

Ontology Visualization

10th International Protégé Conference
July 15, 2007, 11:00 – 12:30PM CEST

Jennifer Vendetti, Stanford University



What is the graph widget?

- Allows visual editing of instances and relationships between instances
- Alternative to Protege's "Forms" for entering instance data

When is the graph widget appropriate?

- When instances are connected through slot values (e.g., a linked list).
- When instances connected by instances of the “:*DIRECTED-BINARY-RELATION*” system class.



When is the graph widget appropriate?

- Speeds knowledge acquisition in ontologies with heavily interconnected concepts.
- Helps convey meaning and organization of acquired knowledge.
- Data that resembles process diagrams, flow charts, organizational charts.

Open the "GraphWidgetExample" project

The screenshot shows the Protégé 3.3 interface. The title bar reads "GraphWidgetExample Protégé 3.3 (file:\C:\Program%20Files\Protege%203.3%20Full\examples\conference\GraphWidgetExample.pprj, Protégé...)". The menu bar includes File, Edit, Project, Window, Tools, and Help. The toolbar contains icons for file operations and editing. The main interface is divided into several panes:

- CLASS BROWSER:** Shows a class hierarchy for the project "GraphWidgetExample". The classes listed are :THING, :SYSTEM-CLASS, Author, Content, Layout_info, Library, Newspaper, Organization (highlighted), and Person.
- CLASS EDITOR:** Shows the details for the selected class "Organization". It includes fields for Name (Organization), Role (Concrete), and Documentation. There are also sections for Constraints and Template Slots.
- Template Slots Table:** A table with columns: Name, Cardinality, Type, and Other Facets.

Name	Cardinality	Type	Other Facets
employee_relationships	multiple	Instance of Manager Supervision Relation	
name	single	String	
organization_chart	multiple	Instance of Employee	

A yellow callout box points to the "Organization" class in the Class Browser with the text: "Select the 'Organization' class".



Classes Slots Forms Instances Queries

CLASS BROWSER

For Project: ● GraphWidgetExample

Class Hierarchy

- :THING
- ▶ ● :SYSTEM-CLASS
- ▶ ● Author
- ▶ ● Content
- ▶ ● Layout_info
- Library
- Newspaper
- Organization
- ▶ ● Person

Superclasses

- :THING

CLASS EDITOR

For Class: ● Organization (instance of :STANDARD-CLASS)

Name

Organization

Documentation

Constraints

Role

Concrete ●

Template Slots

Name	Cardinality	Type	Other Facets
employee_relationships	multiple	Instance of Manager Supervision Relation	
name	single	String	
organization_chart	multiple	Instance of Employee	

Notice the
"organization_chart" slot,
which is the slot that we will
configure to use the graph
widget



FORM BROWSER

For Project: ● GraphWidgetExample

Forms

- :THING
- ▶ ■ :SYSTEM-CLASS
- ▶ ■ Author
- ▶ ■ Content
- ▶ ■ Layout_info
- Library
- Newspaper
- Organization
- ▶ ■ Person

