

# Thai Tattoo Wisdom's Representation of Knowledge by Ontology

Wirapong Chansanam <sup>1,\*</sup>, Kulthida Tuamsuk <sup>1</sup>, Kanyarat Kwiecien <sup>1</sup>, Kittiya Sutthiprapa <sup>1</sup> and Thepchai Supnithi <sup>2</sup>

<sup>1</sup> Department of Information Science, Faculty of Humanities and Social Sciences, Khon Kaen University, Khon Kaen 40002, Thailand; kultua@kku.ac.th (K.T.); kandad@kku.ac.th (K.K.); kittsu@kku.ac.th (K.S.)

<sup>2</sup> Language and Semantic Technology Laboratory, National Electronics and Computer Technology Center, Pathum Thani 12120, Thailand; thepchai.supnithi@nectec.or.th

\* Correspondence: wirach@kku.ac.th; Tel.: +66-90956-2727

**Abstract:** Sak Yan Ontology (SYO) models knowledge derived from Thai tattoos in the design of cultural heritage preservation planning. Ontology Development 101 is a technique of ontology model creation. The aims of this study are to share the performance of ontology development and ontology evaluation. The study is specifically focused on validation from domain experts and automation evaluated using the OOPS! tools (Ontology Pitfall Scanner is a tool that helps detect some of the most common pitfalls appearing when developing ontologies). The results obtained from OOPS! show that SYO is devoid of critical errors; however, it does have one critical, three important, and three minor problems. Four of the problems are fixed, whereas the others are continuous. The combination of automatic and human validation methodologies improves the quality of the ontology being modeled. The tools enhance the traditional methodology with quicker, easier, and smaller amounts of subjective analysis. In conclusion, for the reparation movement, solutions for the above problems are suggested.

**Keywords:** Sak Yan ontology; ontology modeling; ontology evaluation; digital humanities; OOPS!

**Citation:** Chansanam, W.; Tuamsuk, K.; Kwiecien, K.; Sutthiprapa, K.; Supnithi, T. Thai Tattoo Wisdom's Representation of Knowledge by Ontology. *Informatics* **2021**, *8*, 3. <https://doi.org/10.3390/informatics8010003>

Received: 16 October 2020

Accepted: 28 December 2020

Published: 21 January 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Yantras have become commonly recognized by Western people attracted to the mysterious and pious belief systems of East and Southeast Asia. Yantra is a Sanskrit (Sk) word meaning, “A tool used for control”. The Association of Southeast Asian Nations (ASEAN) is a regional grouping that promotes economic, political, and security cooperation among its 10 members: Brunei, Cambodia, Indonesia, Laos, Malaysia, Myanmar, the Philippines, Singapore, Thailand, and Vietnam. Yantras are referred to as the same by the Pali (P) or as Sak Yan in Thai (Th). From the perspective of Western people, Sak Yan might be considered a “magic charm”. However, it is quite sophisticated. Sak Yan combines the use of holy words and a sacred geometry with the magical use of various methods for making the fine art, skills (handicraft), and knowledge of superpowers natural. The preparation of Sak Yan is commonly dependent upon the working experience and knowledge of mantra (Sk), tantra (Sk), and astrology. In reality, Sak Yan is sometimes called the physical expression of a mantra [1].

In the East, a yantra tattoo has a superstitious meaning. Its Sanskrit root term means a “tool” or “weapon.” However, in the Western understanding, a yantra is called a diagram. It appears as a tool symbol and is an automated process such as a machine. Yantras work through structures and organizations. In the West, yantra symbols are widely used in geometry. According to tradition, the Eastern concept of “mantrayanism” applies to religious beliefs such as the Mandara in Tibet. Those who have a talisman tattoo must show various manners that come out of or are expressed by hand as a symbol called

“Mujra”. Sak Yan is a field of spirituality as well as astrology, sorcery, and witchcraft. Sak Yan is a belief that such a tattoo on the skin will help keep the individual safe.

Sak Yan has the power to protect from various dangers and destroy them. Moreover, some types of tattoos can inspire desirable, attractive, and popular elements. In Thai society, a Sak Yan of the Thai people is mostly tattooed as an amulet, particularly for Thai men. These days, Sak Yan for beauty or charm has also begun to play a role in combination with belief. However, in Thai society, Sak Yan is seen as an imprint on a talisman. The motifs are characters, incantations, monks, animals, gods, and religious symbols. In addition, the process of Sak Yan is complex, depending on the school, and a teacher applies a tattoo on the skin because Thai society still sees the tattoo as magical. The belief in supernaturalism has been around for a long time, such as the belief that Sak Yan can withstand all types of weapons. There is an idea that a “Salika pattern” causes mercy, incredible popularity, and love from aging people in society. Some also believe that a Sak Yan on male and female genitals can charm the opposite sex.

In the first period, a Sak Yan tattoo was a belief related to culture and custom, such as tattoos on the legs. During this period, men had tattoos to indicate their bravery for acceptance in a society. Afterward, during periods of war, the tattoo culture preferred soldiers who had to fight their enemies. At the same time, they believed that the tattoo could protect them from weapons, and the Thai men therefore liked to have tattoos on their skin because it gave them confidence during battle. At the present time, Thailand’s tattoo culture only appears in certain areas because, in the era of globalization in Thai society, belief in the power of tattoos is less than in the past. People have been taught that tattoos are only art on the body. However, despite fewer people having spiritual beliefs about tattoos, some groups still believe that tattoos are sacred.

Indeed, ontologies can explain all data collected for realizing a conservation project, and in the meantime, corroborate on how the described information is perceived. To appropriately process the structure of the ontology, knowledge engineers and domain experts need to work on them jointly. Highly qualified in ontology representation language (ontology web language (OWL)), the former informatics experts write programs and manage logical coherence. Experts in cultural heritage contribute to the expertise required to enable the restoration of a historic building from the start of the conservation project to its future use and maintenance. They are responsible for taking care of the system and checking whether the material is reflective and sensitive.

Formalizing such an understanding involves defining the semantics of the concepts and paying careful attention to explaining the logical structure in which the semantics must be expressed. A more comprehensive and broader understanding of the heritage domain is included in testing both the interpretation of a particular object and the relationships, physical and conceptual. In a conservation project, such a complex approach may also deal with the ongoing dialectic between study and interpretation that typifies the path to knowledge [2].

