Engineering and Scientific International Journal (ESIJ)
Volume 9, Issue 2, April – June 2022
ISSN 2394-7187(Online)
ISSN 2394 - 7179 (Print)

MOTunAr Ontology: Creation and Axioms Impacts

Degachi Haifa[*1], Yengui Ameni[2], and Neji Mahmoud[3]
[1]*Ph. D. Scholar,* [2]*Instructor,* [3]*Professor*
[1,2,3]*University of Sfax, Tunisia*
[*]*Corresponding Author Email-id: Haifa.degachi@gmail.com*

Abstract — Due to its potential for supporting heterogeneous and various data, ontologies are used in serval domains, notably the archaeological one. In this paper, we want to present a multimedia ontology that designs the different entities involved in the Tunisian archaeological field. On another hand, the task of creating ontology is error-prone. The quality of ontology should be sequentially evaluated based on various criteria (e.g., coherence, consciences, interoperability, etc.). Axioms present the guarantor to satisfy high quality for a developed ontology. Therefore, we detail in this work a hybrid approach that guarantees the quality of the generated ontology. This approach combines the corrective method that defines the positive axioms, and the constructive method that defines all relevant axioms based on the elimination of model and integrity constraints. The generated ontology is evaluated with the Pellet reasoner and OOPS! Online service.

Keywords — *Axioms; Criteria; Evaluation; MOTunAr; Ontology.*

1. Introduction

In the archaeological field, gathering information is an essential process. It presents the base for interpretation and analysis. The analysis aims to deduce semantic information based on the extracted data. Ontologies have a great capacity for providing obtained significant dynamism due to their great capacity for providing different approaches for maintaining, sharing, managing, retrieving, searching, and viewing knowledge[1]. Due to its important capacities, ontologies are used in serval domains. Our research is focused on developing a multimedia ontology of the Tunisian archaeological field (MOTunAr). This ontology enables the description of all the Tunisian archaeological sites. On another hand, the creation of ontology is based on various sources of knowledge, including unstructured sources (e.g., web pages), semi-structured sources (e.g., dictionaries), and structured sources (e.g., database). For this objective, we propose the two hypotheses to benefit from these various sources:

- The sources have different interests that should be taken into consideration during the development of ontology.
- Confidence related to such information increases when this one appears in serval sources.

In order to take into account these two hypotheses we define two purposes: 1) each source have a score of interest that explains its relevance, dependent on this score, a source of information is used to extracted data or it is rejected from the knowledge base of our ontology, 2) each extracted information have a score of confidence is intimately related to the origin of information (i.e., if the score of interest of this information is raising the score of confidence of information increase automatically.) as well as the presence of this information in many sources. Depending on the score of confidence extracted information is accepted or is rejected.

Based on these hypotheses, we have defined a methodology to develop an ontology that consists of six steps[2]–[4]:1) filtering of data step, 2) Validation of the Classes step, 3) Subclasses step, 4) properties step, 5) axioms step, and 6) Population step. In addition, Despite the important progress in the methodology of creating ontology, most methods existing in the literature define only the positive axioms [5]–[7] . For MOTunAr ontology, we have defined a hybrid approach that combines the corrective method (i.e., defines the positive axioms) and the constructive method (i.e., defines all the relevant axioms) to guarantees a high quality of the MOTunAr ontology. The generated ontology is evaluated based on both the Pellet reasoner [8] from the protégé editor and OOPS! Online service[9]. MOTunAr shows a significant degree of coherence and consciences, due to the implementation of relevant axioms.

The rest of this paper is organized as the following: In section 2, we present our approach to create the MOTunAr ontology and we detail the different steps especially the axioms steps. In section 3, we present the evaluation part that is based on the Pellet reasoner from the protégé editor and the OOPS! Online service. Finally, in section4, we conclude this paper, and we present an outlook for future work.

2. MOTunAr Ontology: Steps of Creation

MOTunAr ontology is a multimedia ontology oriented to the Tunisian archaeological field. This ontology aims to describe all the entities (i.e., classes, subclasses, properties, and axioms) related to the Tunisian cultural heritage field. Therefore, we have defined an approach that contains six

DOI: **10.30726/esij/v9.i2.2022.92005**

ISSN 2394-7187(Online)
ISSN 2394 - 7179 (Print)

steps in order to develop the MOTunAr ontology: 1) filtering of data step, 2) Validation of the Classes step, 3) Subclasses step, 4) properties step, 5) axioms step, and 6) Population step.

