مجلد بحوث المؤتمر العلمي الدولي الاول لكلية التربية غريان ـ جامعة غريان
المنعقد في الفترة
من 8- 9/ 11/ 2025
(المجلد الاول)

Reasoning about Domain Ontology By Protégé-Owl as Software
An Example (Tommor/Date Domain)

Aml Ebrahim Hejaze [1]*, Naima Fouzi Albahloul [2]
[1] Master in Computer Sciences National Institute of Management\\Tripoly-libya
[2]Master in Management Information Systems Faculty of Education / Gharyan
Aml.2401169@gmail.com
Mgdam.naje@gmail.com

التفكير المنطقي حول مجال الأنطولوجيا باستخدام برنامجprotege–OWL (مجال التمور كمثال)

أمل إبراهيم حجازي[1] ، نعيمة فوزي البهلول[2]

تاريخ الاستلام: 2025-06-25، تاريخ القبول: 2025-09-15، تاريخ النشر: 2025-11-08.

ملخص

الانتولوجي لوصف المكونات الاساسية لمصادر المعلومات المختلفة، التي أصبحت القضية الاساسية لاسترجاع والوصول الذكي الي المعلومات، الانتولوجي كمواصفات واضحة تكون واسعة الاستخدام في العديد من التطبيقات، لدعم علم دلالات الالفاظ "سيمانتك" ولإعطاء نماذج غنية المعني من الواقع. مجال الانتولوجي ك تطبيق ناشئ جديد يمتلك امكانيات كبيرة في اعطاء مواصفات واضحة لنموذج المعرفة.

نظام تمثيل المعرفة الذي يعتمد على منطق الوصف يكون قادر علي انجاز أنواع مختلفة من التفكر ويركز عادتاً على طرق لإعطاء مستوي عالي من توصيف العالم الذي يمكن ان يستعمل بفاعليه لبناء تطبيقات ذكية.

تفكر الوصف المنطقي ينجز خدمات الاستدلال، كحساب استدلال الفصائل الام من الفصيل، تحديد مكونات الفصيل من عدمه. معظم التقنيات المتاحة من التفكر الوصف المنطقي يكون PelletRACER ,FaCT++ ,FaCT , .

في هذه الورقة استعملنا مصطلح (منطق الوصف) كلغة شكلية لتمثيل المعرفة والتفكر حول مجال اهتمامنا "مجال التمور".

Abstract

Ontologies to describe the basic contents of different information sources has become the key issue to intelligent retrieval and access of information.

Ontologies, as an explicit specification, are widely used in many applications to support the semantics and to provide a richer model of reality. Domain Ontology as an emerging new discipline has a great potential in providing such explicit specification to knowledge model.

A knowledge representation system based on Description Logic is able to perform specific kinds of reasoning and usually focused on methods for providing high-level descriptions of the world that can be effectively used to build intelligent applications.

A Description Logic reasoner performs various inferencing services, such as computing the inferred super classes of a class, determining whether or not a class is consistent. Some techniques of the popular Description Logic reasoners available are as RACER , FaCT, FaCT++, and Pellet.

In this paper, we mainly use the term "description logic" (DL) as a formal language for the representation knowledge and reasoning about the domain of interest ' Tommor Domain '.

Key Words : *Ontologies, knowledge model, description logic reasoners, inferencing services* , intelligent applications , Tommor Domain.

Introduction

Huge volumes of the data existing in various databases lack meanings and implicit, especially in web applications where searching techniques are key-word based. As a consequence query tools are unable to identify the property of information intelligently .

Ontologies are needed in supporting to organize large information repositories and to access such repositories in an efficient manner. Ontologies can play a major role in the key aspects namely searching and accessing web-based information, and interpretation and reasoning about that information.

Description logics have been used in the implementation of many systems that demonstrate their practical effectiveness as software engineering, configuration, medicine, digital libraries and web-based information systems and other application domains.

A DL knowledge base is analogously typically comprised by two components : a "TBox" and an "ABox" $\Sigma \langle TBox , Abox \rangle$. The TBox contains intentional knowledge in the form of a terminology (hence the term "TBox," but "taxonomy" could be used as well) and is built through declarations that describe general properties of concepts. Because of the nature of the subsumption relationships among the concepts that constitute the terminology, the ABox (or assertion box) contains extensional knowledge that is specific to the individuals. A Description Logic reasoner performs various inferencing services, such as computing the inferred superclasses of a class, determining whether or not a class is consistent. Some of the popular Descrioption Logic reasoners available are as: RACER , FaCT, FaCT++, Pellet.

In a previous paper we built the Ontology of dates from a relational database and our work this is an extension of that.

In this paper, we mainly use the term "description logic" (DL) as a formal language for the representation knowledge and reasoning about the existing domain of interest ' Tommor Ontology '.

Problem Statement

Web-based information systems must provide answers to web-based query from existing information source namely Data Base. End-users, who have an access to the internet, need and should not be aware of all technical details of such information source. An explicit specification of such information sources can be developed to shield end-users from all KB technical, and to remove many ambiguities. Furthermore, capturing knowledge base will provide common understanding of specific domain.

In this paper, we the purposed of Legal case ontology are to retrieve and extract information along with case based reasoning.

A Legal case ontology encompasses different entities, which are brought-up during data collection from the our specific domain.

Their representation is helpful to extract reasoning about the cases, through organizing data in a manner, so that an ontological interface can look into such repository and fetch desired results within short time. No doubt, the data being specific after applying constraints is beneficial for fetching

