Optimization Via the Internet: NEOS 5 and Beyond

Robert Fourer

Industrial Engineering & Management Sciences
Northwestern University
Evanston, Illinois 60208-3119, U.S.A.

4er@iems.northwestern.edu — www.iems.northwestern.edu/~4er/

and many others . . .

INFORMS International

Hong Kong, June 25-28, 2006

Session TC19.3, Web Services & Application Builders for Optimization

Robert Fourer, Jun Ma

Industrial Engineering & Management Sciences
Northwestern University

[4er,maj]@iems.northwestern.edu

Kipp Martin

Graduate School of Business
University of Chicago

kmartin@gsb.uchicago.edu

Jorge J. Moré, Todd S. Munson, Jason Sarich

Mathematics and Computer Science Division
Argonne National Laboratory

[more,tmunson,sarich]@mcs.anl.gov

Dominique Orban

Mathematics and Industrial Engineering Department
École Polytechnique de Montréal

dominique.orban@polymtl.ca

Optimization Challenges

Software

Solver

Server

Optimization: Software Challenges

No one way to solve

- Hundreds of *solvers*
- Competing “free” codes and commercial products
- Competing methods

Models built to order

- Competing *modeling systems*
- Each system supports multiple solvers
- Many solvers work with multiple systems

Result: A tangle of software

- No comprehensive packages as in statistics or simulation

. . . an opportunity for the Internet
to offer guidance and access

Optimization: Solver Challenges

Power

- Faster computers, multiple processors
- More powerful algorithms, better implementations

Ease of use

- Modeling languages and systems
  (AIMMS, AMPL, GAMS, LINGO, MOSEL, MPL, OPL)
- Add-ins to general-purpose systems (Excel, MATLAB)
- Object-oriented programming interfaces

Accessibility

- Unpleasant to download and install
- Trial versions have various limitations
- Few solvers installed at any one site

. . . distributed processing schemes are typically one-of-a-kind

Optimization: Server Challenges

Optimization as a service, not a product

- One remote *server* offering many solvers
- Any local *client* can submit optimization "jobs"

Varied client programs

- General-purpose software: web browsers, e-mail programs
- General optimization environments (already mentioned)
- Specialized tools

Large-scale integration

- ***The NEOS Server***
- Optimization services framework
  - ***XML-based optimization instance standards***
  - Central repository
  - Decentralized sources for solver info & problem analysis
  - Decentralized services for optimization

. . . ***problems special to optimization***

NEOS www-neos.mcs.anl.gov/neos/

A general-purpose optimization server

- Over 45 solvers in all
  - Linear, linear network, linear integer
  - Nonlinear, nonlinear integer, nondifferentiable & global
  - Stochastic, semidefinite, semi-infinite, complementarity
- Commercial as well as experimental solvers
- Central scheduler with distributed solver sites

A research project

- Currently free of charge
- Supported through the Optimization Technology Center
  of Northwestern University and Argonne National Laboratory

NEOS Design

Flexible architecture

- Central controller and scheduler machine
- Distributed solver sites

Numerous formats

- Low-level formats: MPS, SIF, SDPA
- Programming languages: C/ADOL-C, Fortran/ADIFOR
- High-level modeing languages: AMPL, GAMS

Varied submission options

- E-mail – Web forms – Direct function call
- TCP/IP socket-based submission tool: Java or tcl/tk

. . . server processes submissions of new solvers, too

NEOS Frequently Asked Questions

Who uses it?

- Where are its users from?
- How much is it used?

What kinds of solvers does it offer?

- Who supplies them?
- Which are most heavily used?
- Where are they hosted?

How is it supported?

- Who answers user questions?

Who Uses NEOS? *(a sample)*

- We are using NEOS services for duty-scheduling for ground handling activities in a regional airport environment.
- We used NEOS to solve nonlinear optimization problems associated with models of physical properties in chemistry.
- Our company is working with various projects concerning R&D of internal combustion engines for cars and brakes for heavy vehicles.
- We are working on bi-dimensional modeling of earth's conductivity distribution.
- I am dealing with ultimate limit-state analyses of large dams by means of a non-standard approach ("direct method"); this requires solving problems of linear and non-linear programming. The NEOS server is an extraordinary tool to perform parametric tests on small models, in order to choose the best suited solver.
- I have used NEOS with LOQO solver to optimize an interpolator. . . . My domain is digital receivers where the receiver clock is not changed to match the transmitter clock.

Who Uses NEOS? *(more)*

- I have been able to build and solve a prototype combinatorial auction MIP model using AMPL and NEOS in a fraction of the time it would have required me to do this had I needed to requisition a solver and install it locally.
- Our idea is trying to design antennas by using the computer. . . . We have tried various solvers on NEOS to see if this is possible at all.
- I am using the LOQO solver and code written in AMPL to perform numerical optimization of a spinor Bose-Einstein condensate.
- We are using the NEOS Server for solving linear and nonlinear complementarity problems in engineering mechanics and in robotics.
- I have been working on a system for protein structure prediction. . . . I had need to incorporate a nonlinear solver to handle packing of sidechain atoms in the protein.