FORM EDITOR

For Class: ● Organization

Display Slot: name ed Widget Type: GraphWidget

Name

Organization Chart

Configure the "organization_chart" slot to use the graph widget

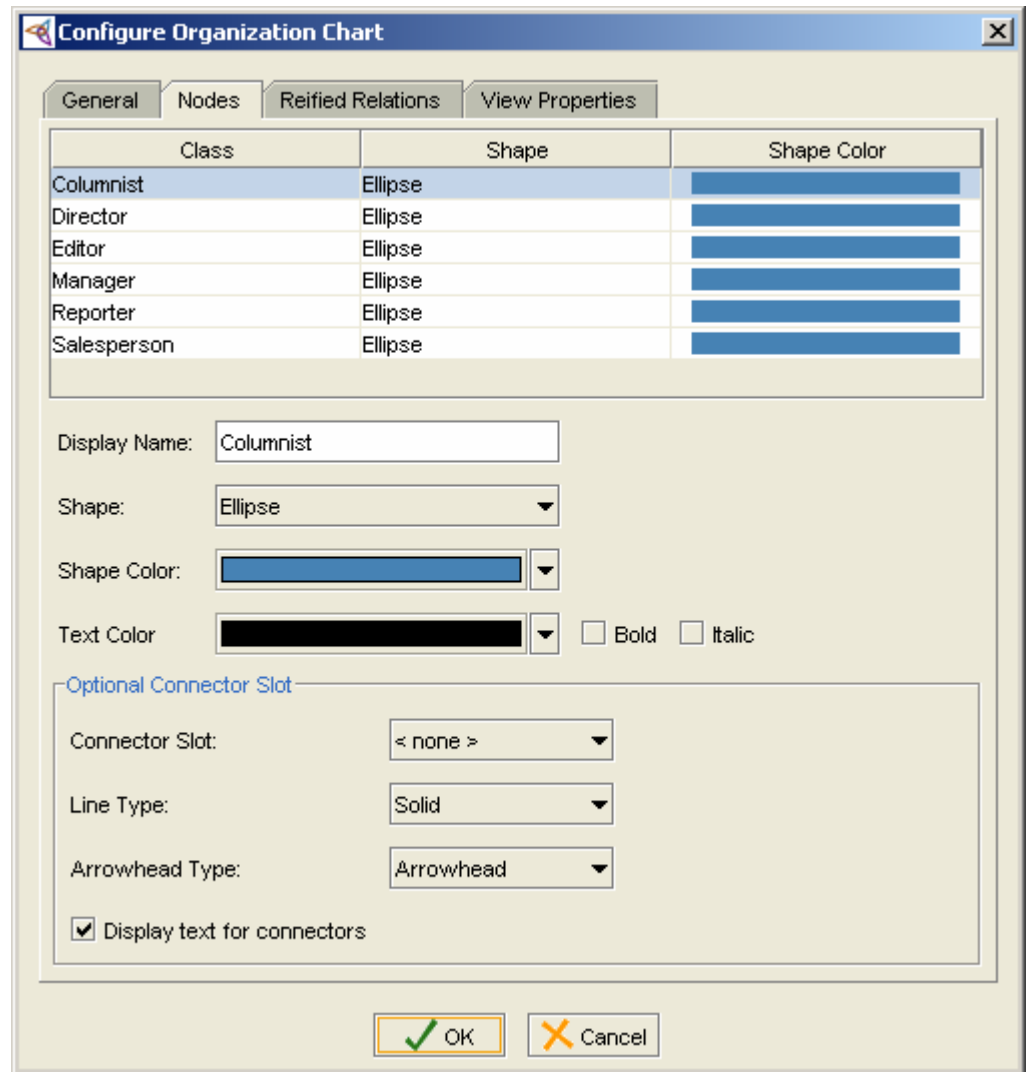


How do I optimize size for the graph widget?

- Double-click on the Form background (anywhere without a widget)
- Use the Layout tab to fill horizontal and vertical space

How do I configure node appearance?

- Double-click on the organization_chart slot to bring up the widget configuration dialog.





How do I configure node appearance?

- Use the “General” tab to change the label or tooltip for the “organization_chart” slot.
- Use the “Nodes” tab to assign shapes, colors, and text properties to nodes.



Configure simple connectors

- Graph widget has two connector types, one of which is a “simple connector”
- Simple connectors have no underlying instances



CLASS BROWSER

For Project: ● GraphWidgetExample

Class Hierarchy

- :THING
- ▶ ● :SYSTEM-CLASS
- ▶ ● Author
- ▶ ● Content
- ▶ ● Layout_info
- ▶ ● Library
- ▶ ● Newspaper
- ▶ ● Organization
- ▶ ● Person
 - ▶ ● Employee
 - Columnist
 - Editor
 - Reporter
 - Salesperson
 - ▶ ● Manager

Superclasses

- Author
- Employee

CLASS EDITOR

For Class: ● Editor (instance of :STANDARD-CLASS)