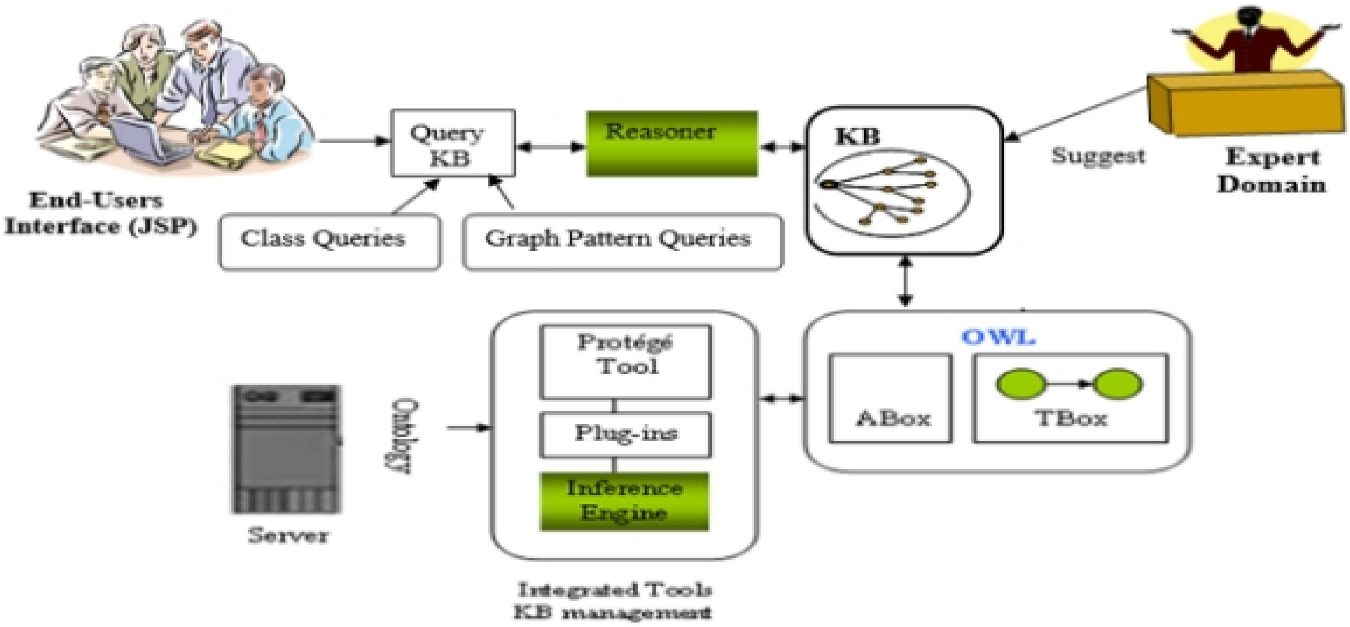
reasoning without mapping it in a local manner.

The Absence of a specialized ontology for the Libyan agriculture sector and its impact on the representation and retrieval of knowledge was the main problem in this paper.

Research Methodology

Research Methodology describes how reasoning about existing domain ontology is a main resource to knowledge component (has been development to KB in the method manually).

Figure (1) System architecture of Reasoning about ontology



Reasoning Ontology by using suited Integrated Tools, which hold data in a form that allowing for inferences with respect to defined concepts as shown in above figure (1).

Consists of Two phases: (i) formalization (Shows the formal visual technologies for Jambalaya and Taxonomy OWL Viz plugins), and (ii) evaluation some previous phases.

Domain expert can have an access to knowledge base, while user can execute queries requires based on knowledgebase focus of class queries and graph patterns query. Class queries are equivalent to define class descriptions and therefore can be used to retrieve super classes, subclasses, and equivalent classes of the class being defined. The formulation of a query is constrained by the entities, relations and individuals defined it by the ontology. These queries were formulated using the Manchester OWL syntax. Class queries are useful when the goal is to retrieve a set of individuals that satisfy a certain restrictions. These restrictions, logically describe the membership requirements of an individual belongs to a class.

The graph pattern based queries are based on graph patterns composed by nodes and edges. Each node can represent an unbound variable that will be bound to a member of a certain class or a variable already bound to a specific (named) individual. Edges represent relations through properties or restrictions in the ontology.

KB modules based OWL language is responsible for creation and initialization of TBox, ABox to store ontology, which managed by integrated tools working under web server as in (Protégé platform + plug-ins + Inference Engine).

Motivation of reasoning

One of the main motivation implemented by a reasoner is to test whether or not one class is a subclass of another class", by performing such tests on all of the classes in ontology. It is possible for a reasoner to compute the inferred ontology class hierarchy. Another motivation implemented by a reasoner is consistency checking 'Satisfiability ', based on the description (Complexity conditions) of a class the reasoner can check whether or not it is possible for the class to have any instances. A class is deemed to be inconsistent if it cannot possibly have any instances .

There are a number of existing systems that have been devised to classify food products and their nutritional properties, and several databases developed with the same purpose. However, very few ontological resources exist that describe food commonly does no exist ontology to describe agriculture products, that have supreme food value such as dates or Tommor. The most well-known food ontology is the wine and pizza and food Ontology, this built ontology was quasi the most suitable pizza and wine, but they do not provide any information about nutritional realities.

Here the applied value of the system appears by adding value to research in the representation of knowledge and the agriculture sector.

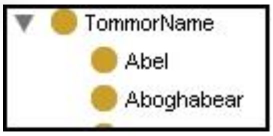
Techniques and algorithms of reasoning

A technique that is executed in this paper named by [Pellet Reasoner] which depends on tableaux algorithms.

Reasoner is an open-source Java based OWL DL Reasoner. It can be used in conjunction with both Jena and OWL API libraries and also provides a DIG" interface. Pellet has a number of features and these features helped in analysis information about Ontology as: Ontology analysis and repair, Entailment, Query answering, Ontology Debugging .

The core of the Pellet Reasoner is the tableaux Reasoner that checks the consistency of a KB (*ABox and a TBox*). These Tableaux algorithms use negation to reduce subsumption to (un)satisfiability of concept descriptions as $C \subseteq D$ iff $C \cap \sim D$ is unsatisfiable, it is similar to

Abel $\subseteq$ TommorName, iff Abel $\cap\sim$ TommorName is unsatisfiable.



To underlying the action of this algorithm we will give an example, in general. Let A, B be concept names, and let R be a role name.

To test known whether $(\exists R.A)\cap(\exists R.B)$ is subsumed by $\exists R.(A\cap B)$ as

$\exists$hasStage.(Abel$\cap$ Aboghabear), This means that we must check whether the concept description C=$(\exists R.A)\cap(\exists R.B)\cap\sim\exists R.(A\cap B)$is unsatisfiable.

Applying Reasoner of Tommor Ontology

The objective of reasoner execution in this paper to give the following results: Synchronize reasoner, check concept consistency, compute inferred hierarchy, compute equivalent classes, and total time by seconds.

Pellet is typically started by clicking on main menu (Reasoning), and clicking the pellet 1.5.2. Which opens a terminal/console window and starts the Reasoner running with HTTP" communication enabled as next figure .