An ontology is a commonly used tool for explaining the representation of knowledge [3,4]. The advantages of an ontology contain [5] its interoperability to share common understanding among people or software agents, it enables the reuse of domain knowledge and facilitates explicit conventions. Some ontologies of cultural knowledge have been developed by During et al. Signore, and Szasz et al. [6–9]. However, such ontologies focus on tangible cultural heritage and are intended to be used as information resources rather than as tools for information retrieval. Although some ontologies such as a belief ontology [10], event ontology for the historical domain [11], OARLCAM (An Ontology-based Approach for Reusing and Learning through Context-aware Annotations Memory) [12], ontology for an e-learning recommendation system [13], the cognitive dimension, and course content modeling [14], and imaginary-being ontology [15] are created through abstract concept modeling. However, some ontologies use a HOZO ontology editor tool (Hozo is a Japanese graphical ontology editor specifically created to produce heavy-weight ontologies and was developed through a partnership between the Department of Knowledge

Systems (Mizoguchi Laboratory), ISIR-Osaka University, and Enegate Co, Ltd.) [16] and are created and evaluated by domain experts and application-based evaluation methods. More significantly, an ontology of the knowledge of yantra has yet to be created.

This paper describes the issue in more depth because it is important for the ontology modeling creation process. It allows common modeling errors to be evaded and rechecks the practical quality of the ontology model being created.

As defined in [17], an ontology evaluation refers to the action of checking the practical quality of an ontology against the settings of a reference. An ontology evaluation contains an ontology validation process and verification activity. An activity is an accomplishment to be implemented, including its required input and output information, whereas a process is a set of ordered activities. The meaning of activity and process are taken from [17]. The ontology validation process can be separated into an ontology analysis and reparation. The first activity must be carried out before the latter. An ontology analysis is used to detect common modeling errors, whereas an ontology reparation is used to resolve such errors. An ontology verification contains an assessment between the ontology and the structure of the reference, which is acquired during the ontology requirement conditioning.

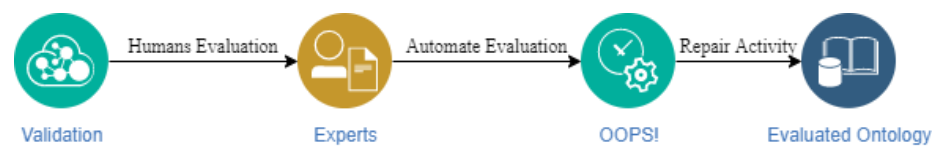
In the Ontology Development 101 technique, an evaluation includes four types of references: competency questions, application-based references, modeling guidelines, and the expert domain. The second is used with the software application, whereas the first, third, and fourth types of references are used during the ontology model creation. In this study, we considered the ontology evaluation between the ontology model creations. Thus, an application-based reference is applied in the next stage.

The questionnaire of competency is used as mentioned for verification. For the validation process, Ontology Development 101 [18] suggests three ways to evaluate the performance of an ontology construction, that is, experiment with it during the application, rectify it with domain experts, or do both. In this study, we designed a framework for evaluating ontology building by both specialists in the Sak Yan domain and ontology engineering experts. An expert evaluation was applied to the model creation design stage evaluation.

An automatic ontology evaluation tool, the ontology pitfall scanner! (OOPS!) [19], was chosen to validate the Sak Yan ontology (SYO) content. It was chosen because of its capability to execute both an ontology analysis and a reparation. The aims of this paper are to share the ontology model building and the procedure for evaluating the SYO by using domain experts and OOPS! This paper is organized as follows: Section 2 describes the research methodology used for the validation process in SYO, whereas Section 3 describes the ontology model creation and validation process operation. Section 4 summarizes the results, and Section 5 provides some concluding remarks.

## 2. Research Methodology

This section briefly explains the activities carried out for the ontology modeling building and validation process. The ontology diagnosis starts with a rough first pass of the ontology modeling. It checks whether the ontology content conforms to the principle of ontology engineering. In this part, the modeling construction is used to evaluate the ontology during the ontology modeling. The ontology content will be repaired if it does not conform to the recommendation by the experts. After the manual validation process is conducted by the experts, the ontology modeling continues with the automatic approach using the OOPS! tools. Figure 1 shows the validation process for SYO.



**Figure 1.** The proposed validation procedure.

### 3. Implementation

This section discusses the ontology modeling building, analysis, and reparation activity implemented by ontology engineers, domain experts, and the OOPS! tools. In Section 3.1, the framework for building the modeling of the class hierarchy to conform to during the analysis is described. Section 3.2 describes the results of SYO after being analyzed by OOPS! and Section 3.3 describes the reparation for important pitfalls according to the guidance generated by OOPS!

#### 3.1. Ontology Model Building

An ontology construction is a complicated and generally domain-focused process that can be promoted using tools. In [20], many ontology building tools are correlated based on certain functions such as modeling features, restrictions of functions, language support, web technology support and application, the transformation format, a graph generated view, inconsistency checking, and multiple user support. It was found that the Protégé ontology editor tool, which is a platform developed based on Java programming, is extensible and makes available a plugin environment as a flexible base for a quick prototype creation and application development [21]. Protégé is an open-source framework and free ontology editor suitable for intelligent system development [22]. Protégé is a tool that transforms data into an RDF/OWL format (the Resource Description Framework (RDF) is a framework for representing information in the Web. OWL is an ontology language for the Semantic Web with formally defined meaning. In this stage, we used the Protégé\_5.5.0 tool (Protege is an OWL ontology development environment) to construct an ontology for the knowledge domain. We chose Sak Yan as the topic (Thai tattoos) of the yantra domain to construct the ontology modeling of future systems. Several stages are involved in ontology creation [10,15,23]. The first stage is to collect comprehensive data on the knowledge domain. The second stage is to classify all classes and subclasses for the ontology to be created. The third stage is to configure the object and data properties between classes and subclasses. Object properties generally define relationships between instances or individuals of classes. Data properties define the relations between instances and values of the data. The domain and range must have properties. The fourth stage is to configure the domain and range of every property. Commentaries can also be added to the classes and properties for the domain description. The fifth stage is to generate instances or individuals of classes and configure their data and object properties to assign relationships between the instances of different classes and subclasses. Finally, the sixth stage is for consistency checking.