Name
Editor

Documentation

Constraints
◆ editor-employees-salary-constraint

Template Slots

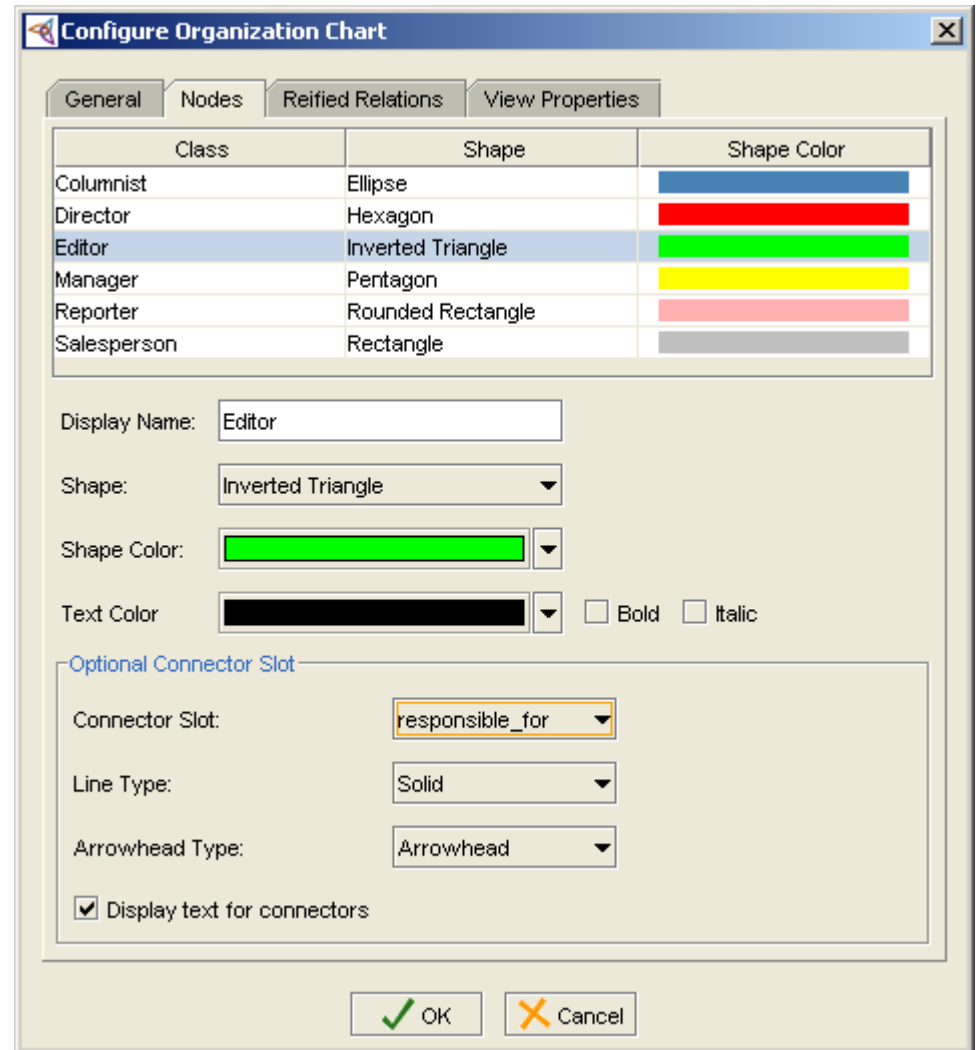
Name	Cardinality	Type	Other Facets
byname	single	String	
current_job_title	single	String	
date_hired	single	String	
name	single	String	
other_information	single	String	
phone_number	single	String	
responsible_for	multiple	Instance of Employee	
salary	single	Float	

Select the Editor class

Notice the "responsible_for" slot, which will be the simple connector

Configure simple connectors

- Choose connector slot for Editor class in widget configuration dialog





CLASS BROWSER

For Project: ● GraphWi...

Class Hierarchy

- :THING
- ▶ ● :SYSTEM-CLASS
- ▶ ● Author
- ▶ ● Content
- ▶ ● Layout_info
- ▶ ● Library (1)
- ▶ ● Newspaper (6)
- ▶ ● Organization (2)
- ▶ ● Person