Figure (2) pellet Startup Screen

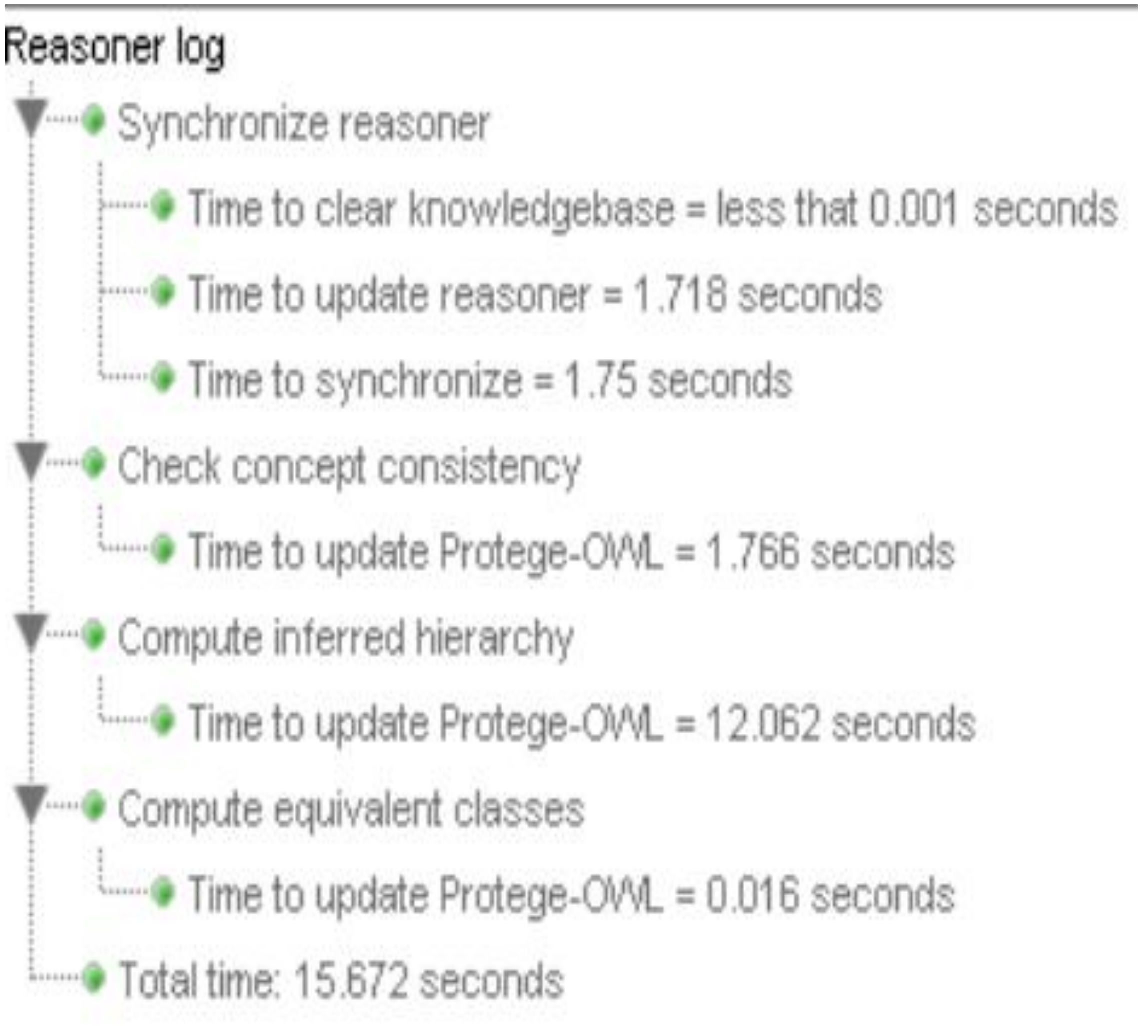
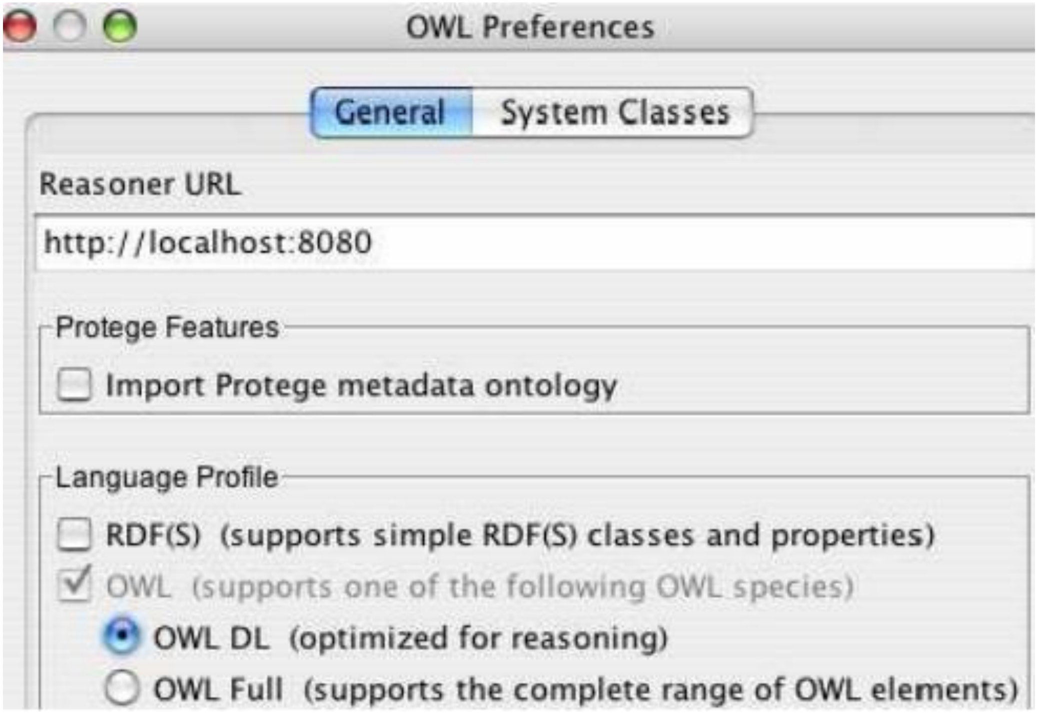


Figure (3) views the OWL Preferences settings. The bottom line indicates that HTTP communication is running, and specifies the I.P. address and port number.