2.1 Filtering of the data step

This step is based on automatic filtering of knowledge. We implement a set of modules that aid to restrict the domain of interest. The extraction task starts with the alignment between the implementation module and the existing sources and resources. As the first result of the extracted task, we obtain a list of candidate classes (concepts). Such a concept is conserved when it exists in the list of the terms aligned with the module. In another way, the extracted class would be rejected and would not be expanded to the final resources. For this purpose, we obtain a list of potential concepts related to the domain of interest. Indeed, these classes describe the various periods in Tunisian history (e.g., the Carthaginian period, the Islamic period, the Roman period, etc.). Based on the resulted list we deduce the relevant classes for the MOTunAr ontology.

2.2 Validation of the Classes Step

Based on the Extraction-Terms algorithm, MOTunAr ontology defines two main classes: 1) T1-Archaeological-Entity that includes as subclasses: a) T11-Site, and b) T12-Monument. And 2) T2-Archaeological-Coordinates that includes as subclasses: a) T21-Appellation, b) T22-Anthroponym, c) T23-People, d) T24-Temporal-Entity,e) T25-Spatial-Entity , f) T26-Physical-Object and, g) T27-Conceptual-Object (fig.1).

Algorithm 1: Extraction-Terms

Input: S, */*S are the set of sources that we based on to extract our candidates terms*/*
Output: Res */*Res is the variable that contains the candidates terms*/*
For i: =1 to n do */*n is the number of sources that we have used*/*
{
Repeat
If (cr1 & cr2 & cr3 & cr4) then */* cr1is the reputation of sources; cr2 is the freshness of sources; cr3 is the adequacy of sources; cr4 is the clarity of sources*/*
Data-Recovery ←retrieves all data related to sources;
Conceptual abstraction ←determines model for data;
Exploration-Information ←defines common format of data;
While (Common-Knowledge);
If (Common-Knowledge) then
{
Repeat
Res← Search pattern in ODP */* ODP Portal (Ontology Designed Patterns). This portal, as its name indicates, groups the designed patterns as well as the transformation patterns*/*
While (not ODP) or (exist); */* if we have tested all the existing ODP (noted) or we have found (exist) a model from ODP)*/*
If (exist) then
Res← Application-Of-Transformation;
Else
Res← Ad-Hoc-Transformation-Method;
Res← Manual-Correction /* enables to retake the updated transformation to manually correct the various errors that can appear*/
If (Res= Alignment of the sources knowledge base) then */* the correspondences between the different pairs exists on the considered S are created*/*
{
CT← final generation of coordinates terms */* CT: Candidates Terms; the candidates of ontological elements or the relation are generated*/*
CT←SC; */* SC: Score of Confidence; A confidence score is associated with each candidate*/*
}
If (SC>=2) then
Res← CT; /* the variable Res receive the set of candidates terms*/}}

2.3 Subclasses Step

In the previous step, we have defined the two main classes: The T1-Archaeological-Entity class aims to define the different archaeological entities involved in the Tunisian archaeological field. This class includes two subclasses a) T11-Site, and b) T12-Monument [2].

The T2-Archaeological-Coordinates class aims to design the various data that can be added to archaeological entities. For this class we have defined the following set of subclasses [3].

- T21-Appellation: it describes the different designations that can be added to such archaeological entities. Three subclasses are added to the T21-Appellation class: 1) T211-Identifier, 2) T212-Old-Name, and 3) T213-Current-Name.
- T22-Anthroponym: it defines persons which are related to the archaeological entities. Three subclasses are defined for this class: 1) T221-Hero, 2) T222-Religious-Person, and 3) T223-Divinity.
- T23-People: it represents the different inhabitants of a mentioned place which are indicated in a historical document. T23-People class defines three subclasses: 1) T231-People-Appellation, 2) T232-Civilization, and 3) T233-Emperor.