##### 3.1.1. Class and Subclass Creation

In the yantra domain, the principal concept of Sak Yan is considered to construct the ontology modeling for the system model. The Sak Yan ontology contains several classes and subclasses, for example, as shown in Figure 2. The root concept comprises classes of several main topics related to the yantra domain, such as Beliefs, AuspiciousTime, Calendar, Disciple, Equipment, Herbs, ImaginaryBeings, Languages, Liquid, Motives, Objectives, Patterns, Procedure, Prohibition, Properties, Ritual, SacredThing, Sakyan, Schools, Status, SurfaceArea, Tattoo, Type, and Yantras. These classes have several subclasses that act as their inheritance. For example, the Objectives class has the seven subclasses To-

Charm, ToFashion, ToGainCharisma, ToIdentity, ToMaturity, ToRemember, and ToTheWayOfLiving. ToCharm has two subclasses, i.e., ToAttractiveness and ToLovingKindness (Figure 3).

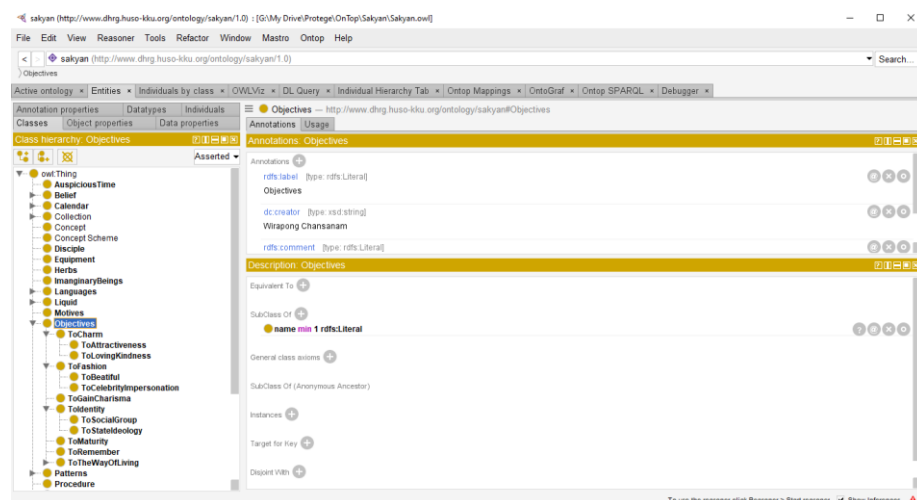


Figure 2. Classes and subclasses of Sak Yan ontology.

### 3.1.2. Object and Data Property Identification

Figures 3–5 show the properties of different classes and subclasses. Many object properties that define the relationship between instances or individuals of classes have been configured. Some of the object properties are Affiliation, hasBelief, hasComponents, hasDateOfCalendar, hasDisciple, hasLanguage, hasMotives, hasObjectives, hasPartOf, hasProdecure, hasProhibition, hasProperties, hasStampArea, hasStatus, hasType, and hasYantras (Figure 4). The data properties are the address, details, eng\_name, isAffiliation, isAuspiciousTime, local\_name, meaning, month, name, pictures, properties, rector, and thai\_month (Figure 5). All properties have domain and range, which need to be configured to create an ontology. For example, the hasPartOf object property has the domain Sakyan and ranges Ritual, Schools, YantrasNonObject, SurfaceArea, Objectives, Sacred-Scripts, Type, and Tattoo (Figure 4).

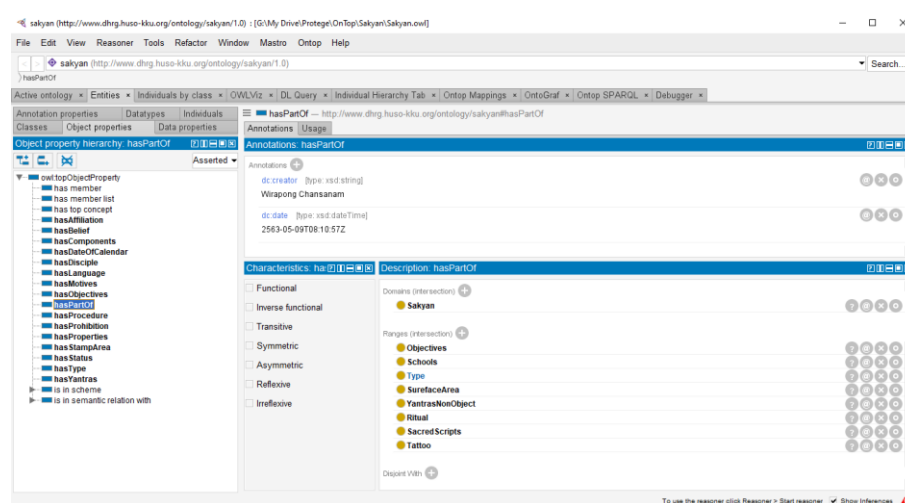


Figure 3. Object properties of Sak Yan ontology.

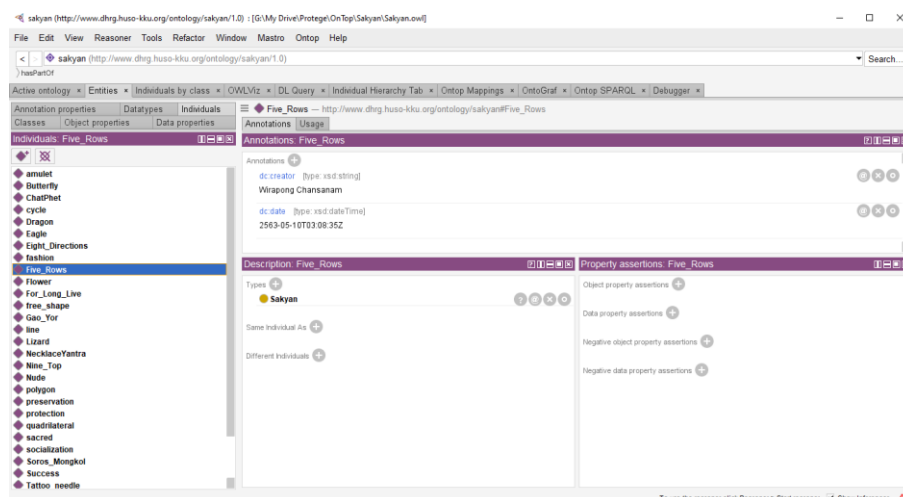


Figure 4. Instances or individuals of Sak Yan ontology.

