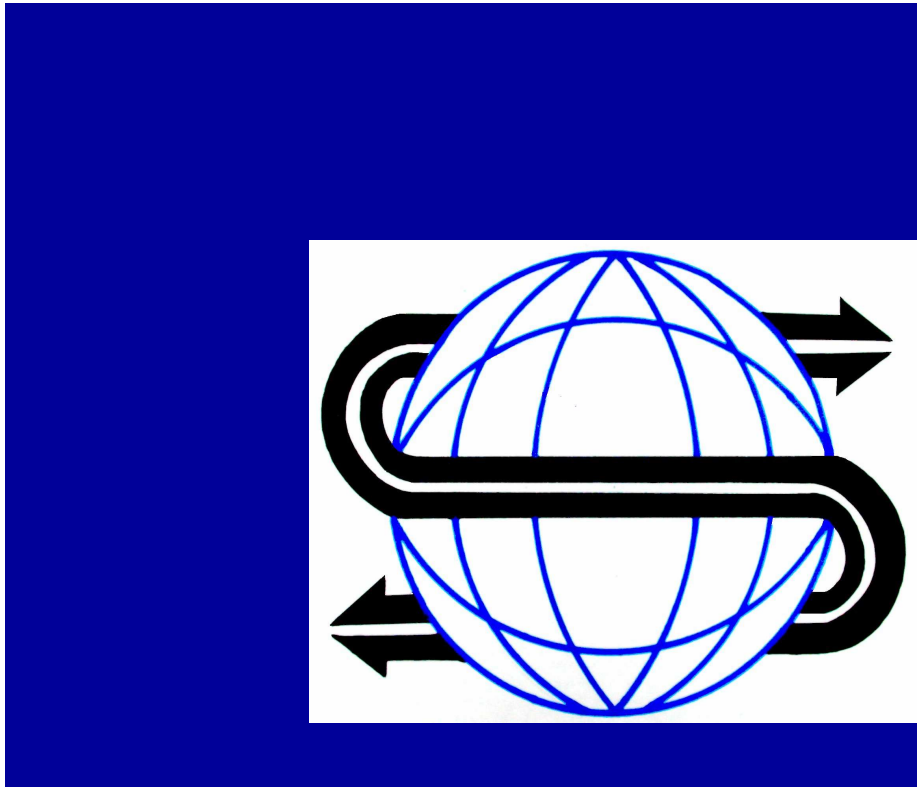


Optimization Services (OS)

