

Feature analysis of ontology visualization methods and tools

Merlin Florence Joseph, Ravi Lourdusamy

Department of Computer Applications (UG), Sacred Heart College, Tirupattur, India

Article Info

Article history:

Received Dec 17, 2019

Revised May 1, 2020

Accepted May 24, 2020

Keywords:

Data visualization

Ontology

Taxonomy

ABSTRACT

Visualization is a technique of creating images, graphs or animations to share knowledge. Different kinds of visualization methods and tools are available to envision the data in an efficient way. The visualization tools and techniques enable the user to understand the knowledge in an easy manner. Nowadays most of the information is presented semantically which provides knowledge based retrieval of the information. Knowledge based visualization tools are required to visualize semantic concepts. This article analyses the existing semantic based visualization tools and plug-ins. The features and characteristics of these tools and plug-ins are analyzed and tabulated.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Merlin Florence Joseph
Department of Computer Applications (UG)
Sacred Heart College, Tirupattur, Vellore, Tamilnadu, India
Email: merlinflorence@gmail.com

1. INTRODUCTION

Diagrammatic representation of a text is used to understand the concept easily and rapidly. Visualizing the text in a graphical mode is called as concept map or knowledge visualization. It is widely used in the education domain which makes the learner to understand the concept in an interactive manner. Numerous tools and techniques are available to convert the text into graphical visualization and they are adopted in different domains. These techniques are applied to develop plug-ins where a plug-in is a piece of software code that enables an application or program to do something. Most of the web pages are developed with a set of plug-ins to support videos, graphics and so on. Most of the information is tied up semantically which is shared in various domains. Knowledge based retrieval method is used over semantic web technology to facilitate the user to utilize the information and resource efficiently. Ontologies play vital role in storing and retrieving information based on knowledge. Ontology is a collection of primitives like classes, properties, axioms etc. Ontology can be defined as, "Ontology is a hierarchically structured set of terms for describing a domain that can be used as a skeletal foundation for a knowledge base" [1]. This skeletal structure of ontology is used to develop knowledge based visualization of ontologies. Ontology provides the means for describing explicitly the conceptualization behind the knowledge represented in a knowledgebase [1]. In ontology, the concepts are aligned in a hierarchical structure and the relationships are established using properties and individuals. Ontologies are used to represent the text in semantic based graph using plug-ins. Ontology visualizing plug-ins are developed by adapting graph visualization techniques like SpaceTree and GraphViz. Many ontology visualization tools (OWLGrEd, NavigOWL, GROW, RDFGravity and so on) and plug-ins (OntoGraf, OWLViz, Knooks, TGViz and so on) are available. The article aims to analyze various data visualization techniques, needs and importance of developing data visualization tools and plug-ins. In this article, types of graph visualizations are presented in section 2. Various ontology visualization plug-ins and tools are elaborated

in section 3. The features of ontology visualization tools and plug-ins are also analyzed and compared in this section. The challenges and the conclusions are proposed in section 4.

2. BACKGROUND AND RELATED WORKS

A graph is a collection of objects that are connected together in some way for the representation of a network where each object is a vertex or node [2]. The graph can be either directed or undirected. Graphical representation of data involves different methods and styles. The structure of diagram will differ depending upon the type of data or concept [2]. According to the type of data and concept it is classified into four categories namely data visualization, information visualization, scientific visualization and software visualization. Data visualization technique is adapted in ontological engineering to visualize ontology concepts. Ontology visualization tools and plug-ins are compared in various research works [3]-[7]. The tools are compared based on the two categories: ontology visualization tools for ontology experts and ontology visualization tools for everyone. All these surveys were compared ontology visualization tools with their characteristics. Ontology visualization methods are classified as: indented list, node-link and tree, zoomable, space-filling, focus with context or distortion, and dimension [8]. These methods are further classified under two dimensions (2D) and three dimensions (3D). The ontology visualization methods display the classes, instances, properties and individual of developed ontologies clearly. *Indented list* displays the class hierarchy in the form of a tree view. The child nodes are displayed under root node or parent node. *Node-link and tree method* represent the ontologies as top to bottom or left to right interconnected node. *Zoomable* visualization method allows the user to zoom all the nodes of the ontology. The visualization space or area is divided according to the size and property of the node in *space filling method*. *Focus with context* methods are used to focus particular concept where all the corresponding concepts are placed around the same.

3. TYPES OF ONTOLOGY VISUALIZATION TOOLS

Ontology visualization tools are classified into various methods as explained in the previous chapter. In this paper, ontology visualization techniques are classified into two categories namely ontology visualization tools and ontology visualization plug-ins. Ontology visualization tools are further classified into Standalone tools and web based tools.

3.1. An overview of ontology visualization tools

Ontology visualization tools are developed to build and visualize ontologies effectively. There are two types of ontology visualization tools are available namely standalone visualization tool and web-based visualization tool. Standalone tools can be downloaded and installed in an end user system. Web-based visualization tool enables the user to upload or select ontology that they want to visualize. Ontology tools are developed to run independently which enables the user to upload and visualize ontology. Ontology visualization tools can be run either on client machine or in a web browser. This section describes the tools that are used to visualize ontologies.

3.1.1. Ontology visualization tools

a. OWLEasyViz

OWLEasyViz is an ontology editor which combines textual representation and graphical representation of OWL ontologies [9]. It provides an easy interface to the users. The textual representation contains class, data properties and object properties in a three-column table. The graphical representation gave “is-a” inheritance relationship of the classes and properties. Hierarchies are represented as a nested set and the child nodes were placed inside of its parent nodes. OWLEasyViz is used to overcome the drawbacks in Semantic DB system. It provides simple working area, abstraction of ontology constructs and advanced options to construct complex ontologies. This tool is integrated with ontology editor, resource editor, searching and filtering tools and inference mechanisms. This integration allowed controlling ontology, resources and information rules. Unfortunately, this tool is not available. The Sample visualization using OWLEasyViz is given in Figure 1.

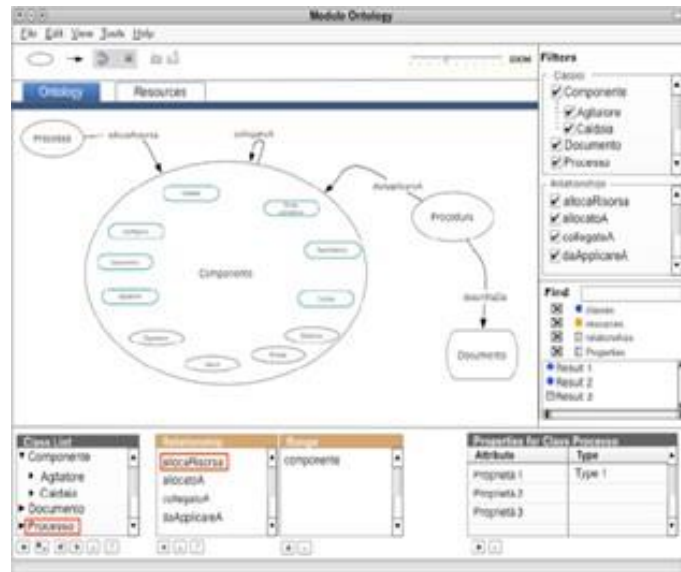


Figure 1. Sample of OWLeasyViz

b. GrOWL

GrOWL is a browser and editor for OWL ontologies [8]. It is used to visualize the description logic semantics of OWL ontologies accurately. It is developed based on Prefuse libraries. It can be used as a protégé plug-in and stand alone application. Protégé plug-in is developed using java applet and stand alone application is developed using java application. GrOWL allows the user to perform ABox mapping and TBox mapping. It uses forced directed layout for an interactive view of class hierarchies. It filters classes, subclasses, super-classes and instances of a selected node. Mathematical notations are used to represent the relations between the classes as given in Figure 2.



