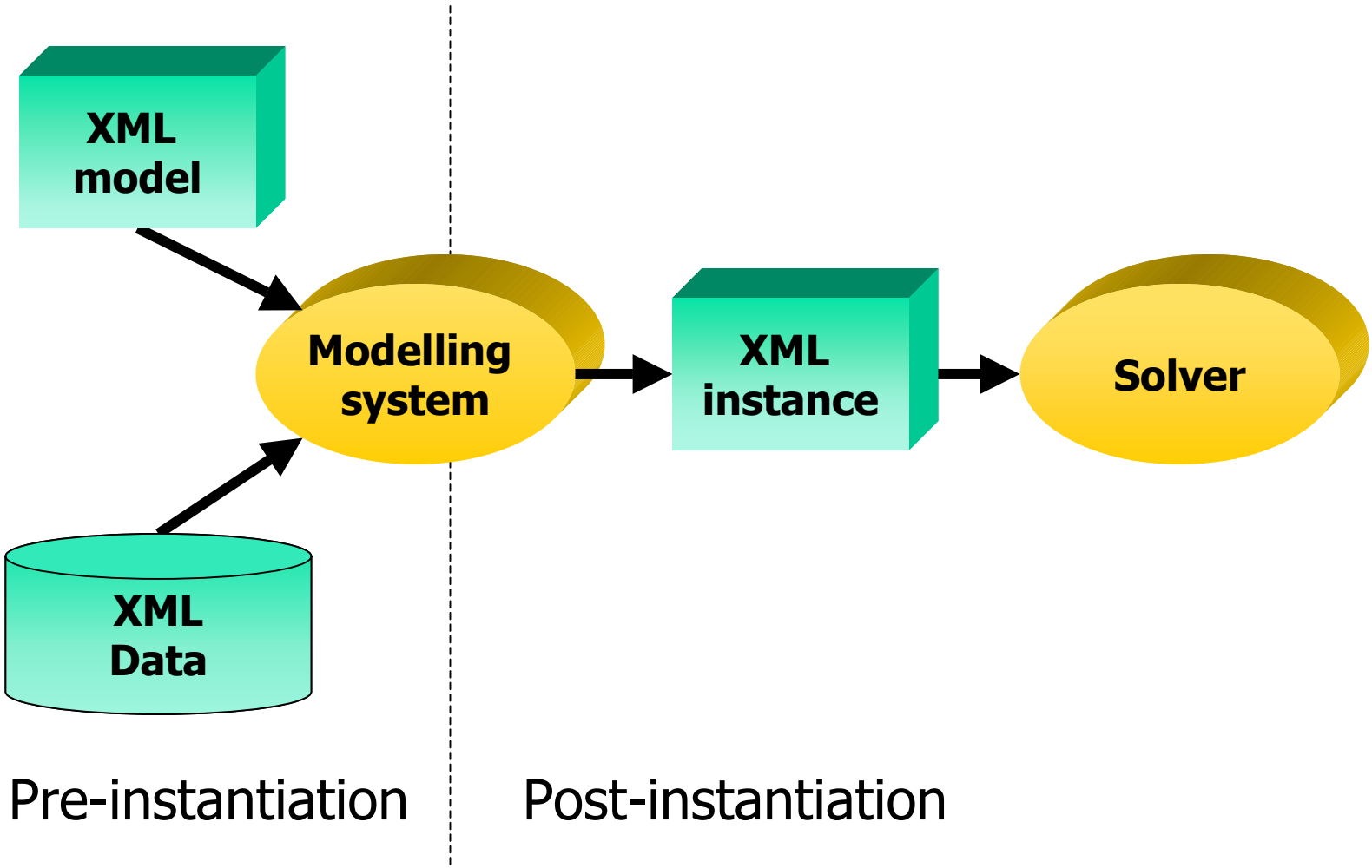


$$3x + 5y \leq 1$$

Representing MP: approaches

- Post-instantiation
  - Representation of model matrices
- Pre-instantiation
  - Separate representation of algebraic model and data instance

Approaches


XML model
Modelling system
XML instance
Solver
XML Data
Pre-instantiation
Post-instantiation

Review

- Post-Instantiation
  - OptML (Kristjansson 2000)
  - SNOML (Lopes, Fourer 2002)
  - FML (Fourer, Lopes, Martin 2002)
  - MathProgML (Hochreiter, Pflug 2003)
- Pre-Instantiation
  - AML (Ezechukwu,Maros)
  - Others?

OptML

```
<problem>
    <creation/>
    <statistics/>
    <matrix>
            <sizes/>
            <rows/>
            < columns/>
      </matrix>
</problem>
```

- Provides models statistics and size
- Sparse matrix representation
- Very close to MPS in representation power
- Supports non linear programming

SNOML

```
<minimize>
    <scenario>
        <subproblem>
            <cols/>
            <irset/>
            <obj/>
            <imatrix/>
            <irhs/>
        </subproblem>
    </scenario>
</minimize>
```

- Uses MathML for non linear expressions
- Keeps track of matrix block structure
- Supports stochastic programming and non linear models
- Follows SMPS to some extent

FML

```
<mathProgram>
    <linearProgram>
        <lpDescription>
        <rows>
        <columns>
        <sparseMatrix>
    </linearProgram>
    <linearProgramSolution>
        <source>
        <primalVector>
        <dualVector>
        <status>
        <solverMessage>
    </linearProgramSolution>
</mathProgram>
```

- Uses MathML for all scalars
- Sparse matrix representation
- Very close to MPS in representation power
- Provides a format for the solution values

MathProgML

```
<mathprog>
   <objective id="1">
         MathML expression
   </objective>
   <constraints>
      <subjectto id="1">
         MathML expression
      </subjectto>
      <subjectto id="2">
         MathML expression
      </subjectto>
         ...
   </constraints>
</mathprog>
```

- Aims to extend MathML
- Introduces sparse matrix representation
- Introduces tree structures
- Supports stochastic programming and non linear models

AML

- This XML schema aims to represent the entire algebraic formulation of a model
- Keeps model and data separated
- Equivalent (?) to an algebraic modelling language
- Part of a set of definitions which include:
  - ORML (Optimisation Reporting Markup Language)
  - WSOP (Optimisation Web Services Protocol)
  - OSCP (Optimisation Service Connectivity Protocol)

Where are we then?

- There is still no agreement on which approach to follow
- Other issues not thoroughly addressed:
  - Encryption
  - Compression
- Alternative: SOAP with Attachments or DIME

WebOpt

- Web site: www.webopt.org
- EU-sponsored ASIA IT&C project
- Promotes knowledge transfer and collaboration between Europe and Asia
- Extends OSP by implementing optimisation web-services
- The aim is to develop complete optimisation e-Services

OptiRisk SYSTEMS

WebOpt: solver web services

- Requirements:
  - Common calling interface
  - Common representation standard for
    - Input models
    - Optimum solutions
    - Messages

CARISMA

OptiRisk SYSTEMS

COIN based solver web service

- Based on the OSI Solver Interface
- Provides a set of methods to:
  - Upload and manipulate models
  - Set algorithmic parameters
  - Solve a model (using SSX, IPM, B&B…)
  - Extract the solution
- Methods are invoked using SOAP messages
- Uses (which?)ML to represent the MP model format and the results

Conclusions

- Distributed optimisation tools are gaining importance
- The representation MP in XML is still an open question
- So is the definition of homogeneous solver APIs
- COIN-OR and OSI are a good starting point
- WebOpt will provide optimisation components for research and B2B applications
- More work is required to consolidate existing propositions