Who Uses NEOS? *(academic)*

- I am regularly suggesting my students to use NEOS as soon as their projects in AMPL cannot be solved with the student edition. **So they debug their AMPL models locally . . . and then they run their real-life projects thanks to NEOS.**
- I didn't even know what nonlinear programming was and after I discovered what it was, it became clear how enormous a task it would be to solve the problems assigned to me. . . . I had extremely complicated objective functions, both convex and nonconvex, an armload of variables, and an armload of convex, nonconvex, equality and inequality constraints, but when I sent off the information via the web submission form, within seconds I received extremely accurate and consistent results. **The results were used for verifying a certain theory in my professor's research** and so accuracy was extremely important.
- NEOS has been a very valuable tool in the two graduate optimization courses that I teach. **NEOS allows students to see a broader variety of solvers than we have available . . .**

. . . more at **`www-neos.mcs.anl.gov/neos/stories.html`**

NEOS Users

Where are They From?

June 2005 through May 2006:
Identifiable domain and >= 20 submissions

(com)	12575
(edu)	69648
(gov)	34547
(net)	21204
(mil)	277
(org)	374

Argentina (ar)	26
Australia (au)	434
Austria (at)	171
Belgium (be)	5973
Brazil (br)	2183
Bulgaria (bg)	60
Canada (ca)	11247
Chile (cl)	2745
Colombia (co)	1201
Croatia (hr)	223
Cyprus (cy)	179
Czech Republic (cz)	1012
Denmark (dk)	94
Finland (fi)	46
France (fr)	2162
Germany ...	

NEOS Users

What Countries are They From?

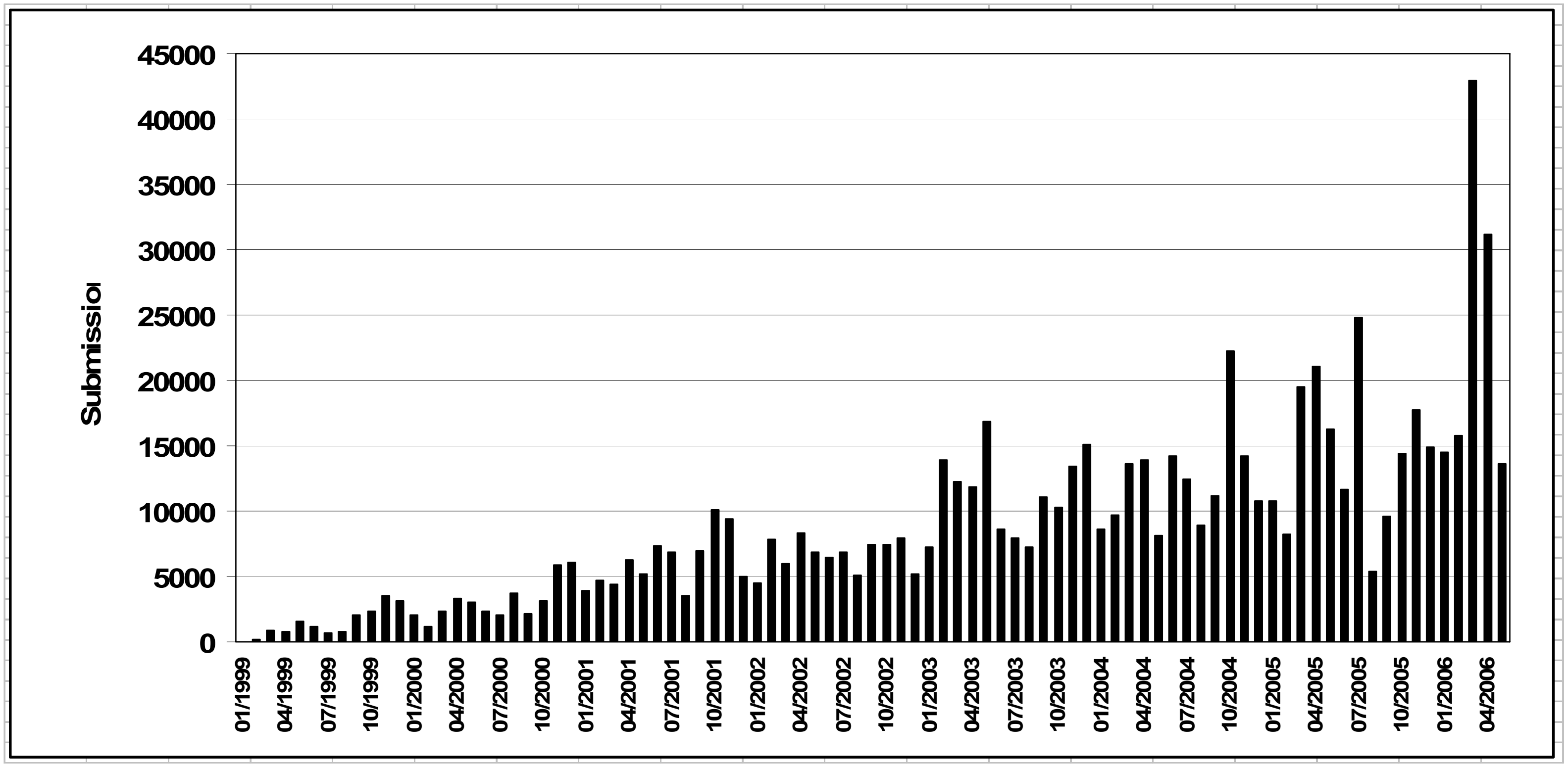
Argentina (ar)	26
Australia (au)	434
Austria (at)	171
Belgium (be)	5973
Brazil (br)	2183
Bulgaria (bg)	60
Canada (ca)	11247
Chile (cl)	2745
Colombia (co)	1201
Croatia (hr)	223
Cyprus (cy)	179
Czech Republic (cz)	1012
Denmark (dk)	94
Finland (fi)	46
France (fr)	2162
Germany (de)	3694
Hong Kong (hk)	45
Hungary (hu)	34
India (in)	704
Ireland (ie)	24
Italy (it)	2401