Figure 2. Screenshot of GrOWL visualization

c. NavigOWL

NavigOWL is designed to visualize semantic net [10]. It gives graph layout that is used to understand the structure of ontologies. It is also used to build mind map of ontologies. It supports RDF and OWL ontologies to visualize. It gives tool tip while mouse hover on the node. User can visualize in five layouts, namely circle layout, random layout, force layout, spring layout and power layout. It represents seven varieties of nodes as: class node, instance node, data property node, objects property node, collection node, literal node and property node. The sample visualization of café shop ontology in NavigOWL tool is shown in Figure 3.

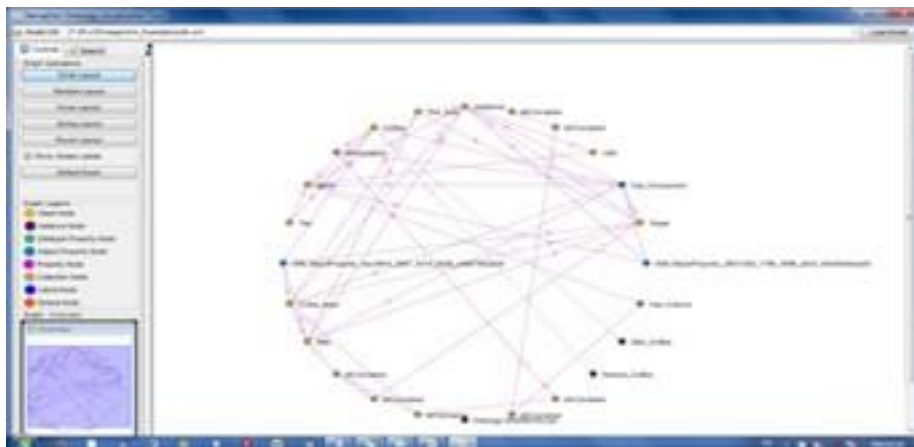


Figure 3. Visualization of café shop in NavigOWL

d. RDF Gravity

RDF Graph Visualization Tool (RDF Gravity) is a RDF and OWL visualization tool provides directed RDF graph structures [11]. Jena semantic web toolkit (Jena 2.0) and JUNG Graph API are used in RDF Gravity. It allows the user to navigate graph easily. User can select more than one node to visualize and move over the graph as shown in Figure 4. It provides two types of filters for visualizing ontology. They are: local filter and global filter. Local Filter is used to hide or view edges of a particular type. Global Filters allow a user to hide specific edges and concepts based on type.

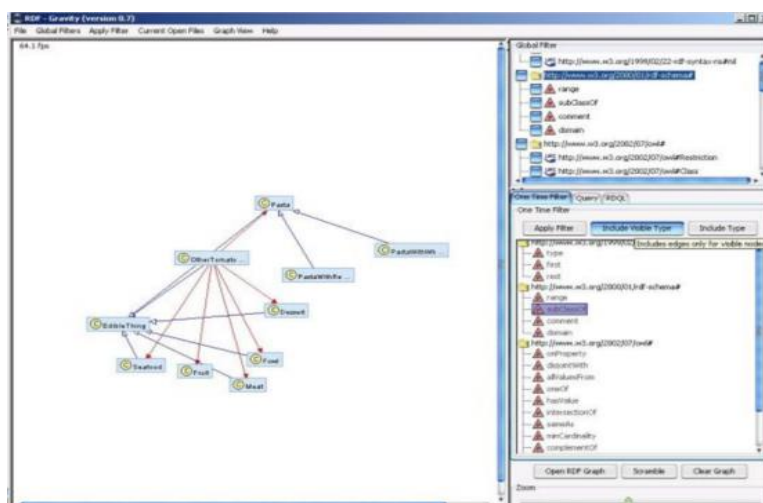


Figure 4. Visualization in RDF gravity

e. IsaViz

IsaViz is a visual authoring tool for RDF [12]. It is used to browse and edit RDF models represented as graphs. It provides 2.5 Dimension user interface and allows zooming and navigating within the graph. RDF/XML, Notation 3 and N-Triple can be imported and visualized using this tool. RDF graphs are rendered using Graph Style Sheets (GSS) which are derived from Cascading Style Sheet (CSS) and Scalable Vector Graphics (SVG). GSS is used to represent RDF as node-link diagrams. This style sheets enable to highlight the selected nodes. It displays the network view of overall ontology within a small window. IsaViz depends on GraphViz/dot libraries. The sample visualization of IsaViz is given in Figure 5.

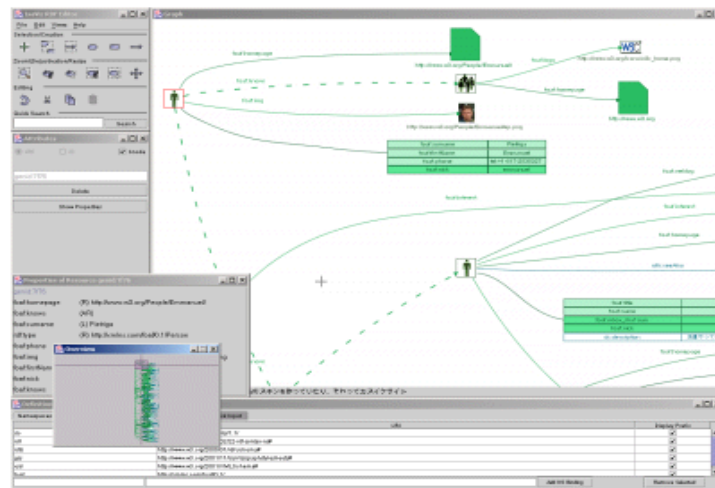


Figure 5. Visualization of ontology in isaViz

f. OntoTrack

OntoTrack is a browsing and editing ontology authoring tool [13]. It is SpaceTree based system implemented in Java2D. OntoTrack expands and de-expands the classes with animation. It provides editing features through mouse-over anchor buttons. OntoTrack combines hierarchical layout technologies with context sensitive zooming features. It allows the user to zoom the tree branches dynamically using linked tree diagram approach. It also allows the user to edit, navigate and manipulate large scale ontologies. All the tools explained above are compared in three perspective namely primitives, user interface and environment. The results are tabulated in Table 1.

Table 1. Comparison of ontology visualization tools (a)

Visualization Tool/Plug-in name	Primitives					
	Class	Object Properties	Data Properties	Individuals	Relations	Annotations
IsAviz	Y	Y	N	Y	Is-a	Y
OntoTrack	Y	N	Y	N	Superclass	Y
GrOWL	Y	Y	Y	Y	User defined	N
RDF Gravity	Y	Y	Y	Y	User defined	N
NavigOWL	Y	Y	N	Y	Subclass-of	Y
OWLEasyViz	Y	Y	Y	Y	IS-A	OWLEasyViz

Table 1. Comparison of ontology visualization tools (b)

Visualization Tool/Plug-in name	Layout	Dimension	Zoomable	Search	User Interface			Representation
					Directions	Adjustable	Movable	
IsAviz	Table Node and Arc	2.5D	Y	Y	Undirected Unidirection	Y	Y	Ellipse Boxes
OntoTrack	Tree	2D	Y	Y	Uni-Direction	Y	Y	Rectangle
GrOWL	Forced directed layout	2D	Y	Y	Uni-Direction	Y	Y	Rectangle
RDF Gravity	Radial	2D	Y	Y	Uni-Direction	Y	Y	Rectangle
NavigOWL	Circle, Random, Force, Spring, Power	2D	Y	Y	Uni-Direction	Y	Y	Circles
OWLEasyViz	Tree	2D	Y	Y	Uni-Direction	Y	Y	Ellipse Rounded rectangle

Table 1. Comparison of ontology visualization tools(c)

