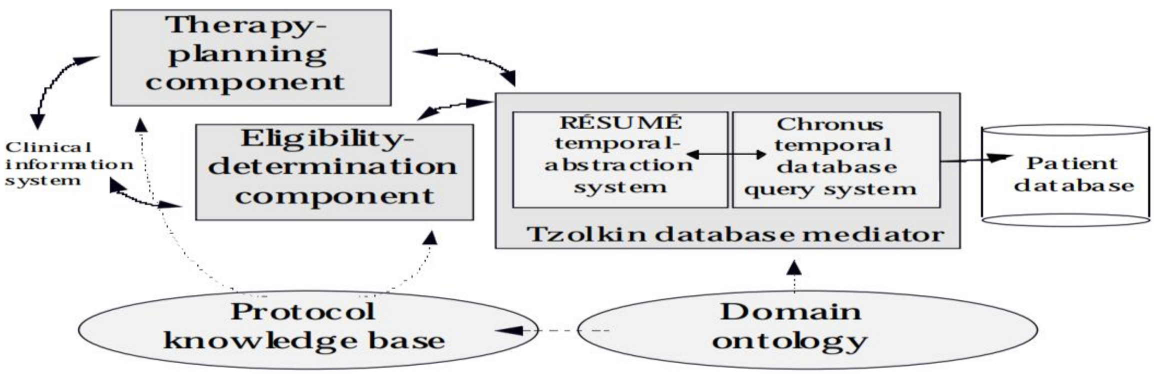
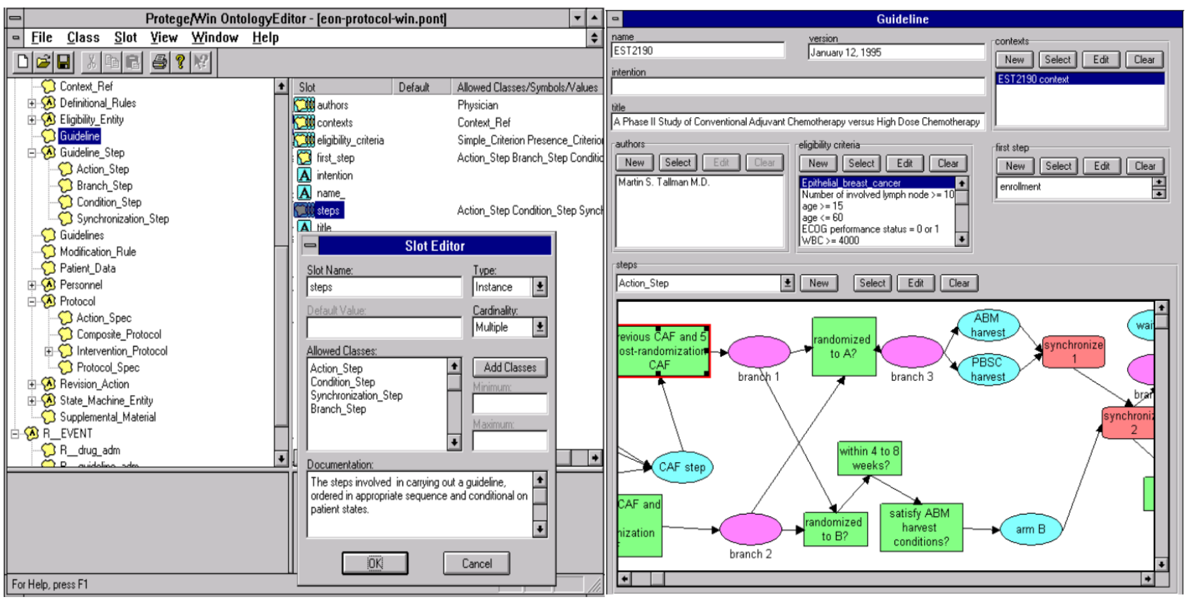
Paper Review

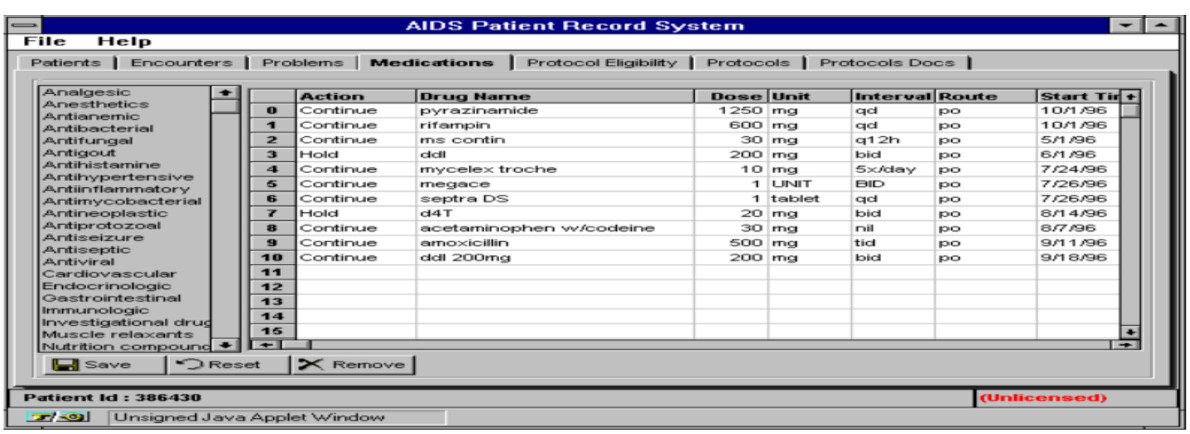
Group: Ontology Graphical UI
Supervised By:
Dr. Mohammad Ashrafuzzaman Khan
Assistant Professor,
ECE Department,North South University.
Submitted By:
Arona Dorin Chowdhury 1520045642
Mohammad Raihan Sarker Razu 1520079042
Tamim Ahmed 1520698642

1. Mark A. Musen proposed EON model architecture using protégé software engineering. EON is a set of middleware components which aspects various types of protocol directed therapy. For this reason, he mainly concentrate on domain ontology works. EON Architecture model is given below.