INSTANCE BROWSER

For Class: ● Organization

name

- ◆ Hungarian Quarterly
- ◆ San Jose Mercury News

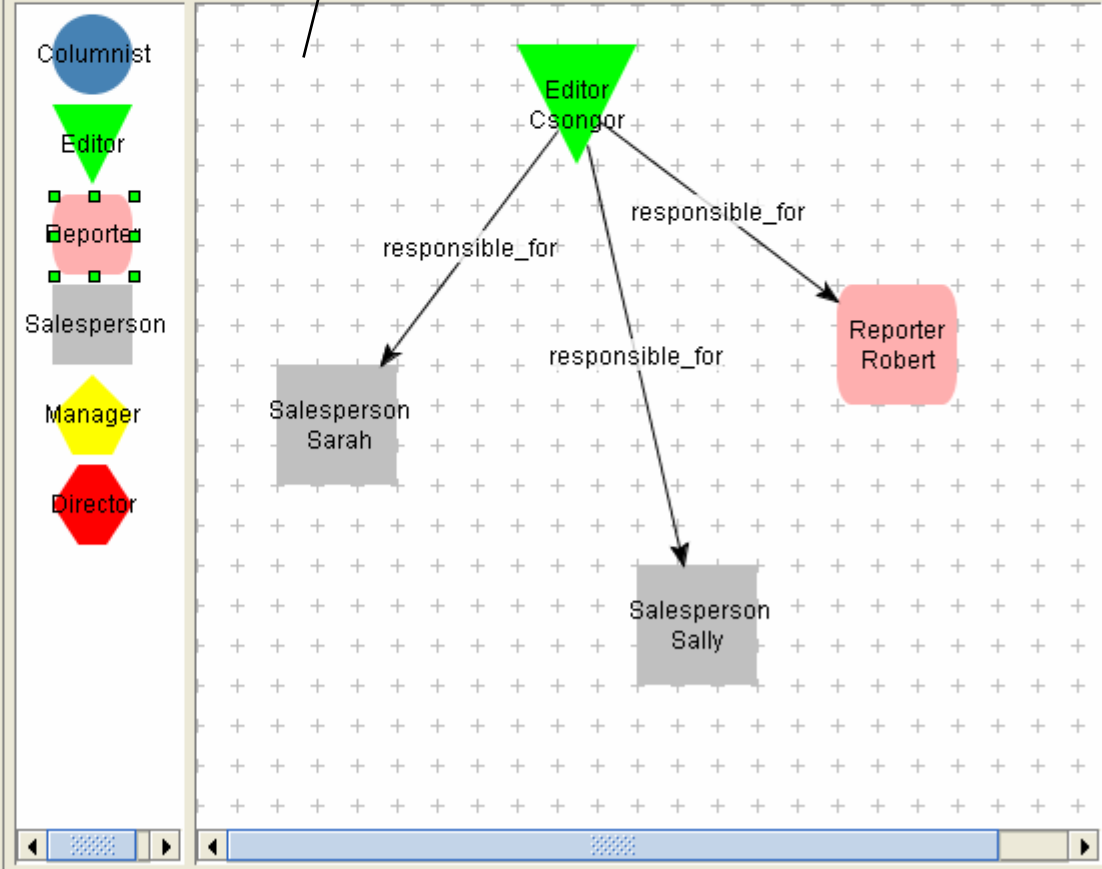
INSTANCE EDITOR

For Instance: ◆ Hungarian Quarterly (instance o

Name

Hungarian Quarterly

Organization Chart



Drag and drop nodes to create instances of employees – draw connectors to fill in slot values

Graph widget UI tips

- Click on node labels to drag nodes
- Nodes are resizable
- Double-click node labels to rename
- Right-click connectors to insert points
- Automatic layout provided
- Save graph as image



Graph widget UI tips

- The graph widget only allows you to draw *valid* connectors between nodes
- You *can* add pre-existing instances to the graph widget



Configure reified relations

- Reified relations are the second connector type offered by the graph widget
- Allows storage of additional information about a relationship between two nodes



CLASS BROWSER

For Project: ● GraphWidgetExample

Class Hierarchy

- :THING
 - ▼ ● :SYSTEM-CLASS
 - ▶ ● :META-CLASS
 - ▶ ● :CONSTRAINT
 - ▶ ● :ANNOTATION
 - ▼ ● :RELATION
 - ▼ ● :DIRECTED-BINARY-RELATION
 - Manager Supervision Relation
 - Director Supervision Relation
 - ▶ ● Author
 - ▶ ● Content
 - ▶ ● Layout_info
 - Library
 - Newspaper
 - Organization
 - ▶ ● Person

