GrowingLeaf: Supporting Requirements Evolution over Time

Alicia M. Grubb, Gary Song, Marsha Chechik
{amgrubb, gary, chechik}@cs.toronto.edu

iStar'16 - Sep. 12, 2016

UNIVERSITY OF TORONTO

Problem

- Assumptions of early-phase requirements modeling:
  - ➡ all model elements have a value
  - ➡ model values are constant
- In reality intentions and relationships in the environments are not constant.

Example Questions

1. Is it possible to satisfy *Goal-A* and partially satisfy *Goal-B*? and how?
2. How does completing *Task-A* and *Task-B* but not *Task-C* affect the top level goals?
3. How do changes in *Actor-A*'s dependums affect the *Actor-A*'s root-level goals over time?
4. Which possible scenarios always satisfy *Goal-A* even if *Goal-B* becomes denied in the future?
5. Does the satisfaction order of *Goal-C* and *Goal-D* matter?

Example Questions

1. Is it possible to satisfy *Goal-A* and partially satisfy *Goal-B*? and how?
2. How does completing *Task-A* and *Task-B* but not *Task-C* affect the top level goals?
3. How do changes in *Actor-A*'s dependums affect the *Actor-A*'s root-level goals over time?
4. Which possible scenarios always satisfy *Goal-A* even if *Goal-B* becomes denied in the future?
5. Does the satisfaction order of *Goal-C* and *Goal-D* matter?

Example Questions

1. Is it possible to satisfy *Goal-A* and partially satisfy *Goal-B*? and how?
2. How does completing *Task-A* and *Task-B* but not *Task-C* affect the top level goals?
3. How do changes in *Acto* s affect the *Actor A*
4. Whic *Goal-A* even i denied in the future?
5. Does the satisfaction order of *Goal-C* and *Goal-D* matter?

Use Qualitative Evaluation Labels with Forward Analysis and Backward Analysis

Example Questions

1. Is it possible to satisfy *Goal-A* and partially satisfy *Goal-B*? and how?
2. How does completing *Task-A* and *Task-B* but not *Task-C* affect the top level goals?
3. How do changes in *Actor-A*'s dependums affect the *Actor-A*'s root-level goals over time?
4. Which possible scenarios always satisfy *Goal-A* even if *Goal-B* becomes denied in the future?
5. Does the satisfaction order of *Goal-C* and *Goal-D* matter?

Example Questions

1. Is it possible to satisfy *Goal-A* and partially satisfy *Goal-B*? and how?
2. How does completing *Task-A* and *Task-B* but not *Task-C* affect the top level goals?
3. How do changes in *Actor-A*'s dependums affect the *Actor-A*'s root-level goals over time?
4. Which possible scenarios always satisfy *Goal-A* even if *Goal-B* becomes denied in the future?
5. Does the satisfaction order of *Goal-C* and *Goal-D* matter?

Example Questions

1. Is it possible to satisfy *Goal-A* and partially satisfy *Goal-B*? and how?
2. How does completing *Task-A* and *Task-B* but not *Task-C* affect the top level goals?
3. How do changes in *Actor-A*'s dependums affect the *Actor-A*'s root-level goals over time?
4. Which possible scenarios always satisfy *Goal-A* even if *Goal-B* becomes denied in the future?
5. Does the satisfaction order of *Goal-C* and *Goal-D* matter?

Example Questions

1. Is it possible to satisfy *Goal-A* and partially satisfy *Goal-B*? and how?

2. How does completing *Task-A* and *Task-B* but not *Task-C* affect the top level goals?

3. How do changes in *Actor-A*'s dependums affect the *Actor-A*'s root-level goals over time?

4. Which possible scenarios always satisfy *Goal-A* even if *Goal-B* becomes denied in the future?

5. Does the satisfaction order of *Goal-C* and *Goal-D* matter?

Contributions

Provide tooling to:

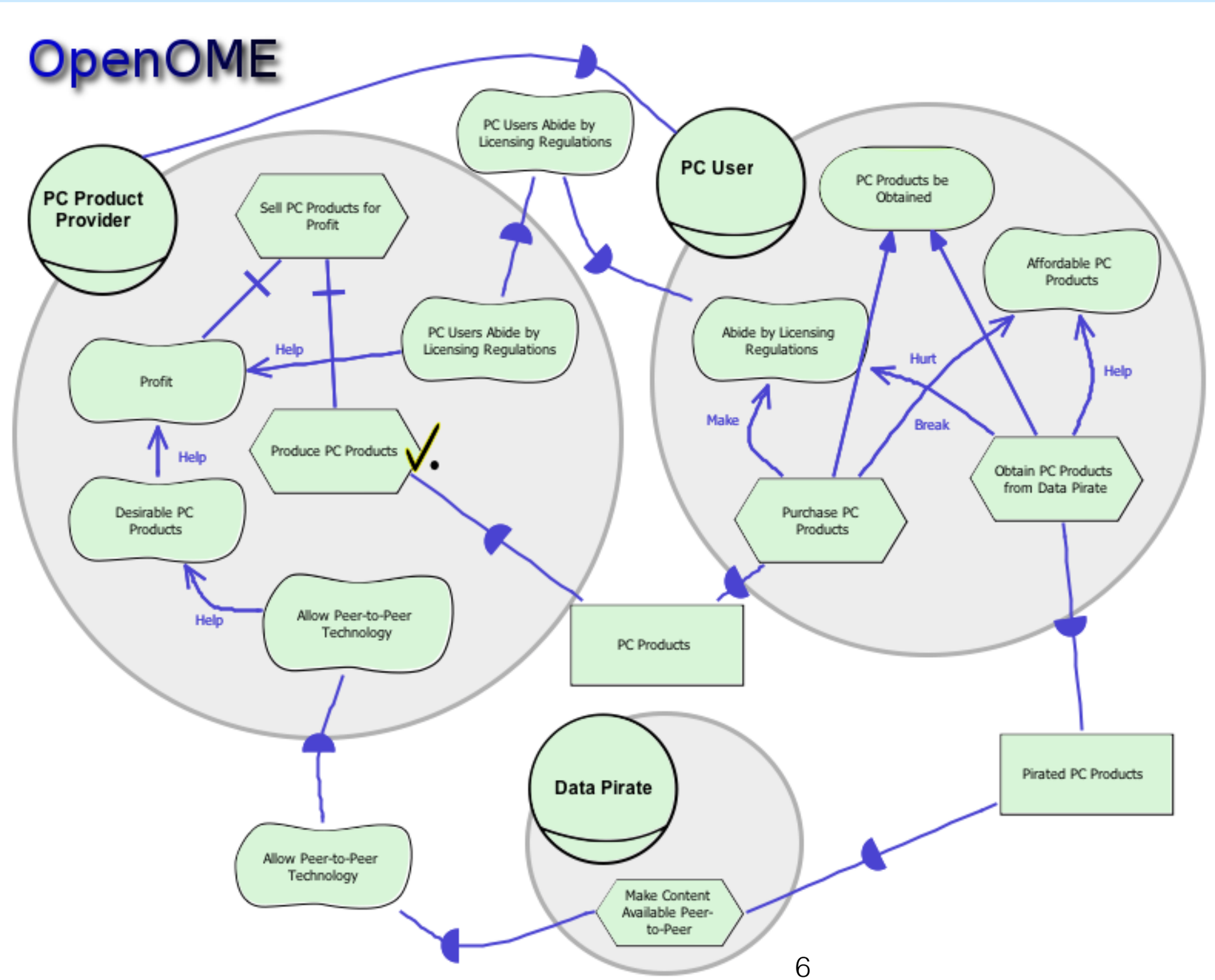
- enrich goal models intentions with dynamically changing evaluation

- analyze the impacts of dynamically changing intentions on decision making

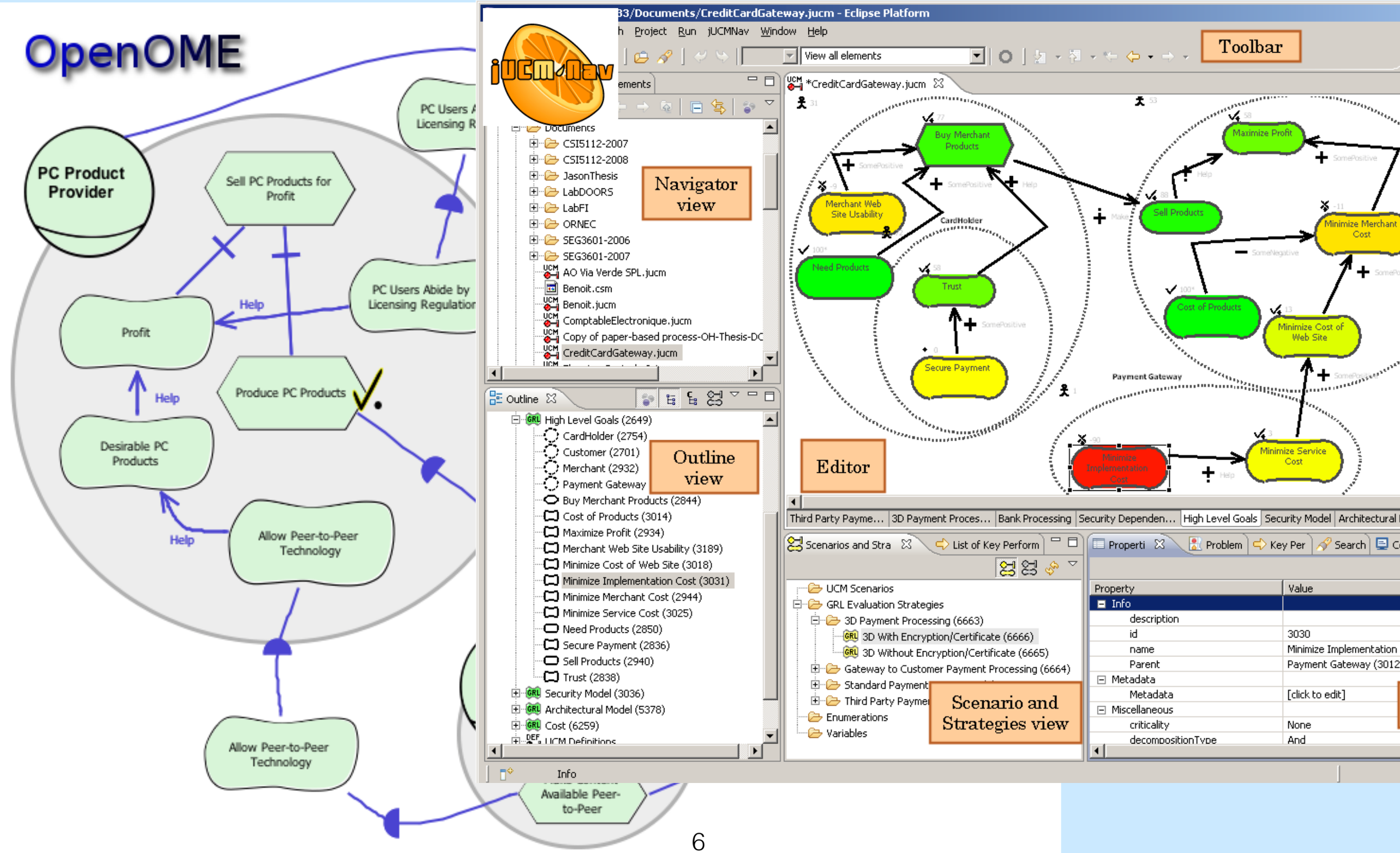
Why another modeling tool?

- Why another modeling tool?

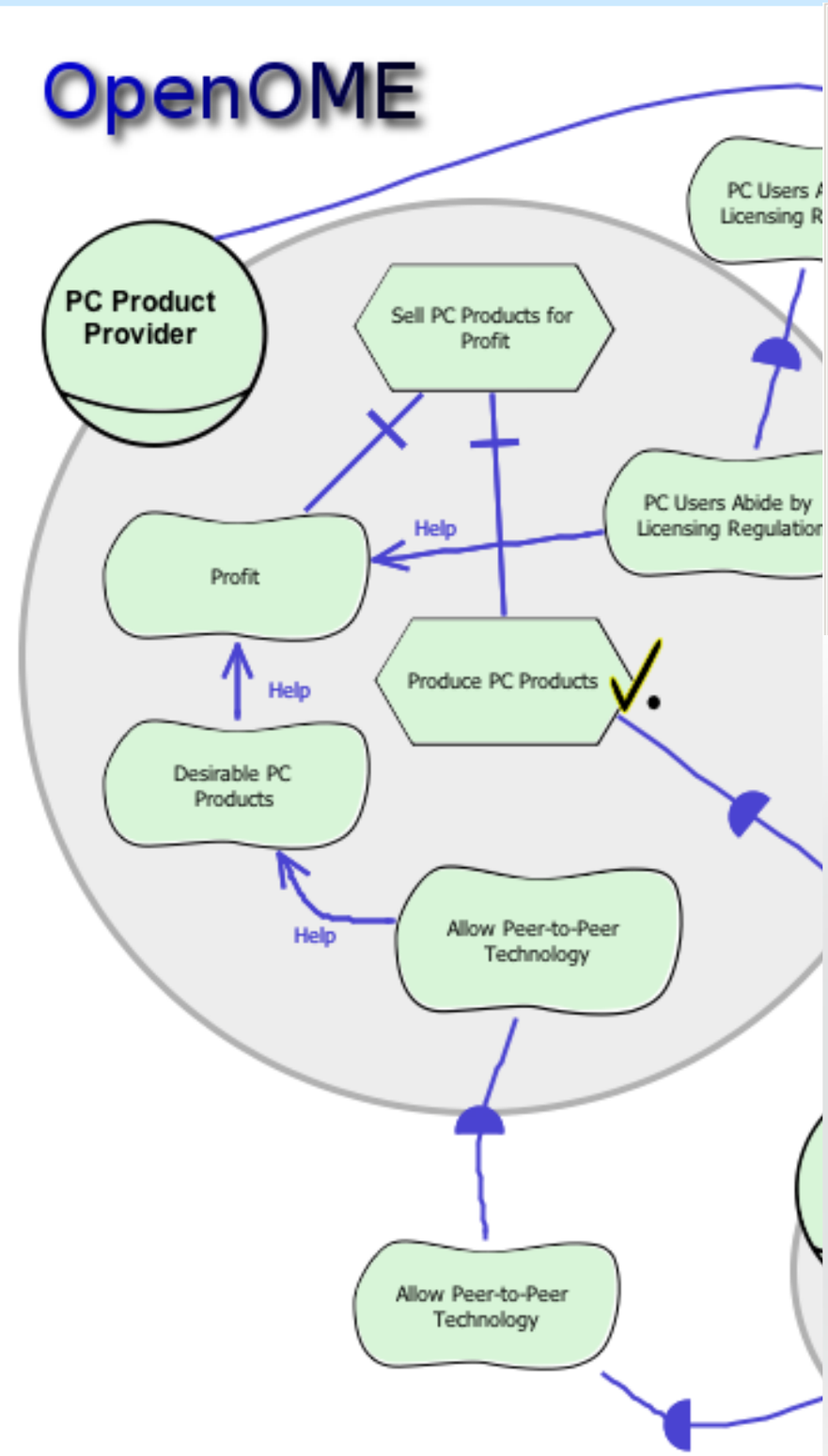
Why another modeling tool?