Figure (3): The OWL Preferences Dialog



Reasoning by Inferred Hierarchy representation:

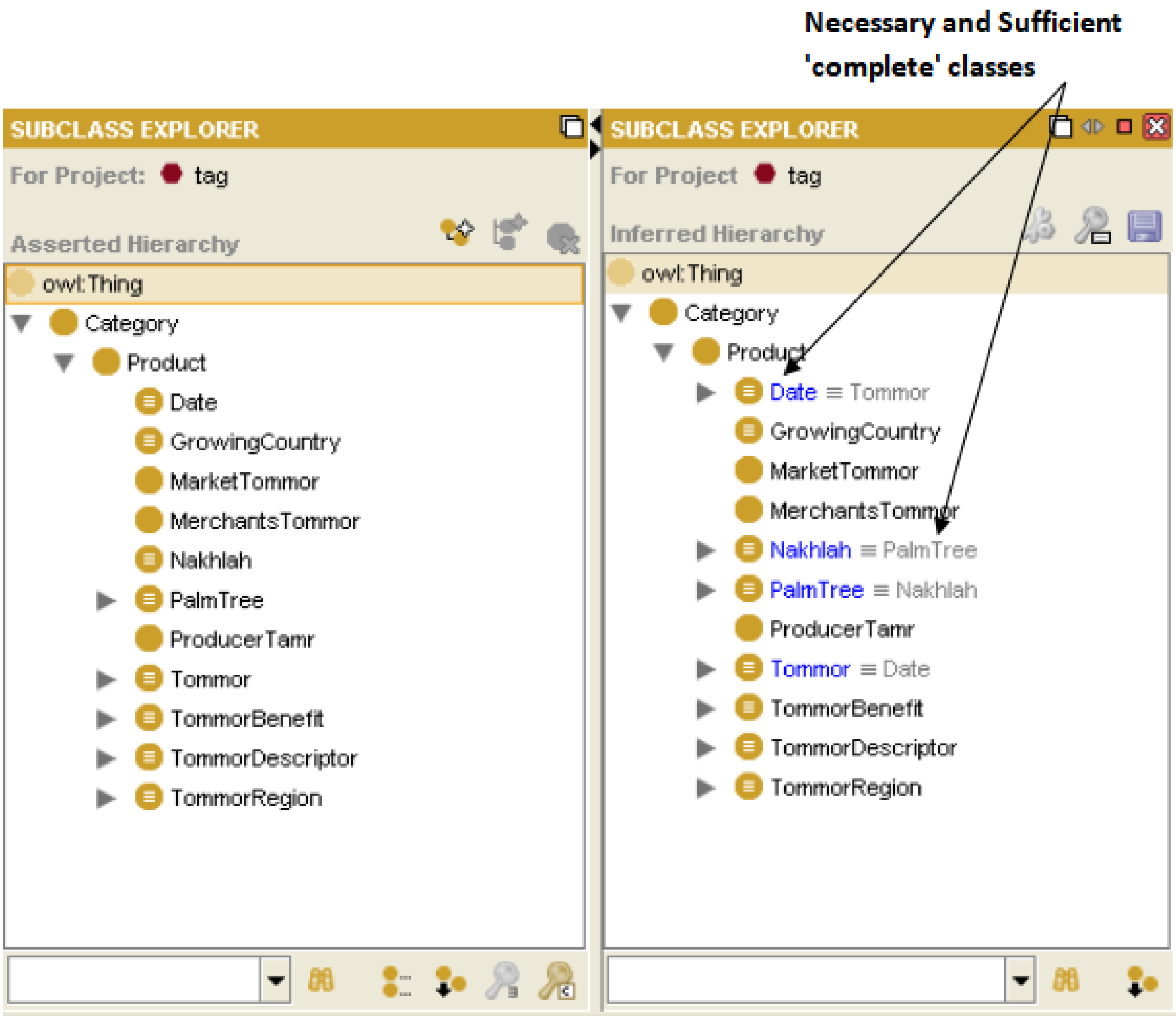
The class hierarchy recalled "asserted hierarchy" is automatically computed by the reasoner is called the inferred hierarchy, to automatically classify and check for inconsistencies ‘Classify Taxonomy’.

They are invoked via the Protégé-OWL menu as shown in Figure (4).

If a class has been reclassified (i.e. if it’s super classes have changed), then the class name will appear in a blue color in the inferred hierarchy.

If a class has been found to be inconsistent then, it's icon will be circled in red (The task of computing the inferred class hierarchy is also know as classifying the ontology).

Figure (4) : The Inferred Hierarchy and the Asserted Hierarchy



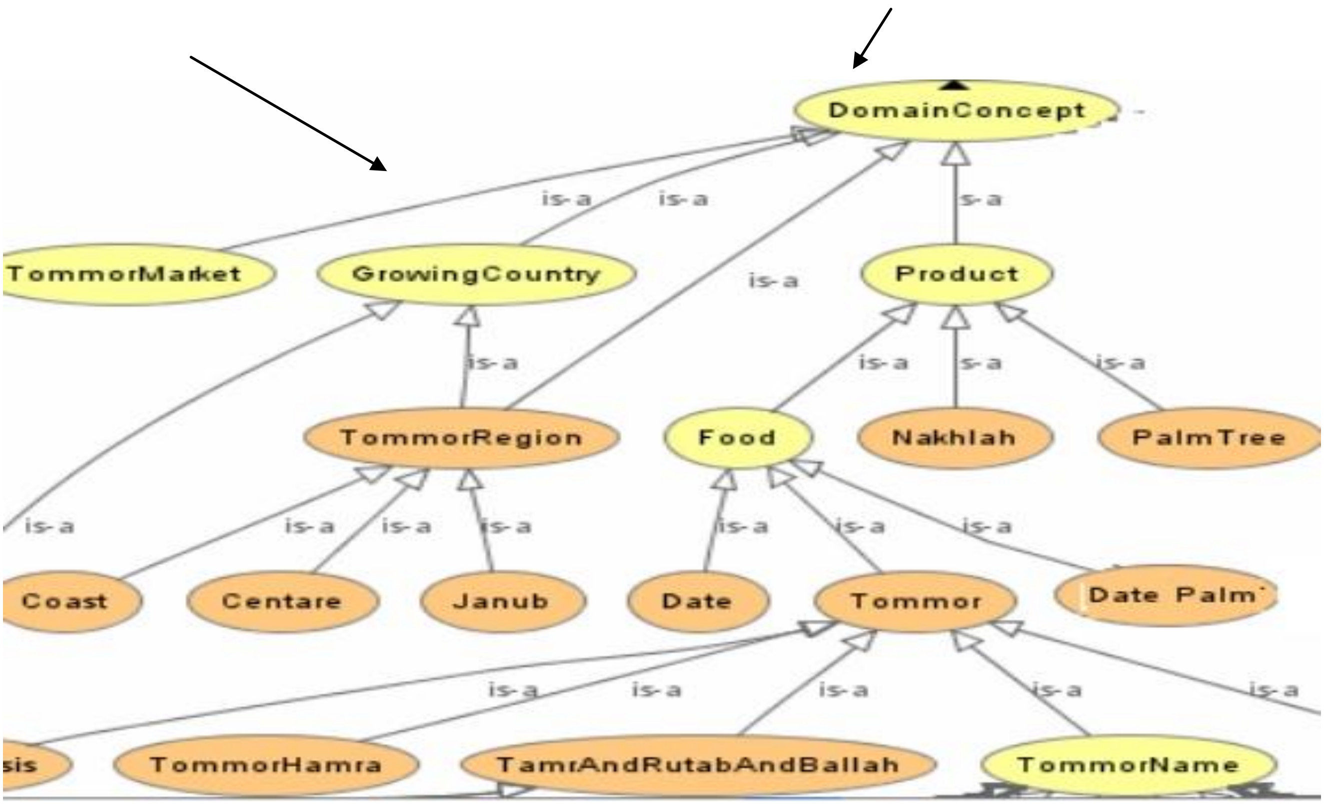
Reasoning by graph representation:

SHOIQ tableaux is a completion graph representing such a model. Each node x in the graph represents an individual, labeled with the set of concepts L(x).

The completion forest can be described as a directed graph $G = (\mathcal{V}, L, \mathcal{E}, P, \neq)$ where each individual $x \in \mathcal{V}$, can be labeled with a set of concept labels L(x) and each edge (x, y) can be labelled with a set of role names $\mathcal{E}$ (x, y). P is a program consisting of DL. Also, keep track of inequalities between nodes of the graph with a symmetric binary relation $\neq$ between the nodes of G.

Figure (5) SHOIQ tableaux as reasoning graph 'Tommor Ontology'

Edge =binary relation=role   Root Node x



In figure (5), defined classes have a class icon with an orange background.

Primitive classes have a class icon that has a yellow background. It is also important to understand that the Reasoner can only automatically classify classes under defined classes - i.e. classes with at least one set of necessary and sufficient conditions.

Tommor Ontology Evaluation

After obtaining results from execution of reasoner run by pellet mechanism, now we'll evaluate the results and analyze them by comparison of Knowledge base . Evaluation of these results will be processed under Protégé with integrated tools.

Ontology evaluation used to minimize human errors that are inherent in maintaining a multiple inheritance hierarchy. The content of ontology should be evaluated before (re)using it on other ontology or applications.

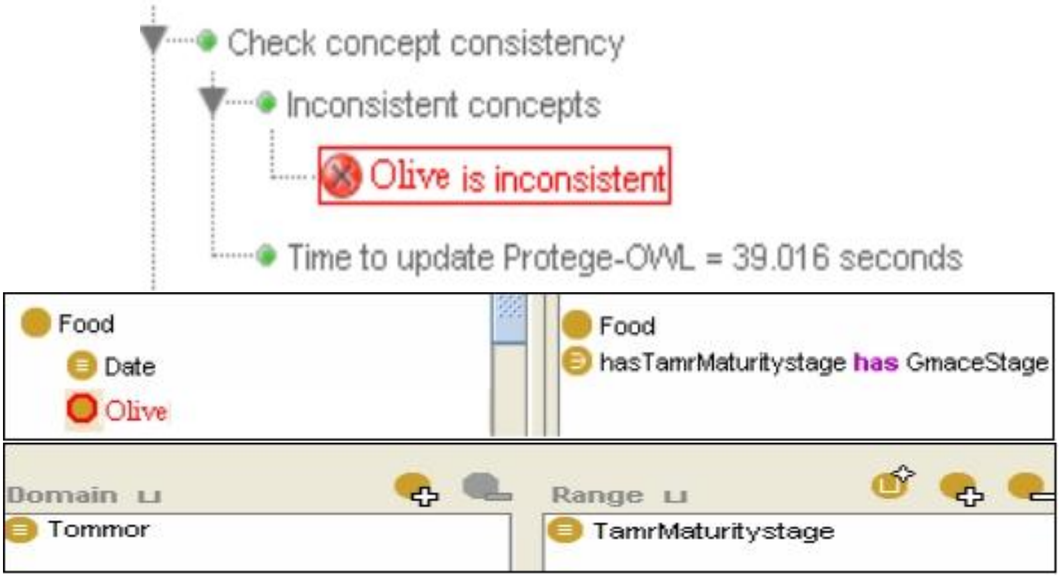
Content Ontology Evaluation

To evaluate the content of a given ontology, the following criteria were identified: consistency, competences and conciseness.

Consistency or ontology verification refers to whether it is possible to obtain contradictory conclusions from valid input definitions. A given definition is consistent if and only if the individual definition is consistent and no contradictory knowledge can be inferred from other definitions and axioms.

We will take a concept that has no relation by Tommor domain, as for example Olive class. What happen to consistency concept?

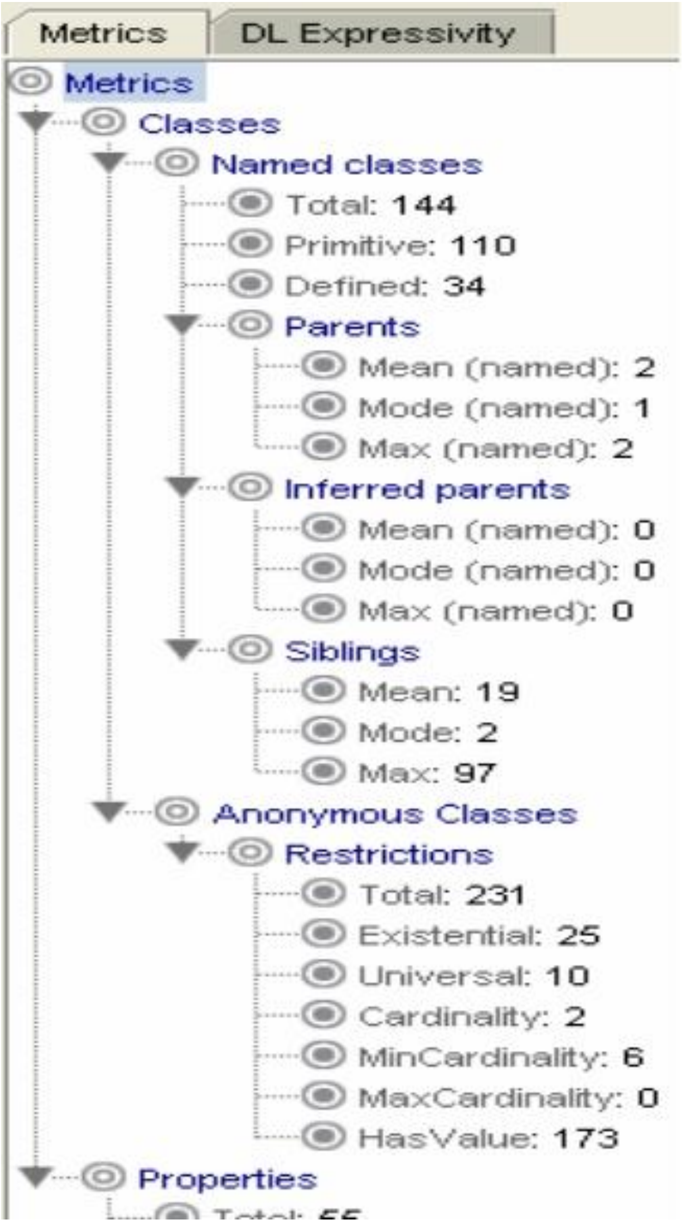
Figure (6) pellet Startup Screen view inconsistent concept



