IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received November 28, 2020, accepted January 20, 2021, date of publication January 25, 2021, date of current version February 1, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3054412

An Improved Methodology for Collaborative Construction of Reusable, Localized, and Shareable Ontology

ABDUL SATTAR[1,3], MOHAMMAD NAZIR AHMAD[1], ELY SALWANA MAT SURIN[1], AND AHMAD KAMIL MAHMOOD[2]

[1]Institute IR4.0, Universiti Kebangsaan Malaysia, Bangi 43600, Malaysia
[2]High Performance Cloud Computing Centre, Department of CS, Universiti Teknologi Petronas, Perak 32610, Malaysia
[3]Department of Computer Science, Khwaja Fareed University of Engineering & IT, Rahim Yar Khan 64200, Pakistan

Corresponding author: Abdul Sattar (sattar56_lakha@yahoo.com)

This work was supported in part by Trans-disciplinary Research Grant Scheme (TRGS), Ministry of Higher Education (MOHE); and in part by the Universiti Kebangsaan Malaysia (UKM), Vot. No: TRGS/1/2020/UKM/02/6/2.

ABSTRACT Interlocking Institutional Worlds (IWs) is a concept explaining the need to interoperate between institutions (or players), to solve problems of common interest in a given domain. Managing knowledge in the IWs domain is complex; however, promoting knowledge sharing based on standards and common terms agreeable to all players is essential and is something that must be established. In this context, ontologies, as a conceptual tool and a key component of knowledge-based systems, have been used by organizations for effective knowledge management, better decision-making, and interoperability among diverse institutions of an IWs domain. The development of ontology involves structural and logical complexity, and requires a well-designed, mature, and widely accepted methodology, to ensure its reliability. Many methodologies for ontology development have been proposed by several researchers; however, most of the developed methodologies have not included several important phases. Furthermore, several methodologies have not provided the complete details of the techniques and activities involved in the ontology construction process. Fewer details make it difficult to follow a methodology for designing ontologies. This study aims to compare existing methodologies based on sixteen important criteria and proposes an improved methodology for ontology development for IWs domains. The proposed methodology has included several important phases such as the Estimation of Human Resources, Re-engineering and Re-using of Resources, Collaborative Ontology Construction, the Conceptualization of both Endurants and Perdurants, Localization of an Ontology, Ontology Integration and Merging, Support for Interoperability, Versioning of an Ontology, and Ontology Population. The proposed methodology has been applied to design an OntoWM domain ontology for the Waste Management (WM) domain. The evaluation of the proposed methodology has been made by utilizing the designed OntoWM domain ontology in an ontology-based web application called SmartBinAnalytics. SmartBinAnalytics has been built using OntoWM, Java Server Pages (JSP), Jena Application Programming Interface (API), and MySQL. The proposed methodology can be used for the construction of domain ontologies in any IWs domain.

INDEX TERMS Domain ontology, survey on methodologies, proposed methodology for ontology development.

I. INTRODUCTION

Semantic web, as a machine-readable web, needs ontologies as its primary and most important component. An ontology describes conceptions and their associations in the domain of discourse [1]. Knowledge-based applications require ontologies, and they serve as formal models and machine-understandable descriptions of a domain. Ontologies assist in sharing a domain's knowledge to other relevant or irrelevant domains. For example, sharing of knowledge can improve individual employee performance as well as team performance [2]. Due to the distributed nature

The associate editor coordinating the review of this manuscript and approving it for publication was Jenny Mahoney.

of organizational knowledge, knowledge-based applications, with the assistance of ontologies, must be able to integrate the knowledge of heterogeneous sources and present an overview of the knowledge available in an organization [3]. In this context, finding a suitable ontology for a domain is one of the bigger research challenges [4].

Ontologies, as a conceptual tool and key component of knowledge-based systems, have been used by organizations for effective knowledge management of the domain of discourse. In the past decade, the development of ontologies [5]–[11] has had a significant stimulus. The industry has shown an interest in developing novel applications in semantic technology, which has resulted in the wide adoption of ontology-based solutions by the government, academia, and commercial industry. Most of the IS are based on databases; therefore, comprehensive and applicable approaches are required to convert existing databases into ontologies [12]. In this context, ontology-based solutions with improved knowledge management assist in better decision-making. Furthermore, an ontological approach makes it easy to share the conceptualization of a domain [1], and this sharing offers more opportunities for stakeholders to solve their real-time problems.