DOI: **10.30726/esij/v9.i2.2022.92005**

ISSN 2394-7187(Online)
ISSN 2394 - 7179 (Print)

- T24-Temporal-Entity: it designs the different time terms. We have added a hierarchy of subclasses to this class: 1) T241-Event that includes T2411-Event-of-modification, T2412-Beginning-Of-Existence, and T2413-End-Of-Existence subclasses, and 2) T242-Chronological Slice that includes T2421-Absolute-Dating, T2422-Periodization, and T2423-Century subclasses.
- T25-Spatial-Entity: it describes the entities related to the location and space of the different archaeological entities. We have added a hierarchy of subclasses to this class: T251-Name-Place, T252-Geographical-Coordinates, T253-Municipality, T254-Surface, T255-Delimitation, T256-Structure, and T257-Current Occupation.
- T26-Physical-Object: it describes all physical entities, and it defines three subclasses: 1) T261-Architectural-Element that includes the subclasses T2611-Decoration, T2612-Architectural-Ornament, T2613-Sculpture, and T2614-Architectural-Style, 2) T262-Materials, and 3) T263-Legal-Statute.
- T27-Conceptual-Object: it includes the different no physical information that describes the archaeological entities. Two main subclasses are added to this class: 1) T271-Description-Of-Site, it includes as subclasses T2711Category, T2712-General-Data, T2713-State-of-Conservation, T2714-State-Of-Research, T2715-Number-Of-Monuments, and T2716-World-Heritage, and 2) T272-Documentation that includes the subclasses T2721-Document, and T2722-Visual-Element.

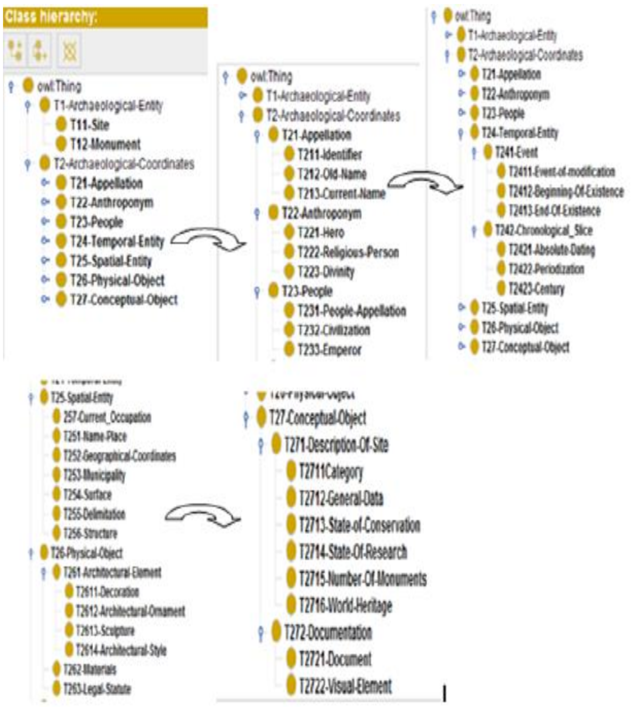


Fig.1: Hierarchy of classes

This set of classes and subclasses are related to the different properties that will be described in the following subsection.

2.4 Properties step

Three kinds of properties are implemented in the MOTunAr ontology[10]:1) Object properties, 2) Data properties, and 3)Annotation properties.

- *Object Properties (Relations):* The Object properties or Relations (R) in ontology represent a type of interaction between the concepts. For our ontology we have distinguished three kinds of relations [11]: 1) Hierarchical relations (HR), 2) Associative relations (AR), and 3) Semantic relations (SR).
- Hierarchical relations: based on these relations we create the structure of the ontologies. In our approach we distinguish four hierarchical relations: 1) Subsumption (P10-is-a), 2) Generalization (P53-is-more-generic-than), 3) Specification (P54-is-more-specific-than), and 4) Part-of (P50-is- Part-of). To define the hierarchical relations we applied the hierarchical relation definition algorithm.