Visualization Tool/Plug-in name	Compatibility	Availability	Import	Export	Environment Dependencies	Platform	OL Support	NL Support
IsAviz	Stand-alone	Y	Y	Y	Graphviz Dot	Java	OWL	N
OntoTrack	Stand-alone	Y	Y	Y	SpaceTree	Java	OWL	N
GrOWL	Protégé Stand-alone	Y	N	Y	Independent	Java	OWL	N
RDF Gravity	Standalone	Y	Y	Y	JVM	Java	RDF, OWL	N
NavigOWL	Standalone	Y	Y	N	Independent	Java	OWL,RDF	N
OWLEasyViz	Standalone	Y	N	Y	Independent	Java	OWL	N

3.1.2. Web-based ontology visualization tools

a. OWLGrEd

OWLGrEd is UML based ontology visualizer. Both online and offline version of OWLGrEd is available [14]. Graphical notations of ontology are represented as UML class diagrams. OWL classes are visualized as UML classes. Data properties are represented as class attributes. Object properties are denoted as associations. Individuals are displayed as objects. Cardinality restrictions on association domain class are expressed as UML cardinalities. OWLGrEd diagrams use the orthogonal layout. The inheritance-defining relations like subclass-of relations between classes and instance-of relations between classes and instances are presented in a hierarchical layout. It runs independently and allows the user to import ontology for better visualization. It also allows the user to develop ontology using UML structures. The screenshot of the OWLGrEd is shown in Figure 6.

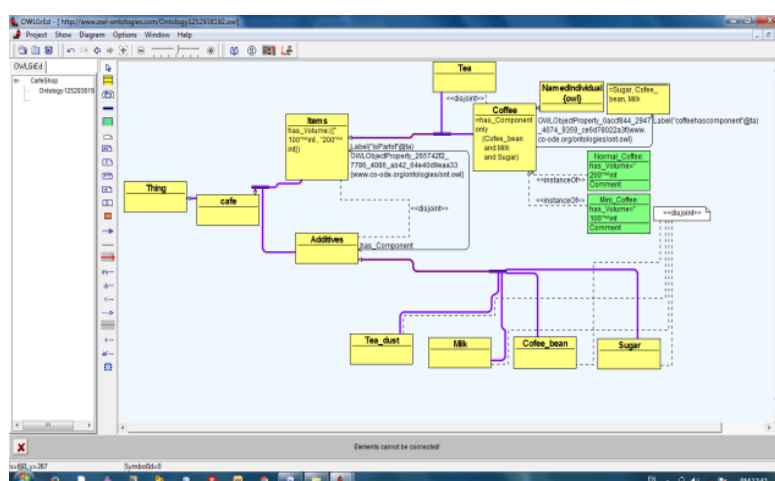


Figure 6. Visualization of café shop ontology in OWLGrEd

b. WebVowl

WebVowl is a web application used to visualize ontology through web [15]. It provides graphical representation of an ontology elements developed in Web Ontological Language (OWL). It is combined to a force-directed graph layout which represents the ontology. The graphical depictions are implemented using Visual Notation of OWL ontologies (VOWL). It has two panels where left side panel represents the graph of the ontology and right side panel represents the annotations of the ontology. It provides five functionalities: export, gravity, filter, modes and graph. Export is used to export the graph as SVG and JSON format. Gravity is used to adjust the distance between classes and data types. Filter has four types of filtering options: data property, solitary subclass, disjointness info and set operators. The VOWL visualizations are automatically generated from JSON files. WebVOWL contains Java based OWL2VOWL converter to effective visualization of ontologies. Friend of a Friend vocabulary ontology is visualized in WebVOWL as shown in Figure 7.

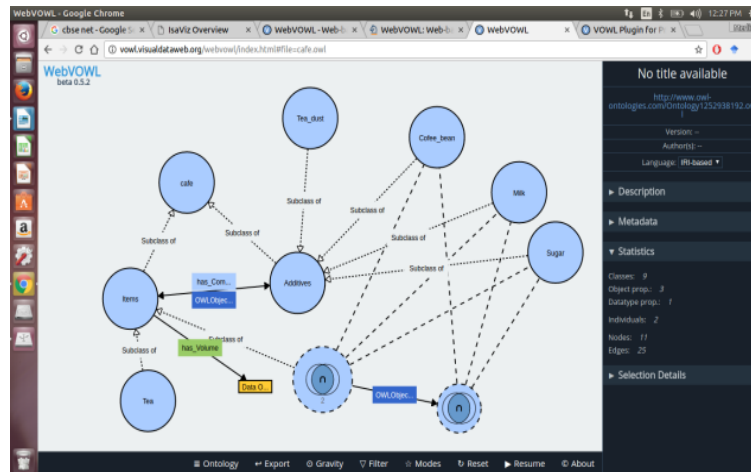


Figure 7. Visualization of café shop in webVOWL

c. WordVis

WordVis is a web application which implant GraphViz module and provides the meanings of the words in a large graph [16]. It uses WordNet as a source to search words and find the meanings of them. It is developed using JavaScript and HTML5. It has the following features: auto-completing search box, node environments, filters, results panel, precalculation and page load speed optimizations. Auto-completing search box is used to find the meaning of the word from the WordNet by typing the word in the search box which also facilitates the user to type comma separated words to focus on several words. Node environment is the panel that is used to display the words and the meaning of the words in three dimensions. Filters are used to customize the visualization of the graph by unfolding or deselecting the options and give the results. Results panel is the place where the graph is displayed. It displays the meaning of the word along with their relations. Precalculation is used to evaluate the levels of the graph connection. This enables the user to visualize 305.000 nodes support in the result panel. Page load speed optimization is used to speed-up the visualization by adding JavaScript into the page. WordVis is an open source online synonym-dictionary explorer that used to find the meaning of the English terms. It is flexible with different browsers. The sample visualization of WordVis is given in Figure 8.

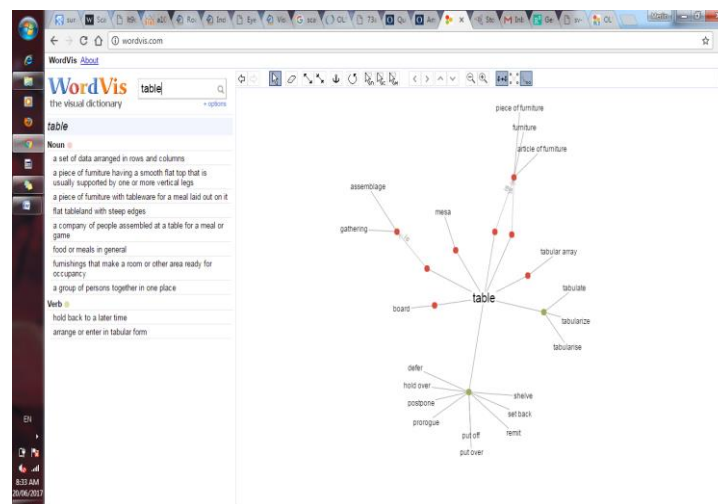


Figure 8. Sample visualization of wordVis

d. OLSVis

OLSVis is an ontology lookup service (OLS) developed to visualize ontology interactively from OLS repository [17]. It is an application of WordVis for ontology. OLSVis provides list of ontologies to be visualized in the form of list and allows the user to either select or search from the search box. OLS is mainly

developed for Bio-medical ontologies and it is updated weekly. It visualizes ontologies as given in Figure 9. It also gives the synonym of the terms. User can visualize and customize ontologies and their subgraphs. Subgraphs use real-time force-based layout algorithm to animate the visualization. When the user searches, adds, drags and hides terms the algorithm used to reorganize the visualization of the graph. At present, OLSVis contains 191 ontologies and 4.891.616 terms.