Superclasses

- :DIRECTED-BINARY-RELATION

CLASS EDITOR

For Class: ● Manager S

Name

Manager Supervision Rel

Role

Concrete ●

Template Slots

Name	Cardinality	Type	Other Facets
name	single	String	
quality	single	Symbol	allowed-values={good,ok,poor}
supervision type	single	Symbol	allowed-values={administrative,editorial}
(c) :FROM	required single	Instance of Manager	
(c) :TO	required single	Instance of Employee	

Notice subclasses of the :DIRECTED-BINARY-RELATION class, which will be our reified relations in the graph widget



CLASS BROWSER

For Project: ● GraphWidgetExample

Class Hierarchy

- :THING
- ▶ ● :SYSTEM-CLASS
- ▶ ● Author
- ▶ ● Content
- ▶ ● Layout_info
- Library
- Newspaper
- Organization
- ▶ ● Person

Search field with a magnifying glass icon.

Superclasses

- :THING

CLASS EDITOR

For Class: ● Organization (instance of :STANDARD-CLASS)

Name: Organization

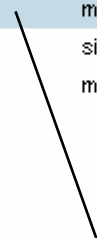
Documentation: [Empty text area]

Constraints: [Empty text area]

Role: Concrete ●

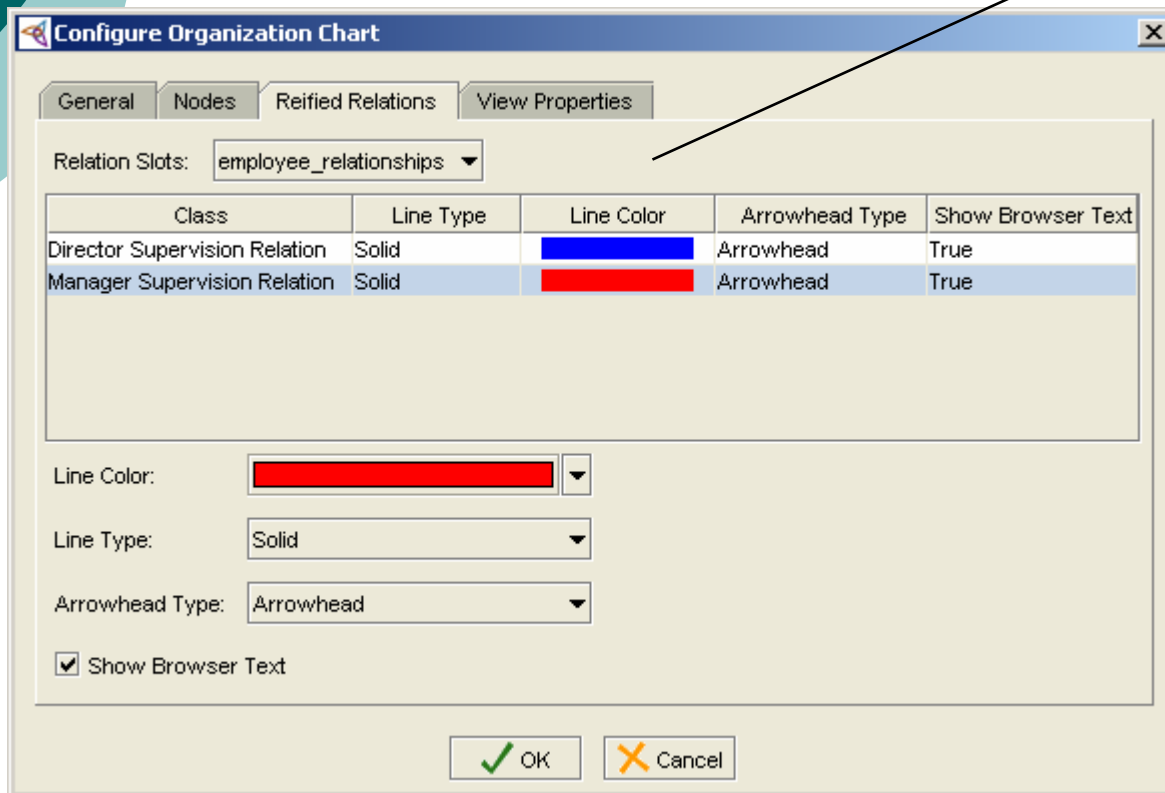
Template Slots

Name	Cardinality	Type	Other Facets
employee_relationships	multiple	Instance of Manager Supervision Relation	
name	single	String	
organization_chart	multiple	Instance of Employee	