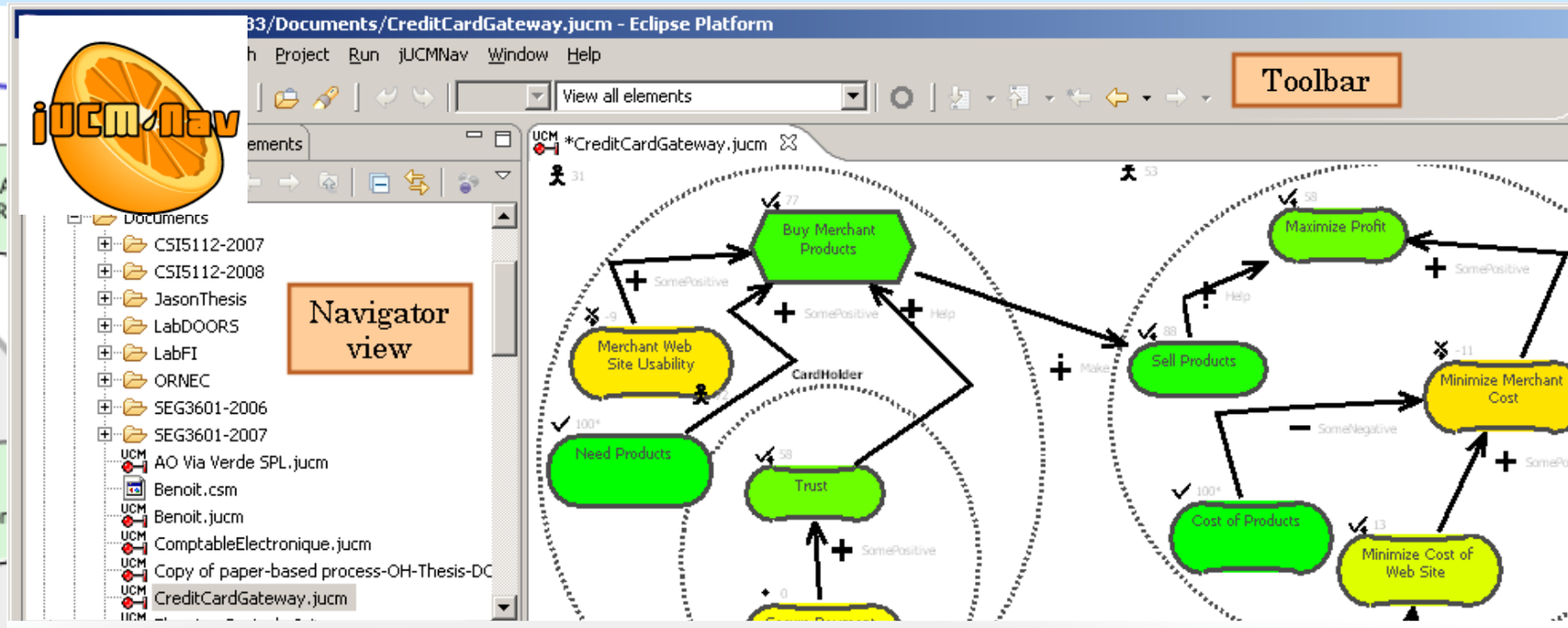


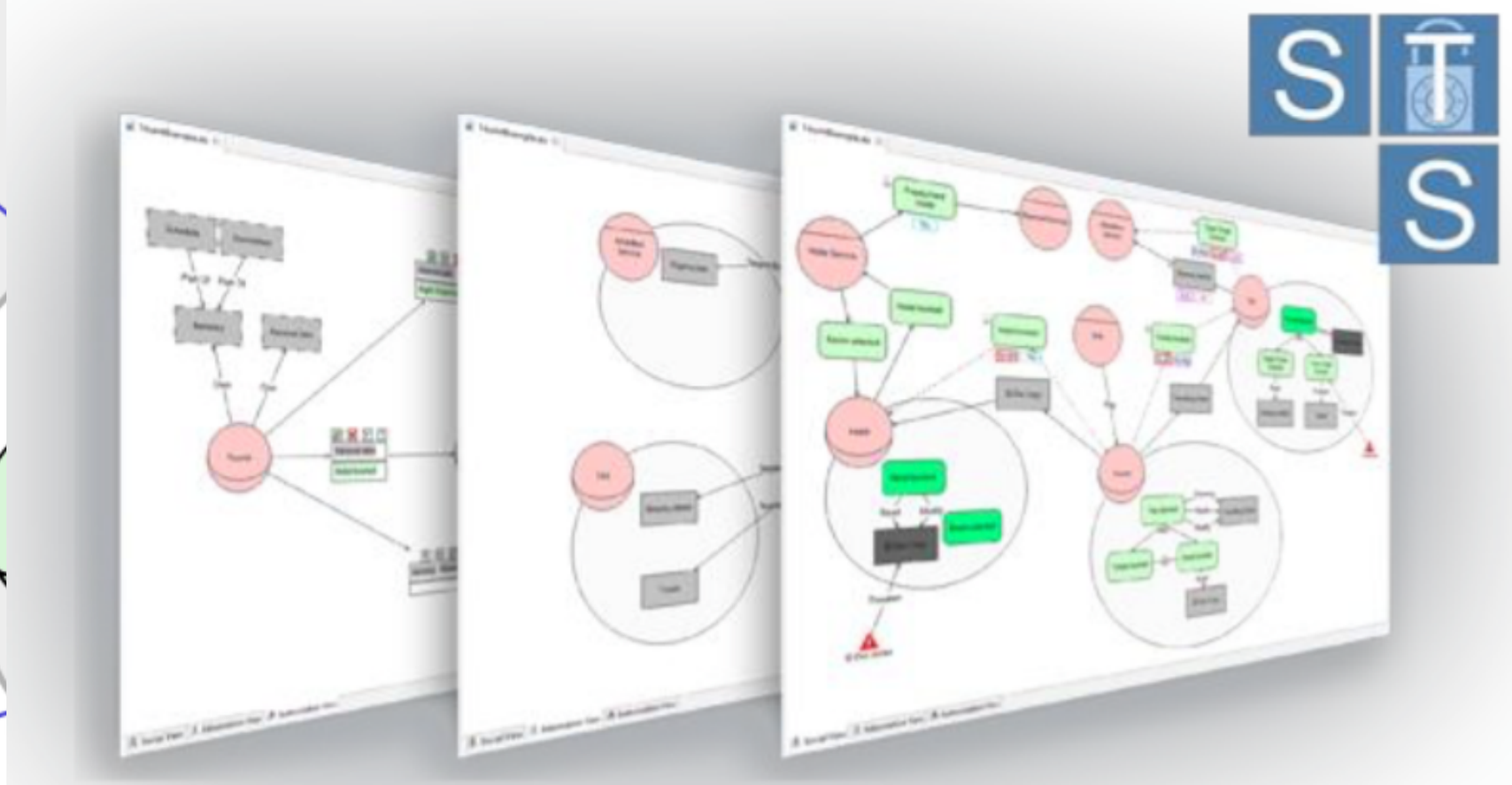
Why another modeling tool?



Why another modeling tool?







Why another modeling tool?

OpenOME

PC Product Provider

Sell PC Products for Profit

PC Users Licensing R

PC Users Abide by Licensing Regulatior

Profit

Help

jUCM-Nav

Toolbar

Navigator view

Documents
CSI5112-2007
CSI5112-2008
JasonThesis
LabDOORS
LabFI
ORNEC
SEG3601-2006
SEG3601-2007
AO Via Verde SPL.jucm
Benoit.csm
Benoit.jucm
ComptableElectronique.jucm
Copy of paper-based process-OH-Thesis-DC
CreditCardGateway.jucm

*CreditCardGateway.jucm

Buy Merchant Products
Merchant Web Site Usability
Need Products
Trust
CardHolder
Sell Products
Maximize Profit
Minimize Merchant Cost
Cost of Products
Minimize Cost of Web Site

tlsam_examle.graffle

Canvases
Models
BUSINESS
APPLICATION
PHYSICAL

Business Layer

Application Layer

Physical Layer

MUSER Control Panel

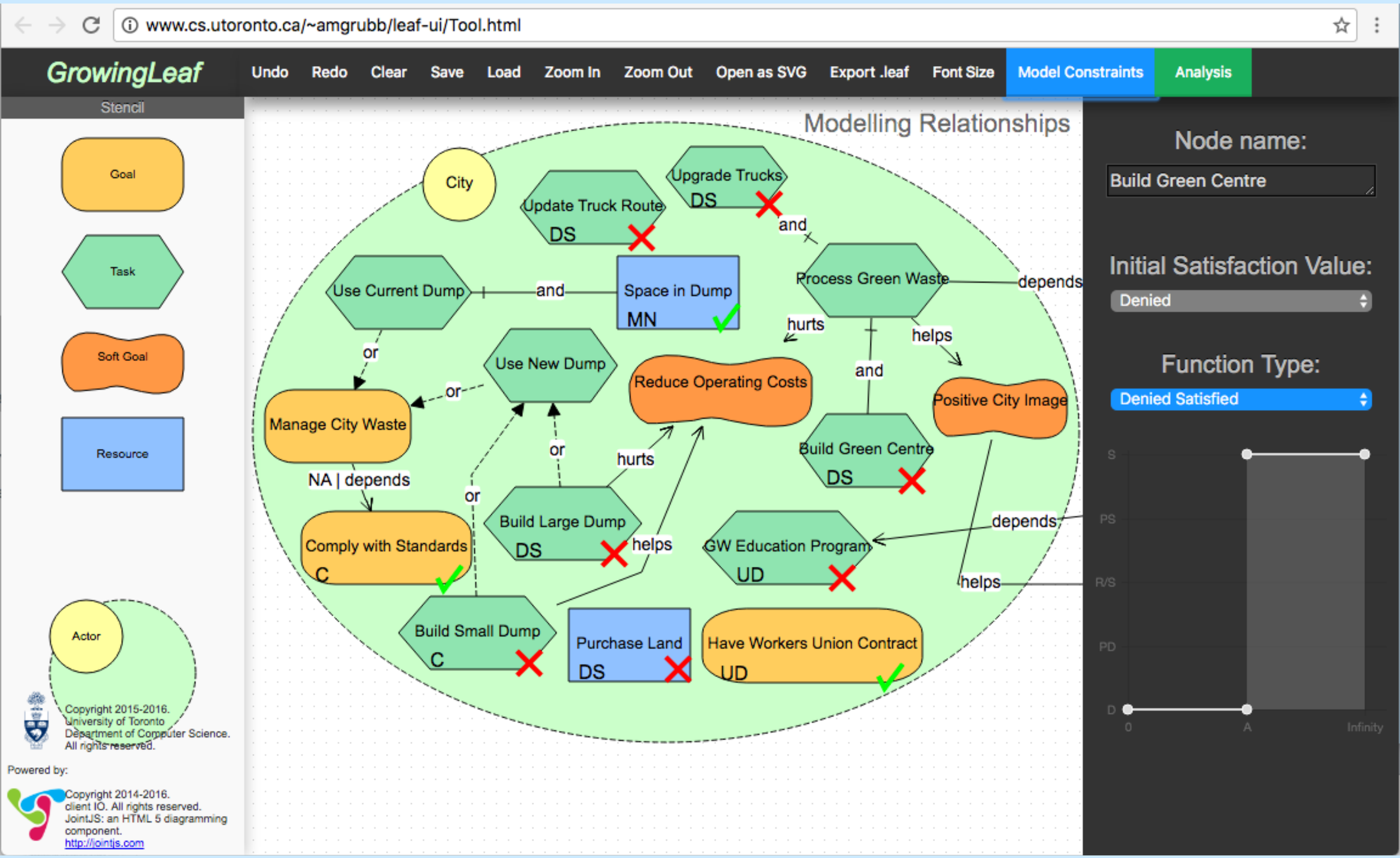
Import parameters
Import Source
Selected element
From files
Model Type
Requirement mod
Trust mode
Import
Delete all
Print mode
Common operation parameters
Layer
All layers
Application
Busines
Physical
Object
All models
Selected mode
Step1: Refine
Step2: Simplify
Step3: Operationalize
Step 4: Transfer security concern
Refinement Mode
One-step
Exhaustive
Refinement Dimension
Attribute
Asset
Interval