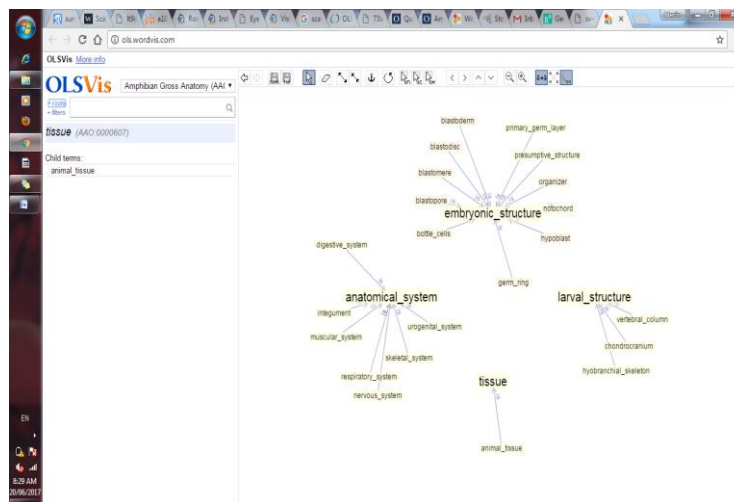


Figure 9. Visualization of OLSVis

e. REVIGO

REVIGO is a web browser that takes lists of gene ontology (GO) terms [18], [19]. It is developed to visualize GO ontology by removing redundant of its terms. The terms are visualized in the form of similarity-based scatterplots, interactive graphs, or tag clouds. Scatterplot displays the GO terms in the form of 2D bubbles. Interactive graphs are used to connect the searched GO terms based on the GO hierarchy. TreeMap displays the terms in clustered manner in colored tiles. Finally REVIGO provides a word cloud which is used to highlight frequently used keywords, name and descriptions of the GO terms. The sample visualization of REVIGO is given in Figure 10. It contains 100 biomedical ontologies. The Features of web-based ontology visualization tools are compared and tabulated in Table 2.

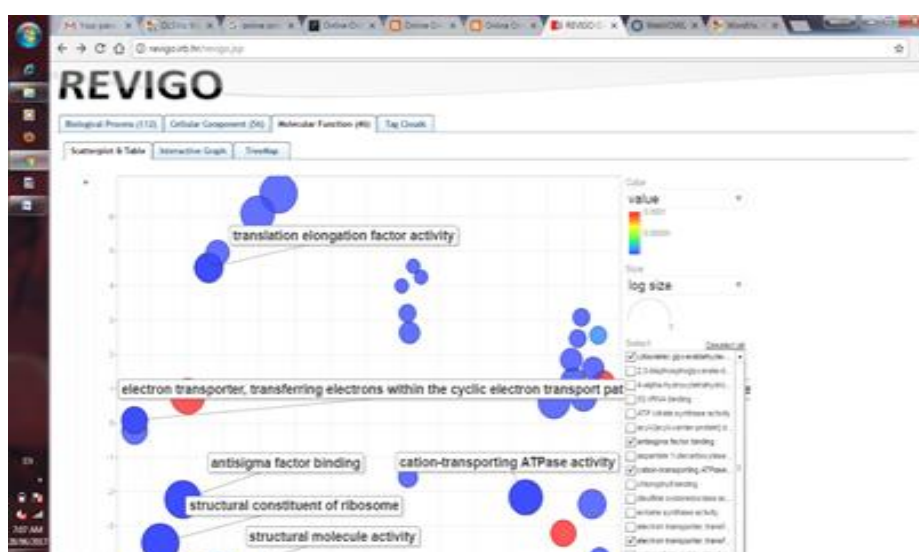


Figure 10. Visualization of GO in REVIGO

Table 2. Comparison of web-based ontology visualization

Web Tools	Source	Domain	Availability	Representation	URL
OwlGrEd	Any ontology files	General	Y	UML Models	http://owlgred.lumii.lv/online_visualization
WebOWL	Any ontology files	General	Y	Round	http://visualdataweb.de/webvowl/
WordVis	WordNet	English Dictionary	Y	Terms	http://wordvis.com/
OLSVIS	BioMedical Ontologies	Medical	Y	Terms	http://ols.wordvis.com/
REVIGO	Gene Ontologies	Medical	Y	scatterplots, interactive graphs, tag clouds	http://revigo.irb.hr/revigo.jsp

3.2. Ontology visualization plug-ins

Ontology visualization plug-ins are developed to integrate with ontology editors. The plug-ins are used to visualize the taxonomical structure, relations and axioms of ontology. In this section, visualization plug-ins used in Protégé [20], NeOn Toolkit [21], SWOOP [22] and ONTOLIS [23] ontology editors is described.

3.2.1. Protégé

Protégé is an ontology editor developed to build ontologies and knowledge management systems. It is used to define ontologies by using the graphical user interface (GUI). It is built with number of plug-ins to perform various operations like evaluation, visualization. It is identified that number of ontology visualization plug-ins are developed in Protégé namely OntoViz, OntoGraf, OWLViz, Jambalaya, TGViz, OntoSphere and ProtegeVOWL. This section elaborates the ontology visualization plug-ins used in Protégé Editor.

a. OntoViz

OntoViz tool is an ontology visualization plug-in which allows the user to visualize ontology [24]. Graphviz software is used to visualize ontology efficiently. This plug-in visualizes a set of classes and instances, slots and its edges. It allows the user to change the colors of the nodes and edges. It produces various closure operators such as sub - subclass closure, sup - superclass closure, slx - slot extension, isx - inverse slot extension, slt-slots, sle-slot edges, ins-instances and sys-system frames to tune the graph. The sample OntoViz visualization of Newspaper ontology is shown in Figure 11.

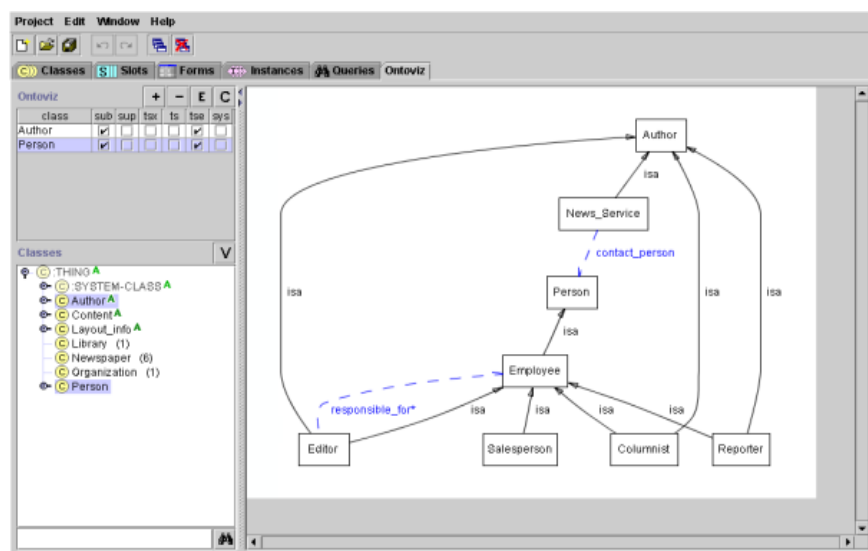


Figure 11. Illustration of ontoviz for newspaper ontology

b. OntoGraph

OntoGraph is one of the visualization plug-ins used in protégé [25]. It supports various relationships like subclass, individual, domain/range object properties, and equivalence of OWL ontologies. Ontologies are

visualized in different layouts such as radial, spring, tree horizontal and tree vertical. Relationships and node types are filtered to create user desired view. Figure 12. Illustrates the visualization of café shop ontology using OntoGraph plug-in.