-- The Internet for OR

-- The Next Generation NEOS

Robert Fourer

Jun Ma

Northwestern University

Kipp Martin

University of Chicago

Jun Ma

Motorola, Schaumburg

01/18/2005

OUTLINE

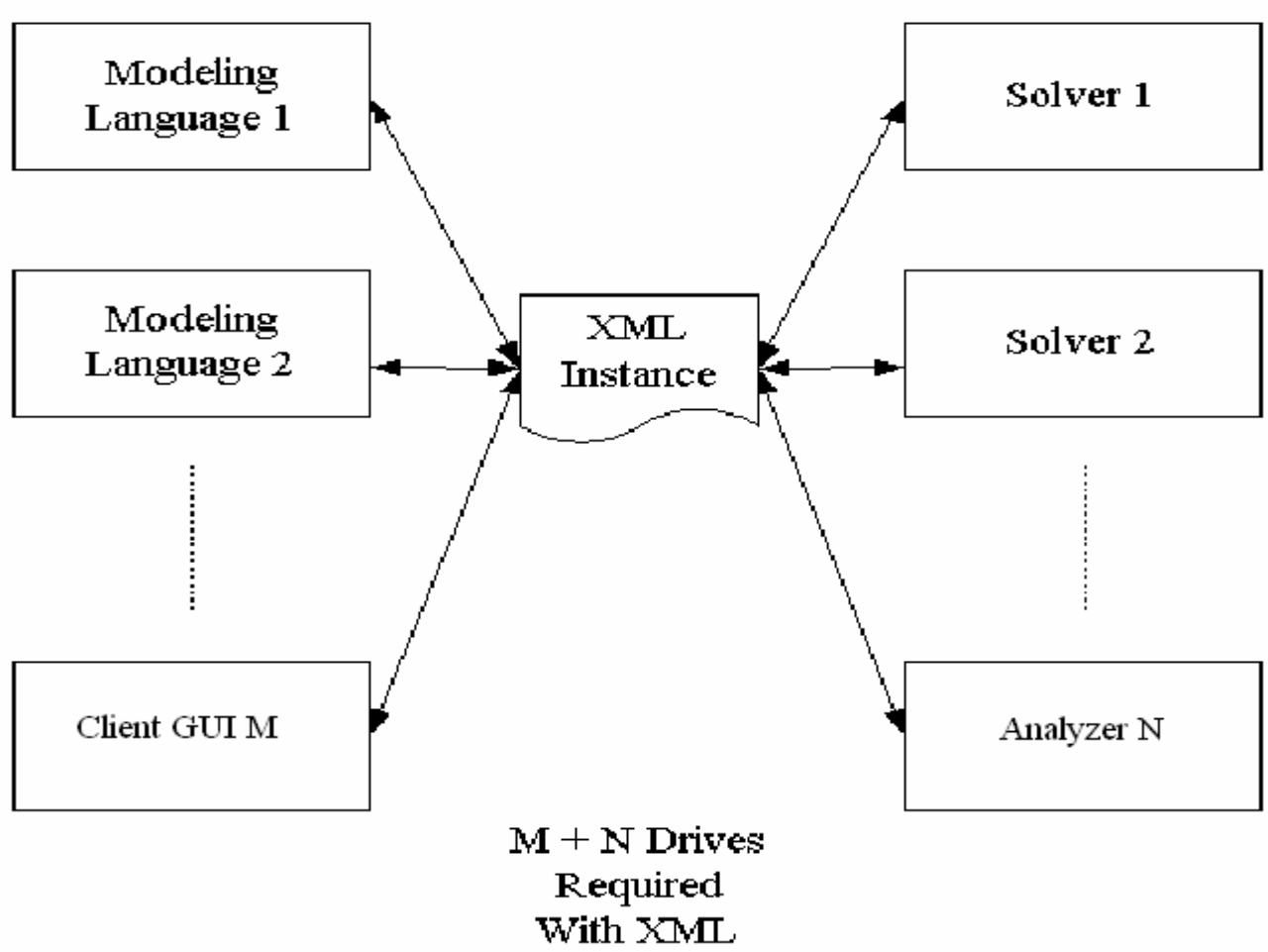
1. Motivations
2. Introduction
3. Optimization Services and OSxL
4. An OSxL Example –
   Optimization Services instance Language (OSiL)
5. Business Values
6. Conclusion

Motivation

- Supply chain modeler should just concentrate on writing a good supply chain model.
- An open, scalable and standard environment that facilitates Development & use of OR software and promotes collaboration and other related researches
- Convenience and power
  - Just like using utility services (therefore the name – Optimization Services)
  - Knowledge in optimization algorithms and software (solvers, options, etc.) required of users should be as little as possible
  - Better and more choices of modeling languages and solvers
  - More types of optimization services (analyzers/preprocessors, problem providers, bench markers, registry, simulation etc.)
  - Solve more types of problems
- Distributed and decentralized environment
  - Automatic optimization services discovery
  - Optimization services development and registration

Motivation

For example, it would be nice to have an instance representation language. This is specified by the Optimization Services instance Language (OSiL)



Introduction

- Optimization Services is

  A framework, NOT a system

  – cf. constitution, NOT government/Court System. Only that the framework specifications are written in XML languages (NOT English).

  – But we are developing the modeling system according to this framework.

  – We are also building libraries for other people to put up their optimization services.

- Distributed environment (Local environment being just a special case)
- Service oriented, optimization centered, decentralized architecture.