In this paper, Musen mainly discussed about patient record system using protégé.





2. Margaret Anne Storey, Mark Musen, John Silva, Nathasha Roy, Rey Ferguson, Neil Ernst, Casey Best proposed SHriMP technic which means Simple Hierarchical Multi Perspective . According to this technic,it enhances how people browse and explore information spaces. Though SHriMP is mainly used for visualizing and documenting softwares.

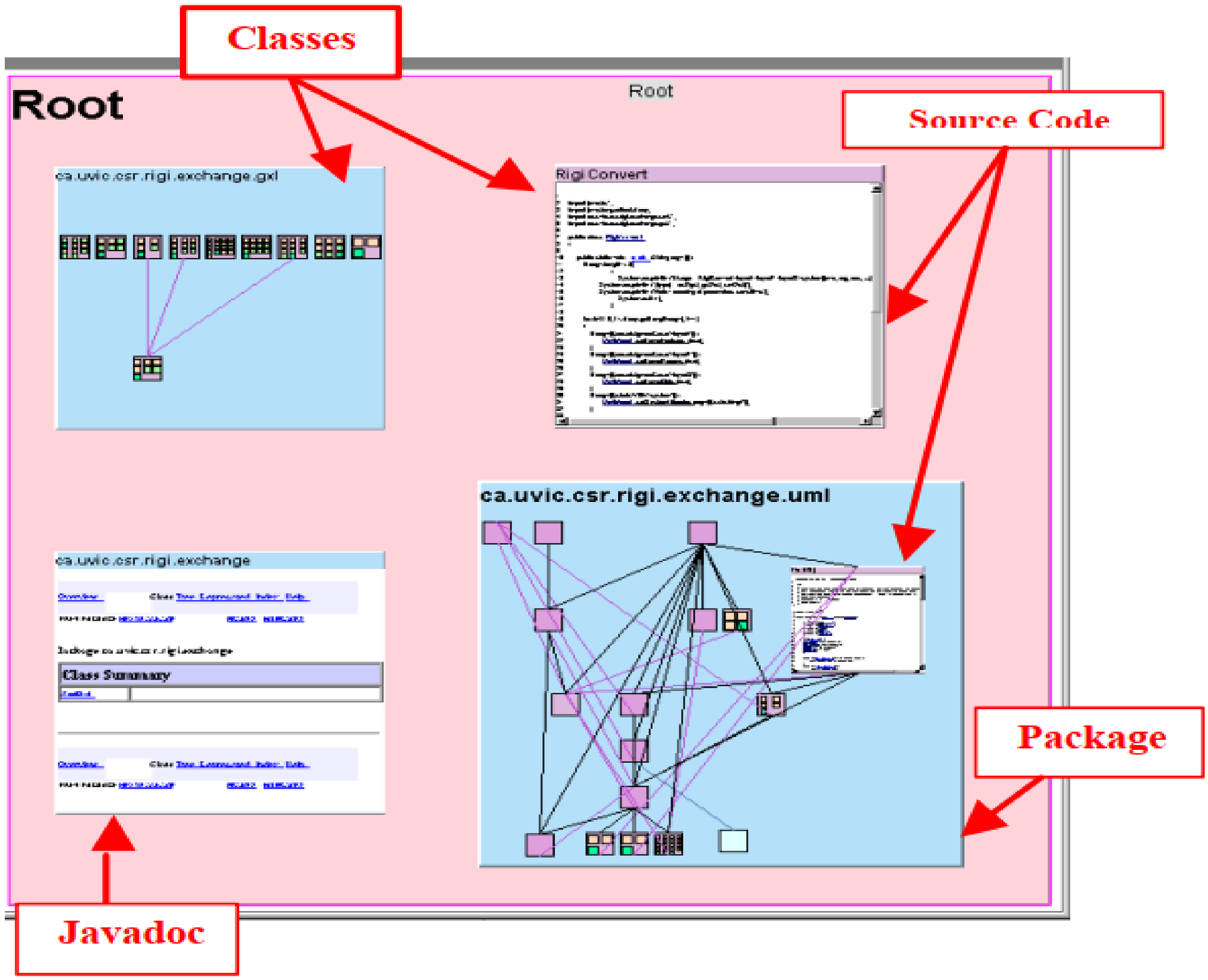


Figure 3: Visualizing a Java program using SHriMP.

3. Junli wang, Zhijung ding,Changjung Jiang proposed ontology based public transport system by using protégé and jena software. The model was mainly established because of differentiating same names of different stations and querying with more semantic information. They also build an algorithm acoording to that problem and got good result.