Pellet reasoner will discover a mistake in ontology consistency. Olive class has not the same properties of Tommor dates. We note from figure (6) that Olive class denotes on a mistake in domain consistency. The property ' hasTamrMaturitystage' has a domain of Tommor. This means that the reasoner can infer that all individuals using the hasTamrMaturitystage property must be of Tommor types. Because of the restriction on this class, all members of Olives must use the hasTamrMaturitystage property, and therefore members must belong to Tommor concept. However,Tommor and Olives are disjoint "an inconsistency". If they were not disjoint, Olives would be inferred to be a subclass of Tommor.

We can summarize Tommor ontology consistency of (number of classes and properties via ontology statistics arrays as demonstrated in next figure.

Figure (7) summary Matrix: Class and properties Page

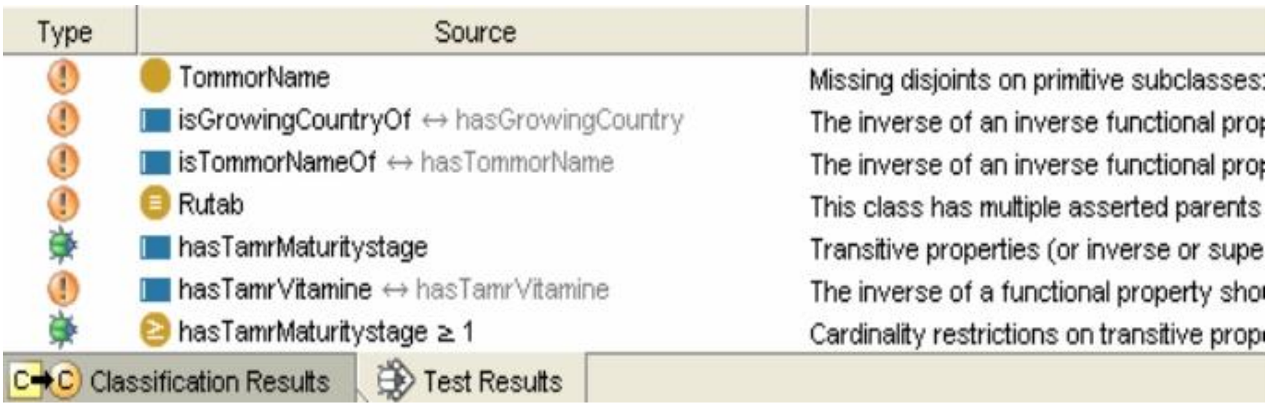


Completeness: incompleteness is a fundamental problem in ontologies. Furthermore, even more when ontologies are available in such an open environment such as Semantic Web. In fact, we cannot prove the Completeness of an ontology, but we can prove the incompleteness of an individual definition. And therefore we can deduce the incompleteness of an ontology if at least one definition is missing in the established reference framework. So, an ontology is complete if and only if: All that is supposed to be in the ontology is explicitly stated in it, or for all the knowledge that is required but not explicit. It should be checked whether it can be inferred from other definition. And axioms. If it can, the definition is complete; other wise, it is incomplete.

The figure (8) shows a test reference framework, which contains various tests that may be running on the ontology being edited.

If we try to delete class definition from Tommor name superclass, then we do a click by mouse on OWL list that found on main menu then we click on Run ontology test command, now the ontology will give framework a result shows a loss of class definition, as appear on next figure.

Figure (8) Ontology Test Results



Type	Source	
	TommorName	Missing disjoints on primitive subclasses:
	isGrowingCountryOf ↔ hasGrowingCountry	The inverse of an inverse functional prop
	isTommorNameOf ↔ hasTommorName	The inverse of an inverse functional prop
	Rutab	This class has multiple asserted parents
	hasTamrMaturitystage	Transitive properties (or inverse or supe
	hasTamrVitamine ↔ hasTamrVitamine	The inverse of a functional property sho
	hasTamrMaturitystage ≥ 1	Cardinality restrictions on transitive prop