Japan (jp)	1376
Korea (kr)	35
Malaysia (my)	640
Mexico (mx)	163
Netherlands (nl)	5026
New Zealand (nz)	299
Norway (no)	156
Poland (pl)	195
Portugal (pt)	1257
Russia (ru)	178
Singapore (sg)	1324
Spain (es)	1265
Sweden (se)	3442
Switzerland (ch)	518
Taiwan (tw)	53
Turkey (tr)	1432
Ukraine (ua)	398
United Kingdom (uk)	2084
Uruguay (uy)	292
Uzbekistan (uz)	98

NEOS Users

How Much Do They Use It?

Submissions by month, 1/1999 through 5/2006



. . . 25 / hour over past year

. . . 50 / hour in peak months

What Solvers Does NEOS Offer?

For familiar problem types

- Linear programming
- Linear network optimization
- Linear integer programming
- Nonlinear programming
- Stochastic linear programming
- Complementarity problems

For emerging problem types

- Nondifferentiable optimization
- Semi-infinite optimization
- Global optimization
- Nonlinear integer programming
- Semidefinite & 2nd-order cone programming

. . . virtually every published semidefinite programming code

NEOS Solvers

Who Supplies Them?

Some commercial solver vendors

- Xpress, FortMP (mixed integer)
- CONOPT, KNITRO, MOSEK (nonlinear)

Universities and their researchers

- BonsaiG (mixed integer)
- DONLP2, FILTER, LANCELOT, LOQO, MINOS, SNOPT (nonlinear)

Open-Source Enthusiasts

- GLPK (mixed integer)

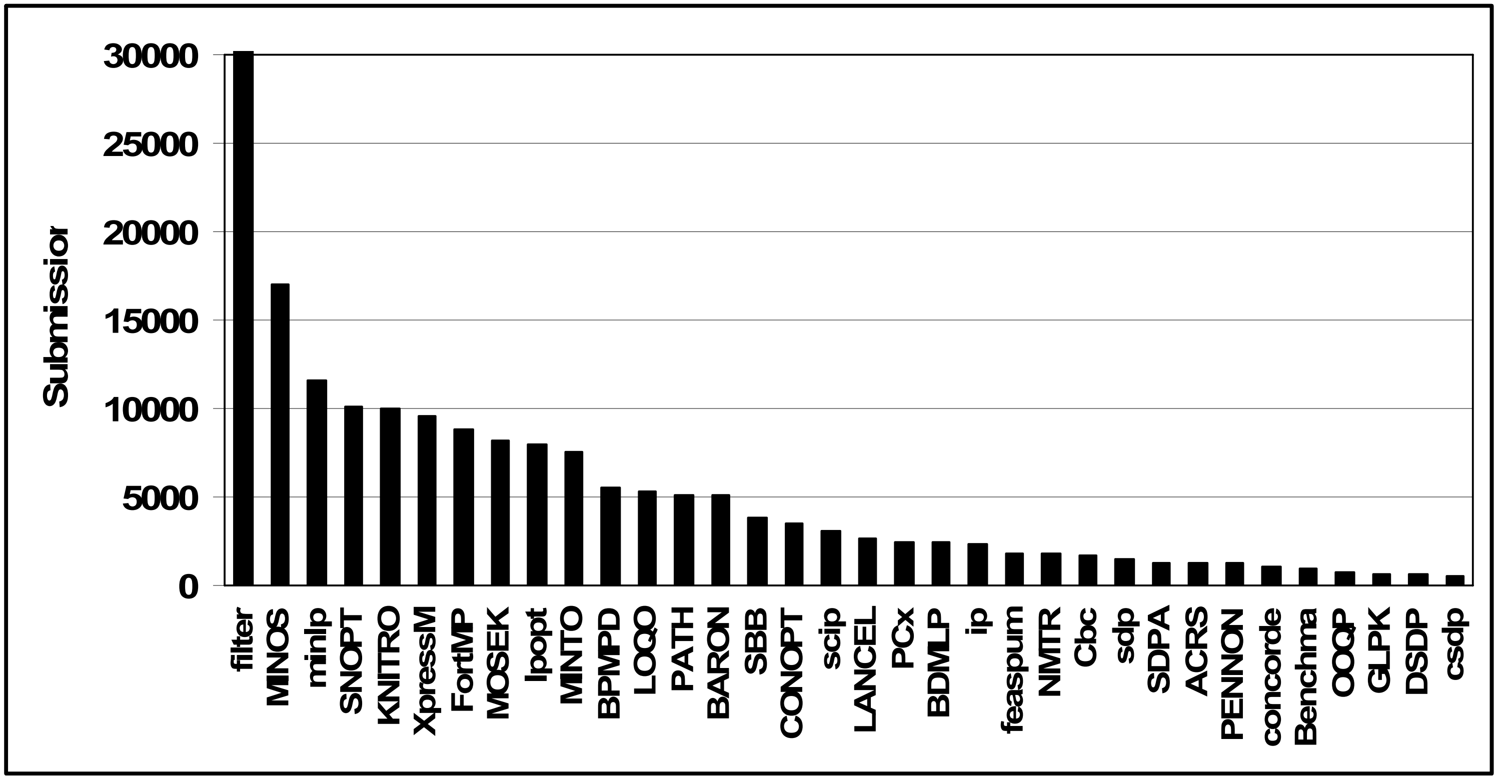
with thanks to . . .