IWs are about inter-agency collaboration, involving multidisciplinary teams to solve a common problem at the domain level [13], [14]. The theoretical foundations of the concepts of IWs are strongly anchored in Institutional Facts and Speech Act theories. Examples of IW domains include WM, the semantic web, the Olympic Games, postal codes, Maps preparation, Geographic information governance, and Flood Management. The integration and merging of ontologies focus on explicit specifications. For supporting integration, merging, and interoperability, conceptualizations of ontologies should have systems of institutional facts. Interlocking systems of institutional facts assist in merging conceptualizations of ontologies [13].

The elements of social reality are known as institutional facts. For example, in a context X, a physical reality Y will be titled as an institutional fact Z. Examples of institutional facts include marital status, citizenship, and the name of an employee or a person. A formal statement created by the officials of an organization is called a speech act, and these speech acts create and destroy the institutional facts in that organization. A nomination speech act is performed in a particular locality by guardians or parents who register the birth of a child in a government office or register. Speech acts can also be utilized in creating institutional facts with similar or dis-similar semantics, based on the context or locality of the occurrence of a particular event. For example, any positive physical posture as a speech act may or may not be positive in another area or jurisdiction. Immaterial reality such as the financial crisis of the subprime mortgages of 2008 had less of an impact on the world as the undeniable physical tsunami of 2004. Furthermore, it often happens that two or more parties participate in a speech act [13], [14].

The WM domain is a good example of an institutional world formed by the interlocking of many institutional worlds, because it is vast and complex. This domain involves the strong coordination of several autonomous institutions. Different small Information Systems (IS) perform different services in the WM domain. Some of the IS are operated by the WM domain itself, such as collection, transportation, treatment, processing, local state-level regulation, and monitoring. Some are operated by other organizations, such as storage and disposal activities, which are performed by affiliated third party contractors who assist the WM domain in managing waste. A Laboratory Information System is an example of a system that does the necessary laboratory experiments to assess the possible harms to the human body caused by waste in a particular locality based on environmental factors. This system integrates a wide variety of testing and administrative activities performed by numerous departments in the laboratory. Furthermore, numerous activities are performed by other agencies, from the regulatory bodies to the outsourced contractors. Therefore, the WM domain is capable of having a mutual agreement that its institutional worlds can interlock to define the entire integrated WM system. In this context, the institutional worlds such as the sets of types of speech acts and instances can interlock other institutional worlds if they are connected in the manner indicated above [13], [14].

A single medium-sized organization can have more than 1000 IS. Most organizations interact with hundreds of others in the supply chain and through other relationships. For example, the UK National Plan for National IT Health Services mentioned in the introduction, attempts to integrate hundreds of thousands of IS. The discussion above on interlocking institutional worlds shows that almost all applications involve multiple institutional worlds belonging to different institutions, that interlock in complex ways. For now, the construction of a domain ontology for such extensive and complex IW domains (for example, WM, Semantic Web, Olympic Games, Postal Codes, Preparation of Maps, and Governance of Geographic Information and Flood Management) is also difficult, laborious, and time-consuming.

The development of an ontology is a complex, tedious, and time-consuming task, and this task requires a well-designed methodology. The methodology for ontology development, a guideline for ontology developers, discusses the processes and methods for ontology development [15]. Several methodologies have been proposed by researchers in the past for designing ontologies [16], [17], [26]–[31], [18]–[25]. Most of the studied methodologies for ontology development provide domain analysis, conceptualization, implementation, evaluation, instantiation, and also an example of a domain ontology (as shown in Table 1); although most of them have not provided all the details of the techniques and activities involved [32]. With fewer details, it is difficult to follow a methodology for designing ontologies for a specific domain. Some exceptions have provided details, in particular, [17]–[20]. Additionally, most of the methodologies studied (as shown in Table 1) do not offer support for maintenance, documentation, construction of collaborative ontologies, support for reuse, support for integration, support for interoperability, ontology localization, and human resources estimation (for example, ontology experts). The methodology proposed in a previous research [16], was an initial attempt to design a methodology for designing a domain ontology for the WM domain. Like other IWs domains, WM is a complex IWs domain, and as such there are several challenges in designing a domain ontology for it [33]. In this study, an ontology development methodology rooted in Design Science Research Methodology (DSRM) has been proposed.