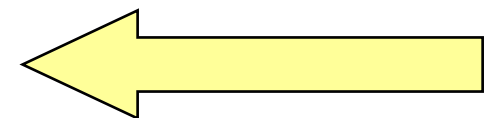
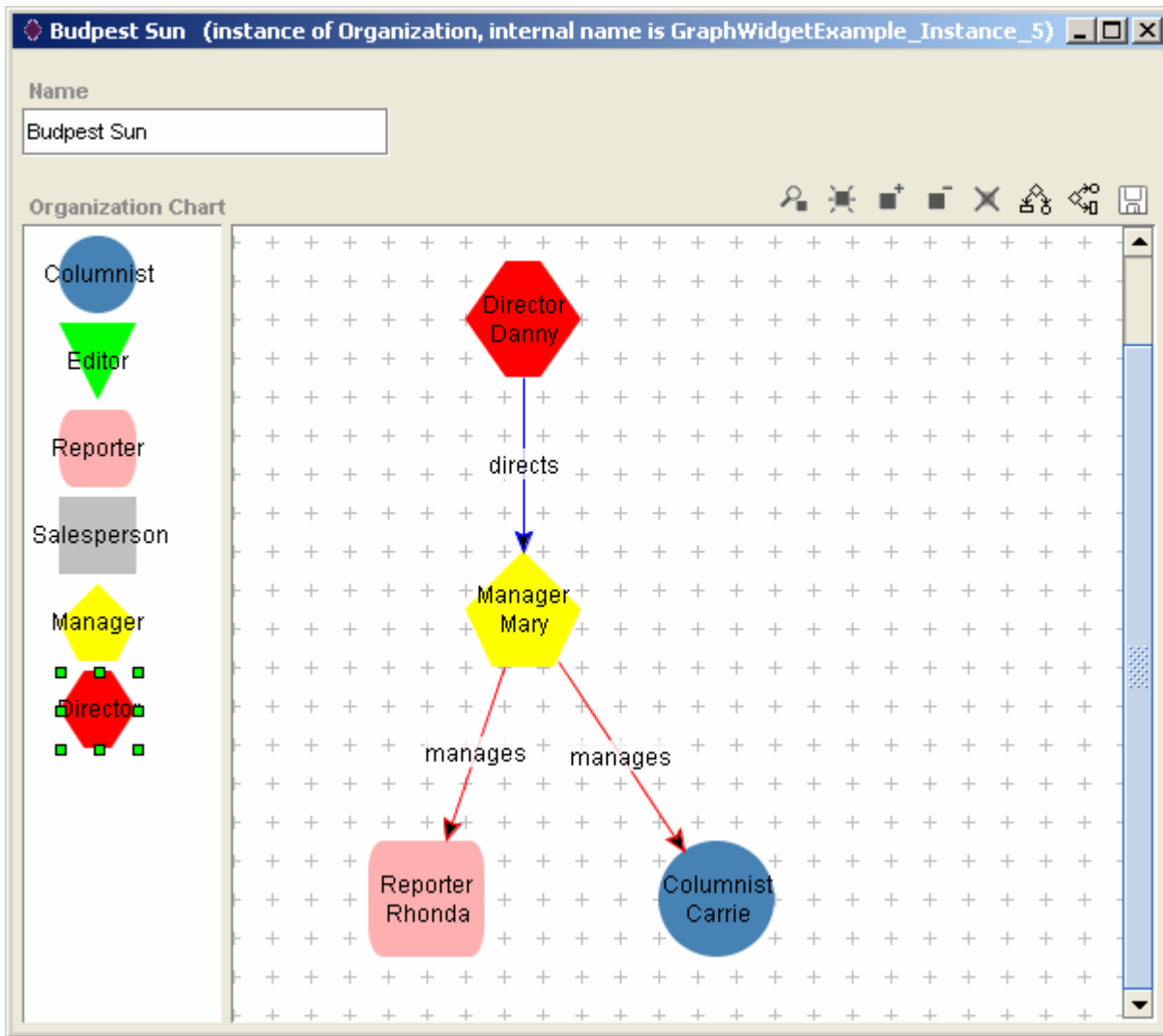


Notice the slot that has been designated to hold instances of reified relations

Configure reified relations



Use the reified relations tab to configure the relation slot, line types, colors, arrowheads, etc.