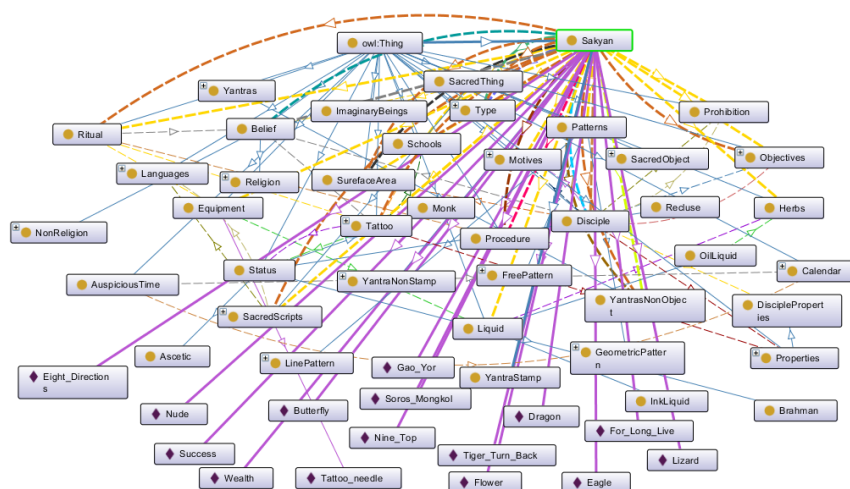


Figure 5. Sak Yan ontology visualization by Onto Graf.

### 3.1.3. Individual or Instance Creation

Individuals act as instances of classes and subclasses. Through individuals (instances), relationships are designed between all entities of the respective ontology. All individuals are an instance of a Sak Yan ontology (Figure 6). Similarly, all individuals are configured for all classes and subclasses. Each class has many memberships that act as their instances. For example, the Sak Yan class includes 14 individuals: Butterflies, Chat-Phet, Dragon, Eagle, Eight\_Directions, Five\_Rows, Flower, Gao\_Yor, Lizard, Neck-laceYantra, Nine\_Top, Nude, Soros\_Mongkol, and Tiger\_Turn\_Back.

|  |                       |
|--|-----------------------|
| Results for P08: Missing annotations.                                | 118 cases   Minor     |
| Results for P11: Missing domain or range in properties.              | 2 cases   Important   |
| Results for P13: Inverse relationships not explicitly declared.      | 16 cases   Minor      |
| Results for P19: Defining multiple domains or ranges in properties.  | 2 cases   Critical    |
| Results for P22: Using different naming conventions in the ontology. | ontology*   Minor     |
| Results for P30: Equivalent classes not explicitly declared.         | 1 case   Important    |
| Results for P41: No license declared.                                | ontology*   Important |

Figure 6. An excerpt from the critical pitfall.



### 3.1.4. Consistency Checking

Protégé offers the reasoner a way to verify the consistency of the created ontology. It can start the reasoning and after a consistency check, it can stop the reasoning.

### 3.2. Evaluation by OOPS!

The analysis results obtained from OOPS! were manually revised in this subsection. The modeling error(s) are categorized into three degrees: critical, important, and minor. These specify the significance of the errors that occur. The first two degrees must be adjusted. Priority is given to the critical degree. The last degree is not necessary to deal with because it is not a problem; however, by doing so, it will improve the ontology's performance. Figure 6 shows the validation results for the SYO. It attains one critical, three important, and three minor pitfalls. First, the critical pitfall is discussed. The reparation recommendations are taken exactly [24].

Figure 7 shows a critical pitfall, that is, P19: defining multiple domains or ranges in the properties. The domain or range (or both) of a property (relationships and attributes) is defined by stating more than one `rdfs:domain` or `rdfs:range` statement. In OWL, multiple `rdfs:domain` or `rdfs:range` axioms are allowed, but they are interpreted as conjunction, being, therefore, equivalent to the construct `owl:intersectionOf`. In fact, the object properties are reusable and shareable among classes within the ontology. Thus, no reparation actions are needed, and this pitfall is continued in SYO.

Results for P19: Defining multiple domains or ranges in properties. 2 cases | Critical

The domain or range (or both) of a property (relationships and attributes) is defined by stating more than one `rdfs:domain` or `rdfs:range` statements. In OWL multiple `rdfs:domain` or `rdfs:range` axioms are allowed, but they are interpreted as conjunction, being, therefore, equivalent to the construct `owl:intersectionOf`. This pitfall is related to the common error that appears when defining domains and ranges described in [7].

- This pitfall appears in the following elements:
  - > <http://www.dh.kku.ac.th/sakyan.owl#hasPartOf>
  - > <http://www.dh.kku.ac.th/sakyan.owl#hasProperties>

**Figure 7.** An excerpt from the critical pitfall.

Figure 8 shows an excerpt of the first important pitfall, that is, P11: Missing domain or range in properties. Two cases were discovered for this pitfall because they represented two object properties without the domain and range. Ontology Development 101 provides guidance regarding the facets of this property. The reparation recommendation by OOPS! is to expose properties with domain and range constraints for a complete meaning. By contrast, scholars [25–27] recommended against the necessity of identifying the domains and ranges of the properties. They do not act as constraints to be checked; instead, they are axioms for reasoning. This may cause two possibilities, that is, unanticipated classification results, where the classes are forced to be subsumed by another class, or unsatisfiable constraints among classes. This condition can be more challenging in significantly large and complicated ontologies. In OWL, the consequence of the range and domain constraints as axioms is a typical problem [25,27]. In SYO, the domain and range of the properties are fixed to avoid the above problems. Thus, two classes of accomplishments were carried out for this pitfall.

Results for P11: Missing domain or range in properties. 2 cases | Important

Object and/or datatype properties without domain or range (or none of them) are included in the ontology.

- This pitfall appears in the following elements:
  - > <http://www.dh.kku.ac.th/sakyan.owl#hasComponents>
  - > <http://www.dh.kku.ac.th/sakyan.owl#hasBelief>
- Tip:** Solving this pitfall may lead to new results for other pitfalls and suggestions. We encourage you to solve all cases when needed and see what else you can get from OOPS!