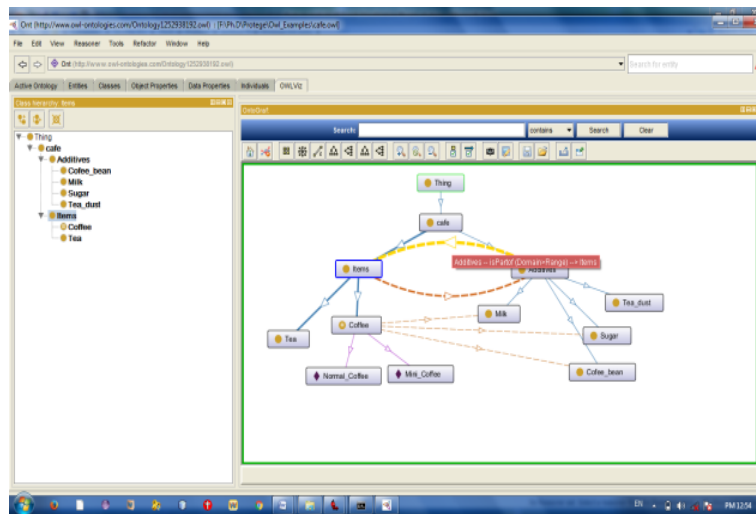


Figure 12. Illustration of ontograf plug-in for café shop

c. OWLViz

OWLViz is an ontology visualization plug-in [26] used to view class hierarchies of OWL ontology. It compares the asserted class hierarchy and the inferred class hierarchy as shown in Figure 13. It represents everything in same color not to distinguish the defined classes and primitives. The class hierarchy is visualized clearly and inconsistent concepts are highlighted in red color. OWLViz allows the user to save both the asserted and inferred views of the class hierarchy to various concrete graphics formats including PNG, JPEG, and SVG.

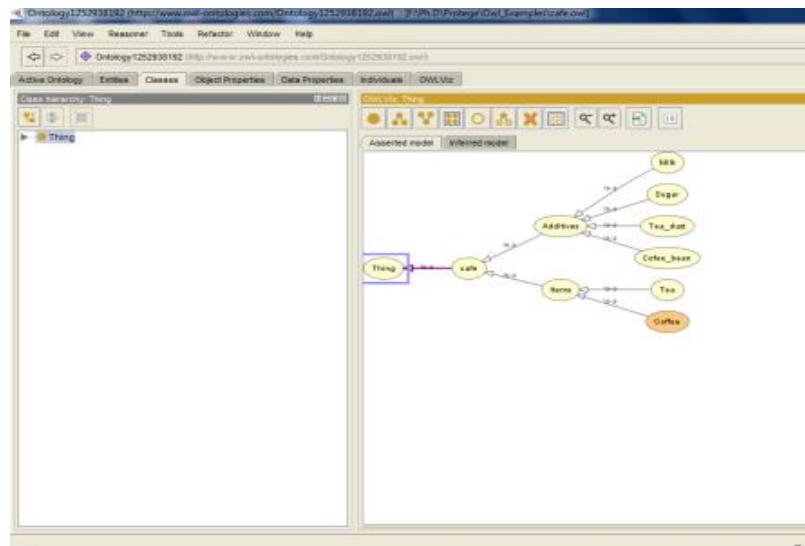


Figure 13. Illustration of OwlViz plug-in for café shop

d. Jambalaya

It is also ontology visualization plug-in used in protégé editor [27]. The classes and instances are visualized in jambalaya main view as depicted in Figure 14. The classes are represented as nodes and slots or

properties are represented as arcs. Jambalaya provides two types of overlapping windows for filters, namely node filter and arc filter. Node type filter is used to sort concepts based on six primitives: defined class, enumeration, individual, logical operation, primitive class and RDF class. They are used to change the visibility of the types of the nodes. Arc filter provides different types of arcs that are corresponding to the slots or properties. All these arc types are classified under four categories: class-instance hierarchy, domain and range, property on individuals and property restriction on classes. It also provides six types of customized views of the ontology: nested view, nested tree map, nested composite view, class and individual tree, class tree and domain/range. It allows the user to sort the graph by alphabetical order, number of children, number of relationship, type and other attributes.

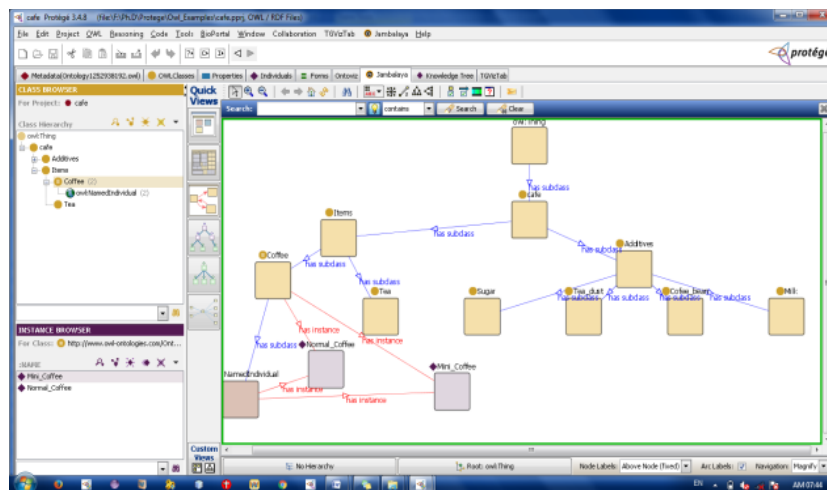


Figure 14. Example of jambalaya for café shop ontology

e. OntoSphere

The OntoSphere is another tool used in protégé to visualize ontologies in 3D view [28]. It displays the classes and properties as node-link and tree method. It provides three types of visualizations: root focus scene, tree focus scene and concept focus scene. Root focus scene provides the sphere like earth which represents the collection of concepts. The concepts are represented as small spheres and placed over the big sphere. Tree focus scene expresses the sub-tree or subclasses of the concepts in hierarchical format. It also represents the relationship between the classes. Concept focus scene is used to produce the entire detail of the particular concept. It displays the concept detail, parent and child class of the concept and relationships related to that concept. OntoSphere used Java 3D API for an effective 3D visualization of ontologies. This API is connected with OpenGL engine to produce the graphical representation. Overview taxonomy structure and properties cannot be viewed in this tool.

f. TGVizTab

Touch Graph Visualization tab (TGVizTab) is ontology visualization tab that uses TouchGraph libraries for visualizing ontologies efficiently [29]. TouchGraph provides Java library for rendering networks as interactive graphs. This tab is used to visualize the classes, instances and slots. It allows the user to change the colors of the graph and nodes. It allows the user to save the graph in extensible markup language (XML) format which is visualized in other TouchGraph applications. Classes and instances are selected based on the demand of the user to visualize. The sample visualization of TGVizTab for newspaper ontology is shown in Figure 15.

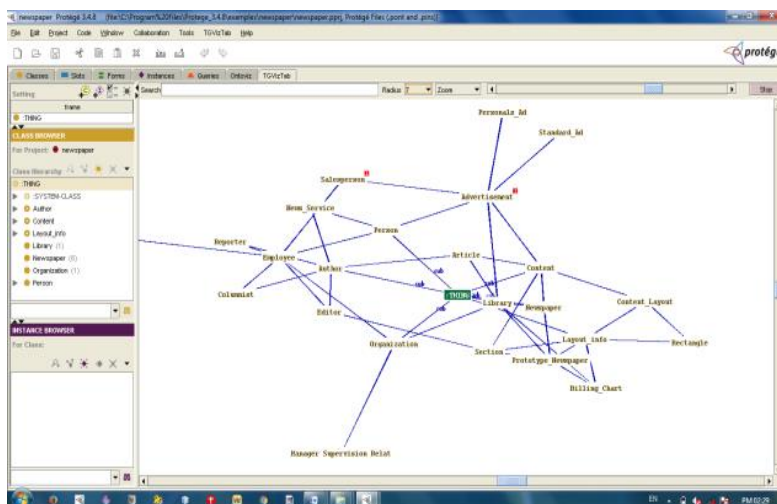


Figure 15. Visualization of newspaper ontology in TGVizTab

g. ProtegeVOWL

ProtégéVOWL is a protégé plugin for the user-oriented visualization of ontologies [30]. It implements the visual notation for owl ontologies (VOWL). VOWL defines a visual language for the user-oriented representation of ontologies. It also provides graphical depictions for the OWL elements. The elements are combined as force-directed graph layout. It is used to visualize ontology schema which is also called as TBox. TBox is a collection of classes, properties and data types. ProtégéVOWL does not support ABox that is individuals and data values. Visualization of café shop ontology in protégéVOWL is depicted in Figure 16.