STS

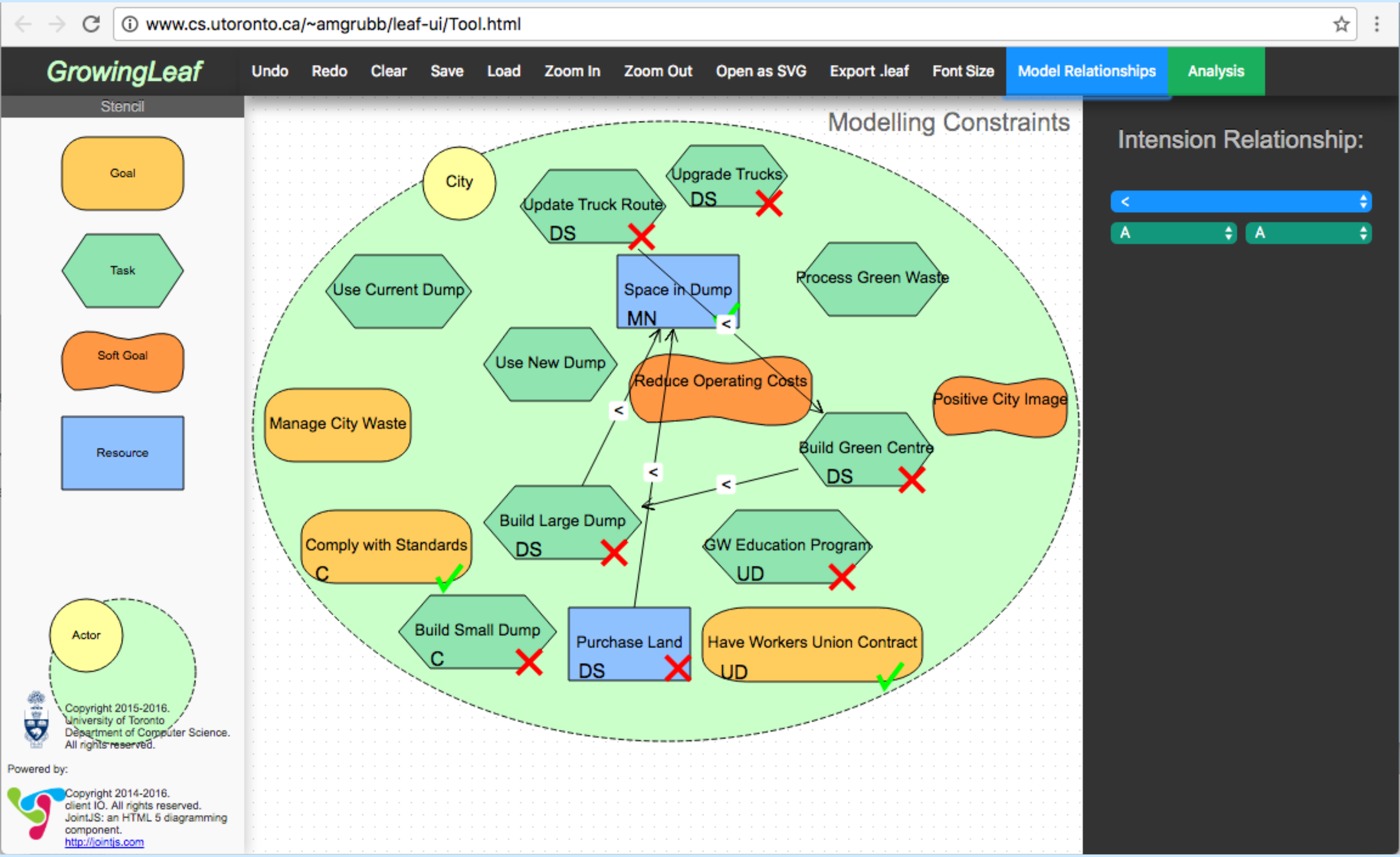
Why another modeling tool?

- Surveyed previous tools
    - Extend their iStar meta-model
    - Add icons/labels on top of intentions
- Web-based tool
    - Framework vs. self-built
    - Multi-view vs. multi-tab