- ***AMPL*** and ***GAMS*** developers
- ***Hans Mittelmann,*** Arizona State

NEOS Solvers

Which are Most Heavily Used?

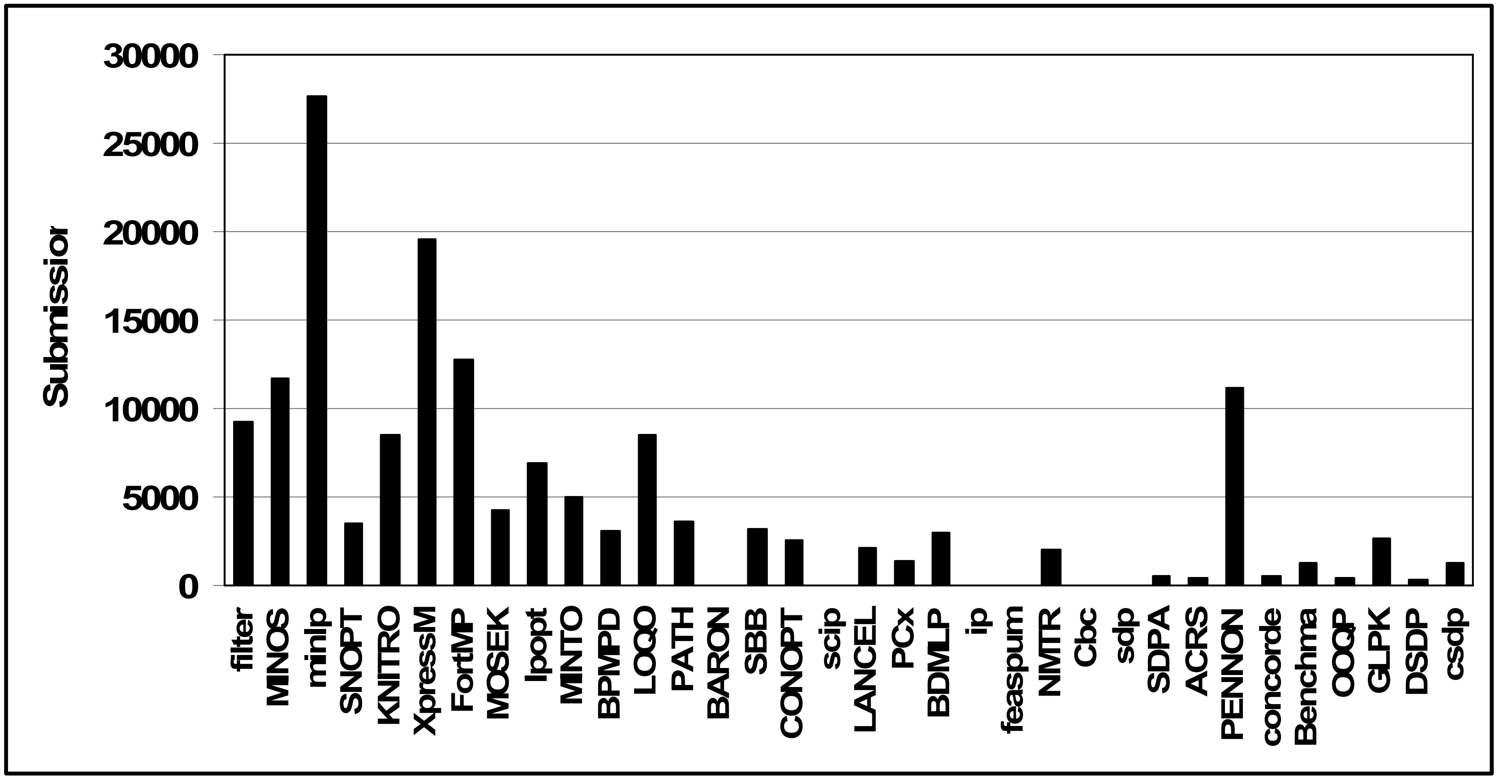
Totals for June 2005 to May 2006



NEOS Solvers

Which are Most Heavily Used?