**Figure 8.** An excerpt from the important pitfall.

Figure 9 shows the second important pitfall, that is, P30: Equivalent classes not explicitly declared. OOPS! discovered one pair of classes that hypothetically had an equal class between them. They were identified because of the doubled classes that exist among them. In Ontology Development 101, the guidance regarding this material falls under the synonym class. Synonyms for the equivalent concept do not represent dissimilar classes.

| Results for P30: Equivalent classes not explicitly declared.   | 1 case   Important |
|--|--------------------|
| <p>This pitfall consists in missing the definition of equivalent classes (<code>owl:equivalentClass</code>) in case of duplicated concepts. When an ontology reuses terms from other ontologies, classes that have the same meaning should be defined as equivalent in order to benefit the interoperability between both ontologies.</p> <ul style="list-style-type: none"> <li>The following classes might be equivalent: <ul style="list-style-type: none"> <li><a href="http://www.dh.kku.ac.th/sakyan.owl#Character">http://www.dh.kku.ac.th/sakyan.owl#Character</a>, <a href="http://www.dh.kku.ac.th/sakyan.owl#Type">http://www.dh.kku.ac.th/sakyan.owl#Type</a></li> </ul> </li> </ul> |                    |

**Figure 9.** Equivalent classes not explicitly declared.

OOPS! provides two reparation recommendations, one for altered namespace and one for equivalent namespace. The SYO uses the namespace because it does not import or integrate with any existing domain ontologies. Thus, the reparation recommendation was to detract one of them and assign its label annotation properties to the remaining class.

However, in the SYO, this pitfall emerged owing to the type of Sak Yan classification found in the real world. For example, the character and numeric groups are an alphabet; thus, Character and Numeric are subclasses of the Alphabet class. Thus, no reparation actions are needed, and this pitfall is continued in SYO.

P41: No license declared is the last important pitfall in SYO, as shown in Figure 10. The pitfall alerts the ontology metadata feature, which does not have any guidance in Ontology Development 101. The reparation recommendation by OOPS! was to contain a declaration containing the license information using any of the following properties: `dc:rights`, `dcterms:rights`, `dcterms:license`, `cc:licenseor`, and `xhv:license`. This action will be discussed further in the ontology reparation section below.

| Results for P41: No license declared.   | ontology*   Important |
|---|-----------------------|
| <p>The ontology metadata omits information about the license that applies to the ontology.</p> <ul style="list-style-type: none"> <li>This pitfall appears in the following elements: <ul style="list-style-type: none"> <li>It has no licence defined</li> </ul> </li> </ul> |                       |

**Figure 10.** No license declared.

The fourth pitfall is P08: Missing annotations. The description of this pitfall was that, in creating an ontology element, human-readable annotations were not attached to it.

The reparation recommendation contains label annotation properties (`rdfs:label`) and description annotation properties (`rdfs:comment`). These are the two most commonly used annotation properties in addition to `owl:versionInfo` [28]. This pitfall will be fixed for further reuse.

P13: Inverse relationships not explicitly declared is the second minor pitfall.

This pitfall occurs when any relationship does not have an inverse relationship (`owl:inverseOf`) defined within the ontology. OOPS! lists all of the object properties in SYO that do not have an inverse relationship (see Figure 5). Ontology Development 101 provides guidance on inverse relationships. Reference [26] specifies that the requirement of the inverse properties is desired for comprehensiveness. This is because all relationships, apart from the symmetric ones, could have an inverse relationship in the standard [24]. The inverse relationships for the object properties will be identified in future tasks.

The final minor pitfall is P22: Using different naming conventions in the ontology. For this pitfall, ontology elements are not named according to the same convention (for example, `thai_month` or use of delimiters such as “-” or “\_”). Because this pitfall is useful to the ontology universally, no particular elements were given. Ontology Development 101 provides guidance on naming conventions. It emphasizes consistency with the chosen



naming conventions. The advantages of consistency assist in evading modeling errors and improving the readability and the simplicity of understanding the ontology. In SYO, the naming convention adherence for the class and individual names are capitalized, whereas the property name has no capitalization. The pitfall occurs in the object property name in which the current researcher fortuitously allocated one of them with a capital letter, that is, IsAuspiciousTime. It likely should be written as isAuspiciousTime. Because the pitfall was quite easy to repair, renaming was executed directly.

### 3.3. Ontology Reparation Activity

The Dublin Core Metadata Initiative (<http://dublincore.org>) is a metadata standard for a wide range of resources, including physical resources such as documents and artwork, and digital resources such as images, videos, and web pages. The annotation property suitable for a license is the rights, which is defined as the information about rights held in and over the resource. It can be used with DC: (<http://purl.org/dc/elements/1.1/>) and dcterms: (<http://purl.org/dc/terms/>) namespaces. Another property of Dublin Core that is related to intellectual property license is a license (a legal document giving official permission to do something with the resource). This property gives rise to creative common metadata (<http://creativecommons.org/ns#>), that is, cc:license. Apart from the Dublin Core vocabulary, XHTML vocabulary also provides license property (<https://www.w3.org/1999/xhtml/vocab>) with the namespace of xhv:.

It is essential for a scholar to define the most suitable license. A study by [29] categorized the most common data license according to its restrictiveness. Among the types of licenses are the public domain, attribution, share-alike, with restrictions, closed, and other. Because the determination of ontology visualization is reusable domain knowledge, the open license is supposed to be the most significant for its rights declaration [29]. It is also similar to the attribution license. The Commons Attribution License (CC BY) is the most popular license used [30,31].

In Protégé, the metadata annotations are under the ontology header view. The interface of the metadata annotations tab is shown in Figure 11, where the ontology license is declared. The establishment for the license declaration of SYO was taken from dcterms:license and given to the CC BY license.

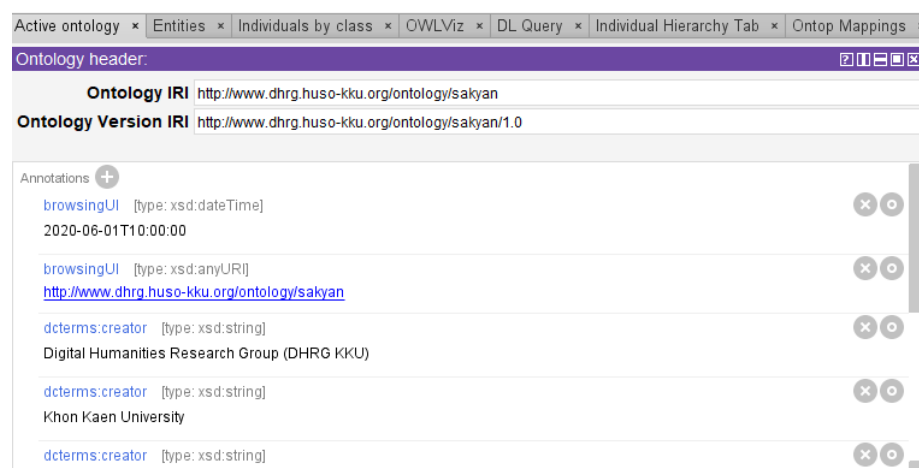






Figure 11. License declaration.