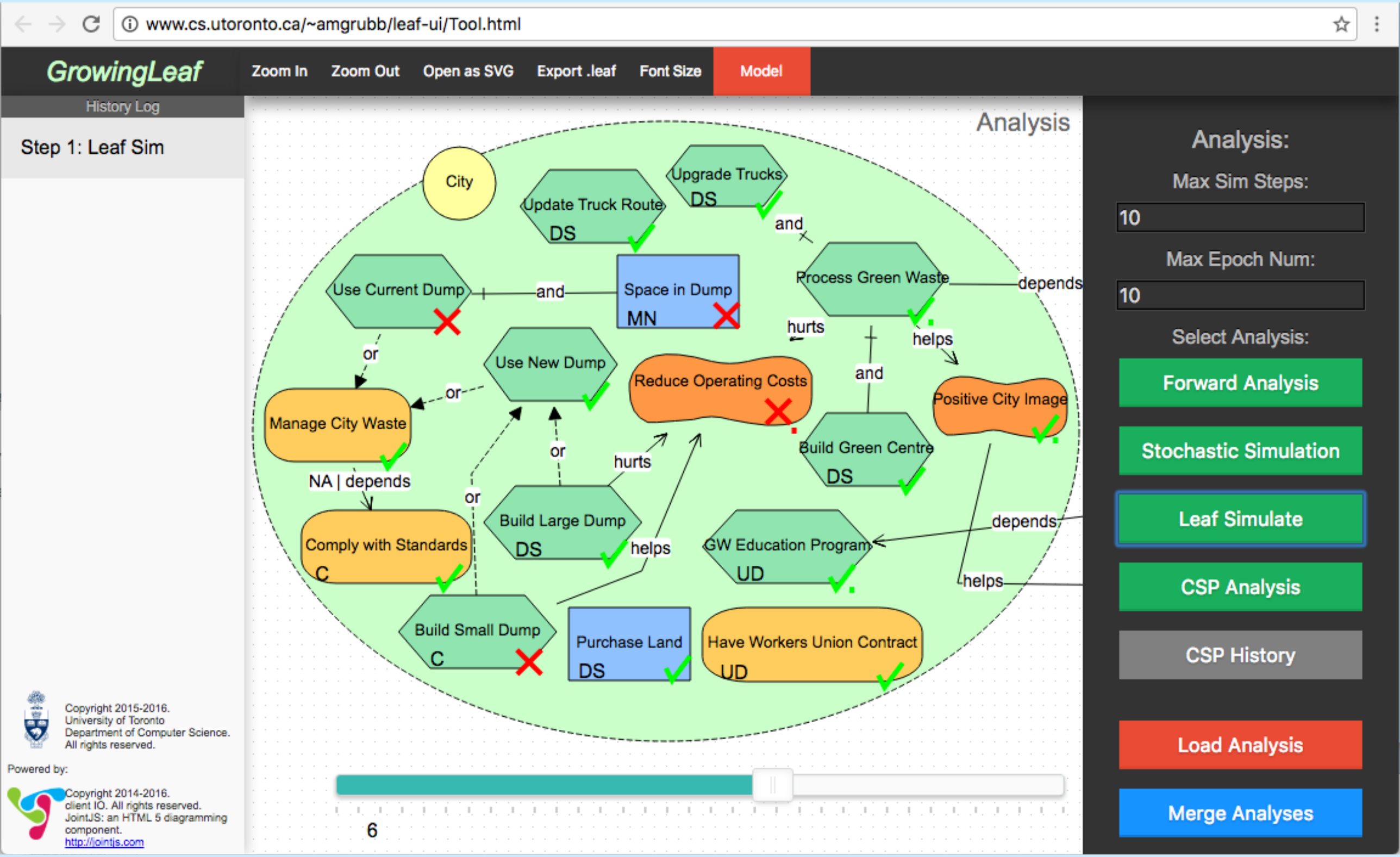
Introducing GrowingLeaf



Introducing GrowingLeaf



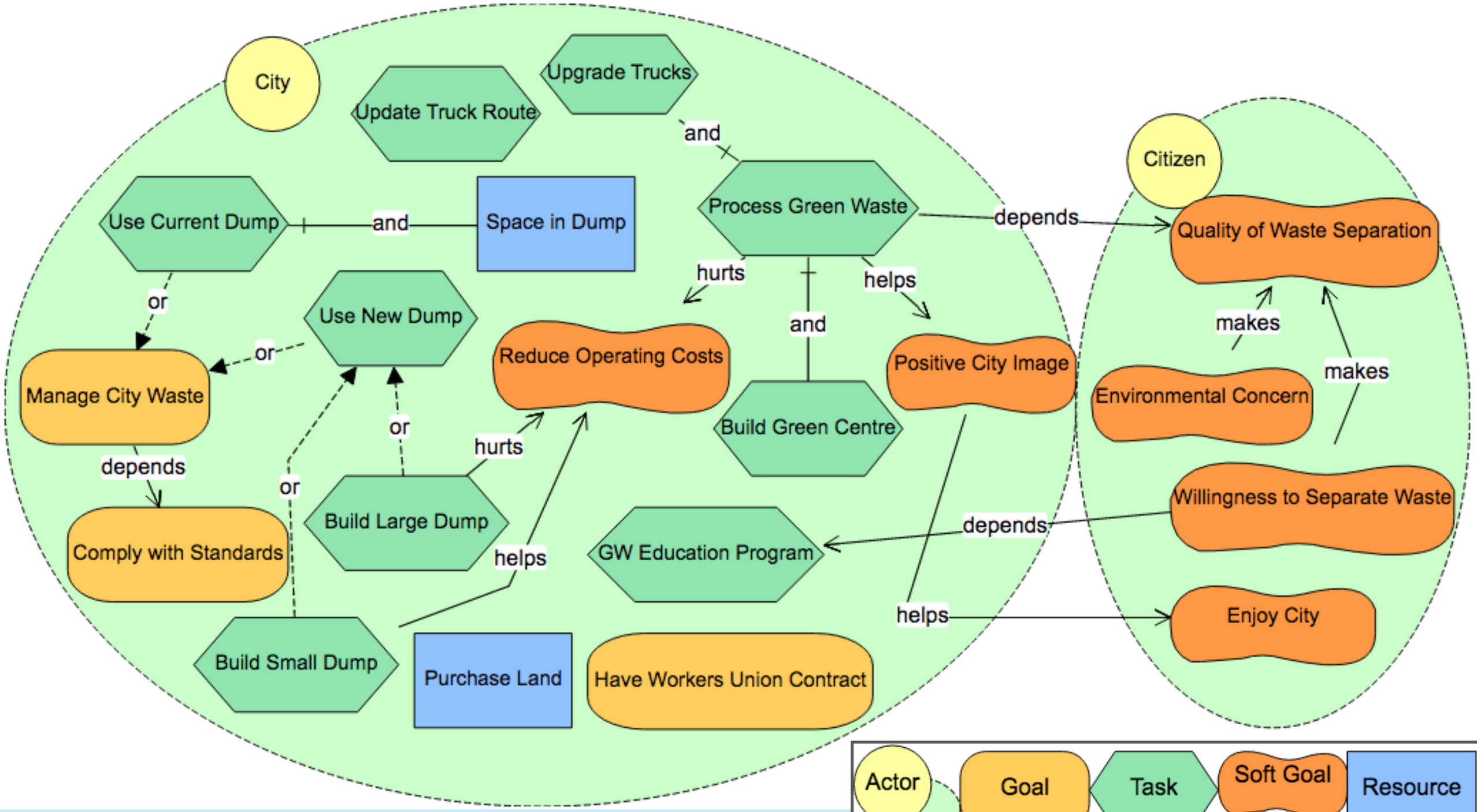
Introducing GrowingLeaf



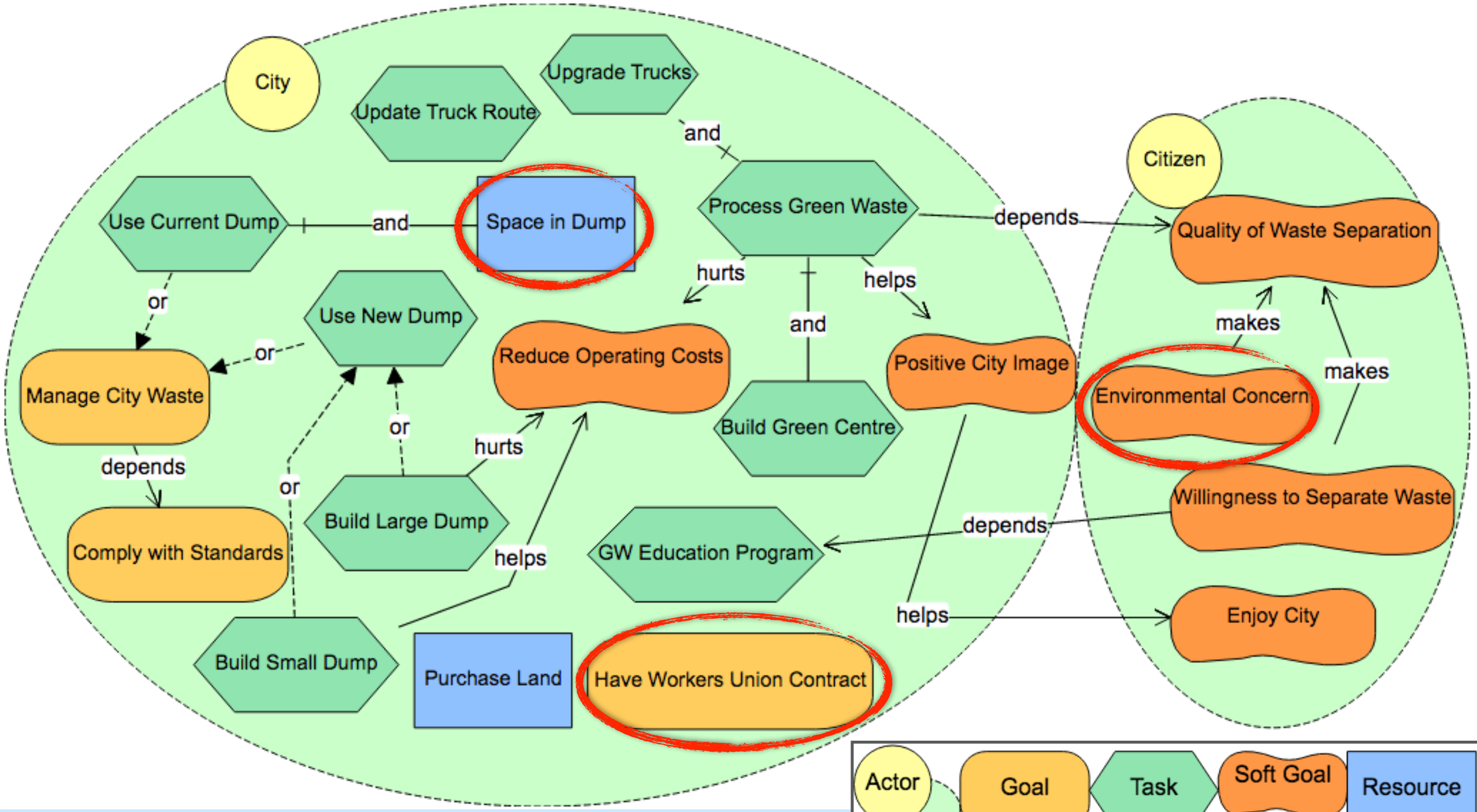
Outline

- Modeling Problem and Tool Justification
- Tool Introduction
- **Dynamic Intentions and Analysis**
- Tool Functionality
- Discussion and Validation
- Status and Future Work

Modeling Dynamic Intentions



Modeling Dynamic Intentions



Modeling Dynamic Intentions

Stochastic (R)

Patterns:

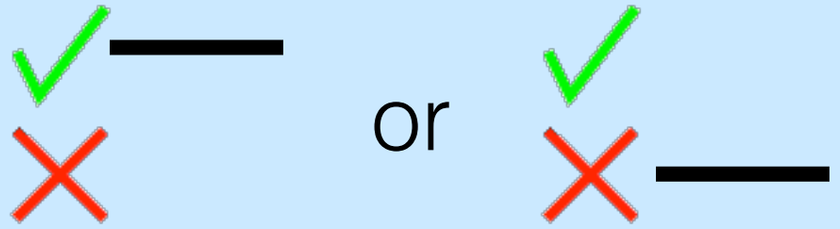
Examples:

Modeling Dynamic Intentions

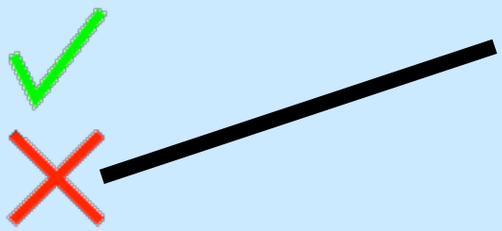
Elementary Functions

Stochastic (R):

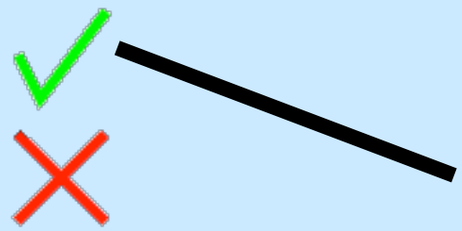
Constant (C):



Increase (I):



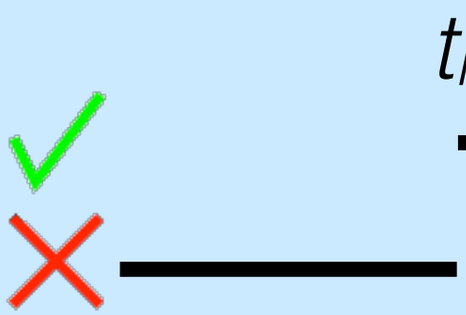
Decrease (D):



Modeling Dynamic Intentions

Denied-Satisfied (DS)

Patterns:



Examples:

Modeling Dynamic Intentions

Denied-Satisfied (DS)

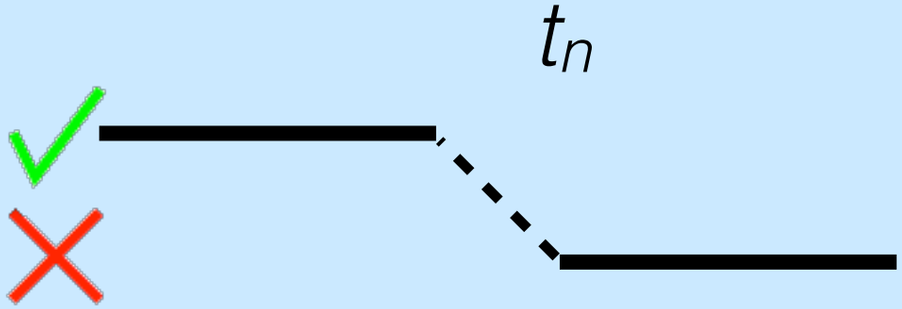
Patterns:

Examples:

Modeling Dynamic Intentions

Monotonic Negative (MN)

Patterns:



Examples:

Space in Dump

Common Compound Functions

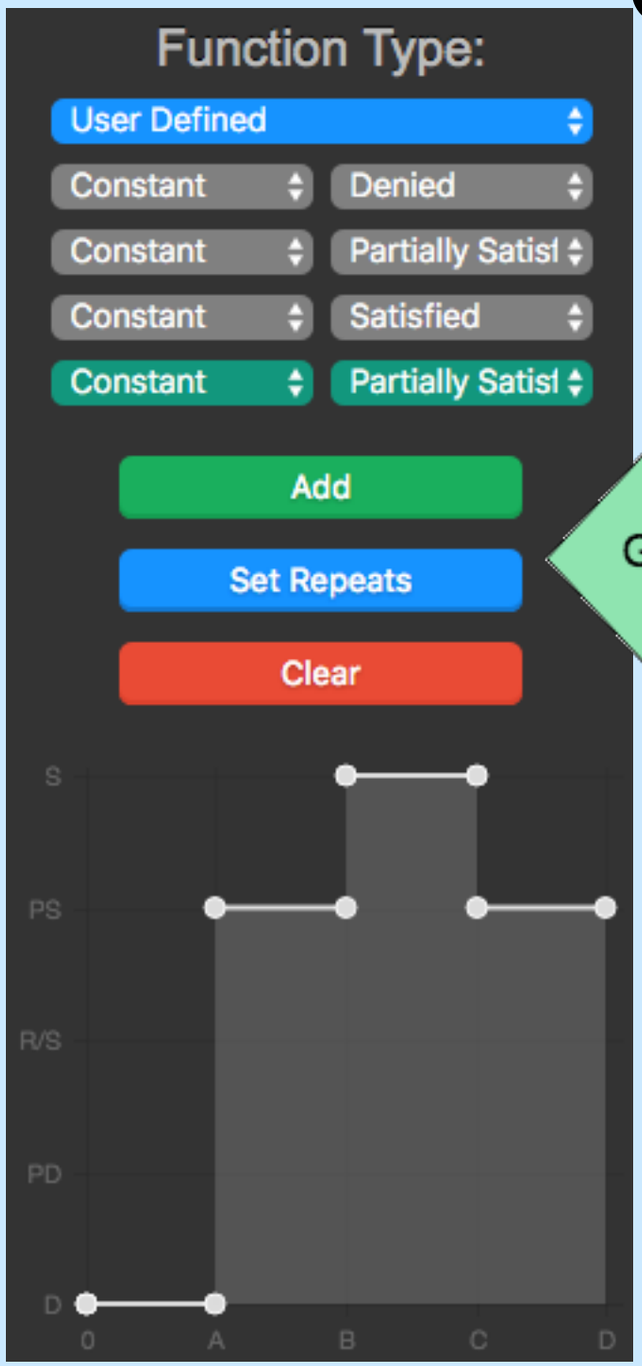
Denied-Satisfied (DS)	the satisfaction evaluation remains *Denied* until $\mathbf{t_i}$ and then remains *Satisfied*
Monotonic Negative (MN)	changes in satisfaction evaluation become "less true" to a *maxValue* at $\mathbf{t_i}$ and then remains constant at *constantValue*

Common Compound Functions

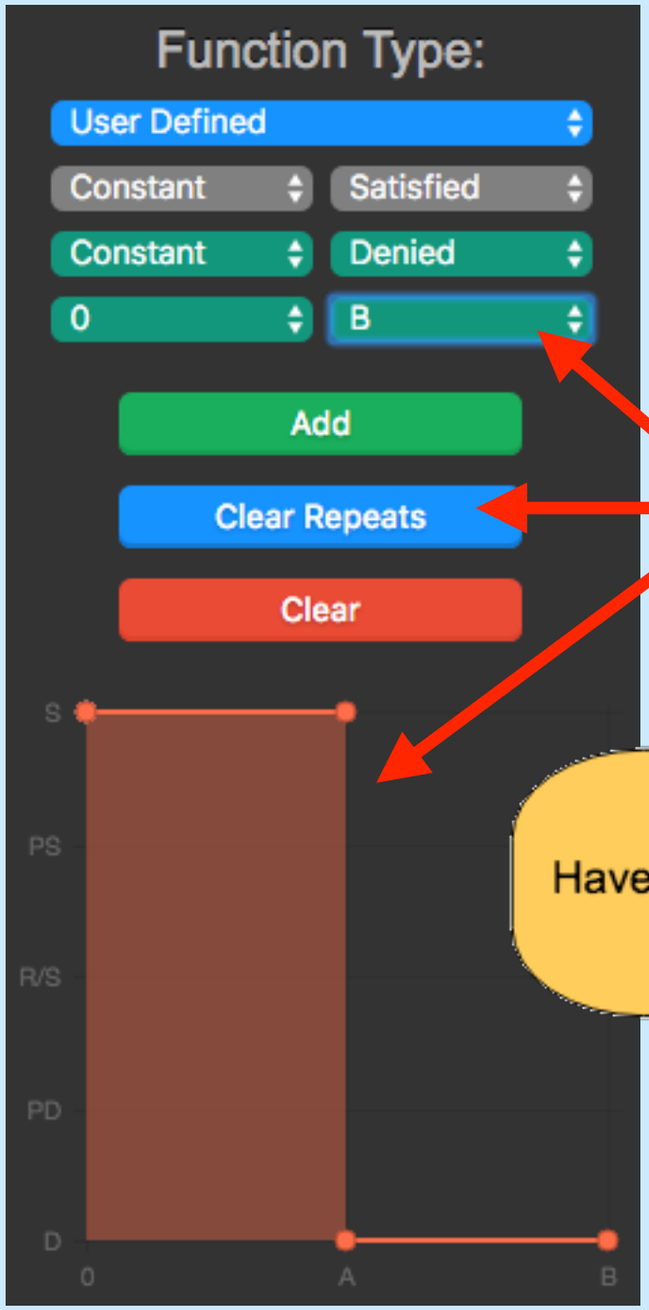
Satisfied-Denied (SD)	the satisfaction evaluation remains *Satisfied* until $\mathbf{t_i}$ and then remains *Denied*
Denied-Satisfied (DS)	the satisfaction evaluation remains *Denied* until $\mathbf{t_i}$ and then remains *Satisfied*
Stochastic-Constant (RC)	changes in satisfaction evaluation are stochastic or random until $\mathbf{t_i}$ and then remains constant at *constantValue*
Constant-Stochastic (CR)	the satisfaction evaluation remains constant at *constantValue* until $\mathbf{t_i}$ and then changes in evaluation are stochastic or random
Monotonic Positive (MP)	changes in satisfaction evaluation become "more true" to a *maxValue* at $\mathbf{t_i}$ and then remains constant at *constantValue*
Monotonic Negative (MN)	changes in satisfaction evaluation become "less true" to a *maxValue* at $\mathbf{t_i}$ and then remains constant at *constantValue*

Modeling Dynamic Intentions

User Defined (UD)



GW Education Program



Repeating Function

Have Workers Union Contract

Analysis Strategies

(Strategy 1: Leaf Simulation) create a **random path** given initial states in the model