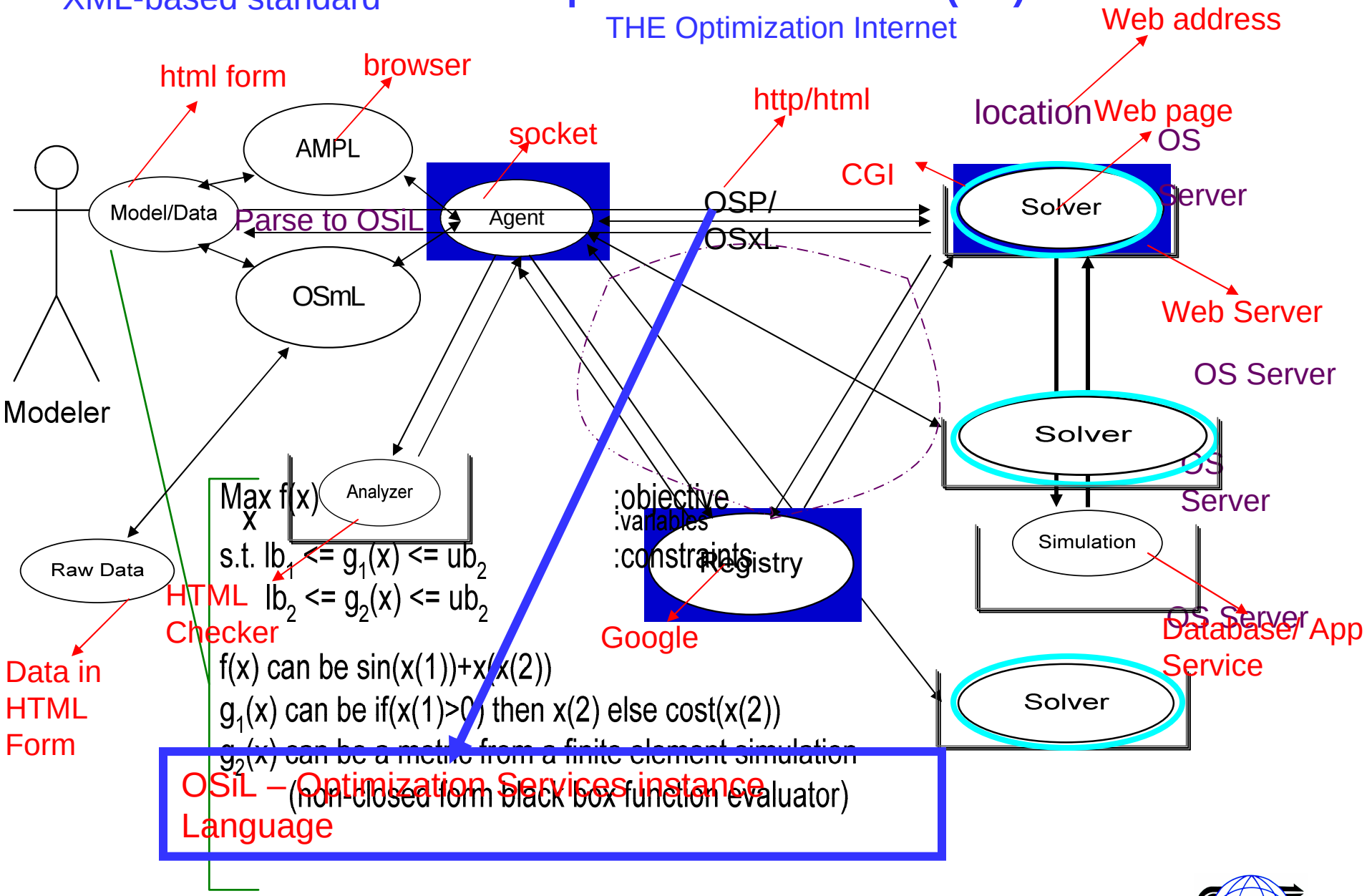
Introduction

- Optimization Services Components
  1. Modeling Language Environment (MLE) (e.g. AMPL, OSmL) (Fourer, Martin, People from MPL, GAMS, AIMMS, LINDO)
  2. Optimization Registries (e.g. The next generation NEOS) (Optimization Technology Center under Argonne and NU)
  3. Analyzers/Preprocessors (e.g. Mprobe, Dr. AMPL) (Chinneck, Orban etc.)
  4. Optimization Solvers (e.g. Lindo) (Mehrotra, Schrage/Lindo, Nocedal, All NEOS Solvers, COIN-OR)
  5. Simulation (e.g. Reliability, Finite Element Analysis) (Companies like Motorola, GM, IBM etc.)
  6. Communication Software Agent – OSAgent (NEOS/OTC, Kestrel/AMPL, Kestrel/GAMS)
  7. All of the above are communicating in common languages (Bradley, Gassman, Birge etc.)