To check whether the fixing activity was executed appropriately, the ontology had to be reexamined. OOPS! also suggests a reexamination because it enables the detection of concealed errors. Figure 12 shows that the SYO is able to repair the pitfall (P41) that is unlisted in the evaluation result. It also shows that another pitfall is being fixed (P11, P19), which is also unlisted as the modeling error in SYO.

|  |  |
|--|--|
| Results for P08: Missing annotations.                                | 118 cases   Minor   |
| Results for P13: Inverse relationships not explicitly declared.      | 16 cases   Minor    |
| Results for P22: Using different naming conventions in the ontology. | ontology*   Minor   |
| Results for P30: Equivalent classes not explicitly declared.         | 1 case   Important  |

**Figure 12.** Reevaluation result after reparation activity.

### 3.4. Evaluation by Experts

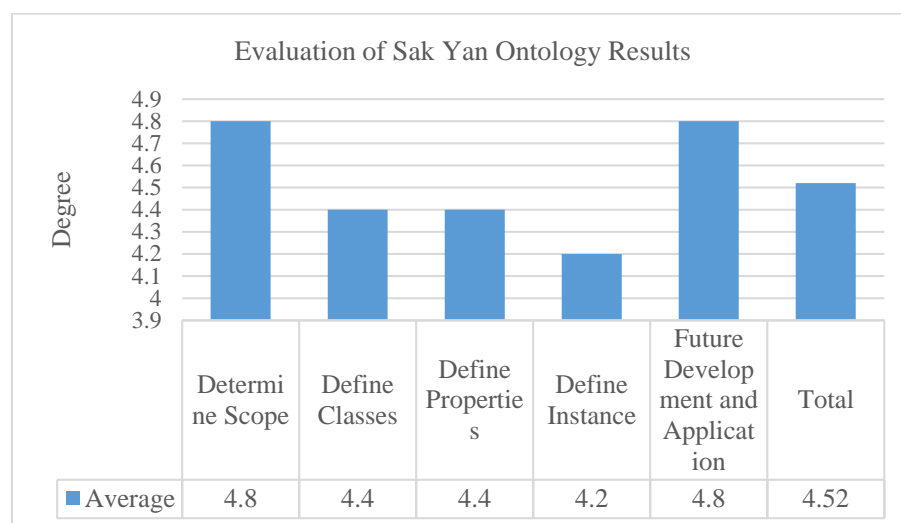
Based on the desired guidelines in the evaluation of the ontology, a generic evaluation framework was developed. In this section, the experts were divided into two parts for evaluation [10]: experts in the yantra domain and knowledge engineers.

A snowball technique was used to seek both domain experts and ontology engineers [32], which first specified one expert and then used these experts to explore another expert until the conditions had been successful. With this technique, we assured no bias from the expert allotment procedure.

The conditions involved two characteristics. The first characteristic was the evaluation of ontology formal structures, for example, its consistency, well-formedness, and completeness of definitions, according to the standards given by Gomez-Perez [33,34]. The second characteristic was the usefulness and adoption of the ontology. This part aims at the statement “how appropriate it is for the task it was generated for” and “how fit it symbolizes the domain of awareness.” The approaches and conditions for the SYO ontology evaluation are as follows:

- Arrange for an inquiry form for ontology evaluation by humans—It is necessary to validate and certify the ontology in an evaluation by humans. We planned an inquiry form by considering three dimensions:
  1. evaluator basic information;
  2. knowledge, structure, and representation in a variety of dimensions, such as scope, concept class, properties, instance, and reusability potential for future development and application;
  3. open-ended questions for recommendation.
- Arrange for evaluation by the evaluator by assigning responsibility to each group.
  1. Knowledge engineers—to verify a comprehensive ontology scheme for computer system agents and ontological mechanisms.
  2. Specialists in the yantra domain—to prove and certify correctness, consistency, properties, and hierarchical relations among concepts in ontology.

The evaluation considered the ontology quality by representation based on semiotic theory [35], taking some metrics into consideration for evaluating the syntactic, semantic, and pragmatic facet of ontology quality. With these conditions and situation, the ontology was evaluated by three experts in the yantra domain and three knowledge engineers (Figure 13).



**Figure 13.** Evaluation of Sak Yan (SYO) ontology results by experts.

#### 4. Results and Discussion

The conclusions attained by SYO were without any critical pitfalls. This showed that the ontology model building assisted the knowledge engineers to model the ontology in a systematic method. Pitfalls can be classified into two categories: continuing and fixed. The first category is usually due to real-world causes. Because an ontology is a representation of the real world, it must replicate this. The pitfalls under this category were P30 and P02. By contrast, the continuing pitfall, P11, was due to an OOPS! Limitation, and the manuals outlined by other scholars. Not all pitfalls in OOPS! are measured as realistic errors. They depend on the scheme decision or requirements. This shows that OOPS! is not the only tool for an ontology evaluation.

P41, P08, and P22 are insidious under the reparation category. The activity started with the easiest pitfall to be fixed, which was P22, followed by the degree of importance. Thus, P41 became the second pitfall to be fixed. The other two continuing pitfalls were taken as future work. Three pitfalls, namely P13, P19, and P22, occurred owing to the inaccuracy made by the current scholars. Ontology Development 101 provided the procedures; however, they were passed over. Here, P41 and P08 occurred because the current scholars did not have any knowledge about them because Ontology Development 101 did not provide any procedures concerning them. Both pitfalls were related to the ontology annotation, which started to gain attention from the community in 2005 [36], whereas Ontology Development 101 was published in 2001.

The OOPS! tool has an important role in confirming that the ontology is free from the common pitfalls by double checking the modeling procedure provided by Ontology Development 101. For example, it supports the latest common modeling errors that are not listed in Ontology Development 101, such as the annotation issue. The main benefit of OOPS! is the repair recommendation. This shows how the ontology component can be repaired to develop the ontology methodological quality. In SYO, the evaluation outcomes from OOPS! enhanced the inferencing, sympathetic, visibility, and metadata aspects. However, OOPS! has a constraint, i.e., it still needs to be revised manually in some pitfalls.