Totals for June 2005 to May 2006



Where are They Hosted?

Varied workstations at

- Aachen University of Technology
- Argonne National Laboratory
- Arizona State University
- Lehigh University
- National Taiwan University
- Northwestern University *(with support from Sun Microsystems)*
- University of Wisconsin at Madison

. . . new hosts are readily added anywhere on the Internet

How is NEOS Supported?

Grants

- National Science Foundation, Operations Research Program, grant DMI-0322580
- National Science Foundation, Information Technology Research Program, grant CCR-0082807
- U.S. Department of Energy, Office of Advanced Scientific Computing, Mathematical, Information, and Computational Sciences Division subprogram, Contract W-31-109-Eng-38
- National Science Foundation, Challenges in Computational Science Program, grant CDA-9726385

Donations

- Processor cycles
- Many people's time

. . . no user charges as yet

Who Answers Users' Questions?

Large mailing list for support questions

- NEOS developers
- Solver developers

Support request buttons on every page

Optimization Services

Decentralized framework

- *Centralized repository, but decentralized . . .*
- Sources for solver information and problem analysis
- Services for optimization

Optimization cyberinfrastructure

- XML-based standards for representation
- Benchmarking and verification services
- High-performance computing on demand

What's special about optimization?

- Independence of modeling, data, and solver software
- Choice of solver based on mathematics
- Huge variation in solver performance

XML-Based Standard Formats

Motivation

- for any standard format
- for an XML-based format

"OSxL" standards

- OSiL: problem instances
- OSoL: solver options
- **OSrL: results**

. . . and a host of others
(see www.optimizationservices.org)

Components of OSiL

- XML schema for text file format, *and*
- Corresponding in-memory data structures
- Libraries for reading and writing the above

XML Means "Tagged" Text Files . . .

Example: html for a popular home page

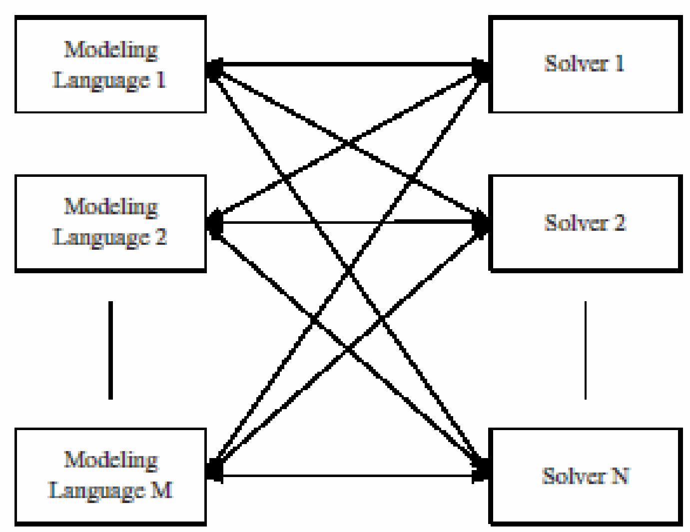
```
<html><head><meta http-equiv="content-type" content="text/html;
charset=UTF-8"><title>Google</title><style><!--
body,td,a,p,.h{font-family:arial,sans-serif;}
.h{font-size: 20px;}
.q{text-decoration:none; color:#0000cc;}
//-->
</style>
</head><body bgcolor=#ffffff text=#000000 link=#0000cc
vlink=#551a8b alink=#ff0000 onLoad=sf()><center><table border=0
cellspacing=0 cellpadding=0><tr><td><img src="/images/logo.gif"
width=276 height=110 alt="Google"></td></tr></table><br>
.......
<font size=-2>&copy;2003 Google - Searching 3,307,998,701 web
pages</font></p></center></body></html>
```