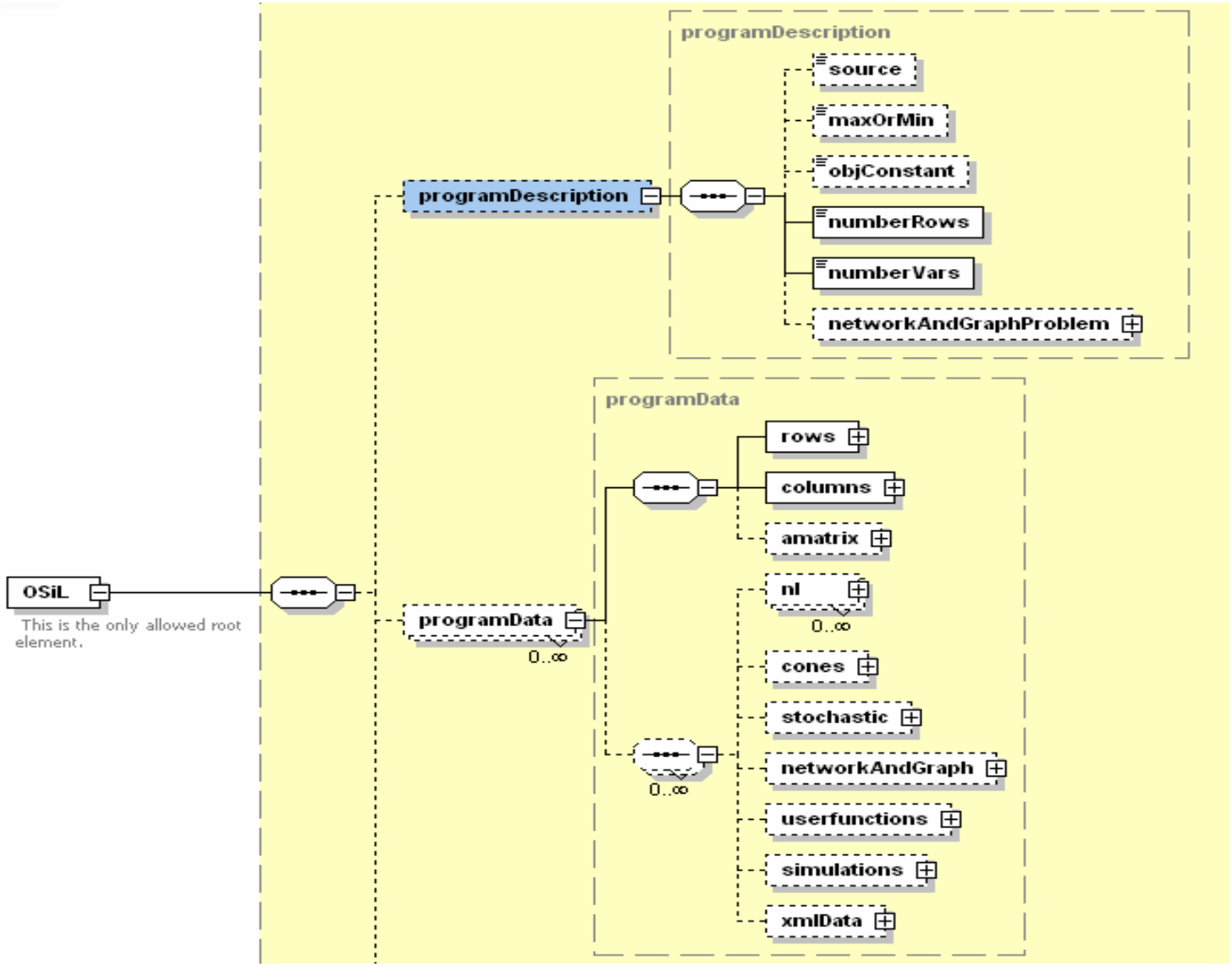
Optimization Services (OS)