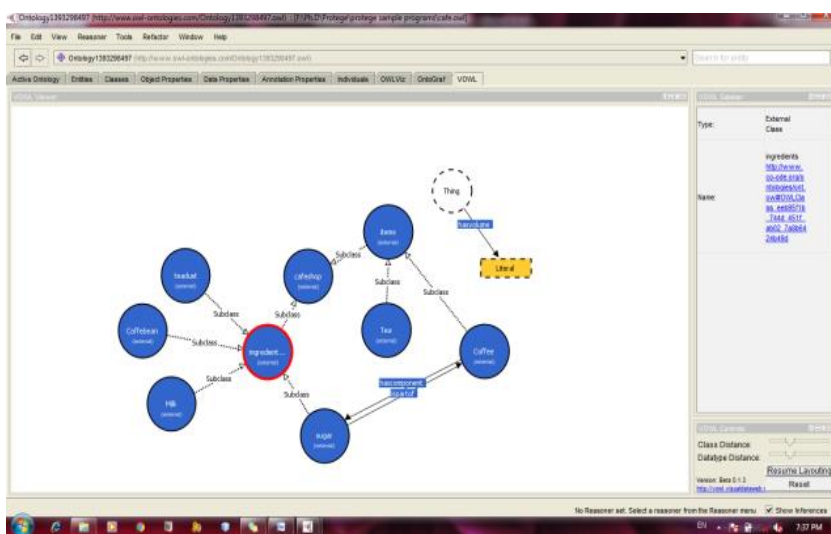


Figure 16. Visualization of café shop in protegeVOWL

3.2.2. SWOOP

a. CropCircles

CropCircles is a visualization plug-in used in SWOOP ontology editor [31]. CropCircles represents the nodes of the ontology in circle. Child circles are nested in its parent circle. The size of the subtree defines the diameter of the circle. The smallest subtree is known as a leaf node which has the size of all circles. CropCircles allows to sort the children node in descending order according to the size of subtrees. The larger nodes are explored first and they occupy more space. CropCircles has two components: panel and visualization

area. Panel contains the details of the ontology or the node to be visualized or focused. Visualization area displays the concept map of the ontology as shown in Figure 17.

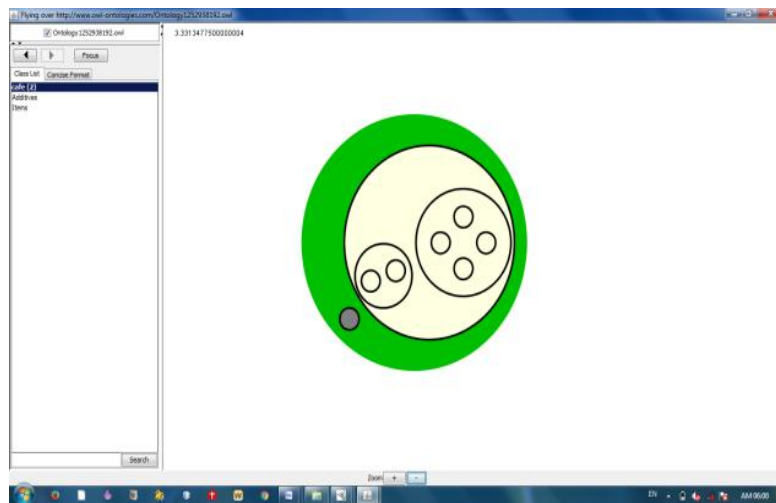


Figure 17. Visualization of cafeshop in cropcircles

3.2.3. Neon toolkit

a. KC-Viz

KC-Viz is a key concept visualization plug-in used in NeOn toolkit [32]. It is used to generate complete view of ontology. Concepts are represented in blue hexagon with specific dimension. The concepts will have two numbers in brackets which represent the number of direct and indirect subclasses of the node. KC-Viz provides two facilities namely KC-Viz toolbar and KC-Viz preferences. KC-Viz toolbar provides various options to change the view of the ontology like undo/redo, adjust gap, adjust font, zoom in/out and reset the view. KC-Viz preferences provide the accessibility such as Maximal size of ontology summary, Maximal number of non-key concepts, and use of sub-optimal algorithm and consideration of imported ontologies. These are used to improve the visualization of ontology efficiently. Figure 18 signifies the sample representation of café shop ontology in KC-Viz.

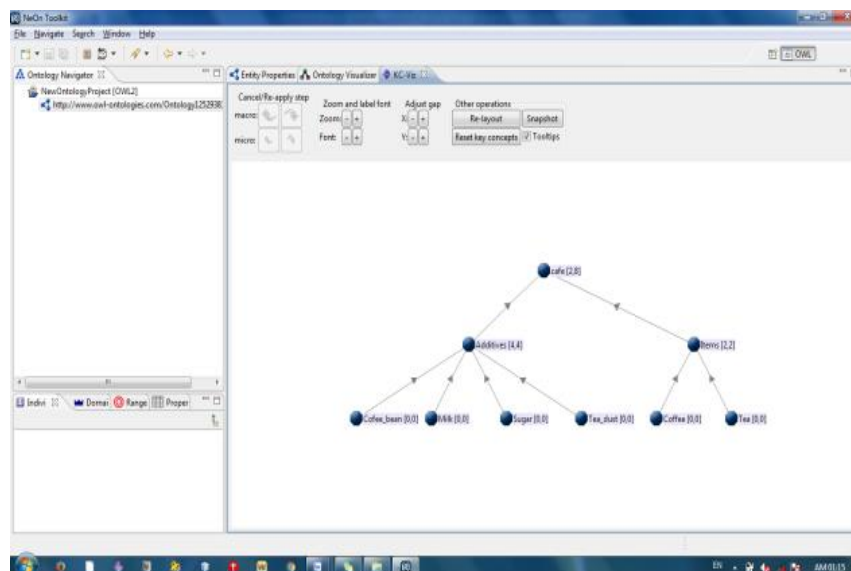


Figure 18. Visualization of café shop in KC-Viz

b. Ontology visualizer

Ontology visualizer panel is used to display ontology elements in NeOn Toolkit [33]. It is used to display the hierarchy of classes, individuals and object properties in a directed graph. It provides the zooming and rotating functions. The layout of the graph is optimized automatically by clicking on an element. The arrows are represented in three colors which denote three various properties. Grey represents the connection between superclasses and subclasses or between a superclass and an ontology root. Orange represents the connection between classes and properties. Red represents the connection between classes and relations. The sample of café shop ontology is visualized in ontology visualizer as shown in Figure 19.

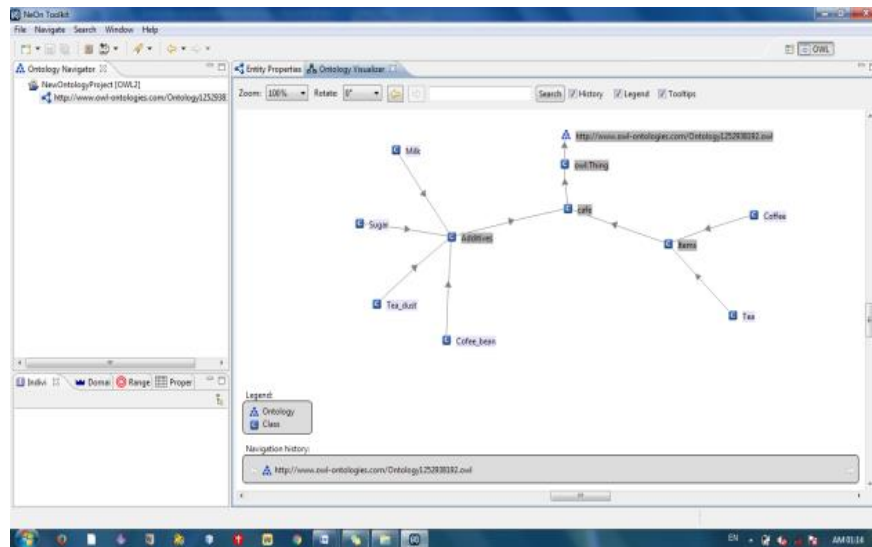


Figure 19. Visualization of café shop ontology in ontology visualizer

3.2.4. ONTOLIS

It's a combination of ontology semantic visualization tool and cognitive graphical representation techniques to ease the task of ontology development. It is implemented with the system called reply, which is used to develop web services. It provides natural language query system to retrieve data effectively. This tool provides the scientific visualization tool called SciViz which enables the students to visualize the ontologies in effective manner. The ontology visualization plug-ins used in different ontology editors like protégé, NeOn Toolkit and SWOOP are compared and tabularized in Table 3.