#Algorithm:

Table 1. Querying Algorithm between stations

```
Input (start, destination, user-priorities)
    {
       Create the relation matrix, R, between stations
and routes;
       //Each value in the matrix marks the order number
of a given station in a certain route;
       for each route i do,
           {if R[start, i] <∞, then
                 if R[destination, i]<∞, and R[destination,
              i]>R[start, i], then
                      Output (route i, the direct route);//
              zero-transfer trip
            else for each station j, do
                 if (R[j, i]) >R[start, i], then
                     for each route k, do
                        if R[j, k]<R[j, Destination], then
                           Output (route i, station j, route k);//
              one-transfer trip
           }
    }
```

#Model and Result:

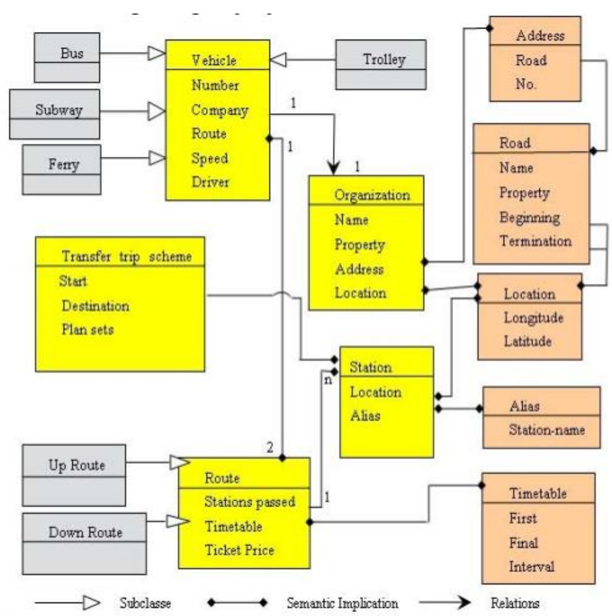


Fig. 1. Urban public transport ontology

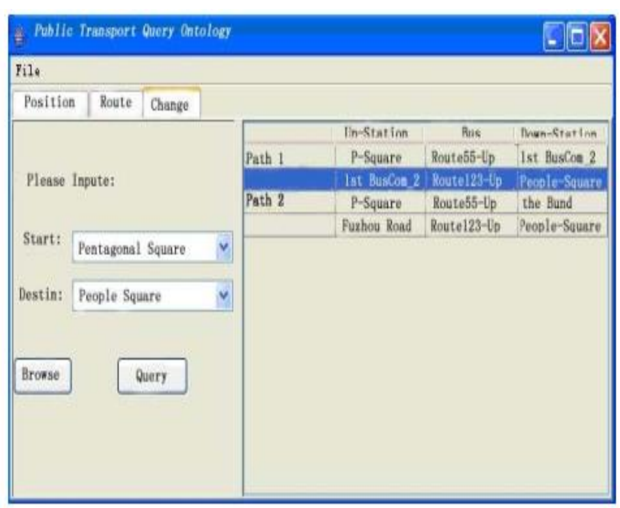


Fig. 2. Interface of the PTQS and experimental result

4. F. Abdoli and M. Kahani discussed about IDS which is Intrusion Detection System. For this reason, they used Ontology model,protégé software and Jena framework to make interactions between MasterAgent and attacks ontology.

#Proposed Model:

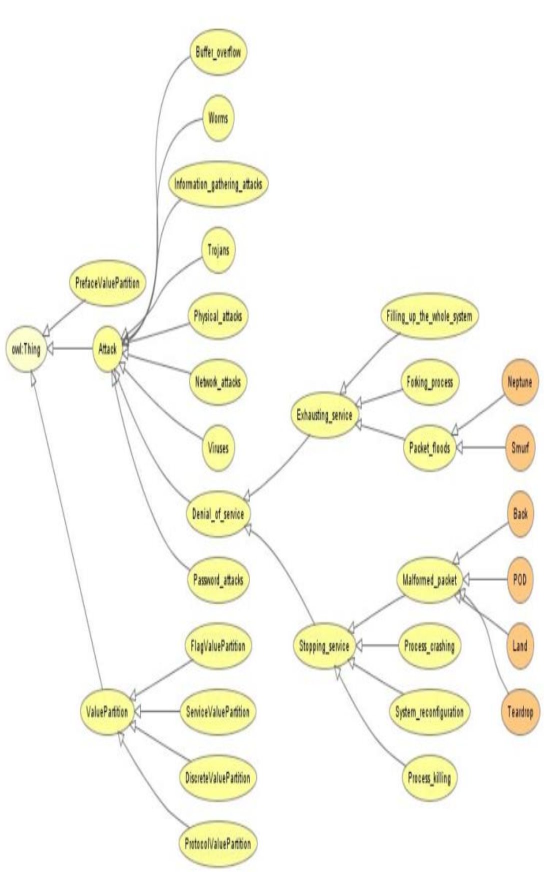


Figure 2. High Level Illustration of the Proposed Ontology

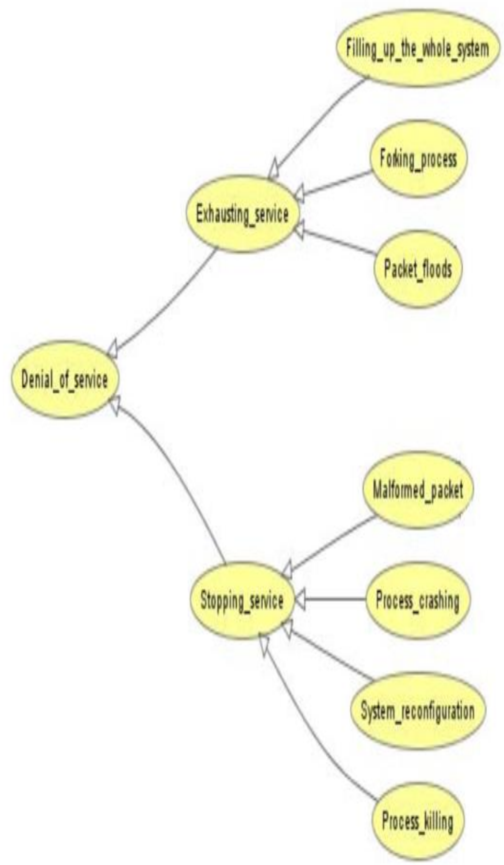


Figure 3. Illustration of the Denial of Service class

#Formula And Results:

$$CPE = \frac{1}{N} \sum_{i=1}^{m} \sum_{j=1}^{m} CM(i, j) * C(i, j)$$

TABLE I.

CPE RESULTS FOR THE ODIDS SYSTEM

The tested dataset	N	CPE	
The labeled 10% KDDcup 99	٤٩٤٠٢٢	0.012	
The labeled KDDcup 99	٢٥٠٠٠	0.016	
The labeled KDDcup 99	٦٤٥٠٠	0.017	
		0.015	(Average) CPE

TABLE II.

COMPARISON OF ODIDS SYSTEM AND OTHER ALGORITHM

CPE	FA	DoS	algorithms
0.015	2.5	99.97	ODIDS (our)
0.1579	1.9	99.5	ESC-IDS[22]
n/r	3.5	99.7	RSS-DSS [26]
0.2024	n/r	96.7	Parzen-Window [23]
0.2331	0.6	97.1	Winner of KDD [24]
0.2356	0.6	97.5	Runner Up of KDD [25]
0.2371	0.4	96.9	PNrule [27]

5. Daniel Schobar, Ilinca Tudose, Vojtech Svatek and Martin Boeker verified ontology naming conventions and metadata completeness using protégé 4.For this reason, they Used OWL ontologies for cleaning up lexical heterogeneity.They checked and used six ontologies which are: Biotop, DCO, NTDO, GoodRelations, Vehicle sales Ontology and Neurist Ontology.

#Result:

Table 2 Exemplary OntoCheck tests with quantification of detected violations

Ontology	Entry Node	Entity	Panel	Check	Classes [abs, %]
BioTop	root	<rdfs:label>	Check	Upper case start	12 (4)
BioTop	root	<owl:Class rdf:about>	Check	CamelCase	34 (8)
DCO	root	<ru-meta:definition>	Check	Min card.=1	37 (8)
DCO	'Disease'	<SNOMED_ID>	Check	Min card.=1	2 (2)
DCO	root	<ru-meta:synonym>	Count	Min card.>2	238 (40)
DCO	root	<ru-meta:shortLabel>	Check	Max Char Count < 20	3 (.5)
DCO	root	n/a	Count	CountClsHavingAtLeast15Subclasses	15 (1)
DCO	root	n/a	Count	CountClsUsedAtLeast15times	48 (3.3)
NTDO	root	<rdfs:label>	Check	Doesn'tContain'Class'or'class'	3 (1)
Good Relations	root	<rdfs:label>	Check	Min card.=1	6 (15)
Vertical Sales Ontology	root	<rdfs:label>	Check	Length regex.{4,50}+	1 (1.5)
Vertical Sales Ontology	root	<rdf:ID>	Check	Doesn'tContain'Or'	7 (10)
Vertical Sales Ontology	root	n/a	Count	ClsUsedOnlyOnce	13 (20)
@neurist	root	n/a	Count	CountClsHavingExactlyOneSubclass	150 (5.3)

'Entry Node' refers to the selected class in the hierarchy for which all descendants are tested. The entity selected to be checked is described via its OWL syntax element. The last column indicates the amount of found classes violating (Check panel) or fulfilling (Count panel) a specified pattern. For the naming checks 'abs' refers to the absolute count of entities of the specified type failing the test. '%' refers to the ratio of abs to the amount of all entry node descendants.

6. Ian Hyland and Renate A.Schimidt discussed in their paper about Protégé-TS which is Protégé Term Selection tool. This tool is actually made for support and automade the process of OWL ontology signatures. They also used Natural Language Processing (NLP) tools.