Example of reified relations between instances of the Employee class

Graph widget UI Tips

- Double-click on complex links to bring up instance forms
- Designate display slots for subclasses of *:DIRECTED-BINARY-RELATION* to enable editing of connector labels

Larger scale visualization

- OntoViz Tab – visualize ontologies with GraphViz
- OWLViz Tab – Visualize OWL ontologies with GraphViz
- Jambalaya – visualize ontologies with SHriMP (Simple Hierarchical Multi-Perspective)
- TGViz – visualize ontologies with TouchGraph

OntoViz Tab: helper documentation for tab abbreviations...

- sub = subclass closure
- sup = superclass closure
- slx = slot extension
- isx = inverse slot extension
- slt = slots
- sle = slot edges
- ins = instances
- sys = system frames

OntoViz Tab

The screenshot shows the Protégé 3.3 interface with the OntoViz tab selected. The main window displays a configuration table for the 'Organization' class and a list of classes. The 'Organization' class is highlighted in the 'Classes' list. The 'Options' button in the 'Config' toolbar is highlighted with a yellow box and an arrow pointing to a callout box.

frame	sub	sup	slx	isx	sit	sle	ins	sys
Organization	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Classes

- :THING
- ▶ • :SYSTEM-CLASS
- ▶ • Author
- ▶ • Content
- ▶ • Layout_info
- Library (1)
- Newspaper (6)
- Organization (1)
- ▶ • Person

Use the
"add class" button to choose
a class to graph

Use the
"Options"
button to
configure
graph

The 'Ontoviz Global Options' dialog box is shown with the 'general' tab selected. It contains several checkboxes and dropdown menus for configuring the graph visualization.

Ontoviz Global Options

general colors

- save as gif
- show io :STANDARD-CLASS edges
- show system own slots
- slot edges dashed
- show instances only
- 3 maximum depth for slot extension (instances)
- 1 maximum depth for slot extension (classes)
- 3 maximum displayed values per slot
- 5 maximum displayed slots per node

Click the "Create Graph" button to generate graph

newspaper Protégé 3.3 (file:\C:\Program%20Files\Protégé Files (.pont and .pin...)

File Edit Project Window Tools TGvizTab Help

Classes Slots Forms Instances TGvizTab Ontoviz

Config

frame	sub	sup	slx	isx	slt	sle	ins	sys
Organization	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Classes

- :THING
- :SYSTEM-CLASS
- Author
- Content
- Layout_info
- Library (1)
- Newspaper (6)
- Organization (1)
- Person
 - Employee
 - Columnist (2)
 - Editor (4)
 - Reporter (3)
 - Salesperson (1)
 - Manager (3)

Organization

Organization		
name	String	
employees	Instance*	Employee
employee_relationships	Instance*	Manager Supervision Relation

Employee

Employee	
salary	Float
date_hired	String
current_job_title	String

Manager Supervision Relation

quality	Symbol	good
		ok
		poor
name		String
supervision_type	Symbol	administrative
		editorial

employees* employee_relationships*

Demo: OWLViz Tab

- Designed specifically for use with the Protégé-OWL editor.
- Allows comparison of asserted and inferred class hierarchy.
 - Computed changes to hierarchy are clearly visible.
 - Inconsistent concepts are shown in red.