Algorithm 2: Hierarchical Relation Definition

Input: Training set O= {(C1, C2, R}, */* C1, C2 are the concepts for which we try to define a hierarchical relation; R is anonym relation among C1, C2 */*
Output: HR
Randomly initialize θ;
While training is not terminated do
Switch R
Case Subsumption:
HR← is-a;
Is-a (C1, C2);
Break;
Case Generalization:
HR← is-more-generic-than;
Is-more-generic-than (C1, C2);
Break;
Case Specification:
HR← is-more-specific-than;
Is-more-specific-than (C1, C2);
Break;
Case Part:
HR← is-part-of;
Is-part-of (C1, C2);
Break;
Default:
HR← ϴ;

DOI: **10.30726/esij/v9.i2.2022.92005**

ISSN 2394-7187(Online)
ISSN 2394 - 7179 (Print)

- *Associative relations:* Based on these relations we create the conceptual representations between terms. We benefit from the existing sources and resources that describe our interest field. Candidate relations are extracted and a score of confidence (SOC) is associated with each one of these relations. Depending on the SOC a candidate relation can be conserved or rejected. To define the associative relations we applied the associative relation definition algorithm.

Algorithm 3: Associative Relation Definition

Input: Training set O= {(C1, C2, R}, RES, S/* *C1, C2 are the concepts for which we try to define an associative relation; R is anonym relation among C1, C2 */*
/ resources RES and sources S*/*
Output: AR, SOC /* AR is the associative relation which we want to extract; SOC is the score of confidence*/
Randomly initialize θ;
While training is not terminated do
If exit(R) ==true then/* exist is a function which search the candidate relation R on the resources RES and sources S*/
SOC++; /* score of confidence of the relation R increase when it exists in different documents*/
If SOC>=2 then
AR← R;
Else
R is rejected; /*if the score of confidence is less than 2 the relation R is rejected from the list of relations candidates*/

- Semantic relations: based on these relations we create the semantic representations between terms. The definition of semantic relations is carried out between each pair of concepts. A double-entry matrix is then produced [12]. This matrix contains in a row and a column all the different concepts already defined. Each cell of the matrix contains semantic relations which can link two concepts together. To define the semantic relations we applied the semantic relation definition algorithm.

Algorithm 4: Semantic Relation Definition

Input: Training set O= {(C1, C2,, Cn, R}, n, m/* *C1, C2 are the concepts for which we try to define a hierarchical relation; R is anonym relation among C1, C2 */*
/ n is the number of lines; m is the number of column*/*
Output: SR,
Randomly initialize θ;
While training is not terminated do
For i := 1 to n do
For j:= 1 to m do
If R (Ci, Cj) then
SR←R;

- *Data Properties (Attributes):* Data properties or Attributes **(A)** relate an instance to data values (e.g., XML Schema, RDF literal, Datatype value, etc.) [13]. For MOTunAr we have defined a set of Data Properties that added value to, especially entities.

For each Data property, we defined the rdfs: domain and rdfs: range. Domain presents one of the existing classes or subclasses (e.g., T222-Religious-Person, T261-Architectural-Element, T271-Description-Of-Site, etc.). And the ranges express the Datatype that can be defined by the data properties (e.g., XSD: string, RDF: XML literal, rdfs: literal, owl: real, etc.).

Table 1. Examples of Data properties

Data properties	Domain	Range
has-designation	T212-Old-Name T213-Currenr-Name	xsd :string
has-number	T212-Old-Name T213-Currenr-Name	xsd :integer
Has-origin	T212-Old-Name T213-Currenr-Name	xsd :string
has-birth-day	T221-Hero T222-Religious-Person T233-Emperor	xsd :date Time
has-death-day	T221-Hero T222-Religious-Person T233-Emperor	xsd :date Time

Table 1 shows some data properties related to the MOTunAr ontology as well as their ranges and their domains. Data properties, as well as the Object properties, aim to relate the different entities and to transform data into knowledge. The annotation properties aim to add information to the different entities (i.e., concepts, relations, attributes)[10].

- *Annotation Properties:* Annotation properties are used to more describe such entities and add metadata to the different declared objects [10].