#MODELS:

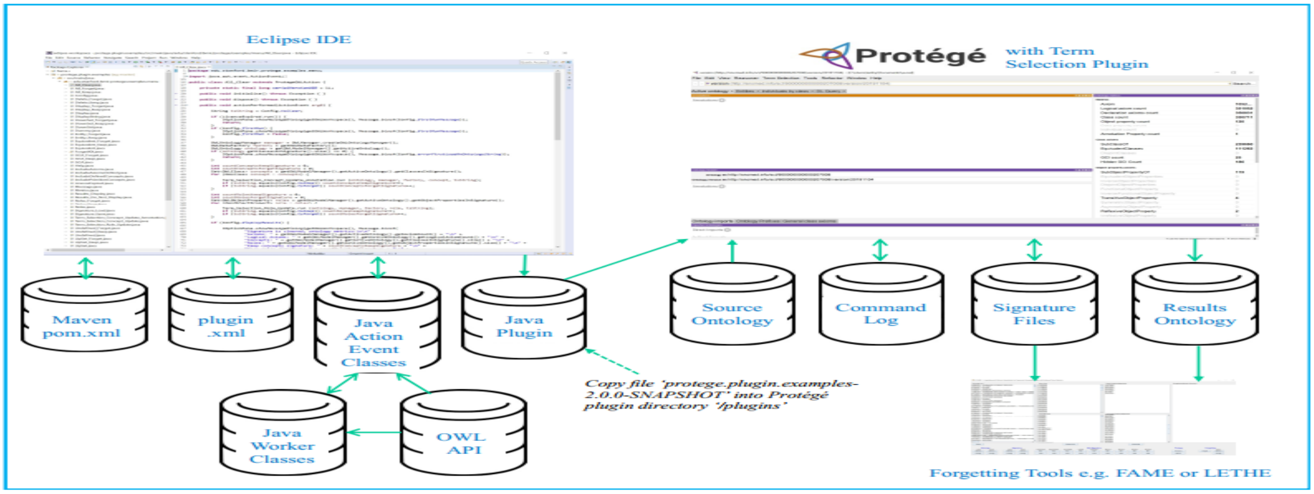


Fig. 5. Protégé-TS Software Architecture

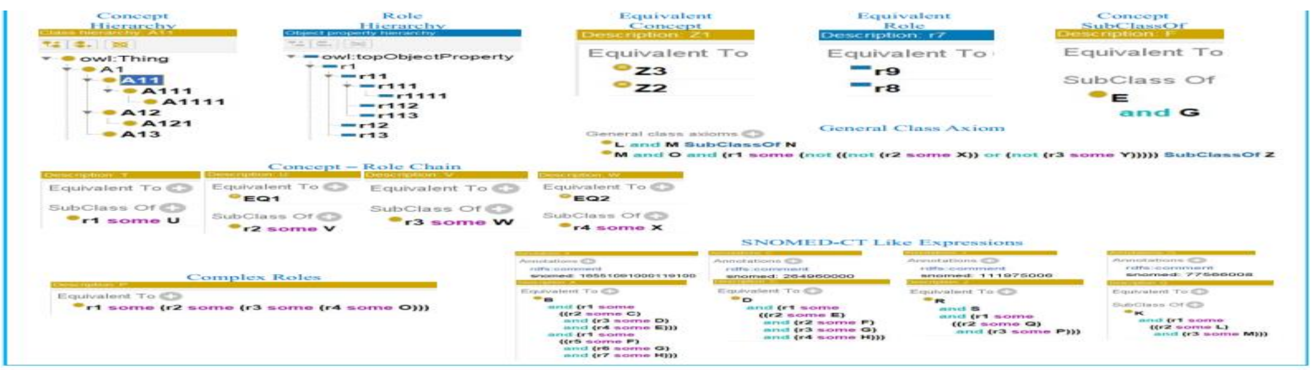


Fig. 6. Test Ontology OWL Expression Types.

#Result:

Action	Run-Time (secs)	Concepts Added	Roles Added	Test Comment
Restart Protégé	17			Pass
Load SNOMED-CT	52			Pass
All-Forget	5	350711	120	Pass
Search 'sunburn skin analysis'	1			Pass
Entity-Keep w/o Axioms	2	1	0	Pass
Entity-Keep w/ Axioms	9	5	4	Pass
DownSet w/ Axioms	21	5	4	Pass
UpSet w/ Axioms	49	39	7	Partial Pass
Equivalent-Keep w/ Axioms	17	5	4	Pass
GCA-Keep w/ Axioms	2	1	0	Pass
Role-Keep 1	7	5	4	Pass
Role-Keep 2	7	13	4	Pass
Role-Keep 3	7	24	4	Pass
Role-Keep 4	6	31	4	Pass
Role-Keep 5	6	33	4	Pass
Role-Keep 6	8	34	4	Pass
Role-Keep 7	6	34	4	Pass
Role-Keep 8	6	34	4	Pass

7. M.P.S Bhatia, Akshi Kumar, Rohit Beniwal and Tushar Malik Developed ontology based software automatic detection and update of software requirement specifications. They provide mechanism which automatically generates. However, They have used Software Design Development (SDD) and Software Requirements Specifications (SRS) ontologies. They have taken different approaches in their research phases.

Results:

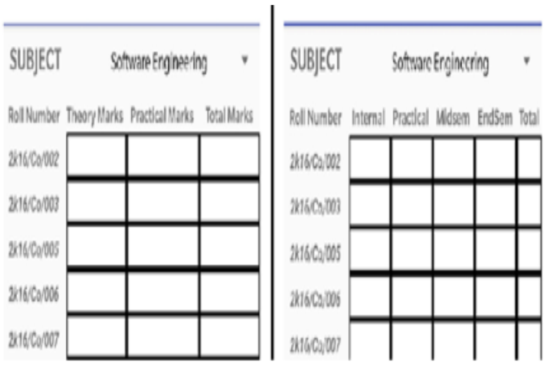


Figure 6

Snapshots of Old as Well as Updated Marks Entry Screen of the Earlier and Latest Version of RMS.