Figure (9) Asserted model with out reasoning

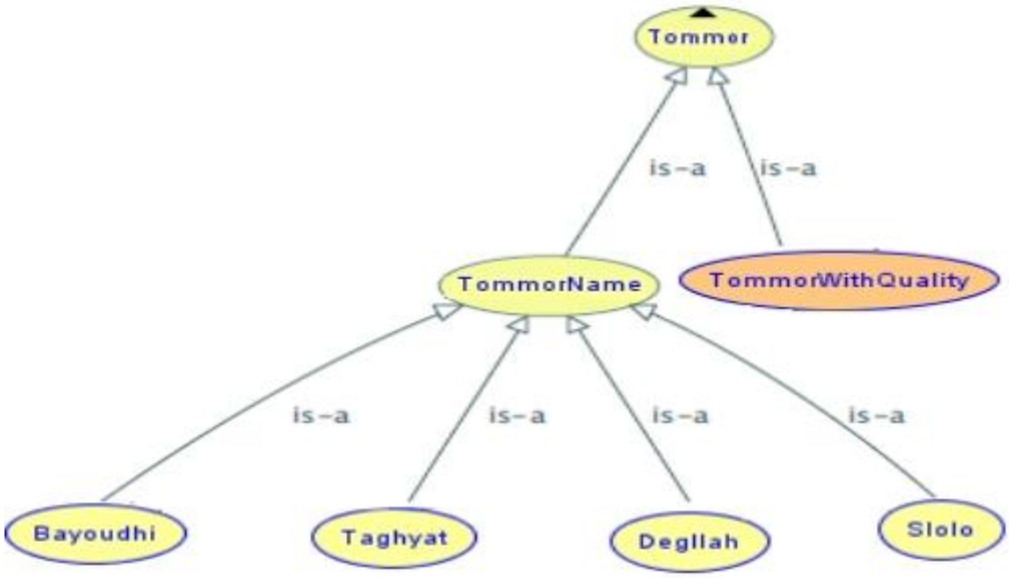
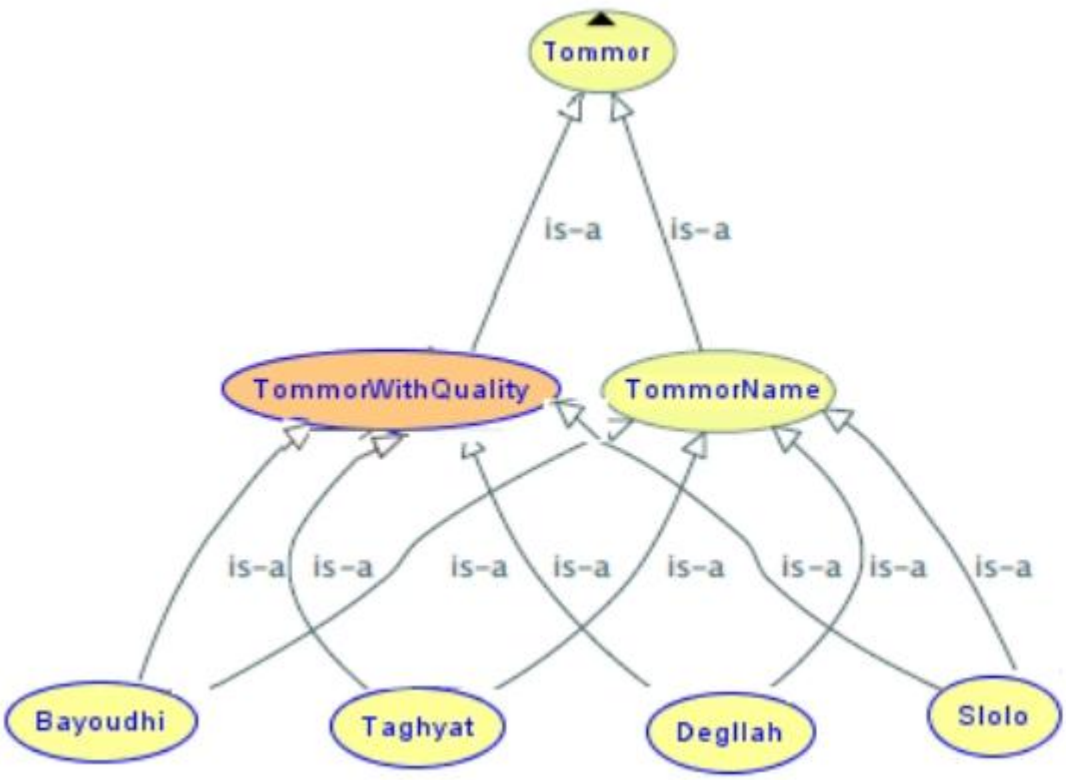


Figure (10) Inferred model with reasoning



Tommor domain after execution of reasoner can discover an inferred on other class definition found on tommor domain, reasoner work according to SHOIQ.

In last example, we have two models, the last one before executions of reasoner, the 2nd one after execution of it . We note in 2nd case how defined class which named tommor with quality how discover primitive class from Tommor name class. Defined classes or (inferred model) has a class icon with an orange background. Primitive classes have a class icon that has a yellow background.

It is also important to understand that the Reasoner can be applied the automatically to classify classes under defined classes.

Conciseness (ontology validation): An ontology is concise if: it does not store any unnecessary or useless definitions, and redundancies cannot be inferred from other definitions and axioms. The goal is to prove the real world model.

After revision of expert to Tommor domain, we say that on a domain there is not unwanted define, but all defines in this domain are [just needs].

Conclusion and Future Work

The Semantic Web technology now is one of the most active research topics during the recent years. It becomes widely used in the internet technology.

This new technology aims at the sharing and exchange of information where the semantic issue appears to be essential. Ontologies have been widely accepted as the primary method of representing knowledge in the Semantic Web. The most significant area is ontology building. An ontology as a form of knowledge base is exploited to provide common shared understanding of systems. Ontologies provide an explicit specification of specific domain, are widely used in many applications to support the semantics and to provide a richer model of reality.

In this papers, we aimed at investigating the issue of reasoning of ontology development from an existing information sources.

Therefore, the concepts of knowledge bases and semantics become the savior of many database shortcomings.

In a network, distributed environment, a Protégé platform (ontology editor), was used to extract concepts (element of knowledge) in a Libyan date (Tommor) domain is explored as an example to building knowledge model.

Our work can be further extended to cover a large database with several applications. The process of extraction can also be fully automated.

To verify and validate the extracted ontology, we performed the following steps. First, we took in consideration three criteria: consistency, completencee and conciseness by using a reasoner engine (pallet); we verified that the ontology satisfied the three criteria mentioned a above.

Second: A comparative analysis between the extracted knowledgebase and its relevant database was performed.

A set of criteria factors were applied namely model expressiveness, Assumption' Closed- vs. Open-World Semantics ', Simple Model vs. Rich Model, and Query Optimization. For future work, we plane to upload the derived knowledge in Tommor web site. Such knowledge can assist end-users to have an answer to several knowledge-based queries. We, also feel that task ontology instead of domain ontology can enrich the knowledgebase concepts .This becomes very important when we consider the E-Commerce applications of Tommor domain.

Protégé tool is a power open-ended tool. In fact, a set of plug-ins can be development and added to Protégé such as thesaurus and WordNet.

Related Work

There are many recent researches that contribute in development of ontology tools like (XML, RDFS, OWL, OIL, DAML+ OIL), to representing the taxonomy and classification hierarchies, and there are different approach for each one to capture knowledge and building ontology; however , all of these approach the same testing of ontology by reasoning based DL.