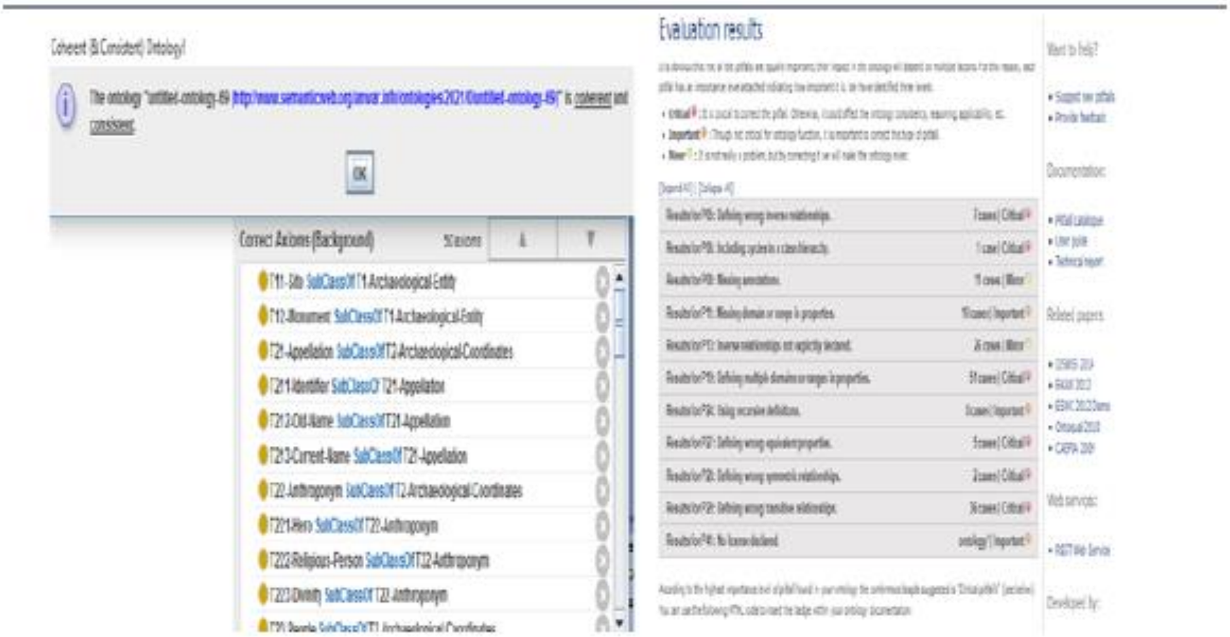
Many formats of annotations are implemented into the MOTunAr ontology in order to add metadata to different

DOI: **10.30726/esij/v9.i2.2022.92005**

ISSN 2394-7187(Online)
ISSN 2394 - 7179 (Print)

Table 2. Positive Axioms Examples

Designation	Explication	Syntax	Protégé declaration examples
Reflexive property	Class C is in relation R with the same class C.	(R^I) -is reflexive	Reflexive : P1-is-associated-with
Irreflexive property	C can't be in relation R with the same class C.	(R^I) - is irreflexive	Irreflexive :P 18-has-constructed
Symmetric property	C_1 is in R with C_2 and C_2 is in R with C_1.	$R^I = (R^I)^-$	Symmetric: P3-is-related-to
Asymmetric property	C_1 is in R with $C_2$and C_2isn't in R with C_1.	$R^I =(R^I)$	Asymetric : P53-is-identified-by
Transitive property	C_1 is in R with C_2 and C_2 is in R with C_3 then C_1 is in R with C_3.	$R^I = (R^I)^+$	Transitive :P10-is-a
Inverse property	Each relation R can have an inverse relation.	$R^I = (R^I_0)^-$	P12-is-operated-by InverseOf P13-has-operated
Functional Property	A relation R is declared functional when no more than one individual can be linked to it.	R^I is Functional	Functional: P53-is-identified-by
Inverse Functional	The inverse relationship is declared functional.	(R^I) - is Functional	InverseFunctional: P54-has-identified
Domain/Domain	If an individual x is connected by R with some other individual, then x is an instance of C.	$R^I \subseteq C^I_i \times \Delta^I_D$	P1-is-associated-with Domain T2-Archaeological-Coordinates
Range/Range	if some individual is connected by R with an individual x, then x is an instance of C.	$R^I \subseteq \Delta^I \times C^I_i$	P1-is-associated-with Range T1-Archaeological-Entity



(a) (b)

Fig.2: (a) Pellet reasoner, (b) OOPS! service

DOI: **10.30726/esij/v9.i2.2022.92005**

objects including in the MOTunAr ontology. Such as the rdfs: IsDefinedBy annotation that is used to give the definition for each object from MOTunAr ontology. Also, we have used rdfs: label in order to add the French name for each entity (i.e., the French version of MOTunAr is available when we use the option « render by label » from the view menu in the protégé editor).