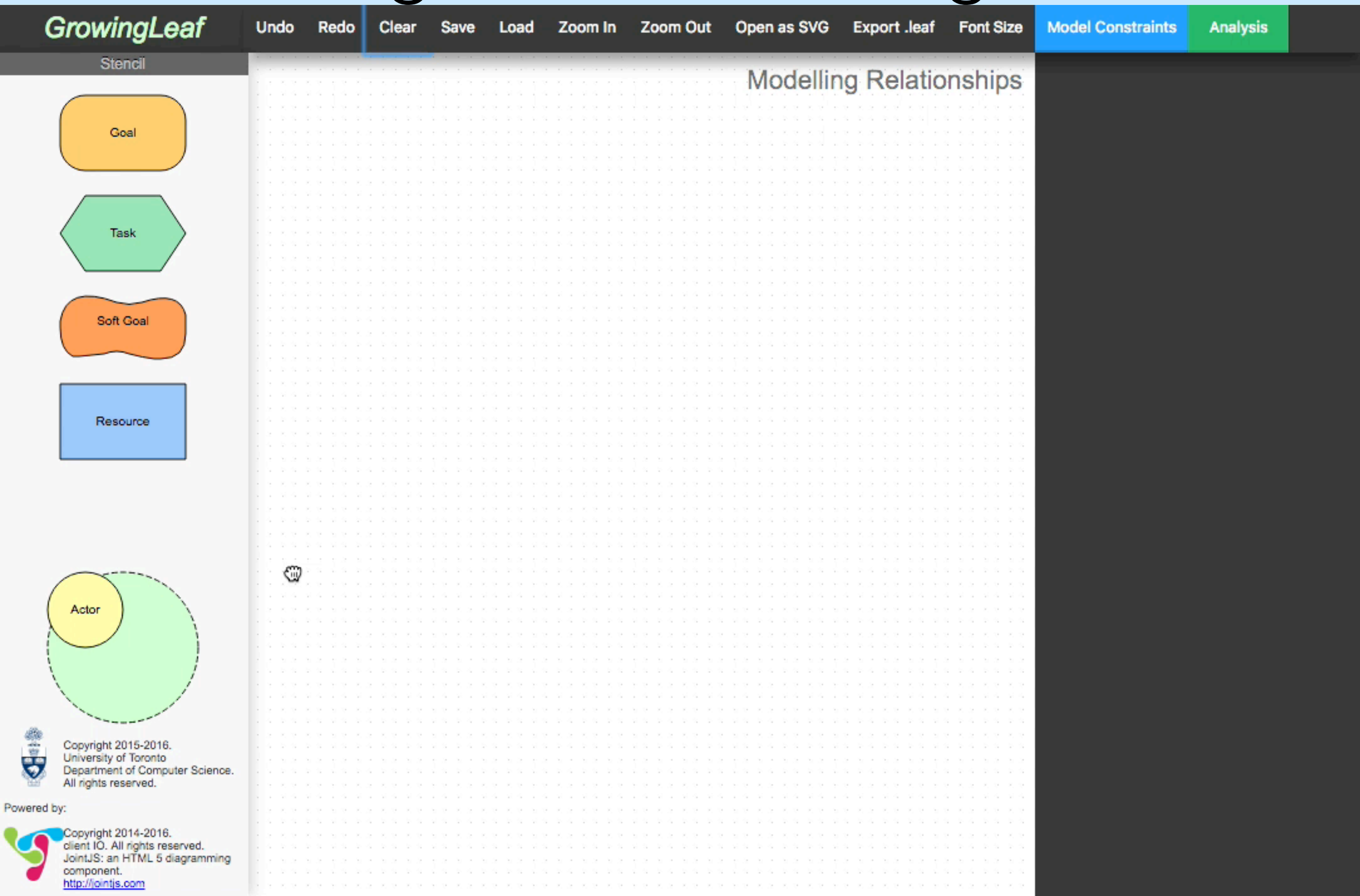
(Strategy 2: CSP Analysis) create a path given **desired properties** of the **intermediate state** (with optional properties over the initial or final state)

(Strategy 3: CSP History) create a path which is **different than the previously seen path** over the same constraints

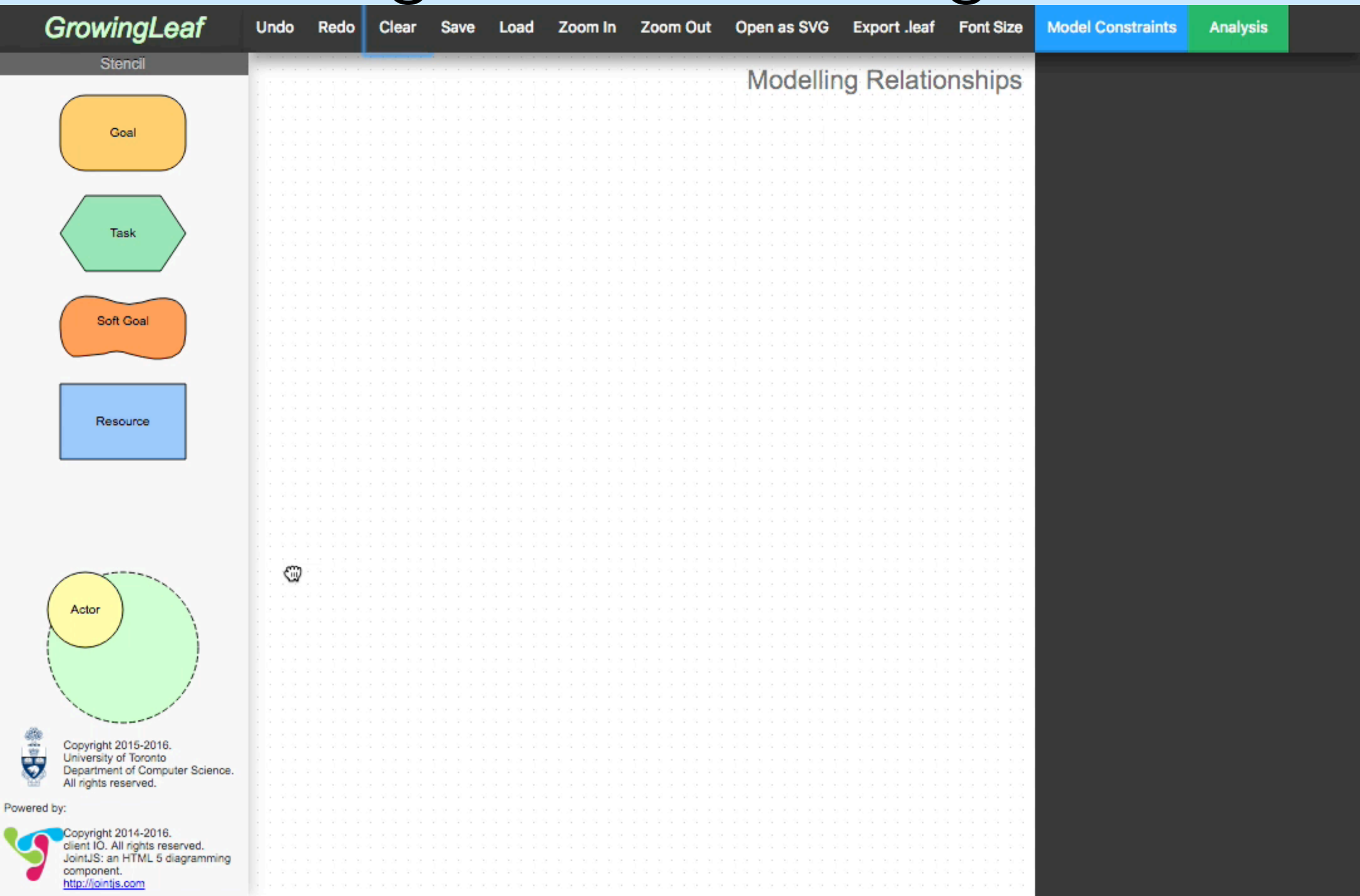
Outline

- Modeling Problem and Tool Justification
- Tool Introduction
- Dynamic Intentions and Analysis
- **Tool Functionality**
- Discussion and Validation
- Status and Future Work

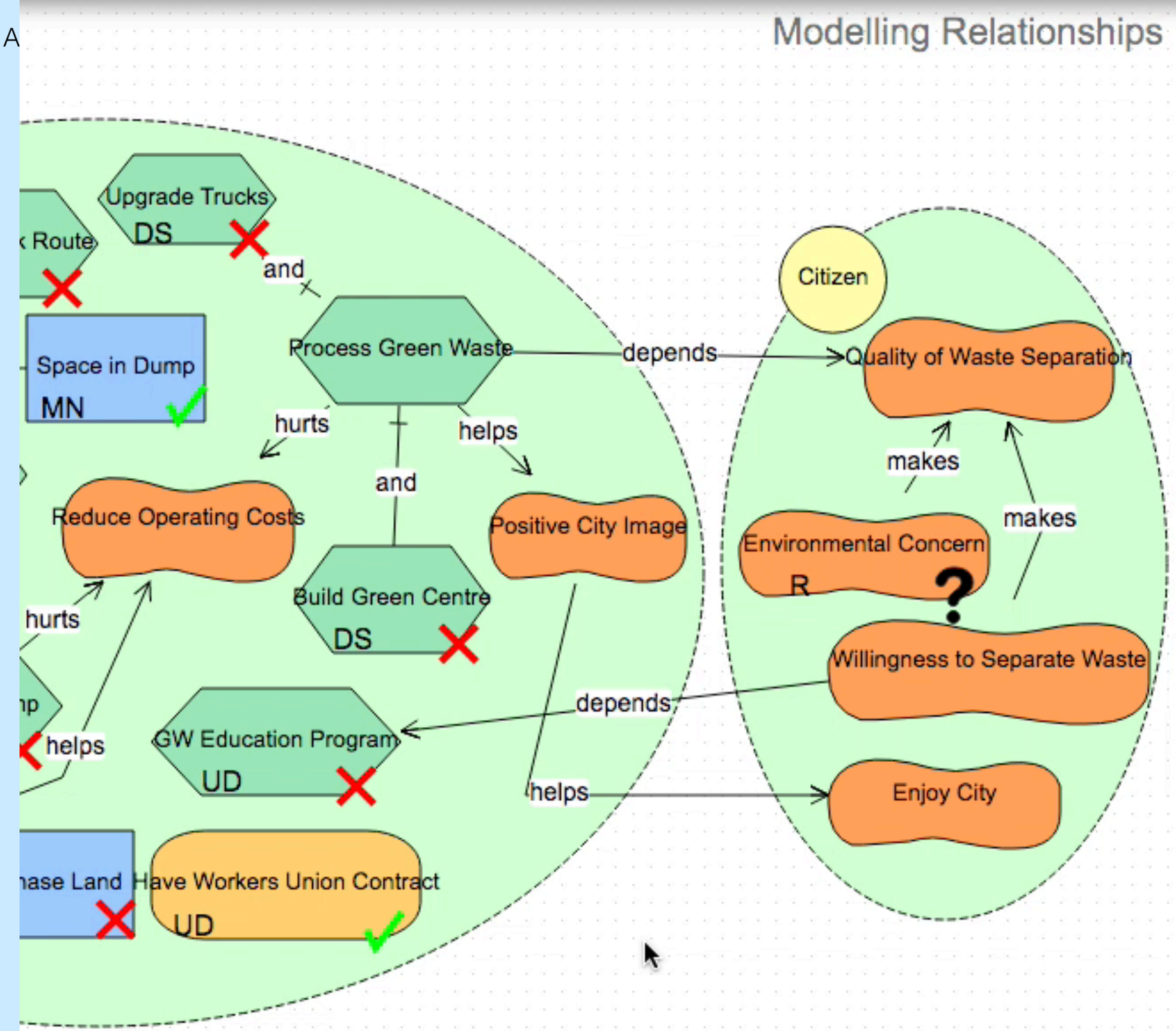
GrowingLeaf - Modeling Demo



GrowingLeaf - Modeling Demo



Modelling Relationships



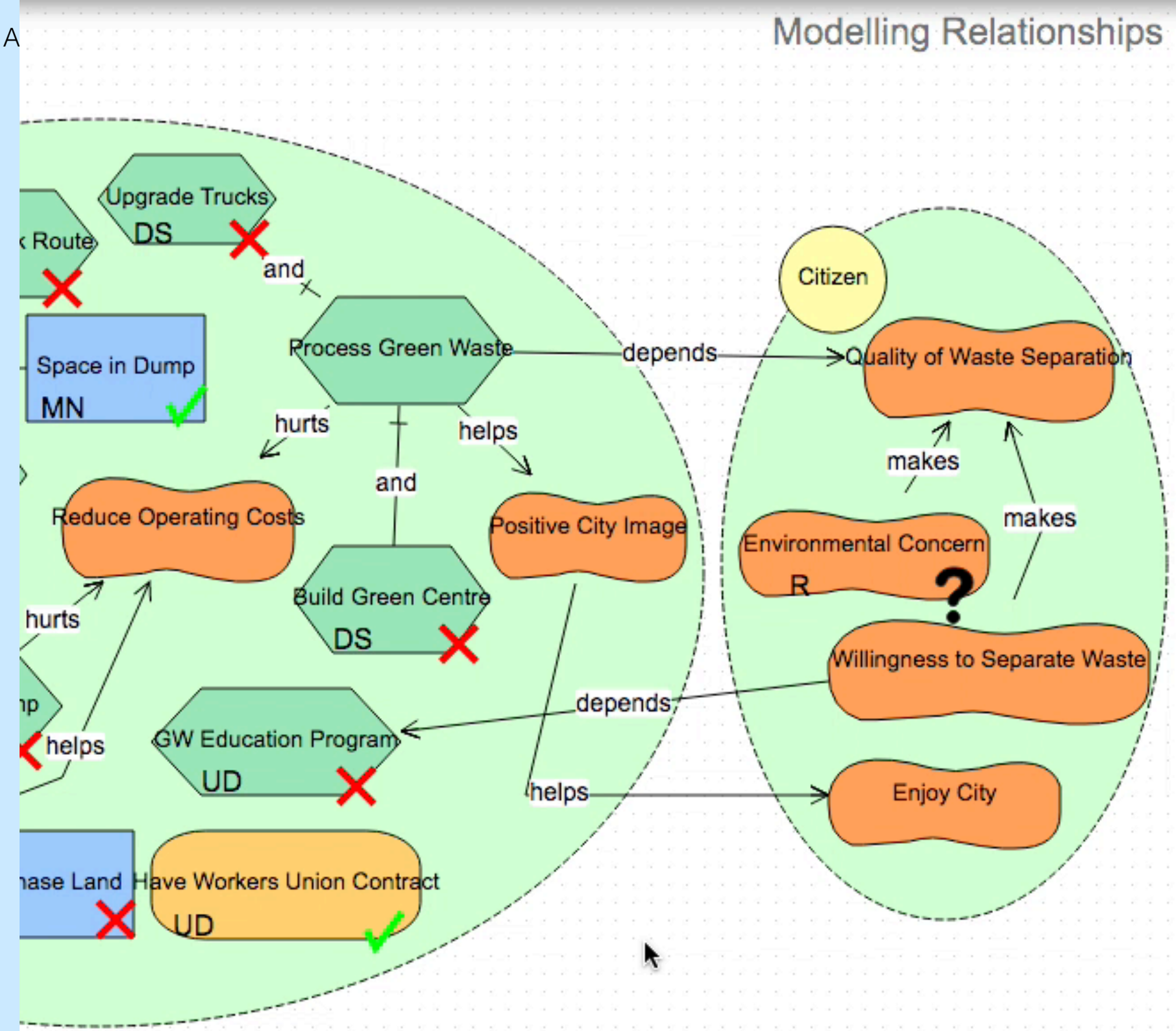
Node name:

Positive City Image

Initial Satisfaction Value:

None

Modelling Relationships



Node name:

Positive City Image

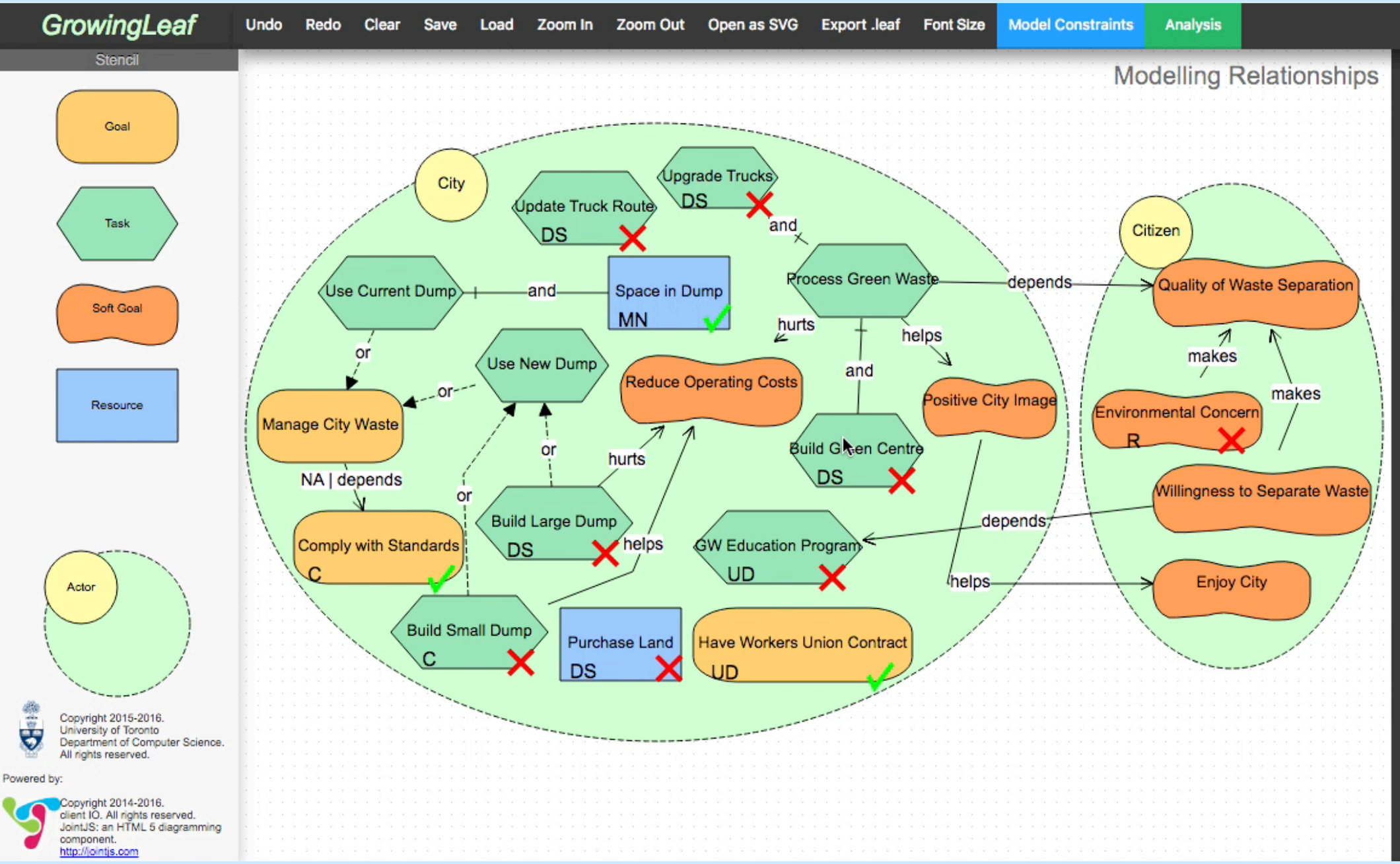
Initial Satisfaction Value:

None

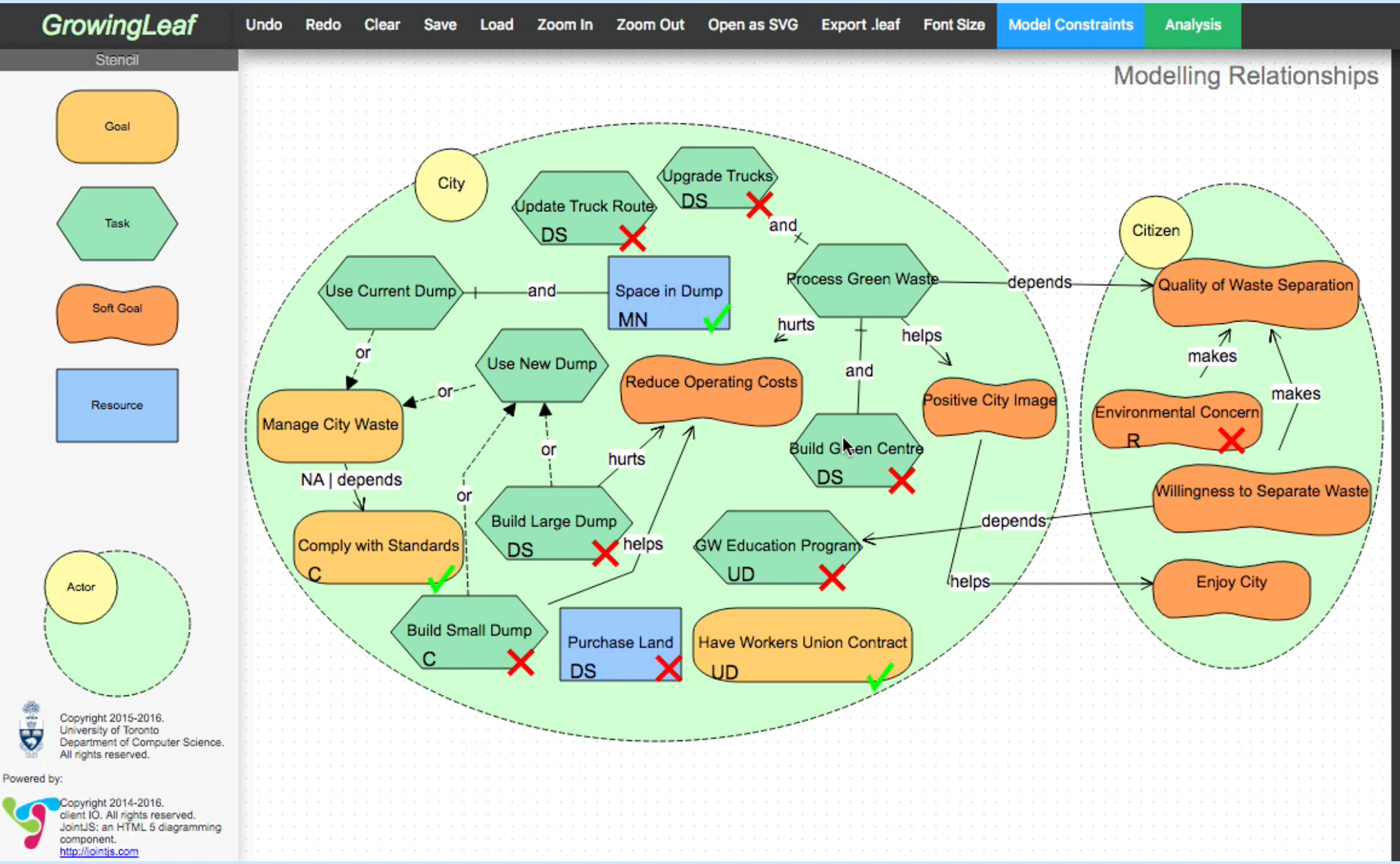
GrowingLeaf - Modeling Demo Summary

- Drag and drop interface
- Naming and adding elements
- Loading, saving, exporting, and zooming models
- Resizing label fonts
- Changing initial satisfaction values
- Changing dynamic function types
- Creating User Defined functions

GrowingLeaf - Analysis Demo



GrowingLeaf - Analysis Demo

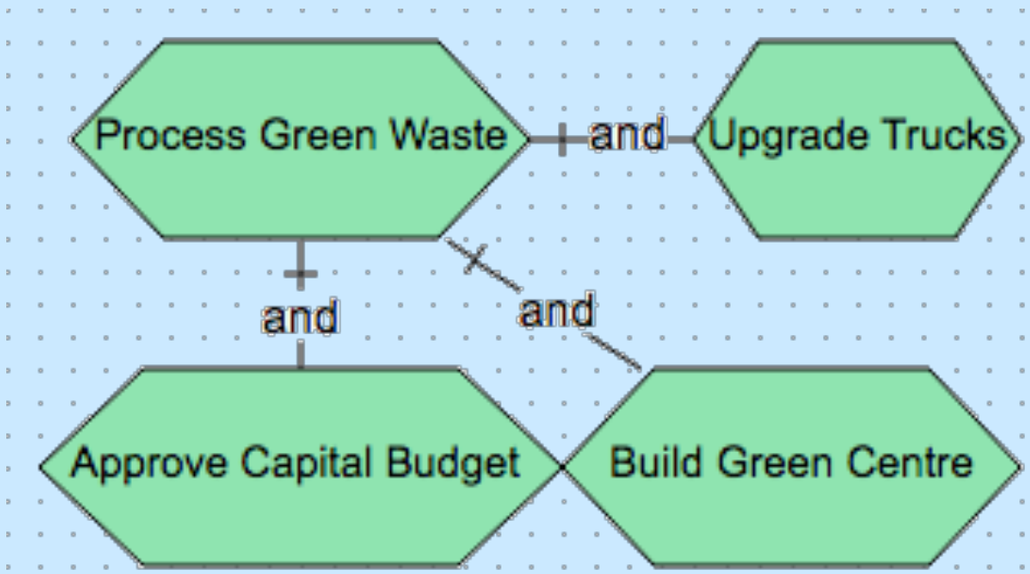


GrowingLeaf - Analysis Demo Summary

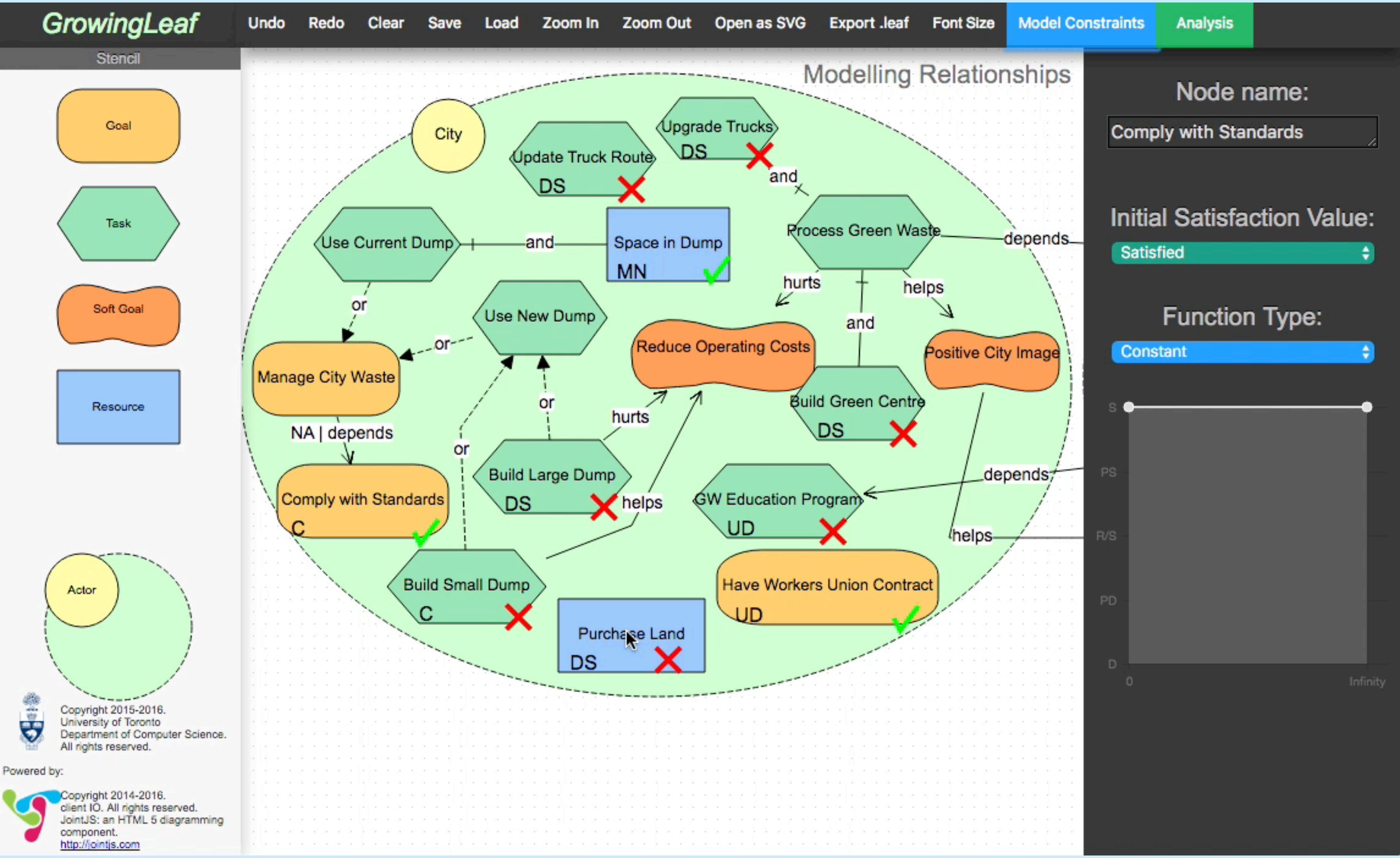
- How to run analysis
- Adjust simulation length
- Types of analysis
- Scrolling through analysis results