#### 5. Conclusions

This paper presented approaches, ontology modeling construction, and an ontology evaluation by manual and automatic methods, that is, creating an ontology by knowledge engineering, a human ontology evaluation, and OOPS! for validating SYO. The ontology consisted of 150 concepts from a thesaurus and was extended with data particular to cul-

tural heritage resources, with 700 axioms, 90 classes, 16 object properties, 13 data properties, and 31 individuals. Its contributions include mapping the structure of reference to Ontology Development 101 onto the sub-activities in an ontology evaluation and their suitable time for evaluation. It also proposes a validation process by combining manual and automatic methodologies. In addition, it demonstrates how to resolve important pitfalls based on its recommendations. In conclusion, it generates a better quality of Sak Yan ontology. For future studies, apart from the repair activity for the two remaining pitfalls (P08 and P13), SYO will be developed with semantic search and linked with open data to support knowledge sharing and reuse.

This paper presented an information model for supporting the representation of knowledge in cultural heritage conservation processes. At its core, the suggested model has an ontology-based representation of the knowledge collected. While ontologies have only been used to model data and knowledge on very narrow disciplines and cultural heritage aspects, this study presents a broader model that focuses on the entire yantra domain and its development process. Although many class declarations have been incorporated into the model, they are related to a new framework that matches the conservation process logic structure to be easily managed by a conservation professional. The use of ontology and its integration with a knowledge modeling environment enables the structured formalization of direct and indirect knowledge necessary to fully understand cultural heritage to be homogeneous, accessible, and computable.

The proposed ontology, which focuses on the entire validation process, can be configured to serve as a supporting guide for conservation practice information and knowledge management, particularly within the relevant institutions.

From this perspective, the research presented here opens up new possibilities for further research on the use of ontology to support the conservation process of local wisdom, and in particular, the phases of preliminary planning and conservation design.

**Author Contributions:** Conceptualization, W.C., and K.T.; methodology, W.C. and T.S.; validation, K.T.; formal analysis, W.C.; investigation, W.C., K.K., and T.S.; resources, W.C. and T.S.; data curation, K.S.; writing—original draft preparation, W.C.; writing—review and editing, W.C. and K.T.; visualization, T.S.; supervision, K.T.; project administration, W.C.; funding acquisition, K.T. and K.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Digital Humanities Research Group (DHRG), Khon Kaen University, grant number 2563.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are openly available at <https://www.dhrghuso-kku.org/sakyan/index-en.html>.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

## References

1. Chuanchom, W. Belief in Five Columns Tattoo: A Case Study of Tattoo Group of Ajarn Noo Khanpai. Master's Thesis, Arts Program in Sociology, Chulalongkorn University, Bangkok, Thailand, 2010.
2. Acierno, M.; Cursi, S.; Simeone, D.; Fiorani, D. Architectural heritage knowledge modelling: An ontology-based framework for conservation process. *J. Cult. Herit.* **2017**, *24*, 124–133. doi:10.1016/j.culher.2016.09.010
3. Gruber, T. What is an ontology. In *Encyclopedia of Database Systems*; Liu, L., Ozsu, M., Eds.; Springer: New York, NY, USA, 2009.
4. Chandrasekaran, B.; Josephson, J.R. The Ontology of Tasks and Methods. AAAI 1997 Spring Symposium on Ontological Engineering, 24–26 March 1997, Stanford University, Stanford, CA, USA, 1997. Available online: <http://web.cse.ohio-state.edu/~chandrasekaran.1/Ontology-of-Tasks-Methods.PDF> (accessed on 14 May 2020).
5. Milton, S.; Keen, C.; Kurnia, S. "Understanding the Benefits of Ontology Use for Australian Industry: A Conceptual Study", AIS Electronic Library (AISeL). 2010. Available online: <https://aisel.aisnet.org/acis2010/64/> (accessed on 9 March 2020).
6. During, R. *Cultural Heritage Discourses and Europeanisation*; Thesis Wageningen University: Wageningen, The Netherlands, 2010.