Some of these systems concentrate on programs and tools used in building ontology but neglect reasoner, and some concentrate on UI "User Interface", but neglect information field. However, we concentrate on using the most developed tools (Protégé-OWL) with presenting UI, at the same time, that to create user inquires, and testing ontology by reasoning to make our system more clear. And to forget [Holger, et al., 2004] who contribute to development of many ontology tools and by his researches that help us to execute building and reasoning about ontology. Reasoning with domain ontologies are the core topic in knowledge representation, semantic web technologies, AI, etc. A domain ontology defines concepts and relations in a domain and for representation of knowledge and for automated reasoning.

Ontology-based reasoning has been widely applied in domain–specific contexts. In the biomedical domain, projects like SNOMED CT and the Gene Ontology leverage reasoning to detect inconsistencies and infer implicit relationships. In the Hôtel Domain, A research approche combines both quantitative and qualitative méthodes as used to develop a Hôtel room personalization framework. That supports well–defined machine–readable descriptions of hotel rooms and guest profiles [Ronald et al., 2022]. The legal and Educational sectors, ontologies support regulatory compliance and adaptive learning, respectively.

Modular ontology reasoning addresses scalability by dividing large ontologies into manageable components. Techniques such as ontology segmentation, import-by-query, and distributed reasoning have been proposed to support reasoning in complex domains [Grau et al., 2008].

More recently, **hybrid approaches** combining symbolic reasoning with machine learning have emerged. Embedding-based reasoning techniques (e.g., RDF2Vec, OWL2Vec) approximate logical inference using vector space models, improving scalability but at the cost of exactness [Chen et al., 2021].

References (APA Style)

- Alberto Olivares-Alarcosa,, Sergi Foixa , J´ulia Borr`asa , Gerard Canalb , Guillem Aleny`aa.(2025). Ontological Foundations for Contrastive Explanatory Narration of Robot Plans. reprint submitted to Information Sciences September 29, 2025. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://arxiv.org/pdf/2509.22493.
- S.Izquierdo-Badiola,G. Canal, Rizzo, G.Alenya, PlanCollabNL(2024) :Leveraging Large Language Models for Adaptive Plan Generation in Human-Robot Collaboration, in:IEEE EEE International Conference on Robotics and Automation (ICRA), 2024, pp. 17344–17350.
- Ronald Ojino, Luisa Mich (2022). Hotel room personalization via ontology and rule-based reasoning. nternational Journal of Web Information Systems Vol. 18 No. 5/6, 2022.pp. 369-387.© Emerald Publishing Limited .1744-0084.DOI 10.1108/IJWIS-02-2022-0045.
- Aml Ebrahim Hejaze (2022). Building Ontology Based on legacy database By Protégé as Software An Example (Tommor/Date Domain). General Authorlty Office for culture-Gharyan, Aljable Journal for Applied Science and Humanities.WWW.MAJLTALOULUM.ORG.ISSN.2709-5800 No (9) (62-83) .
- Ronald Ojino. F. ,Lusia Mich, D.(2022). Hotel room personalization via ontology and rule-based reasoning, *by Cooperative University of Kenya user on 25 August 2025.*http://www.emerald.com/ijwis/article-pdf/18/5-6/369/2814124/ijwis-02-2022-0045.

- Chen, J., Hu, Z., Ji, S., & Qu, Y. (2021). OWL2Vec*: Embedding OWL Ontologies with Semantic Enrichment. *Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(7), 6384–6392. https://doi.org/10.1609/aaai.v35i7.16760.

- Wajid Ahmed Channa1 , Qamar Uddin Khand2 , Sajid Ahmed Ghanghro3 , Abdul Moiz Ghanghro4 .(2019). Reasoning and Representation of Legal Cases Through Ontology IJCSNS International Journal of Computer Science and Network Security, VOL.19 No.8, August. https://www.researchgate.net/publication/336019837.

- Rector, A. L., Brandt, S., & Schneider, T. (2011). Getting the foot out of the pelvis: Modeling problems affecting use of SNOMED CT hierarchies in practical applications. *Journal of the American Medical Informatics Association*, *18*(4), 432–440. https://doi.org/10.1136/amiajnl-2011-000087

- Grau, B. C., Horrocks, I., Kazakov, Y., & Sattler, U. (2008). Modular reuse of ontologies: Theory and practice. *Journal of Artificial Intelligence Research*, *31*, 273–318. https://doi.org/10.1613/jair.2420

- Holger Knublauch, Alan Rector, Robert Stevens, Chris Wroe1.A Practical Guide To Building OWL Ontologies Using The Prot×eg×e-OWL Plugin and CO-ODE Tools Edition 1.0 .Copyright c The University Of Manchester. August 27, 2004.

- Horridge, M., Knublauch, H., Rector, A., Stevens, R., & Wroe, C. (2004). A practical guide to building OWL ontologies using the Protégé-OWL plugin and CO-ODE tools. *University of Manchester*. Retrieved from http://protege.stanford.edu.

- Baader, F., Calvanese, D., McGuinness, D., Nardi, D., & Patel-Schneider, P. F. (Eds.). (2003). *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press.

- Dr. D.D. Breimer, holger .Building a knowledge-based ontology of the legal domain. (Applied legal epistemology), 20 juni 2002 te klokke 15.15 uur door Laurens Mommers geboren te Doetinchem in 1972 .pdf

- Holger Knublauch . Ontology-Driven Software Development in the Context of the Semantic Web: An Example Scenario with Prot g OWL. Stanford Medical Informatics, Stanford University, CA, holger@smi.stanford.edu.

- معرض الزهور ونباتات الزينة الثالث (2025). قطاع الزراعة والثروة الحيوانية غريان, (14-15-16 يونيو/2025م) تحت شعار"غريان السلام والجمال", المجلس البلدي غريان.http://agriculture.gov.ly/
- المؤتمر العلمي الدولي الأول للثقافات المحلية في ضوء المتغيرات الحالية (22-23 /06/ 2021م) "تحت شعار معاً من أجل ثقافة تجمعنا" , وزارة الثقافة والتنمية المعرفية , مكتب الثقافة والتنمية المعرفية /غريان.

Tools for Ontology Engineering

- http://protege.stanford.edu/
- http://www-ksl.stanford.edu
- http://www.isi.edu/isd/ontosaurus.html
- http://ontoserver.aifb.unikarlsruhe.de/ontoedit/
- http://kmi.open.ac.uk/projects/webonto/
- http://webode.dia.fi.upm.es/webODE/