Improving Analysis with Constraints

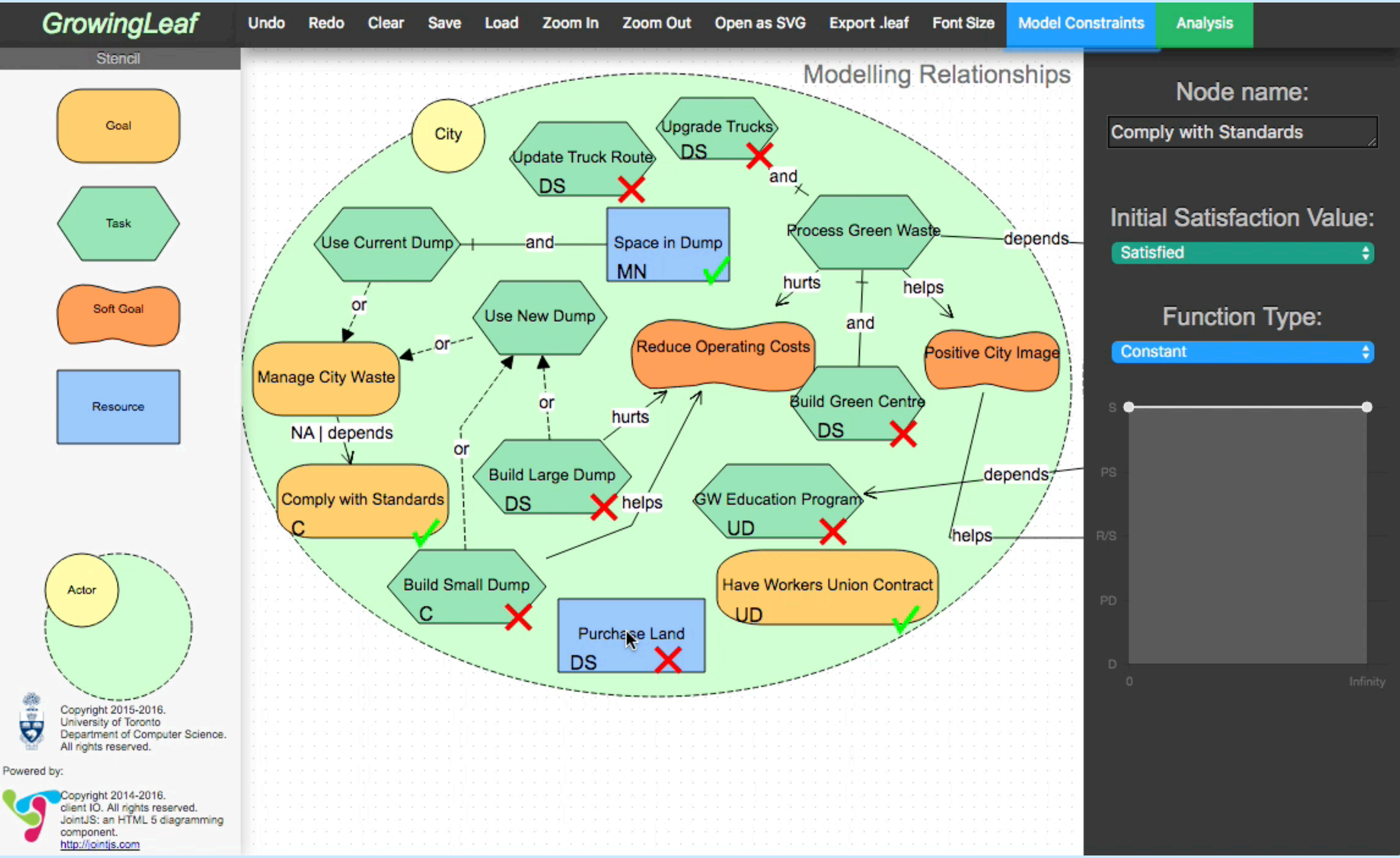
- Undesirable results due to EB ordering
- Add constraints over EB order
    - Adding model links is inappropriate
        - Test relationship before updating the model
- Used on rare occasions



GrowingLeaf - Model Constraints Demo



GrowingLeaf - Model Constraints Demo



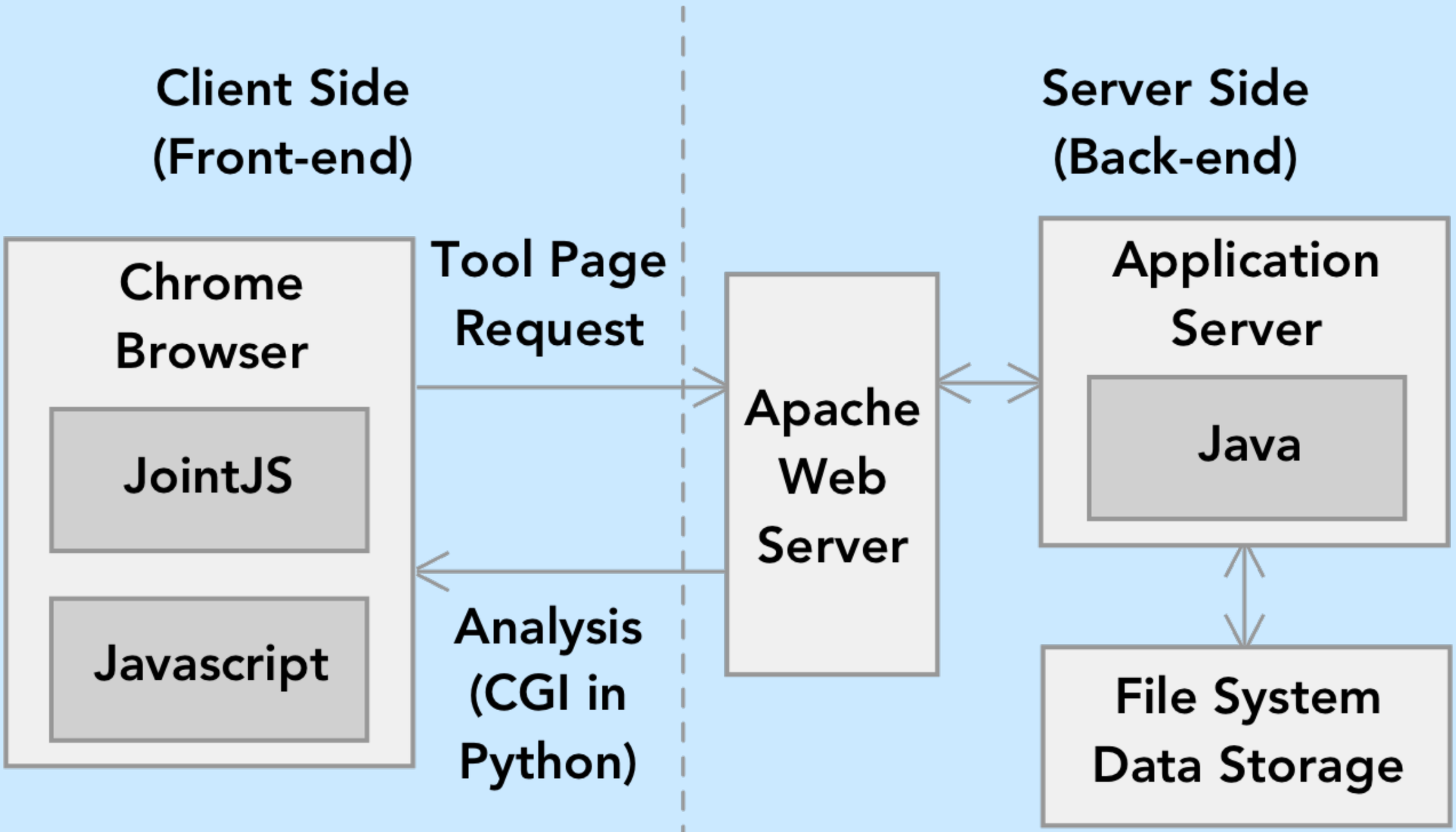
GrowingLeaf - Constraints Demo Summary

- Adding constraints between EBs

Outline

- Modeling Problem and Tool Justification
- Tool Introduction
- Dynamic Intentions and Analysis
- Tool Functionality
- **Discussion and Validation**
- Status and Future Work

Architecture



Design Decisions

- Browser versions and updates
- JointJS data model and constrains

Usability

- Two rounds user testing
- Found issues with
  - resizing
  - ‘enter’ key
  - ‘backspace’/‘delete’ key
  - selecting analysis techniques
- Further user studies are ongoing
- Built several models and examples

Examples and Case Studies

- City transportation planning
- Network maintenance
- Software supply chains
- Technical debt
- Compliance
- Sustainability

Further case studies are ongoing….

Ongoing Validation

- Evaluate usability / effectiveness with controlled experiment

- Prototype study at this week at iStar and RE

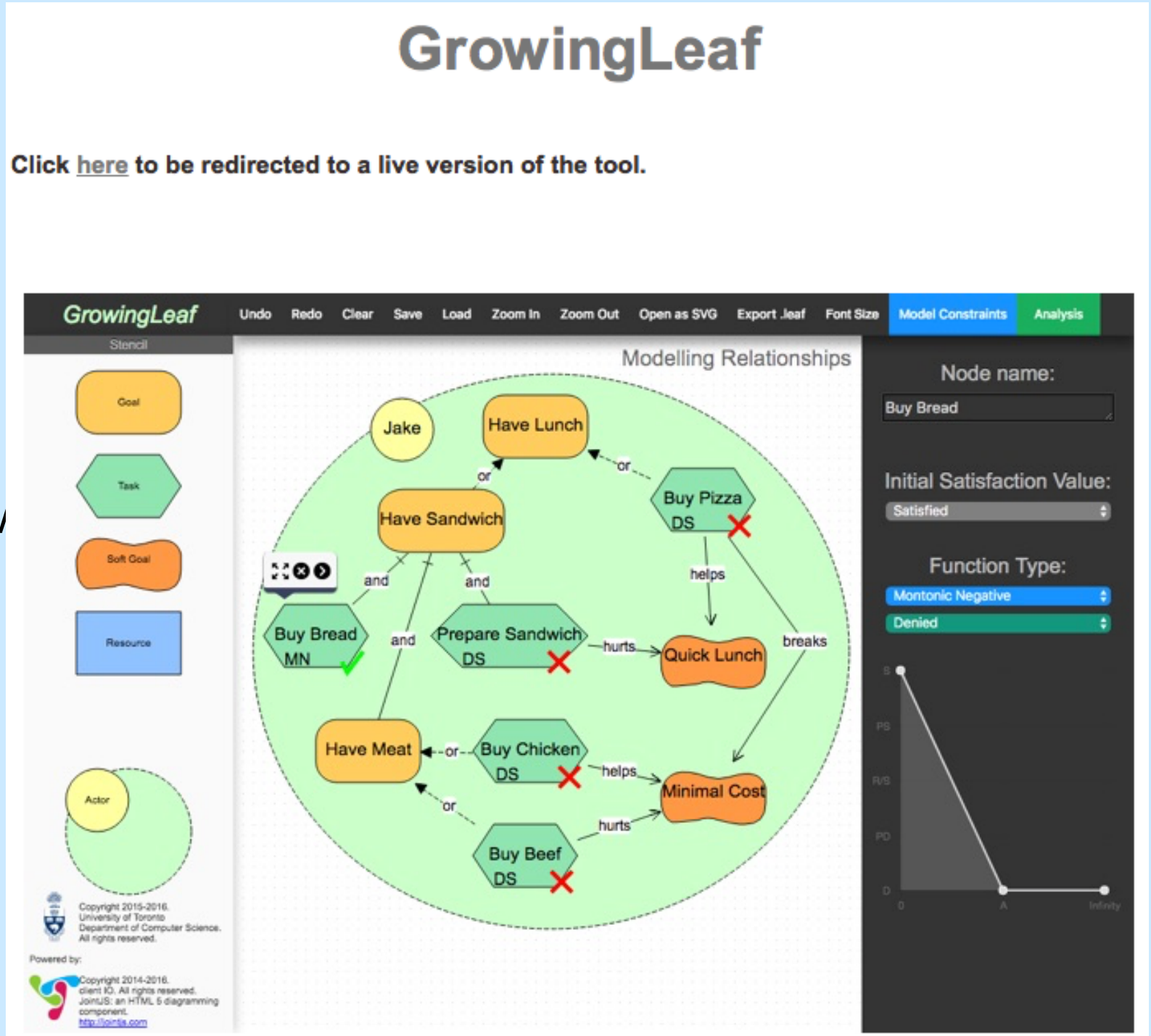
- Please Participate!!