2.5 Axioms Step

Two approaches are combined in order to define the axioms of the MOTunAr ontology:

- *Corrective approach:* in order to guarantee the coherence and the excitability of ontology, in this step, we have defined a set of assertions that give meaning to the already defined entities and allow their good exploitation. Eventually, the different editors of ontology existing in the literature assure the definition of various axioms depending on their capacities. For MOTunAr we have based on Protégé editor due to its efficiency and simplicity of use [14]. Protégé editor defines especially the positive constraint, i.e. axioms define positive inferences (e.g., a hierarchy of classes, domains and co-domains of properties, etc.). Via protégé editor, we have added a set of positive axioms to MOTunAr ontology. The definition of the axioms is depending on the entity that it describes. For MOTunAr three types of positive axioms are defined: 1) Class Expression Axioms, 2) Object Property Axioms, and 3) Cardinality Restrictions Axioms.
- Class Expression Axioms: In our work four axioms are declared [15] :
  - Class declaration axioms: *Declaration (Class (T223-Divinity)).*
  - SubClassOf axiom: *SubClassOf (T212-Old-Name, T21-Appellation).*
  - Equivalent Classes axioms: *T231-People-Appellation EquivalentTo T234-People-Name.*
  - Disjoint Classes axioms: *T11-Site DisjointWith T12-Monument.*
- Object Property Axioms: it tends to determine the axioms that could be exploited to find relations between item property terms. Table 2 shows the same examples of the Object Property Axioms.
- Cardinality Restrictions Axioms: In the MOTunAr ontology we have added the following cardinality restrictions [16]: 1) SomeValues (e.g., P42-has-triggered some T241-Event), 2) AllValues (e.g.,P18-has-starting-during only (T232-Civilization or T242-Chronological-Slice)), 3) HasValues (e.g.,P15-has-existed-during exactly 1 T232-Civilization), 4) MinCardinality (e.g.,P16-has-coincided-with min 1 T223-Divinity), and 5) MaxCardinality (e.g.,P15-has-existed-during max 1 T232-Civilization).

This approach allows to clearly defining the positive axioms but for the negative ones, it is limited to the separation constraints. We define in the following section the constructive approach that allows the definition of all relevant axioms that can be added to the MOTunAr ontology.

2.5.1 Constructive approach

This approach is under development. On other hand, under this step, the axioms are generated by the elimination of models. The ontology resulting from the application of the corrective approach is considered as reference ontology. This ontology will be transformed into a set of models (i.e., class diagram). Two possible contexts are presented:

- Elimination of Model therefore a negative axiom is automatically generated.
- Conservation of model therefore OCL constraints are adopted. These constraints will be transformed to OWL axioms and added to the existing MOTunAr axioms.

Once the implementation of declared axioms is achieved; we want to define the instance: the population step.

2.6 Population Step

The population step is a future step for MOTunAr ontology. We want to define the instances (individuals) under this task. For our ontology, we will be based on the BOEMIE methodology to define the instance for classes and subclasses as well as for the properties. BOEMIE methodology offers such advantages in comparison to the existing methodology in literature [17] :

- The engine of extraction instances of concept/relation is not anticipated to extract instances of the composite classes. It is anticipated to extract just instances of the primitive classes. An evident benefit is a modification in the structure of the ontology becomes immune. The adaptation of the extraction engine is a necessity only when such \primitive" classes or relations related to primitive classes are modified.
- The ontology is applied to extract an instance of a \composite" class from an instance of populated \primitive" class of either of the populated relation instances. Two main advantages of this approach that are:1) the instances of the \composite" classes are constantly synchronized with the available formal definition added to the relevant classes, and 2) the generation of \composite" instances, depending on the constraints and the rules imposed by the implemented ontology these rules ensure the consistency and efficiency of the ontology.