The remainder of this article is organized as follows. Section II summarizes the related work. Section III presents the proposed methodology for ontology development. Section IV describes the utilization of the designed OntoWM domain ontology using SmartBinAnalytics. Section V discusses the results of this study. Finally, Section VI concludes the article.

II. RELATED WORK

A methodology for ontology development, a guideline for ontology developers, discusses processes and methods for ontology development [15]. Different ontology development methodologies have been defined in research [16], [18], [21]–[24], [34], [35]. The methodology proposed in the previous work was an initial attempt to design a methodology for designing a domain ontology for the WM domain [16]. In this context, different researchers design their domain ontologies in different ways and as such there is no single viable method for doing so, as discussed by Brusa *et al.* [25].

Additionally, different methodologies focus on the development of ontologies differently, and include distinct aspects of ontology development. For example, a few focus on the scope of an ontology and domain analysis, and so there is no design phase and no details about activities they perform during ontology development. This study made a comparison of several methodologies, along with the identification of a criterion, based on which it was necessary to analyze and compare the methodologies. Selected evaluation criteria were defined by observing needs and trends that have evolved over the last decade, which cover sixteen different aspects of the methodology for developing domain ontologies. This evaluation criteria will assist the reader in understanding the different ontology development methodologies, and will also assist in selecting a suitable methodology for designing their domain ontologies of the domain of discourse.

This study has defined below, the criteria used to compare methodologies for ontology development. The five facets of the criteria are the coarse-grained level of a methodology, namely localization of an ontology, support for reusability, support for integration, support for interoperability, methodology rooted in established methodologies, and estimation of human resources. The nine facets of the criteria, namely domain analysis, conceptualization, level of detail, collaborative construction, implementation, evaluation, instantiation, maintenance, and documentation discuss the technical fine-grained level of a methodology and assist the reader in understanding a particular methodology for ontology development.

One of the important facets from the aforementioned ones is Domain analysis, which requires numerous resources and lacks domain-specific recommendations. The situation becomes more complex when an ontology integrates knowledge from different fields. The necessary training seminars are important for understanding the terminology of the core of ontologies for team members who are unfamiliar or have a controversial understanding. Another important aspect is the availability of details of techniques and methods applied in designing ontologies. If the researchers of a proposed methodology are unable to provide the details of the techniques and methods, it is very difficult for others to follow the same methodology for designing ontologies. Furthermore, it is not possible to make physically available, all the stakeholders in the ontology design process. In this regard, collaborative construction involves several stakeholders of the ontology development group working from the same location or different locations, on the same ontology simultaneously, and without affecting the overall efficiency of the ontology development task.

Furthermore, implementation needs a semantic representation language for the implementation of a conceptual ontology model to ontology. In this context, manual construction from a formal conceptual model to ontology is laborious and ontology developers do not prefer it. However, the construction of a domain ontology from a formal model requires paying attention to semantic differences between the formal model and the representation language. Additionally, the evaluation aspect requires the evaluator to have a better understanding of the ontology, and a proper understanding of the ontology is a key success factor of the ontology evaluation process. The population of an ontology can be processed easily as compared to the processing of poorly structured sources such as documents, relational tables, and Extensible Markup Language (XML) data. Ontologies are constantly confronted with the problem of evolution. Due to the complexity of the changes to be made, a maintenance process, manual or semi-automatic or automatic, is increasingly necessary to facilitate this task and ensure its reliability. In this context, every designed ontology needs to be well-documented. Documentation assists in the understanding of the ontology, its use, reuse, and future revisions. Another aspect regarding ontology development is the localization of the developed ontology. Adaption of the ontology into a natural language is called the localization of ontology. Finally, ontology development is a tedious and time-consuming task. It is usually costly to reuse application dependent ontologies. In this context, the utilization of upper-level ontologies assists in reducing reusability costs, by providing a common conceptual structure in domain ontologies developed using the same upper-level ontology.