  http://www.cs.toronto.edu/~amgrubb/restudy.htm

Where do I get the tool?

http://www.cs.toronto.edu/~amgrubb/growing-leaf

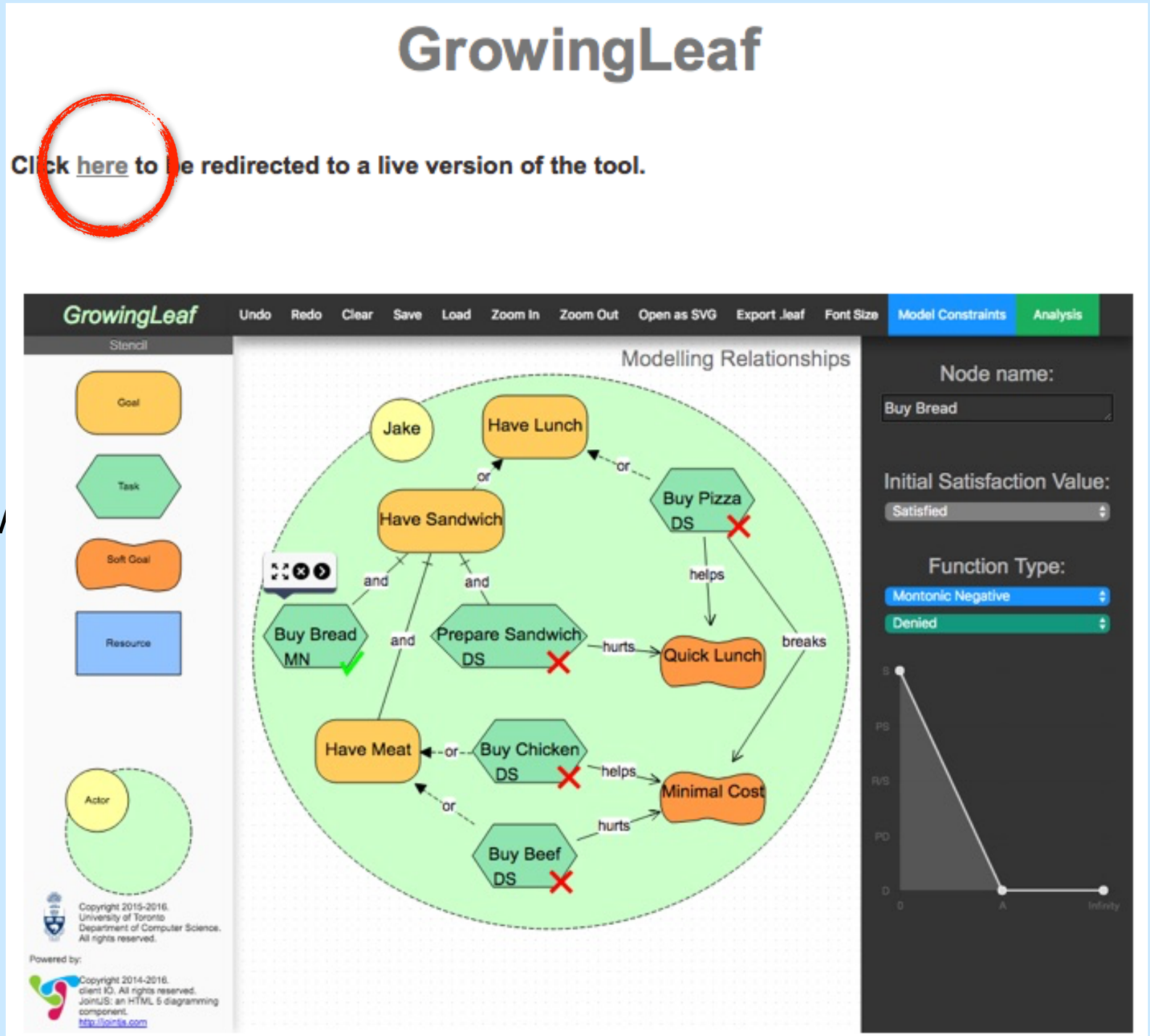
Where do I get the tool?



http://w g-leaf

GrowingLeaf: is an iStar modeling and analysis tool focused on understanding model evolution and how the evaluations of intentional elements change over time. GrowingLeaf was developed as an extension to Leaf (beta).

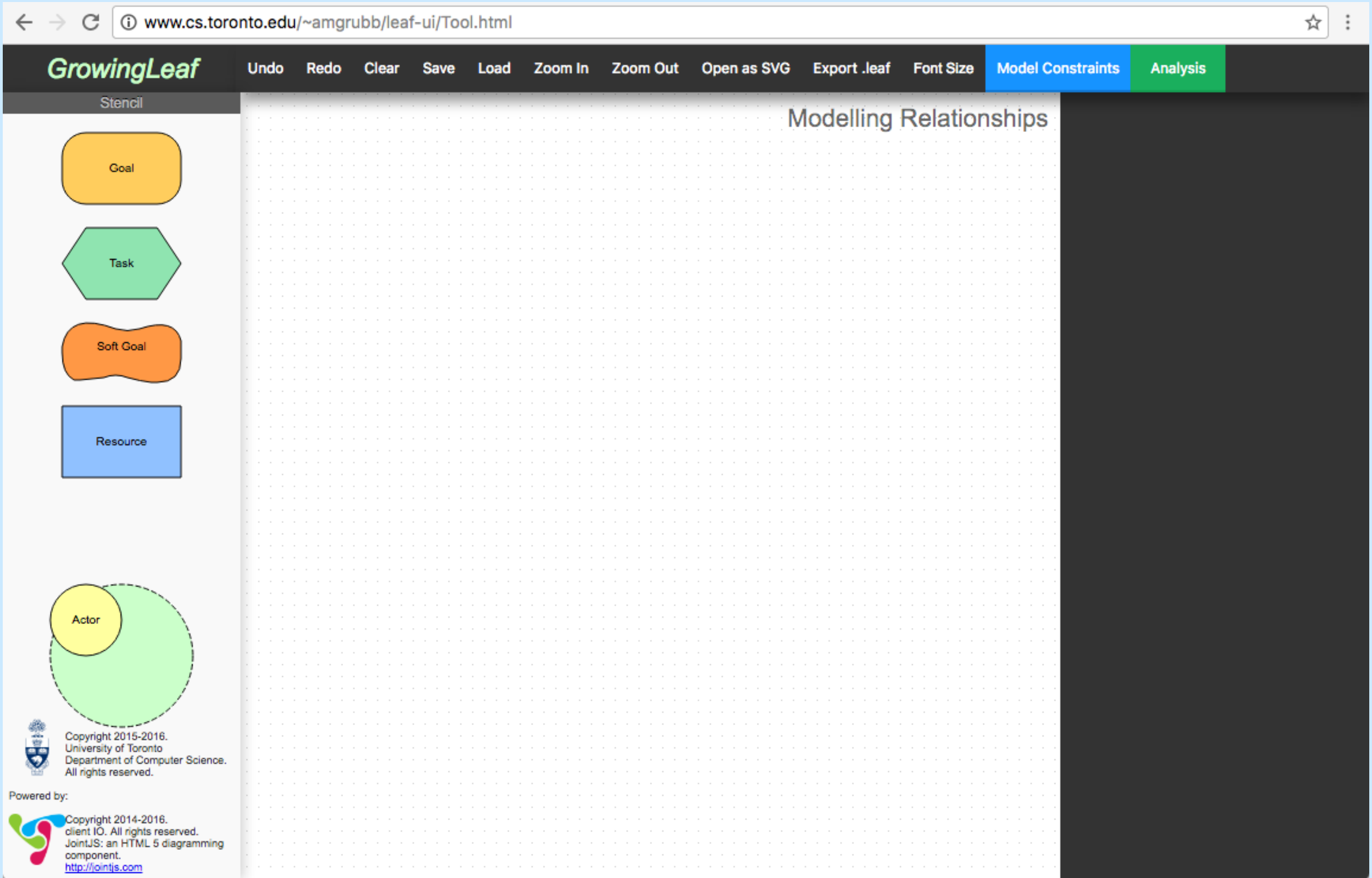
Where do I get the tool?



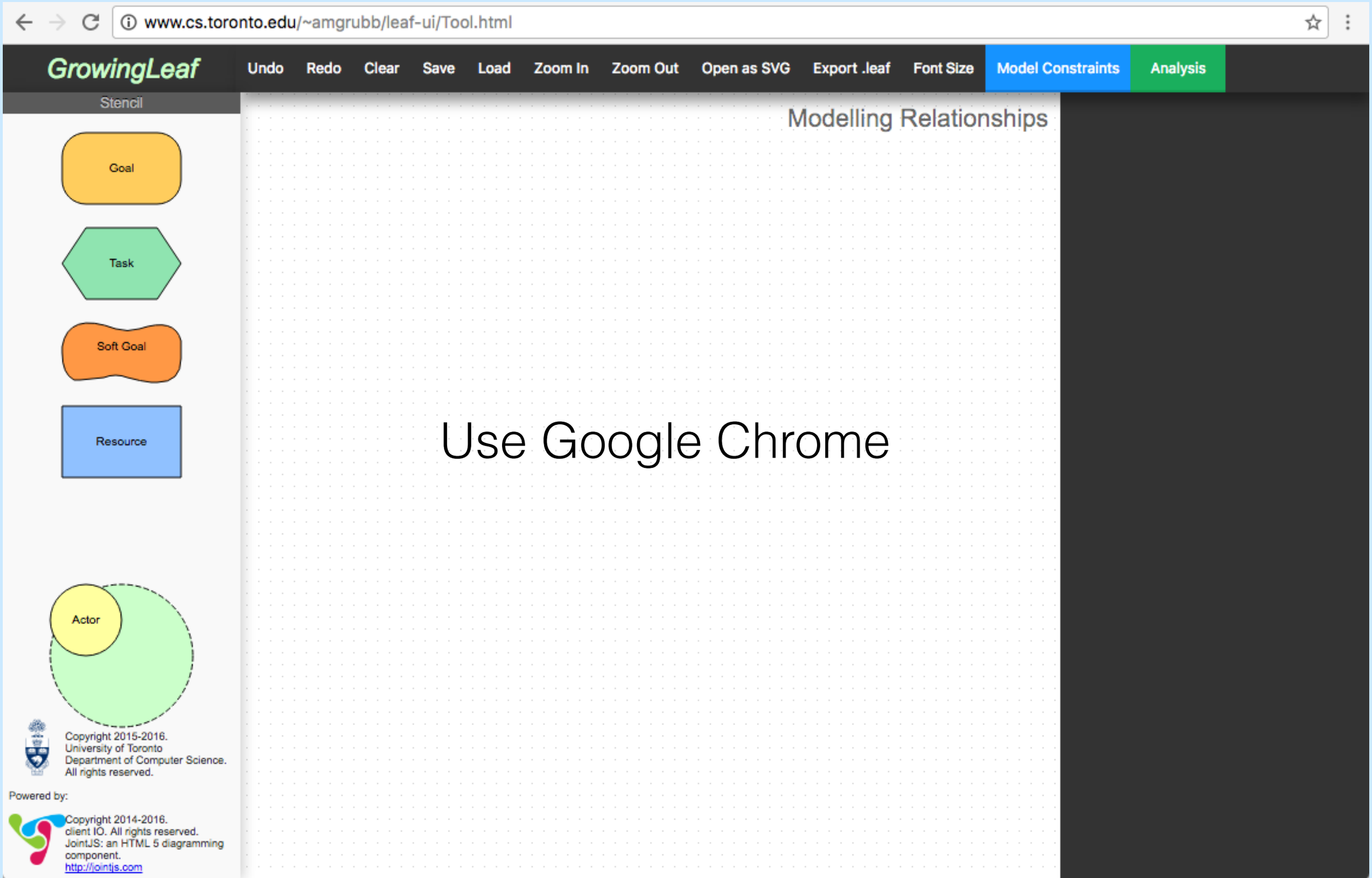
http://w g-leaf

GrowingLeaf: is an iStar modeling and analysis tool focused on understanding model evolution and how the evaluations of intentional elements change over time. GrowingLeaf was developed as an extension to Leaf (beta).

Where do I get the tool?



Where do I get the tool?



Where do I get the tool?

http://www.cs.toronto.edu/~amgrubb/growing-leaf

Join the development team.

Future Work

- Update tool to use iStar 2.0 Language Guide
- External industrial case study
- Improve server connection (security)
- Multiple users to simultaneously edit
- Development for other browsers

Questions?

GrowingLeaf: Supporting Requirements Evolution over Time

GrowingLeaf

http://www.cs.toronto.edu/~amgrubb/growing-leaf

Tool Study at RE'16:

http://www.cs.toronto.edu/~amgrubb/restudy.htm

UNIVERSITY OF TORONTO

Alicia M. Grubb
amgrubb@cs.toronto.edu