DOI: 10.30726/esij/v9.i2.2022.92005

ISSN 2394-7187(Online)
ISSN 2394 - 7179 (Print)

The population step is the final step in our methodology. Thus, in order to verify the effectiveness of the developed ontology, the following section detail the evaluations applied to the MOTunAr based on different tools existing in the literature.

3. Evaluations of MOTunAr ontology

MOTunAr ontology has been developed based on a flexible approach, which allows the coherence, clarity, and consciences of each declared term. Despite this, the implementation of a pertinent methodology for developing ontologies does not promise to obtain ontologies free of errors. In addition, the task of developed ontologies is error-prone; some anomalies can appear on the developed ontology. Therefore, the evaluation of ontology is considered as a fundamental step to guarantee the quality of generated ontology and reduce future maintenance.

For our study we have applied two tools to evaluate MOTunAr ontology: 1) Pellet reasoner from Protégé editor and [9] 2) OOPS! Service [10].

Pellet does expertness analysis of ontology. The service reasoning find by Pellet help to debug for incompatibility and incoherence that can be detected Fig.2.a. To assure the quality of MOTunAr ontology, we have applied the OOPS! Service with the Pellet reasoner OntOlogy Pitfall Scanner! (OOPS!), is a service available at, http://oops.linkeddata.es/response.jsp. It aims to aid the ontology designer during the ontology evaluation task. OOPS! Executed independently of the ontology creation platform, this service produces pitfalls (i.e., bad practices in the task of development of ontology) that describe the errors detected among the tested ontology Fig.2.b.

Despite MOTunAr being under development, both, pellet reasoner, and OOPS! Service shows a good result for this ontology. The implementation of a constructive approach will be allowed:

- More conscience ontology by defining all relevant axioms and controlling redundant axioms.
- More coherent ontology by manipulating all contradictions provoke by axioms.
- More interoperability ontology by reusing and interacting axioms from other ontologies.

4. Conclusion and Future Works

In this paper, we have presented a new approach to developing a multimedia ontology. This approach is realized among six steps: 1) filtering of data step, 2) Validation of the Classes step, 3) Subclasses step, 4) properties step, 5) axioms step, and 6) Population step. To verify the quality of the MOTunAr ontology, we have based on the Pellet reasoner and OOPS! Online service. The result of the evaluation shows the effectiveness of the developed ontology. In addition, we benefited from the experience of a domain expert (i.e., Tunisian archaeologist) to have a correct sense for the different defined terms and to cover our studied domain (i.e., Clair ontology). The use of annotation properties, as well as the use of simple entities appellation (e.g., people, hero, event, was-associated-to, is-a, etc.), make the MOTunAr ontology understandable either by an expert or by a simple user (i.e., easy to use ontology). The definitions of the relevant axioms guarantee the satisfaction of coherence, conscience, and interoperability criteria of evaluation. Despite, the axioms step is under development, the achievement of this step will be added more effectiveness to the MOTunAr ontology due to the implementation of all pertinent axioms. To have a completed, coherent, and Clair multimedia ontology that defines all entities related to the Tunisian archaeological field. As ongoing steps we plan to apply the BOEMIE methodology to achieve the population step as well as the MOWL (Multimedia Ontology Web Language) to take into account the multimedia format of such entities.