TABLE 1. Comparison of methodologies for creating domain ontologies from 2015-2020[32].

No.	Methodology	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16
1	OntoDI: The Methodology for Ontology Development on Data Integration [24]	Y	Y	4	N	Y	Y	Y	N	Y	N	N	Y	N	N	Y	N
2	An Agile Methodology for Ontology Development [21]	Y	Y	4	N	Y	Y	Y	Y	Y	N	N	Y	N	N	Y	N
3	A Domain Ontology for Software Requirements and Change Management in a Global Software Development Environment [22]	Y	Y	4	N	Y	Y	Y	N	N	N	N	N	N	N	Y	N
4	Design and Implementation of E-Campus Ontology with a Hybrid Software Engineering Methodology [23]	Y	Y	3	N	Y	N	Y	N	N	N	N	N	Y	N	Y	N
5	An Ontology for the Waste Management Domain [16]	Y	Y	4	N	N	Y	Y	N	N	N	Y	N	N	N	Y	Y
6	Towards a Software Centric Approach for Ontology Development: Novel Methodology and its Application [18]	Y	Y	5	N	N	Y	Y	N	N	N	N	N	N	N	Y	N
7	Design and Development of a Biocultural Ontology for Personalized Diabetes Self-Management of American Indians [19]	Y	Y	5	N	Y	Y	Y	N	N	N	N	N	N	N	Y	N
8	Towards a Methodology for Reusable Ontology Engineering: Application to the Process Engineering Domain [26]	Y	Y	3	N	Y	Y	N	N	N	N	Y	Y	N	N	Y	N
9	A Methodology for a Criminal Law and Procedure Ontology for Legal Question Answering [17]	Y	Y	5	N	Y	Y	Y	N	N	N	N	N	N	N	Y	N
10	Fine Construction of HIV Protein Ontology [27]	Y	Y	3	N	Y	Y	Y	N	N	N	N	N	N	N	Y	N
11	The Methodology for Ontology Development in Lesson Plan Domain [28]	Y	Y	4	N	Y	Y	Y	Y	N	N	Y	Y	N	N	Y	N
12	A Lightweight Methodology for Rapid Ontology engineering - UPON Lite [29]	Y	Y	4	Y	Y	Y	Y	Y	Y	N	Y	Y	N	Y	Y	N
13	The NeOn Methodology for Ontology engineering [35]	Y	Y	5	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
14	YAMO: Yet Another Methodology for large-scale faceted Ontology construction [30]	Y	Y	4	N	N	Y	Y	N	N	N	N	N	N	N	Y	N
15	Methodologies to build ontologies for terminological purposes [31]	Y	Y	2	N	Y	N	N	N	N	Y	N	N	N	N	N	N

Table 1 presents a detailed and complete comparison of the methodologies based on the established criteria. This study will compare the selected methodologies for ontology development based on the following criteria (C1-C16), which are crucial when it comes to the creation of domain ontologies (a few are taken from Simperl *et al.* [36]).