Table 3. Comparative analyzes of ontology visualization tools and plug-ins (a)

Visualization Tool/Plug-in name	Primitives		Data Properties	Individuals	Relations	Annotations
	Class	Object Properties				
OntoViz	Y	N	N	N	IS-A	N
OntoGraph	Y	Y	N	Y	hasSubclass hasIndividuals	Y
OwlViz	Y	Y	N	N	Is-A	N
Jambalaya	Y	Y	Y	Y	hasSubclass hasInstance	Y
OntoSphere	Y	Y	Y	Y	Subclass-of	N
CropCircles	Y	Y	N	N	Parent-child	N
IsAviz	Y	Y	N	Y	Is-a	Y
GrOWL	Y	Y	Y	Y	User defined	N
TGVizTab	Y	Y	N	Y	Super	N
KC-Viz	Y	Y	N	N	SubclassOf Part-of	Y
Ontology Visualizer	Y	Y	Y	Y	OBO_REL: is_a	N
RDF Gravity	Y	Y	Y	Y	User defined	N
OWLEasyViz	Y	Y	Y	Y	IS-A	N
Protégé VOWL	Y	Y	Y	N	Subclass-of	Y

Table 3. Comparative analyzes of ontology visualization tools and plug-ins (b)

Visualization Tool/Plug-in name	User Interface							
	Layout	Dimension	Zoomable	Search	Directions	Adjustable	Movable	Representation
OntoViz	Tree	2D	Y	N	Uni-direction	N	N	Ellipse
OntoGraph	Grid-Alphabetical Radial Spring Tree	2D	Y	Y	Uni-direction	N	Y	Rectangle box
OwlViz	Tree	2D	Y	N	Uni-direction	N	N	Ellipse
Jambalaya	Grid Radial Spring Tree	2D	Y	Y	Uni-Direction	N	Y	Rounded Rectangle
OntoSphere	Sphere Tree	3D	Y	N	Uni-Direction	Y	Y	Sphere
CropCircles	Circle	2D	Y	Y	Uni-Direction	Y	Y	Circle
IsAviz	Table Node and Arc Forced directed layout	2.5D	Y	Y	Undirected Unidirection	Y	Y	Ellipse Boxes
GrOWL		2D	Y	Y	Uni-Direction	Y	Y	Rectangle
TGVizTab	Tree	2D	Y	Y	Uni-Direction	N	Y	Text
KC-Viz	Tree	2D	Y	N	Uni-Direction	Y	Y	Hexagon
Ontology Visualizer	Spring	2D	Y	Y	Uni-Direction	Y	Y	Box
RDF Gravity	Radial	2D	Y	Y	Uni-Direction	Y	Y	Rectangle
OWLEasyViz	Tree	2D	Y	Y	Uni-Direction	Y	Y	Ellipse Rounded rectangle
Protégé VOWL	Spring	2D	Y	N	Uni-Direction	Y	Y	Circles

Table 3. Comparative analyzes of ontology visualization tools and plug-ins(c)

Visualization Tool/Plug-in name	Environment							
	Compatibility	Availability	Import	Export	Dependencies	Platform	OL Support	NL Support
OntoViz	Protégé 3.4	Y	N	Y	Independent	Java	OWL RDF	N
OntoGraph	Protégé 4.0	Y	N	Y	Independent	Java	OWL	N
OwlViz	Protégé 3.4	Y	N	Y	GraphViz	Java	OWL	N
Jambalaya	Protégé 3.4.8	Y	Y	Y	Protégé-Owl	Java	OWL	N
OntoSphere	Protégé	Y	Y	N	Java 3D	Java	OWL	N
CropCircles	SWOOP	Y	Y	N	Independent	Java	OWL	N
IsAviz	Stand-alone	Y	Y	Y	Graphviz Dot	Java	OWL	N
GrOWL	Protégé Stand-alone	Y	N	Y	Independent	Java	OWL	N
TGVizTab	Protégé	Y	N	N	Independent	Java	OWL	N
KC-Viz	NeOn	Y	Y	Y	Independent	Java	OWL	Y
Ontology Visualizer	NeOn OntoStudio	Y	N	N	Independent	Java	OWL	Y
RDF Gravity	Standalone	Y	Y	Y	JVM	Java	RDF, OWL	N
OWLEasyViz	Standalone	Y	N	Y	Independent	Java	OWL	N
Protégé VOWL	Protégé	Y	Y	N	Independent	Java	OWL	N

4. RESULTS AND DISCUSSION

In this article, ontology visualization plug-ins used in protégé, SWOOP, NeOn Toolkit and stand alone ontology visualization tools are collected for review. The common features and characteristics of these tools and plug-ins are classified into three categories namely primitives, user interface and environment. These categories are further subdivided into 22 criteria: class, object properties, data properties, individual, relations, layout, dimensions, zoom, search, annotations, compatibility, availability, import, export, directions,

Feature analysis of ontology visualization methods and tools (Merlin Florence Joseph)

representation, resizable, movable, dependencies, platform, ontology language support, and natural language support. Primitives contain the elements such as classes, object properties, data properties, individual, annotations and relations which are used to represent ontology. Classes represent the concepts of the domain. Object properties are used to define binary relation between the concepts. Data properties are used to define concept attributes. Individual represents the instances of classes. Annotation properties are used to give the additional details about classes and properties.

User interface depicts the features given by the ontology visualizations to view the ontology effectively. It includes the layout, dimensions (2D/3D), zoom, search, representation, direction, resize and movable and different kinds of layouts like radial, spring, tree, matrix, circle and so on. The types of layout used in the visualization are either two dimensions or three dimensions. It also describes that the classes are zoomable, searchable, resizable, and movable of the nodes. The environment contains details about the visualization tools and plug-ins. It reveals the compatibility and availability of the visualizations, the programming languages used to develop the tools and plug-ins. The ontology file format, which is supported by the tool, is also given. This environment criterion comprises the natural languages supported by the visualization tools. All the tools and plug-ins considered for the study are compared under these criteria and the features are tabulated in Table 3. It is learned from the analysis, that very few ontology visualization tools are developed to support non-English languages. Most of the visualization tools and plug-ins need to be configured to support natural languages.

5. CONCLUSIONS

Data visualizations are commonly used in education domain to attract the user towards learning and understanding the content easily and rapidly. Semantic web techniques are applied in data visualization to perform knowledge based visualization. Many tools and plug-ins are developed to visualize ontologies effectively. This article analyzed the features of existing ontology visualization tools and plug-ins. the characteristics of these tools and plug-ins are grouped into three major categories. The common features of these visualizations are compared with several criteria. From this survey, it is identified that very few ontology visualization tools are developed to support multilingual. Some of the tools like jambalaya, OntoGraph require additional configuration to support non-English languages. Most of the tools are implemented as plug-ins to work with ontology editors like protégé, OBOEdit, NeOn toolkit etc. GrOWL, OWLGrEd, Glow and NavigOWL are developed as standalone applications which run independently without ontology editors. The existing visualization plug-ins and tools are not clear in case of large number of concepts. This can be addressed in future and the problems can be resolved by improving the workspaces. The tools and plug-ins can be developed to support multilingual. Web-based ontology visualization tools can also be implemented to visualize ontology using cloud services.