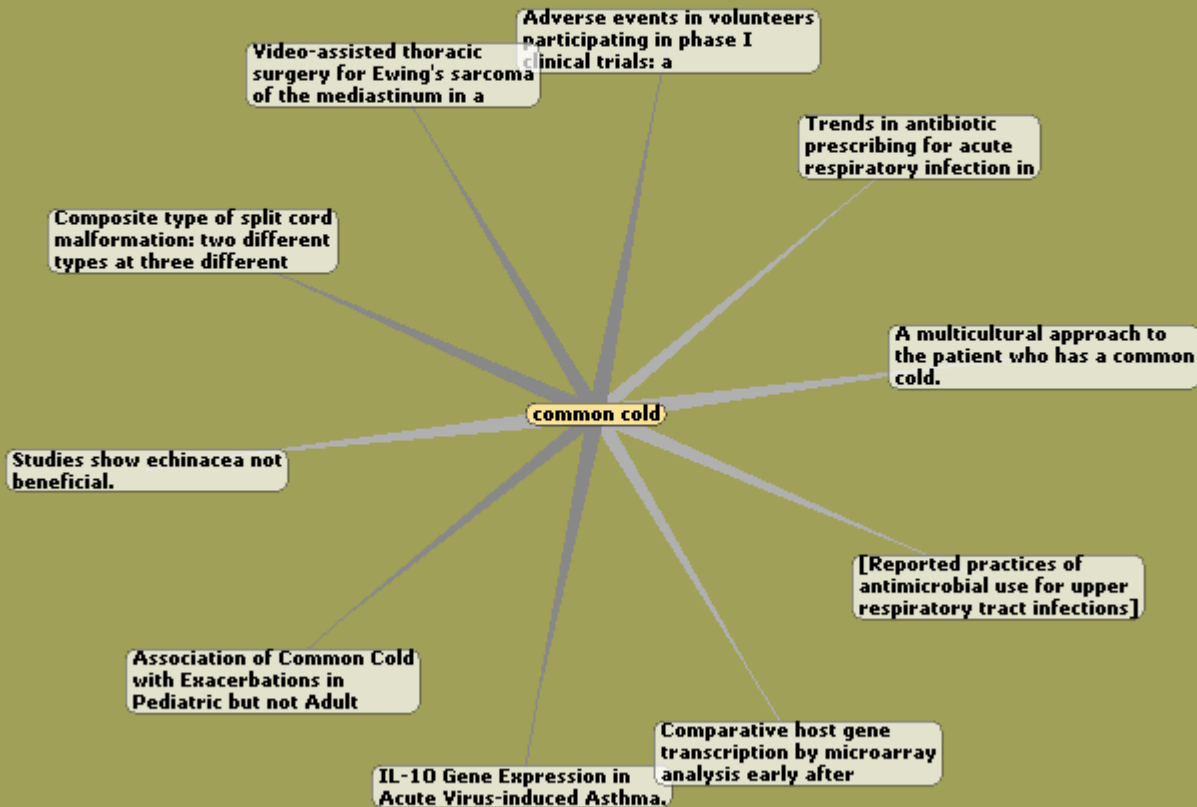
Demo: Jambalaya Tab

- Uses SHriMP (Simple Hierarchical Multi-Perspective)
- SHriMP is designed to help people browse complex information spaces
- Upside: very feature rich
- Downside: bigger learning curve than other tools
- Documentation/tutorials:
<http://www.thechiselgroup.org/jambalaya>



TGViz Tab

- Utilizes TouchGraph (renders networks as interactive graphs)
- TouchGraph uses “Spring Layout”
- PubMed uses TouchGraph to visualize graphs of related documents in medical libraries



Use the "add class" and "add instance" buttons to add items – click the "create graph" button to graph

The screenshot shows the Protégé 3.3 interface with a project named "newspaper". The main workspace displays an ontology graph with the following structure:

- Organization** (highlighted in green) is the root class.
- Employee** is a subclass of **Organization**.
- Person** is a subclass of **Organization**.
- Library** is a subclass of **Organization**.
- Advertisement** is a subclass of **Person**.
- Manager Supervision Relat** is a subclass of **Organization**.
- Library** is connected to **Article**, **Prototype Newspaper**, **Newspaper**, and **Billing_Chart**.
- Employee** is connected to **Reporter**, **Editor**, **Manager**, **Columnist**, and **Salesperson**.

The left sidebar contains the following panels:

- Setting**: Includes icons for adding and deleting classes and instances, and a search field.
- CLASS BROWSER**: Shows the class hierarchy for the project "newspaper". The hierarchy includes:
 - :SYSTEM-CLASS
 - Author
 - Content
 - Layout_info
 - Library (1)
 - Newspaper (6)
 - Organization (1)
 - Person
- INSTANCE BROWSER**: Shows instances for the selected class "Organization", including "San Jose Mercury News".