1) *Criteria C1: Domai*n *Analysis (specification, knowledg*e *acquisitio*n). Domain analysis requires numerous resources and lacks domain-specific recommendations. The situation becomes more complex when an ontology integrates knowledge from different fields. The necessary training seminars are important for understanding the terminology of the core of the ontologies for team members who are unfamiliar or have a controversial understanding.
2) *C2: Conceptualization*. Does the proposed methodology support or include the conceptualization activity, as described in [37]?
3) *C3: Level of Detail*. Does the methodology provide details about techniques and methods for various activities? In this case, the level of detail criteria is evaluated on a scale of 1 to 5, from very few details to detailed.
4) *C4: Collaborative Construction.* Collaborative construction involves several stakeholders of the ontology development group to work from the same location or different locations, on the same ontology, simultaneously, and without affecting the overall efficiency of the ontology development task.
5) *C5: Implementation (model to language)*. Implementation needs a representation language for the implementation of a conceptual ontology model to ontology. In this context, the manual construction of an ontology from a formal conceptual model is laborious, and ontology developers do not prefer it. However, the construction of a domain ontology from a formal model requires paying attention to semantic differences between the formal model and the representation language.
6) *C6: Evaluation (Refinement).* The evaluation aspect requires the evaluator to have a better understanding of the ontology, and a proper understanding of the ontology is a key success factor of the ontology evaluation process.
7) *C7: Instantiation*. Does the methodology apply any technique and method for instantiation? The population of an ontology can be processed easily as compared to the processing of poorly structured sources, such as documents, relational tables, and XML data.
8) *C8: Maintenance*. Ontologies are constantly confronted with the problem of evolution. Due to the complexity of the changes to be made, a maintenance process, manual or semi-automatic or automatic, has become increasingly necessary to facilitate this task and to ensure its reliability.
9) *C9: Documentation*. Does the methodology provide detailed documentation of the methods and activities for ontology construction? Documentation assists in the understanding of the ontology, its use, reuse, and future revisions.
10) *C10: Ontology Localization*. Does the methodology support localization of the ontology? Adaption of the ontology into a natural language is called the localization of the ontology.
11) *C11: Support for Reusability*. Ontology development is a tedious and time-consuming task. It is usually costly to reuse application dependent ontologies. In this context, the utilization of upper-level ontologies assists in reducing reusability costs by providing a common conceptual structure in domain ontologies, developed using the same upper-level ontology.
12) *C12: Support for integration and merging.* The integration of an ontology is dependent on the degree of overlapping between two integrating ontologies with the same representing language. The quality of integration

is measured in terms of accuracy (correct mapping rate) and recall (discovered mapping rate).

13) *C13: Support for interoperability.* Interoperability between systems requires the same supporting structure (same high-level concepts). The resulting ontologies will hence have a common global conceptual structure, and it will assist them in sharing their concepts.
14) *C14: Estimation of human resources*. Team needs (people with required skills, experience, and team size): ontology engineers, domain experts, a technical writer for documentation, software engineers.
15) *C15: Sample Application*. Is any sample ontology designed for the planned project?
16) *C16: Methodology rooted in established methodologies*. Is the developed methodology rooted in any well-established methodology like Design Science Research (DSR)?

Most of the methodologies studied provide domain analysis, conceptualization, implementation, evaluation, instantiation, and provide an example of a domain ontology; although most of them have not provided all the details of the techniques and activities involved. Fewer details make it difficult to follow a methodology for designing ontologies for a specific domain. Some exceptions have provided details, in particular, [17]–[20]. Additionally, most of the methodologies studied (compared to Table 1) do not offer support for maintenance, documentation, construction of collaborative ontologies, support for reuse, support for integration, support for interoperability, ontology localization, and human resources estimation (for example, ontology experts).

Ontologies require change over time; support for modification and maintenance is mandatory for an ontology. As such, maintenance and support should be factored in for a domain ontology. A few studied methodologies provide maintenance, collaborative construction of ontologies, documentation, ontology localization, support for reuse, support for integration, support for interoperability, and estimation of human resources (for example ontology experts). Maintenance is discussed by [20], [21], [28], and [29]. Likewise, a few researchers provide documentation; for example, documentation is provided by [20], [21], [24], [29], and [38]. Documentation assists in the understanding of an ontology.

The development of an ontology is an evolutionary process and the availability (in the same place) of domain experts, ontology engineers, technical writers, and other human resources is not always possible. The collaborative construction of ontologies can therefore make valuable human resources available worldwide. In this regard, support for the collaborative construction of ontologies can assist in the correct execution of the ontology construction plan [39]. Suárez-Figueroa *et al.* [20], and De Nicola *et al.* [29] provide collaborative ontology development.