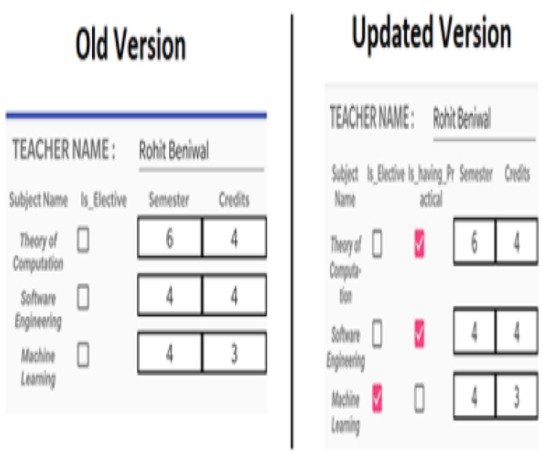


Figure 7

Snapshots of Old as Well as Updated Subject Entry Screen of the Old and Updated Version of RMS.

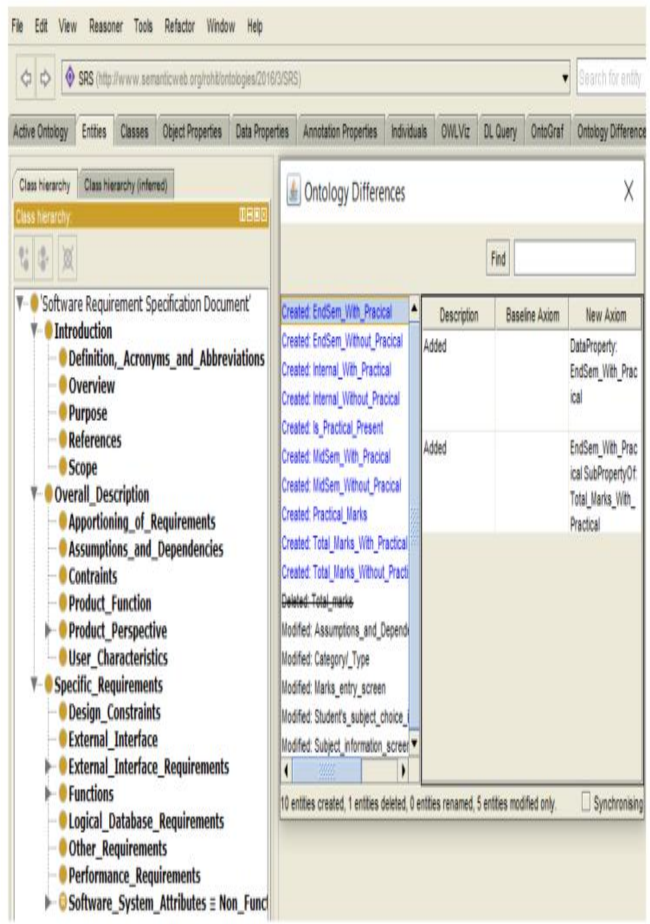


Figure 8

The Outcome of Comparison between the Updated SRS Ontology with its Old Version.

Table 1
Few added, deleted, and modified entities of the updated RMS.

Description	Baseline Axiom	New Axiom
Added		Marks_entry_screen Category/_Type xsd:string[pattern\" Applied Science and Mathematics(ASC), Allied Engineering Course(AEC),Department Core course(DCC),Foundation Elective(FEC),Department Elective Course(DEC),,Open Elective Course(OEC),Humanitarian Course(HMC), B.Tech project,Training Seminar,Mooc \"]"^^string
Added		Marks_entry_screen EndSem_With_Pracical "xsd:unsignedByte[>= 0, <= 40]"^^string
Deleted	Marks_entry_screen External_marks "xsd:unsignedByte[>= 0, <= 60]"	
Deleted	Student's_subject_choice_information_screen Category/_Type "xsd:string[pattern \" Elective 1, Elective 2, Core, Lab, Term Paper, Seminar, Dissertation \"]"^^string	
Added		Subject_information_screen Is_Practical_Present " xsd:string[pattern \"Yes,No\"]"^^string

8. Meng Huang, Tao li, Hui Zhao, Xiaojie liu, Zhan gao Proposed Immune Based network dynamic risk control strategy knowledge by using ontology construction. This paper illustrates domain knowledge concepts, attributes and instances. Concepts are expressed in graph.

#Model:

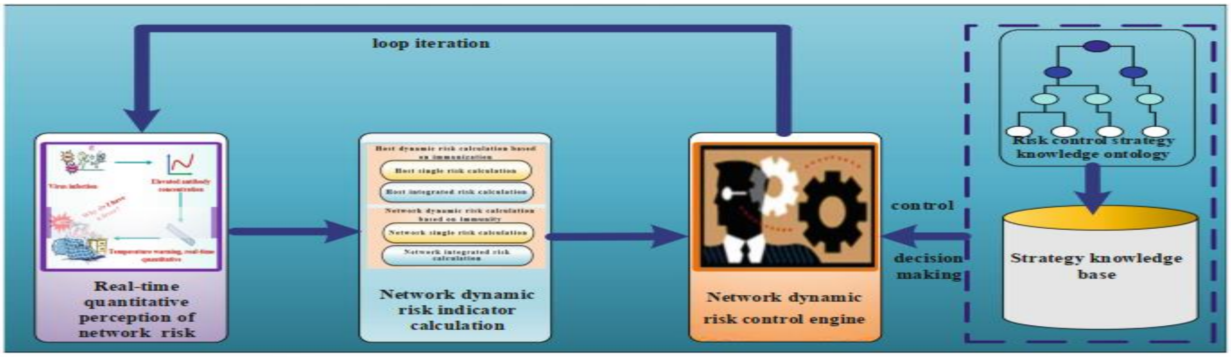


Fig. 1. Immune-based network dynamic risk control model

#Formula:

When a network attack is detected, the antibody concentration increases, and the formula is as follows:

$$Ac(t) = \alpha + \beta Ac(t-1) \tag{1}$$

When no network attack is detected, the antibody concentration is reduced and the formula is as follows:

$$Ac(t) = \begin{cases} Ac(t-1) - \frac{Ac(t-1)}{\lambda - \theta(t-1)}, & \theta(t-1) < \lambda \\ 0, & \theta(t-1) \geq \lambda \end{cases} \tag{2}$$

The risk calculation is expressed as Eq. 3 to Eq. 6:
The risk value of the ith attack that host m receives at time t:

$$r_{m,i}(t) = \frac{2}{1 + e^{-\omega_i \bullet Ac_i}} - 1 \tag{3}$$

The risk value of the type I attack that host m receives at time t:

$$r_m(t) = \frac{2}{1 + e^{\left(-\sum_{i=1}^{I} \omega_i \bullet Ac_i\right)}} - 1 \tag{4}$$

The risk value of the i th attack that network n receives at time t:

$$R_{n,i}(t) = \frac{2}{1 + e^{\left(-\omega_i \bullet \sum_{m=1}^{M} \mu_m \bullet Ac_i\right)}} - 1 \tag{5}$$

Network overall risk value:

$$R(t) = \frac{2}{1 + e^{\left(-\sum_{i=1}^{I}\left(-\omega_i \bullet \left(\sum_{m=1}^{M} \mu_m \bullet Ac_i\right)\right)\right)}} - 1 \tag{6}$$

#Result:

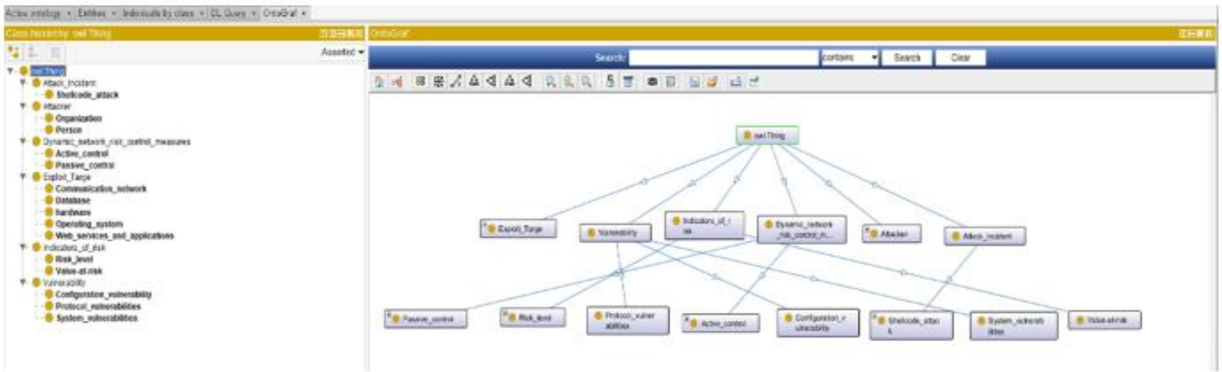
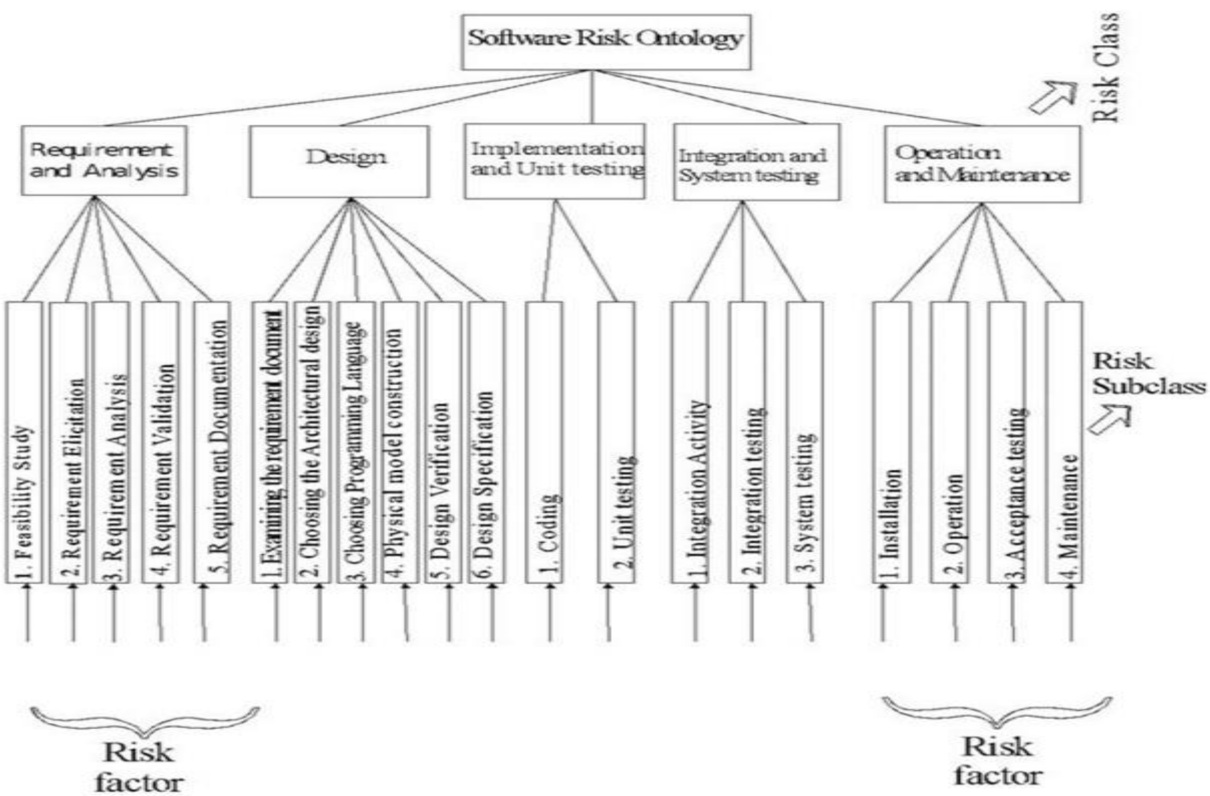