. . . a collection of XML tags is designed for a special purpose

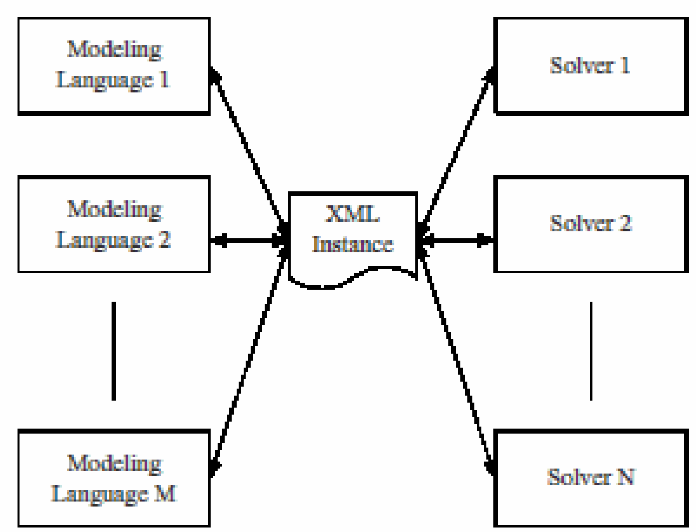
. . . by use of a schema written itself in XML

Advantage of any standard

MN drivers
without a standard



M + N drivers
with a standard



Advantages of an XML Standard

Specifying it

- Unambiguous definition via a *schema*
- Provision for *keys* and *data typing*
- Well-defined expansion to new *name spaces*

Working with it

- Parsing and validation via standard *utilities*
- Amenability to *compression* and *encryption*
- Transformation and display via XSLT *style sheets*
- Compatibility with *web services*

What about "MPS Form"?

Weaknesses

- Standard only for LP and MIP, not for
  nonlinear, network, complementarity, logical, . . .
- Standard not uniform (especially for SP extension)
- Verbose ASCII form, with much repetition of names
- Limited precision for some numerical values

Used for

- Collections of (mostly anonymous) test problems
- Bug reports to solver vendors

Not used for

- **Communication between modeling systems and solvers**

Standards

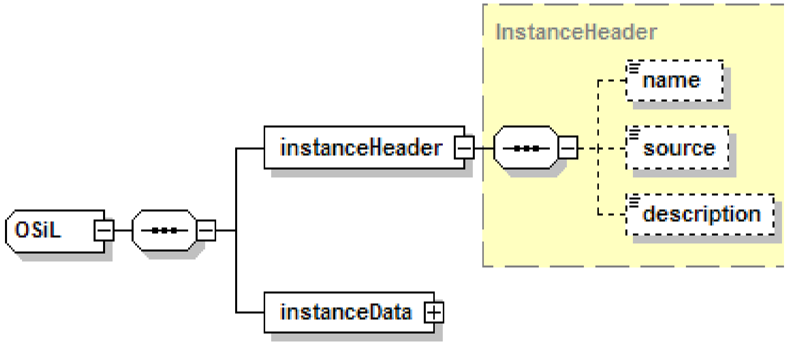
Text from the OSiL Schema

```
<xs:complexType name="Variables">
  <xs:sequence>
    <xs:element name="var" type="Variable" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="number" type="xs:positiveInteger" use="required"/>
</xs:complexType>
```

```
<xs:complexType name="Variable">
  <xs:attribute name="name" type="xs:string" use="optional"/>
  <xs:attribute name="init" type="xs:string" use="optional"/>
  <xs:attribute name="type" use="optional" default="C">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="C"/>
      <xs:enumeration value="B"/>
      <xs:enumeration value="I"/>
      <xs:enumeration value="S"/>
    </xs:restriction>
  </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="lb" type="xs:double" use="optional" default="0"/>
  <xs:attribute name="ub" type="xs:double" use="optional" default="INF"/>
</xs:complexType>
```