Additionally, for an extensive and complex domain, a single heavy ontology cannot be created. There would be multiple subdomains, and each subdomain would need an ontology. In this context, the interoperability of IS of sub-domains is needed. Interoperability between systems requires the same support structure (same high-level concepts). The resulting ontologies will hence have a common global conceptual structure, and it will assist them in sharing their concepts. The methodologies proposed by Suárez-Figueroa *et al.* [20], and Jacksi [23] support interoperability. Similarly, ontology integration of sub-domain ontologies is needed. This can be realized in the integration of the ontologies discussed by [20], [21], [24], [28], and [29].

Creating an ontology is a tedious and boring activity, and reusability allows one to reuse the existing ontology and add additional components to the target ontology. A few researchers including Ahmad *et al.* [16], Saad and Shaharin [28], and Nicola and M. Missikoff [29] provide a mechanism to reuse an ontology. Similarly, it is important to translate an ontology into different natural languages, as it is not possible to conceive a separate ontology for each natural language (English, Malay, Urdu, Hindi, German, French, and so on). Only two of the studied methodologies, by Suárez-Figueroa *et al.* [20], and Zambrana [31], provide support for the localization of the ontology. Additionally, estimating the costs of the ontology building process requires estimating the required human resources, such as domain experts, ontology engineers, and API developers. Only a few researchers discuss little on the estimation of human resources [18], [20], and [29]. Finally, most of the methodologies examined in the study are not based on any well-established methodology. There are some exceptions, mainly Ahmad *et al.* [16], who rooted their methodology in DSR.

In this article, an ontology development methodology rooted in DSR has been proposed. The strength of this methodology lies in its ability to be customized to fit several factors, including ontology complexity, a domain of interest, and ontology size. The proposed methodology will assist in the estimation of human resources, re-engineering and re-using of resources, collaborative ontology construction, ontology construction of both endurants and perdurants, localization of an ontology, ontology integration and merging, support for interoperability, versioning of an ontology, and ontology population. Moreover, it showed that the utilization of DSR methodology simplifies the implementation of the ontology development activities. The proposed methodology was applied to design a domain ontology (OntoWM) for the WM domain. The experience from building OntoWM ontology indicated its applicability and a high degree of acceptance by ontology engineers.

The methodology proposed in a previous work by Ahmad *et al.* [16] was an initial attempt to design a methodology for designing a domain ontology for the WM domain. The improved methodology has included several important missing phases, such as the estimation of human resources, re-engineering and re-using of resources, collaborative ontology construction, the conceptualization of both endurants and perdurants, localization of an ontology, ontology integration and merging, support for interoperability, versioning of an ontology, and ontology population.

III. METHODOLOGY FOR DEVELOPING AN IWs DOMAIN ONTOLOGY

The proposed methodology aims to adapt the best practices used in the development of ontologies. Under the design and development activity, the phases are divided into three stages. These include Stage 1 (Pre-Conceptualization), including Definition, Gantt chart, Tools and Techniques, Estimation of Human Resources, Ontological and non-ontological resources, Re-engineering and Reusing, Support for Interoperability, Domain Analysis, Knowledge Acquisition, and Validation. Stage 2 (Conceptualization) has the Conceptualization of both Endurants and Perdurants. Stage 3 (Post-Conceptualization) is the last stage, and is divided into three sub-stages, including Demonstration activity (Stage 3 Part A), which consists of Collaborative Construction, Localization, Implementation, Instantiation, Verification, Versioning, and Integration and Merging; Evaluation activity (Stage 3 Part B) which involves Evaluation and Maintenance; and Communication (Stage 3 Part C), which involves Documentation (as shown in Table 2).

Returning to Table 2 , in general, the proposed methodology is composed of two major parts, which are a blend of Design Science Research Methodology (DSRM) activities and the phases of ontology development. The phases are rooted in the DSRM of Peffers [40] because through the lens of DSR, OntoWM is an ontology that is a designed artifact, as explained by Ahmad *et al.* [41]. The goal of design science research is the utilization of the designed artifact [42]. To agree with Geerts [43], the arrows on the left side of Table 2 emphasize the importance of useful iterations as part of DSRM. They demonstrate that activities, including evaluation and communication, often result in revising an artifact's objectives and design. Iteration is ingrained in design science research and has been illustrated by Hevner *et al.* [42] through their build-and-evaluate loop. Here, evaluation provides feedback information regarding the designed artifact, and offers a better understanding of the problem, which leads to a re-iteration of the design process.