Fig. 8. Part of ontology visualization diagram

9. Temitope Elizabeth Abioye, Oluwasefunmi Tale Arogundade, Sanjay Misra, Adio T. Akinwale and Olusola John Adeniran have done an empirical study on ontology. They worked on ontological risk management framework. They analyze the risks and prioritize by using precision recall and F-measure matrix.Mean score from the precision score and performance compared with the evaluation scale.

#Model:



#Result:

TABLE 8 Evaluation concept scale[55]

Overall score	Interpretation
0–0.99	Poor
1–1.99	Marginal
2–2.99	Satisfactory
3–3.99	Good
4–5	Outstanding

TABLE 9 Summary of software expert evaluation using selected indicators with mean value

	Cost	Duration	Complexity	Quality	Scope	Functional requirement
	2	3	4	3	3	3
	4	3	3	4	3	3
	4	3	2	2	4	3
	4	3	4	3	3	3
	3	3	2	3	3	4
	3	3	3	5	3	4
	3	4	2	4	4	4
	4	3	3	3	3	3
	2	4	3	3	4	3
	3	3	3	3	2	4
Mean	3.20	3.20	2.90	3.30	3.20	3.30

TABLE 10 Perception evaluation results

	UA	RD	BA	UF	AC	PA	SA	US	CA
	4	3	3	4	4	4	3	3	3
	3	3	3	2	3	4	4	4	3
	4	3	3	4	3	2	3	3	3
	3	3	3	4	4	5	5	4	4
	4	3	3	4	4	5	4	4	3
	3	2	3	4	4	5	4	4	4
	2	3	3	3	3	4	4	3	3
	3	3	4	3	3	4	4	3	4
	2	3	3	4	4	3	4	3	3
	5	3	3	3	4	4	4	3	5
Mean	3.3	2.9	3.1	3.5	3.6	4.0	3.9	3.4	3.5

#Reference:

1. Musen, M. A. (1998). Domain ontologies in software engineering: use of Protege with the EON architecture. *Methods of Information in Medicine-Methodik der Information in der Medizin*, *37*(4), 540-550.
2. Storey, M. A., Musen, M., Silva, J., Best, C., Ernst, N., Fergerson, R., & Noy, N. (2001, October). Jambalaya: Interactive visualization to enhance ontology authoring and knowledge acquisition in Protégé. In *Workshop on interactive tools for knowledge capture* (Vol. 73).
3. Wang, J., Ding, Z., & Jiang, C. (2005, November). An ontology-based public transport query system. In *2005 First International Conference on Semantics, Knowledge and Grid* (pp. 62-62). IEEE.
4. Abdoli, F., & Kahani, M. (2009, October). Ontology-based distributed intrusion detection system. In *2009 14th International CSI Computer Conference* (pp. 65-70). IEEE.
5. Schober, D., Tudose, I., Svatek, V., & Boeker, M. (2012, September). OntoCheck: verifying ontology naming conventions and metadata completeness in Protégé 4. In *Journal of biomedical semantics* (Vol. 3, No. S2, p. S4). BioMed Central.
6. Hyland, I., & Schmidt, R. A. Protégé-TS: An OWL Ontology Term Selection Tool.
7. Bhatia, M. P. S., Kumar, A., Beniwal, R., & Malik, T. (2020). Ontology driven software development for automatic detection and updation of software requirement specifications. *Journal of Discrete Mathematical Sciences and Cryptography*, *23*(1), 197-208.
8. Huang, M., Li, T., Zhao, H., Liu, X., & Gao, Z. (2020, July). Immune-Based Network Dynamic Risk Control Strategy Knowledge Ontology Construction. In *Science and Information Conference* (pp. 420-430). Springer, Cham.
9. Abioye, T. E., Arogundade, O. T., Misra, S., Akinwale, A. T., & Adeniran, O. J. (2020). Toward ontology-based risk management framework for software projects: An empirical study. *Journal of Software: Evolution and Process*, e2269.