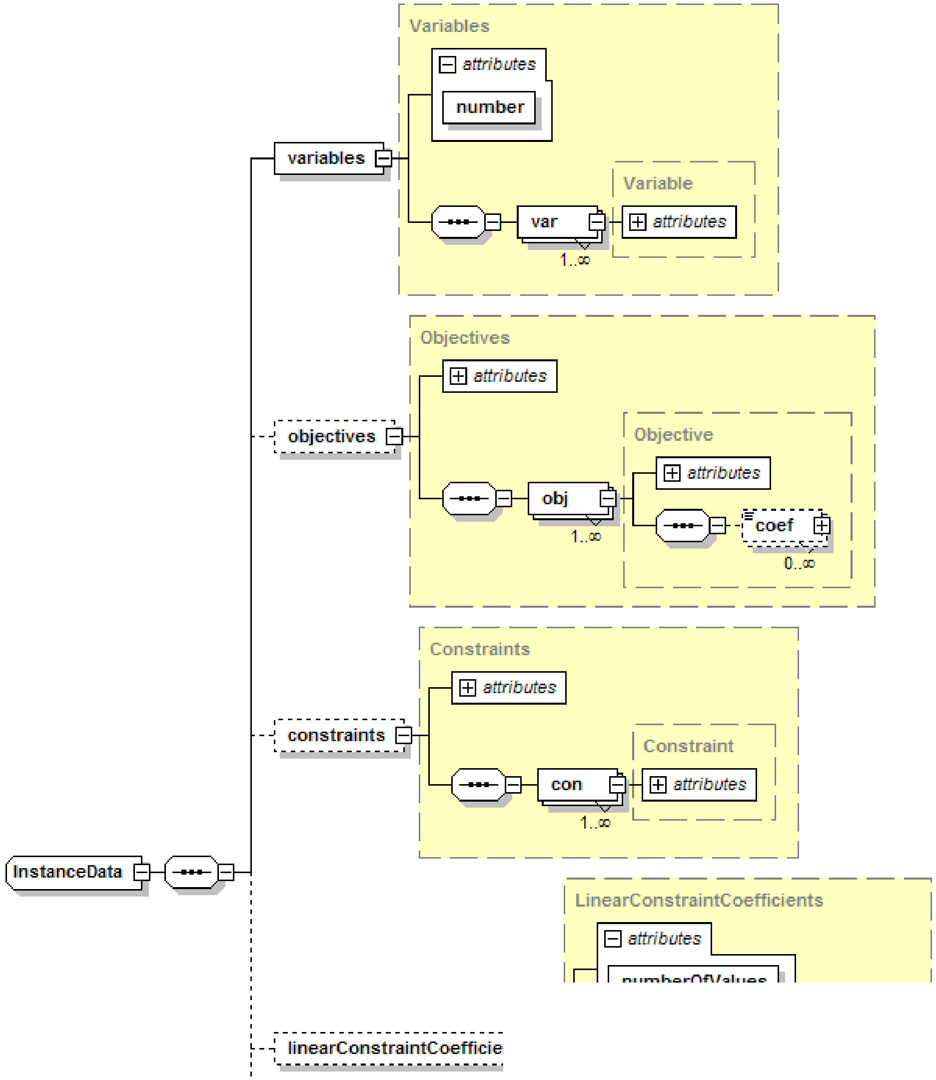
Standards

Diagram of the OSiL Schema



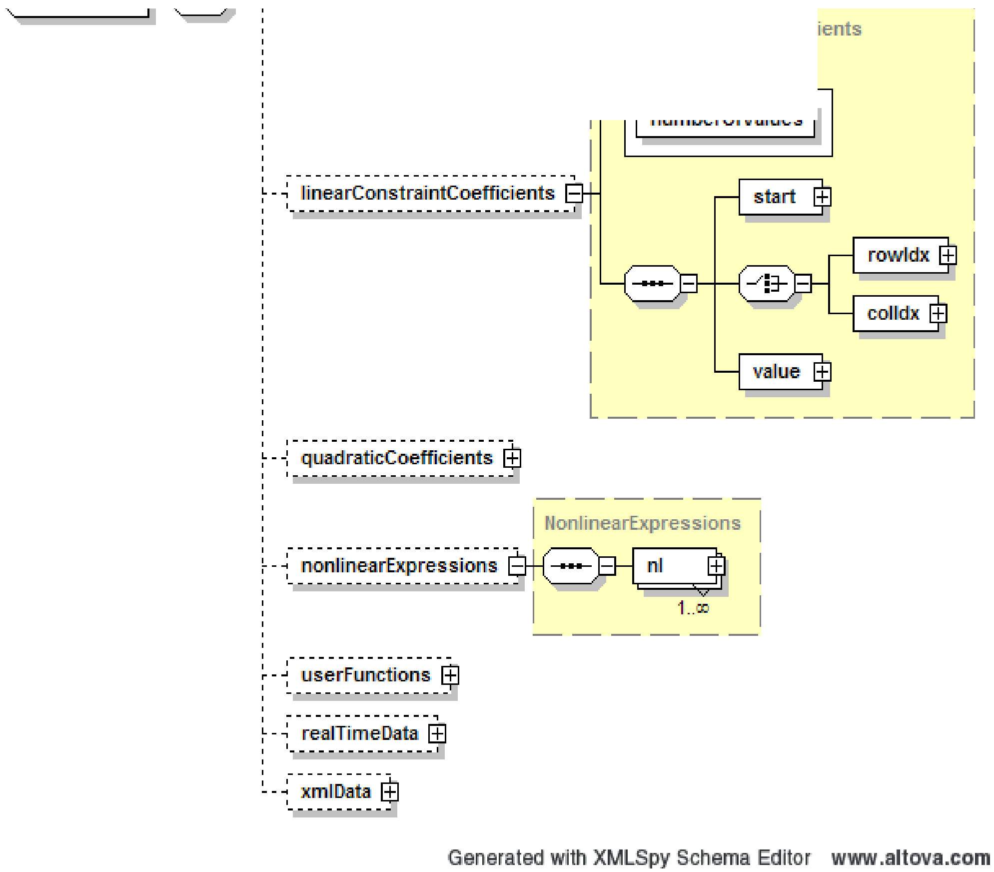
Standards

Details of OSiL's instanceData Element



Standards

Details of OSiL's instanceData Element



Example: A Problem Instance (in AMPL)

```
ampl: expand _var;

Coefficients of x[0]:
        Con1   1 + nonlinear
        Con2   7 + nonlinear
        Obj    0 + nonlinear

Coefficients of x[1]:
        Con1   0 + nonlinear
        Con2   5 + nonlinear
        Obj    9 + nonlinear

ampl: expand _obj;

minimize Obj:
        (1 - x[0])^2 + 100*(x[1] - x[0]^2)^2 + 9*x[1];

ampl: expand _con;

subject to Con1:
        10*x[0]^2 + 11*x[1]^2 + 3*x[0]*x[1] + x[0] <= 10;

subject to Con2:
        log(x[0]*x[1]) + 7*x[0] + 5*x[1] >= 10;
```