Peffers *et al.* [40] designed a research methodology which consists of six activities, which include i) problem identification and motivation, ii) definition of objectives of a solution, iii) design and development, iv) demonstration, v) evaluation, and vi) communication. In the first activity, a specific research problem should be defined, and the value of a solution should be justified. Therefore, the research must demonstrate knowledge of the current state of the art, and also the relevance of the identified problem. The second activity infers the objectives of a solution from the problem definition and provides knowledge of what is possible and feasible. In the third activity, the artifact is designed. Such an artifact can potentially include concept-constructs, models, methods, instantiations, or new properties of technical, social, and/or information resources. The fourth activity of the methodology demonstrates the utilization of artifacts in solving the problem. This could involve its utilization in experimentation, simulation, case study development, proofing, or in other appropriate activities. The fifth activity, namely evaluation, observes and measures how well an artifact supports a solution to a problem. This activity involves comparing the objectives of a solution to the results observed through the utilization of the artifact in the methodological demonstration. It requires knowledge of relevant metrics, and analysis techniques. The last activity, communication, involves consulting the dissemination of new knowledge obtained by the research, such as dissertations or journal articles.

To be specific, the first column in Table 2 shows the DSRM activity that makes up the DSRM, as a nominal sequence. The second column further describes each activity in detail, in the sense of "What to do?". The third column links the respective activities, tools, or techniques to be adopted in creating the artifact, to solve the problem at hand. This column presents varied activities, tools, or techniques, depending on the ontology context being developed by the researchers. For instance, with regards to the third, fourth, and fifth activities, namely design and development, demonstration, and evaluation, these could involve the utilization of varied methods or techniques, which include quantitative and qualitative methods, prototyping, experimentation, simulation, case studies, proof, or other appropriate activities.

Finally, this study has used this analysis to improve the previously proposed methodology [16]. This study has proposed to perform the following activities:

A. STAGE 1 - PRE-CONCEPTUALIZATION

1) DEFINITION

In this phase, an Ontology Requirements Specification Document (ORSD) will be prepared [44]. This document will answer on why an ontology is being built, who the intended users are, and what functional or non-functional requirements will be fulfilled by this ontology [45].

The general aspects of ontologies that are not related to the contents of ontologies are called non-functional requirements. For example, tools utilized to design ontologies (for example, WebProtege), the semantic representation languages, and the naming conventions that will be utilized for creating the target ontology. The content-centric requirements of ontologies are called functional requirements.

One of the key techniques, in this case, is Competency Questions (CQs), which the developed ontology will be able to answer. Answering these questions will produce an ORSD, which will serve as a conceptual bridge and an agreement between ontology owners, ontology engineers, users, and domain experts, in terms of what requirements the ontology should meet. A suitable set of CQs has helped in identifying functional requirements. Table 3 presents a sample ORSD for the OntoWM domain ontology. The OntoWM domain ontology is required to represent a taxonomy of waste bins, contracts, agents, collectors, transporters, routes, and the location area for all bins.

The designed OntoWM domain ontology will serve as a knowledge base for the intended users, and it can be utilized in different IS related to the WM domain. Furthermore, an

TABLE 2. Proposed Methodology for Ontology Development.