REFERENCES

- [1] A. Gomez-Perez, M. Fernández-López, & O. Corcho, "Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web," *Springer Science & Business Media*, 2006.
- [2] M. Afzal, "Graph-Based Visualization of Ontology-Based Competence Profiles for Research Collaboration," Independent thesis Advanced level (degree of Master (Two Years)), JTH. Research area Information Engineering 2012.
- [3] A. Katifori, C. Halatsis, G. Lepouras, C. Vassilakis, & E. Giannopoulou, "Ontology visualization methods a survey," *ACM Computing Surveys (CSUR)*, vol. 39, no. 4, pp. 10, 2007, doi: 10.1145/1287620.1287621.
- [4] R. Sivakumar, & P. V.Arivoli, "Ontology visualization PROTÉGÉ tools—a review," *International Journal of Advanced Information Technology (IJAIT)*, vol. 1, no. 4, Aug 2011, doi: 10.5121/ijait.2011.1401.
- [5] V. Swaminathan, & R. Sivakumar, "A comparative study of recent ontology visualization tools with a case of diabetes data," *International Journal of Research in Computer Science*, vol. 2, no. 3, pp. 31-36, 2012, doi: 10.7815/ijorcs.23.2012.026.
- [6] R. Lourdusamy, V. Arulnathan, & G. Ganapathy, "A Survey on Ontology Visualizations, Tirupattur, Vellore," *National Conference on Software Engineering and Applications*, 2008.
- [7] S. Mikhailov, M. Petrov, & B. Lantow, "Ontology Visualization: A Systematic Literature Analysis," In *BIR Workshops*, 2016.
- [8] S. Krivov, R. Williams, & F. Villa, "GrOWL: A tool for visualization and editing of OWL ontologies," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 5, no. 2, pp. 54-57, 2007, doi: 10.1016/j.websem.2007.03.005.
- [9] N. Catenazzi, L. Sommaruga and R. Mazza, "User-Friendly Ontology Editing and Visualization Tools: The OWLeasyViz Approach," *2009 13th International Conference Information Visualisation*, 2009, pp. 283-288, doi: 10.1109/IV.2009.34.

- [10] A. Hussain, K. Latif, A. T. Rextin, A. Hayat, & M. Alam, "Scalable visualization of semantic nets using power-law graphs," *Applied Mathematics & Information Sciences*, vol. 8, no. 1, pp. 355-367, Jan 2014, doi: 10.12785/amis/080145.
- [11] S. Goyal, & R. Westenthaler, "RDF gravity (RDF graph visualization tool)," Salzburg Research, Austria, 2004.
- [12] Pietriga, E. Isaviz, "A visual authoring tool for RDF. World Wide Web Consortium," [Online] Available: <http://www.w3.org/2001/11/IsaViz>, 2003.
- [13] T. Liebig, & O. Noppens, "OntoTrack: Fast Browsing and Easy Editing of Large Ontologie," *Proceedings of the 2nd International Workshop on Evaluation of Ontology-based Tools held at the 2nd International Semantic Web Conference (ISWC)*, 2003.
- [14] K. Cerans, J. Ovcinnikova, R. Liepins, & A. Sprogis, "Advanced OWL 2.0 ontology visualization in OWLGrEd." In *Databases and Information Systems VII—Selected Papers from the 10th Int. Baltic Conference (DB&IS)*, pp. 41-54, 2012, doi: 10.3233/978-1-61499-161-8-41.
- [15] S. Lohmann, V. Link, E. Marbach, & S. Negru, "WebVOWL: Web-based visualization of ontologies," In *International Conference on Knowledge Engineering and Knowledge Management*, Springer, Cham. pp. 154-158, 2014, doi: 10.1007/978-3-319-17966-7_21.
- [16] S. Vercruysse, & M. Kuiper, "WordVis: JavaScript and animation to visualize the WordNet relational dictionary," In *Proceedings of the Third International Conference on Intelligent Human Computer Interaction (IHCI)*, Prague, Czech Republic, Springer, Berlin, Heidelberg. pp. 137-145, 2011, doi: 10.1007/978-3-642-31603-6_12.
- [17] S. Vercruysse, A. Venkatesan, & M. Kuiper, "OLSViz: an animated, interactive visual browser for bio-ontologies," *BMC bioinformatics*, vol. 13, no. 1, pp. 116, 2012, doi: 10.1186/1471-2105-13-116.
- [18] F. Supek, M. Bošnjak, N. Škunca, & T. Šmuc, "REVIGO summarizes and visualizes long lists of gene ontology terms," *PloS one*, vol. 6, pp. e21800, 2011, doi: 10.1371/journal.pone.0021800
- [19] TouchGraph, LLC, "TouchGraph Navigator," [Online] Available: <http://www.Touchgraph.com/>, 2008.
- [20] J. H. Gennari *et al.*, "The evolution of Protégé: an environment for knowledge-based systems development," *International Journal of Human-computer studies*, vol. 58, no. 1, pp. 89-123, 2003, doi: 10.1016/S1071-5819(02)00127-1.
- [21] P. Haase *et al.*, *The neon ontology engineering toolkit*, WWW 2008, 2008.
- [22] A. Kalyanpur, B. Parsia, E. Sirin, B. C. Grau, & J. Hendler, "Swoop: A web ontology editing browser," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 4, no. 2, pp. 144-153, 2006, doi: 10.1016/j.websem.2005.10.001.
- [23] S. Chuprina, O. Nasraoui, "Using Ontology-Based Adaptable Scientific Visualization and Cognitive Graphics Tools to Transform Traditional Information Systems into Intelligent Systems," *Journal on Scientific Visualization*, vol. 8, pp. 23-44, 2016.
- [24] M. Sintek, "OntoViz," [Online] Available: <https://protegewiki.stanford.edu/wiki/OntoViz>, 2007.
- [25] S. Falconer, "OntoGraf," [Online] Available at: <https://protegewiki.stanford.edu/wiki/OntoGraf>, 2010.
- [26] M. Horridge, *OWLviz-A visualisation plugin for the Protégé OWL Plugin*, The University of Manchester, 2008.
- [27] M. A. Storey *et al.*, "Jambalaya: Interactive visualization to enhance ontology authoring and knowledge acquisition in Protégé," In *Workshop on interactive tools for knowledge capture*, 2001.
- [28] A. B. Bosca, D. Bonino, & D. Pellegrino, "OntoSphere: more than a 3D ontology visualization tool," *SWAP 2005*, in *the 2nd Italian Semantic Web Workshop*, 2005.
- [29] H. Alani, "TGVizTab: An ontology visualization extension for protégé," in *Knowledge Capture 03-Workshop on Visualizing Information in Knowledge Engineering*, Sanibel Island, FL, 2003.
- [30] Lohmann, S., Negru, S., Haag, F., & Ertl, T., "VOWL 2: User-oriented visualization of ontologies," In *International Conference on Knowledge Engineering and Knowledge Management*, Springer, Cham, pp. 266-281, 2014, doi: 10.1007/978-3-319-13704-9_21.
- [31] B. Parsia, T. Wang, & J. Golbeck, "Visualizing web ontologies with cropcircles," In *Proceedings of the 4th International Semantic Web Conference*, 2005.
- [32] M. Dzbor, S. Peroni, E. Motta, & M. d'Aquin, "D4.5.4 NeOn Toolkit plug-in for visualization and navigation in ontologies and ontology networks based on concept summarization and categorization," *NeOn: Lifecycle Support for Networked Ontologies*, *NeOn Integrated Project EU-IST-027595*, 2006-2009.
- [33] M. Erdmann, & W. Waterfeld, *Overview of the neon toolkit*. In *Ontology Engineering in a Networked World*, Springer Berlin Heidelberg, pp. 281-301, 2012.