7. Ore, C.-E.; Eide, Ø. TEI and cultural heritage ontologies: Exchange of information? *Liter. Linguist. Comput.* **2009**, *24*, 161–172.
8. Signore, O. The Semantic Web and Cultural Heritage: Ontologies and Technologies Help in Accessing Museum Information, In: Information Technology for the Virtual Museum. 6–7 December 2006. Available online: <https://pdfs.semanticscholar.org/ab4f/0ee826bbea097f4d7aa4d5244b67c0caea6.pdf> (accessed on 14 May 2020).
9. Koutsomitropoulos, D.; Hyvönen, E.; Papatheodorou, T. Semantic Web and reasoning for cultural heritage and digital libraries. *Semant. Web* **2012**, *3*, doi:10.3233/sw-2012-0050.
10. Chansanam, W.; Tuamsuk, K.; Kwiecien, K.; Ruangrajitpakorn, T.; Supnithi, T. Development of the belief culture ontology and its application: Case study of the Greater Mekong subregion. In *Lecture Notes in Computer Science, Proceedings of the Joint International Semantic Technology Conference, Chiang Mai, Thailand, 9–11 November 2014*; Springer: Berlin/Heidelberg, Germany, 2014; Volume 8943, pp. 297–310, doi:10.1007/978-3-319-15615-6\_22.
11. Ramli, F.; Noah, S.A.M. Building an Event Ontology for Historical Domain to Support Semantic Document Retrieval. *Int. J. Adv. Sci. Eng. Inf. Technol.* **2016**, *6*, 1154.
12. Aloui, N.; Gargouri, F. OARLCAM: An Ontology-based Approach for Reusing and Learning through Context-aware Annotations Memory. *Int. J. Emerg. Technol. Learn.* **2014**, *9*, 4–16.
13. Bouchra, B.; Mohamed, B. Ontology and Rule-Based Recommender System for E-learning Applications. *Int. J. Emerg. Technol. Learn.* **2019**, *14*, 4–13.
14. Madhusudhana, K. The Cognitive Dimension and Course Content Modeling: An Ontological Approach. *Int. J. Emerg. Technol. Learn.* **2017**, *12*, 181–188.
15. Chansanam, W.; Tuamsuk, K. Development of an imaginary beings Ontology. In *Lecture Notes in Computer Science, Proceedings of the Digital Libraries: Knowledge, Information, and Data in an Open Access Society, Tsukuba, Japan, 7–9 December 2016*; Morishima A., Rauber A., Liew C., Eds; Springer: Cham, Switzerland, 2016; Volume 10075, pp. 1–12, doi:10.1007/978-3-319-49304-6\_28.
16. Hozo-Ontology Editor. 2011. Available online: [http://www.ei.sanken.osaka-u.ac.jp/hozo/eng/HozoManual\\_en\\_20140317.pdf](http://www.ei.sanken.osaka-u.ac.jp/hozo/eng/HozoManual_en_20140317.pdf) (accessed on 14 February 2020).
17. Suárez-Figueroa, M.C.; de Guadalupe A.C.; Asunción, G.-P. Lights and shadows in creating a glossary about ontology engineering. *Terminol. Int. J. Theor. Appl. Issues Spec. Commun.* **2013**, *19*, 202–236.
18. Noy, N.F.; McGuinness, D.L. Ontology Development 101: A Guide to Creating Your First Ontology. 2001. Available online: [https://protege.stanford.edu/publications/ontology\\_development/ontology101.pdf](https://protege.stanford.edu/publications/ontology_development/ontology101.pdf) (accessed on 14 January 2020).
19. Poveda-Villalón, M.; Gómez-Pérez, A.; Suárez-Figueroa, M. OOPS! (Ontology Pitfall Scanner!). *Int. J. Semant. Web Inf. Syst.* **2014**, *10*, 7–34, doi:10.4018/ijswis.2014040102.
20. Ritika, B.; Bansal, C. An Approach for Semantic Information Retrieval from Ontology in Computer Science Domain. *Int. J. Eng. Adv. Technol. (IJEAT)* **2014**, *4*, 58–65.
21. Biomedical Informatics Research (BMIR). A Free, Open-Source Ontology Editor and Framework for Building Intelligent Systems. Available online: <http://protege.stanford.edu/> (accessed on 16 October 2020).
22. Manola, F.; Miller, E. RDF 1.1 Primer. Available online: <https://www.w3.org/TR/rdf-primer/> (accessed on 16 October 2020).
23. Chansanam, W.; Tuamsuk, K. Development of an imaginary beings knowledge structure. In *Lecture Notes in Computer Science, Proceedings of the International Conference on Asian Digital Libraries, Seoul, Korea, 9–12 December 2015*; Springer: Cham, Switzerland, 2015; Volume 9469, pp. 291–293.
24. Villalón, M.P. Ontology Evaluation: A Pitfall-Based Approach to Ontology Diagnosis. Ph.D. Thesis, Department of Artificial Intelligence, Polytechnic University of Madrid, Madrid, Spain, 2016.
25. Rector, A.; Drummond, N.; Horridge, M.; Rogers, J.; Knublauch, H.; Stevens, R.; Wang, H.; Wroe, C. OWL pizzas: Practical experience of teaching OWL-DL: Common errors and common patterns. In *Lecture Notes in Computer Science, Proceedings of the Engineering Knowledge in the Age of the Semantic Web, Whittlebury Hall, UK, 5–8 October 2004*; Motta E., Shadbolt N.R., Stutt A., Gibbins N., Eds.; Springer: Berlin/Heidelberg, Germany, 2004; Volume 3257, pp. 63–81.
26. Horridge, M. *A Practical Guide to Building OWL Ontologies using Protege 4 and CO-ODE Tools Edition 1.3*; The University of Manchester: Manchester, England, 2011.
27. Allemang, D.; Hendler, J. *Semantic Web for the Working Ontologist Effective Modeling in RDFS and OWL*, 2nd ed.; Morgan Kaufmann: San Francisco, CA, USA, 2011.
28. Hebel, J.; Fisher, M.; Perez-lopez, A.; Blace, R. *Semantic Web Programming*, Indianapolis; Wiley Publishing: Hoboken, NJ, USA, 2009.
29. Rodríguez-Doncel, V.; Gómez-Pérez, A.; Mihindukulasooriya, N. Rights declaration in Linked Data. In Proceedings of the Fourth International Workshop on Consuming Linked Data (COLD), Sydney, Australia, 22 October 2013; EUR-WS.org: Aachen, Germany, 2013.
30. Australian Research Data Commons (ARDC). *Research Data Rights Management Guide*; ARDC: Caulfield East, VIC, Australia, 2019.
31. Poblet, M.; Aryani, A.; Manghi, P.; Unsworth, K.; Wang, J.; Hausstein, B.; Dallmeier-Tiessen, S.; Klas, C.P.; Casanovas, P.; Doncel, V.R. Assigning creative commons licenses to research metadata: Issues and cases. In *AI Approaches to the Complexity of Legal Systems*; Springer: Cham, Switzerland, 2015; pp. 245–256.
32. Etikan, I. Comparison of Snowball Sampling and Sequential Sampling Technique. *Biom. Biostat. Int. J.* **2016**, *3*, 55. doi:10.15406/bbij.2016.03.00055.

33. Gómez-Pérez, A.; Fernández-López, M.; Corcho, O. *Ontological Engineering: With Examples from the Areas of Knowledge Management, e-Commerce and the Semantic Web*, 2nd ed.; Springer: London, UK, 2004.
34. Gómez-pérez, A.; Rojas-amaya, M.D. Ontological Reengineering for Reuse. In *Lecture Notes in Computer Science, Proceedings of the 11th European Workshop on Knowledge Acquisition, Modeling and Management, Dagstuhl Castle, Germany, 26–29 May 1999*; Springer: Berlin/Heidelberg, Germany, 1999; pp. 139–156.
35. Stamper, R.; Liu, K.; Hafkamp, M.; Ades, Y. Understanding the roles of signs and norms in organizations—A semiotic approach to information systems design. *Behav. Inf. Technol.* **2000**, *19*, 15–27.
36. Hartmann, J.; Palma, R.; Sure, Y.; Suarez-Figueroa, M.C.; Haase, P.; Gómez-Pérez, A.; Studer, R. Ontology metadata vocabulary and applications. In *Lecture Notes in Computer Science, Proceedings of the OTM Confederated International Conferences “On the Move to Meaningful Internet Systems, Agia Napa, Cyprus, 31 October–4 November 2005*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 906–915.