XML-based standard

THE Optimization Internet



Optimization Services instance Language (OSiL) Schema

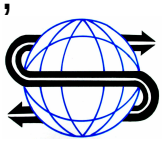


OSiL Schema

- Nonlinear Expressions and OSnL Schema
  - 220 major nodes (operators/operands)
  - Arithmetic operators, elementary functions, statistical and probability functions, constants, operands, logic and relational operator, trigonometric function, special elements
  - User defined functions
  - simulations
  - XMLData and xPath elements
  - Quadratic programming nodes
- OS API (OSiLReader/OSiLWriter) and OS Expression Tree
- Connecting to solvers
- All major optimization types supported

OSiL Schema: Supported Optimization Types

-Linear
-Mixed integer
-Bound constrained optimization
-General quadratic optimization
-Nonlinear unconstrained/constrained
-General mixed integer nonlinear
-General nonlinear with user-defined functions
-Global optimization
-General nonlinear with simulations (black-box functions)
-Optimization over simulation/nondifferentiable optimization
-General nonlinear with xml data (either within OSiL or remotely located)
-General nonlinear with data look up (XPath)
-Network and graph definition
-Network programming
-Constraint programming
-Semidefinite programming
-Semi-infinite programming
-Cone programming
-Complementarity problems
-Stochastic linear/nonlinear (distribution based recourse problem, scenario based recourse problem, chance constrained)
-Combinatorial optimization/Heuristic Optimization (TSP, MST, SP, MF, MCF, VRP, Set Covering, Coloring etc. etc.)



Business Values

- Market
  - Operations Research in general
  - Everything that involves optimization of some kind (Knowledge Management, Business Process Management, Supply Chain)
- Technology
  - An OR Internet of solvers that can easily be adapted within an Intranet
  - Goes beyond the VP System
  - New standards
  - A Complete Suite
  - Computing on Demand/Charge by Solution
  - Highly publicized Optimization Services Registry
  - Solve More Types of Problems
  - High entry barriers
- Strategy
  - Authorities
  - A Wide Level of Corporation and Support Being Formed around the Critical Core of Optimization Serivices
  - Control over Optimization Services Registry and Domain
  - Derived Consulting Services
  - Presences at all major conferences -- INFORMS, ICS IFORS (two sessions) etc.

NEOS Excerpts

- I have been using the NEOS site for about a month now. It has greatly helped in the work I am doing here at General Motors. 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. Because of this, internal GM customers have been able to see the benefits of optimization in this business context, and will most likely give the go ahead for a full scale development project. This will help GM, and will also help the solver vendor as they will probably get a sale at the end of the project. I have also built two prototype equilibrium models for two separate projects and solved them using the PATH solver. I wouldn't have even attempted going down this road had it not been for NEOS since GM has no complementarity solvers in-house to the best of my knowledge. As it is, I've been able to bring this class of models to bear on these problems, and, as a result, created a valuable solution for two different internal GM customers. These prototype models will also, hopefully, be eventually embedded into full blown enterprise applications which will, ultimately, result in a sale to the solver vendor.

- http://www-neos.mcs.anl.gov/neos/stories.html

Conclusion

- Sufficient Motivation for Optimization Services
- Optimization Services as the Internet for OR
  - Simple
  - Scalable
  - Standard
  - Smooth
- An OSxL Example – Optimization Services instance Language
  - Cleanly Designed from Scratch
  - Highly Extendable
  - State-of-art Expression Tree Design
  - Supports All Major Optimization Types
  - Built for Distributed and Decentralized Systems
  - Comes with Natively Designed OSiL APIs (OSiLReader/Writer)
  - Already Connected with Solvers