Standard formats

Example in OSiL

```
<instanceHeader>
   <name>Modified Rosenbrock</name>
   <source>Computing Journal3:175-184, 1960</source>
   <description>Rosenbrock problem with constraints</description>
</instanceHeader>

<variables number="2">
   <var lb="0" name="x0" type="C"/>
   <var lb="0" name="x1" type="C"/>
</variables>

<objectives number="1">
   <obj maxOrMin="min" name="minCost" numberOfObjCoef="1">
      <coef idx="1">9</coef>
   </obj>
</objectives>

<constraints number="2">
   <con ub="10.0"/>
   <con lb="10.0"/>
</constraints>
```

Standard formats

Example in OSiL *(continued)*

```
<linearConstraintCoefficients numberOfValues="3">
   <start>
      <el>0</el>
      <el>2</el>
      <el>3</el>
   </start>
   <rowIdx>
      <el>0</el>
      <el>1</el>
      <el>1</el>
   </rowIdx>
   <value>
      <el>1.0</el>
      <el>7.0</el>
      <el>5.0</el>
   </value>
</linearConstraintCoefficients>

<quadraticCoefficients numberOfQPTerms="3">
   <qpTerm idx="0" idxOne="0" idxTwo="0" coef="10"/>
   <qpTerm idx="0" idxOne="1" idxTwo="1" coef="11"/>
   <qpTerm idx="0" idxOne="0" idxTwo="1" coef="3"/>
</quadraticCoefficients>
```

Standard formats

Example in OSiL *(continued)*

```
<nl idx="-1">
   <plus>
      <power>
         <minus>
            <number type="real" value="1.0"/>
            <variable coef="1.0" idx="1"/>
         </minus>
         <number type="real" value="2.0"/>
      </power>
      <times>
         <power>
            <minus>
               <variable coef="1.0" idx="0"/>
               <power>
                  <variable coef="1.0" idx="1"/>
                  <number type="real" value="2.0"/>
               </power>
            </minus>
            <number type="real" value="2.0"/>
         </power>
         <number type="real" value="100"/>
      </times>
   </plus>
</nl>
```

Standard formats

Example in OSiL *(continued)*

```
<nl idx="1">
   <ln>
      <times>
         <variable idx="0"/>
         <variable idx="1"/>
      </times>
   </ln>
</nl>
```

Compression

Specific to OSiL

- Collapse sequences of row/column numbers
- Collapse repeated element values
- Encode portions using base-64 datatype

General for XML

- Compression schemes designed for XML files

Comparisons

- XML base-64 $<$ MPS
- XML with multiple values collapsed $<$ $2 \times$ MPS
- Compressed XML < Compressed MPS

Other Features in OSiL . . .

In current specification

- Real-time data
- Functions defined by the user

In process of design

- Stochastic programming / optimization under uncertainty
- Logical / combinatorial constraints
- Semidefinite / cone programming

Associated languages

- OSoL for communicating options to solvers
- OSrL for communicating results from solvers

. . . broader family of "optimization services" languages

In-Memory Data Structures

OSInstance object class

- Parallels the OSiL schema
- complexType in schema ←→ class in OSInstance
- attributes / children of an element ←→ members of a class
- choices / sequences in the schema arrays ←→ array members

OS expression tree

- Parallels the *nonlinear* part of the OSiL schema
- Designed to avoid lengthy "switch" statements

Advantages

- One standard instead of two
- Complements COIN-OR's OSI

Libraries (APIs, Interfaces)

Use by client

- OSInstance `set()` methods generate instance in memory
- `OSiLWriter` writes instance to a file in OSiL format
- Using SOAP over HTTP, instance is sent to a solver

Use by solver

- OSiLReader in solver interface
  reads instance from OSiL format back to memory
- OSInstance get() methods extract instance data
  as needed by solver
- Solver works on the problem
- Results are sent back similarly, using OSrL

. . . OSiL can be skipped
when instance is passed in memory

For More Information

- E.D. Dolan, R. Fourer, J.J. Moré and T.S. Munson, Optimization on the NEOS Server. SIA*M News* **35:**6 (July/August 2002) 4, 8–9. `www.siam.org/siamnews/07-02/neos.pdf`
- NEOS Server: `www-neos.mcs.anl.gov/neos`
- R. Fourer, L.B. Lopes and K. Martin, LPFML: A W3C XML Schema for Linear and Integer Programming. *INFORMS Journal on Computing* **17** (2005) 139–158.
- R. Fourer, J. Ma and K. Martin, OSiL: An Instance Language for Optimization. ***Draft available soon.***