References

[1] B. Ben Mahria, I. Chaker, and A. Zahi, "A novel approach for lesarning ontology from relational database : from the construction to the evaluation," *J. Big Data*, 2021, doi: 10.1186/s40537-021-00412-2.
[2] D. Haifa, Y. Ameni, and N. Mahmoud, "Accomplishment of multimedia ontology for the tunisian archaeology field," in *Advances in Intelligent Systems and Computing*, vol. 1184, Springer Science and Business Media Deutschland GmbH, 2021, pp. 64–72.
[3] D. Haifa, Y. Ameni, and N. Mahmoud, "Multimedia Ontology of the Tunisian Archaeology Field," in *Advances in Intelligent Systems and Computing*, Jul. 2020, vol. 1105 AISC, pp. 321–330, doi: 10.1007/978-3-030-36674-2_33.
[4] D. Haifa, Y. Ameni, and N. Mahmoud, "Towards a Multimedia Ontology for Tunisian Archaeology Field," Dec. 2019, doi: 10.1109/ICTA49490.2019.9144900.
[5] S. Belabbes, S. Benferhat, and J. Chomicki, "Elect : Une méthode de gestion des incohérences dans des ontologies légères partiellement préordonnées *," Jul. 2019, Accessed: Oct. 01, 2021. [Online]. Available: https://hal.archives-ouvertes.fr/hal-02301980.
[6] S. Salminawati, "ONTOLOGICAL BASIS OF SCIENCE CLASSIFICATION (Study on the Philosophy of Islamic Education)," *Edukasi Islam. J. Pendidik. Islam*, vol. 9, no. 02, pp. 683–700, Aug. 2020, doi: 10.30868/EI.V9I02.1581.
[7] M. Ben Salah, A. Yengui, M. Neji, and N. Mahmoud, "Construction of OTunAr : ontology of Tunisian archeology," 2017. Accessed: Mar. 02, 2021. [Online]. Available: https://www.researchgate.net/publication/340922998.
[8] S. Gherbi and M. T. Khadir, "ONTMAT1: Ontology matching using a reasoner and property restriction," *Int. J. Web Eng. Technol.*, vol. 15, no. 2, pp. 119–139, 2020, doi: 10.1504/IJWET.2020.109728.
[9] A. Mathews, S. Chen, M. T. Bigham, and K. Mansel, "Oops: rapid Deterioration of the Transport Patient Admitted to the General Care Floor," *Pediatrics*, vol. 141, no. 1 MeetingAbstract, pp. 728–728, Jan. 2018, doi: 10.1542/PEDS.141.1_MEETINGABSTRACT.728.
[10] L. Olsina, "ThingFO v1.2's Terms, Properties, Relationships and Axioms -- Foundational Ontology for Things," Jul. 2021, Accessed: Jan. 30, 2022. [Online]. Available: http://arxiv.org/abs/2107.09129.
[11] C. Roche and M. Papadopoulou, "Terminologie et ontologie pour les humanités numériques : le cas des vêtements de la Grèce

DOI: **10.30726/esij/v9.i2.2022.92005**

antiqueTerminology and Ontology for Digital Humanities: The Case of Ancient Greek Dress," *Humanit. numériques*, no. 2, Jun. 2020, doi: 10.4000/revuehn.462.

[12] M. Rasmussen, M. Lefrançois, M. Bonduel, C. Hviid, and J. Karlshøj, "OPM: An ontology for describing properties that evolve over time," Jun. 2018, Accessed: Oct. 01, 2021. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01885248.

[13] H. J. Pandit, K. Fatema, D. O'Sullivan, and D. Lewis, *GDPRtEXT - GDPR as a Linked Data Resource*, vol. 10843 LNCS. 2018.

[14] M. Florrence, "MLGrafViz: Multilingual ontology visualization plug-in for protégé," *Comput. Sci. Inf. Technol.*, vol. 2, no. 1, pp. 43–48, Mar. 2021, doi: 10.11591/CSIT.V2I1.P43-48.

[15] F. Z. Smaili, X. Gao, and R. Hoehndorf, "Formal axioms in biomedical ontologies improve analysis and interpretation of associated data," 2018, doi: 10.1093/bioinformatics/xxxxxx.

[16] F. Freitas and I. Varzinczak, "Cardinality Restrictions Within Description Logic Connection Calculi," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11092 LNCS, pp. 65–80, Sep. 2018, doi: 10.1007/978-3-319-99906-7_5.

[17] F. Song and E. De La Clergerie, "Clustering-based Automatic Construction of Legal Entity Knowledge Base from Contracts," *Proc. - 2020 IEEE Int. Conf. Big Data, Big Data 2020*, pp. 2149–2152, Dec. 2020, doi: 10.1109/BIGDATA50022.2020.9378166.

DOI: **10.30726/esij/v9.i2.2022.92005**