DSRM Activity	Activity Description		Activities / Tools / Techniques
Problem Identification	Define the research problem and justify the value of a solution.		Understanding the weaknesses and identifying a research problem that can be resolved by utilizing OntoWM.
Define objectives	How should the problem be solved?		Set the applications area of OntoWM driven for Waste Management System development.
Design and development	Create an artifact that solves the problem. Create the required ontology.		
	Stage 1 – Pre-Conceptualization	Definition	Identify the purpose, scope, and cases to be covered by OntoWM.
		Gantt Chart	Design a Gantt Chart for all stages, phases, and activities.
		Tools and Techniques	Identify techniques and tools for knowledge capturing.
		Estimation of Human resources	How many ontologists, domain experts, technical writers, and developers are required to design the ontology?
		Ontological and non-ontological resources	Identify ontological and non-ontological resources for knowledge acquisition.
		Re-engineering and Reusing	Reorganize and harmonize identified knowledge for OntoWM.
		Support for interoperability	Use relevant parts of a suitable upper ontology (like UFO) for providing interoperability of OntoWM to other sub-ontologies
		Knowledge acquisition	Conduct interviews, surveys, observations, and discussions, for identifying resources as sources for OntoWM.
		Domain Analysis	Use of Interlocking Institutional Worlds, for an understanding of a domain of discourse.
		Validation	Validation of the Ontology Requirements Document.
	Stage 2 – Conceptualization	Conceptualization Of both endurants and perdurants	Construct (specification) of OntoWM, using visual-based conceptual modeling languages, UFO, and OntoUML.
Demonstration	Demonstrate the artifact's use Prove that the artifact works, by solving one or more instances of the problem.		
	Stage 3 Part A – Post Conceptualization	Collaborative construction	Collaborative construction support allows different members of the ontology development team to work on a single ontology, at the same time.
		Localization	Translate an ontology to local natural languages.
		Implementation	Transform OntoWM using formal conceptual modeling languages (OWL, RDF, and so on).
		Instantiation	Populate OntoWM using an available dataset.
		Verification	Demonstrate a valid OntoWM structure, by developing an OntoWM and conforming it to a specific SmartBin system function.
		Versioning, Integration & Merging	Maintain different versions of an ontology Integrate OntoWM with other sub-domain ontologies.
Evaluation	How well does the artifact work? Observe and measure how well the artifact supports a solution to the problem, by comparing the objectives with the observed results.		
	Stage 3 Part B – Post Conceptualization	Evaluation	Evaluate the ontology with stakeholders, in terms of its applicability regarding the OntoWM for developing a SmartBin system.
		Maintenance	Ontology development is an evolutionary process. Evaluate the ontology with stakeholders, in term of its additions, improvements, and updates, regarding the OntoWM for developing a SmartBin system.
Communication	Communicate the problem, its solution, and the utility, novelty, and effectiveness of the solution to researchers and other relevant audiences		
	Stage 3 Part C – Post Conceptualization	Documentation	Document all necessary items of the developed OntoWM.

ontology-based web application has been developed. This system will assist the stakeholders in knowledge management and sharing in the waste collection sub-domain of the WM domain.

2) GANTT CHART

This phase will be performed by domain experts, ontology designers, ontology developers, technical writers, and ontology translators. The schedule for the design and development of domain ontologies is called the ontology development calendar. This calendar links the phases, activities, and tasks to be executed with the available resources. This schedule will support the resources with their performance. Gantt chart is one of the important ways to represent the ontology calendar, as shown in Table 4 .

3) TOOLS AND TECHNIQUES

This phase will be performed by ontology designers, ontology translators, ontology developers, and application developers. This phase will be performed after the definition of the domain ontology. In this phase, the latest well-equipped compatible tools should be selected, and for the implementation of ontologies, the latest applicable techniques for knowledge capturing should be identified. A tool that allows

TABLE 3. A sample ORSD for OntoWM domain ontology.

1	**Purpose**
	The purpose of the OntoWM domain ontology is to facilitate knowledge management and sharing in the waste collection sub-domain of the WM domain.
2	**Scope**
	The scope of the OntoWM domain ontology is the waste collection sub-domain of the WM domain.
3	**Implementation Language**
	The ontology has to be implemented in the OWL semantic language.
4	**Intended Users**
	The central institution in Malaysia, which is the National Solid Waste Management Department (NSWMD). Solid Waste Management and Public Cleansing (SWMPC). WM Institutions at the State level. Local Institutions in every State. Waste Collectors. Waste Transporters. Waste Recycling Industry. Other WM Departments. Citizens/Companies/Institutions.
5	**Intended Uses**
	Keep a record of waste bins. Keep a record of the collection history of waste bins. Keep a record of the collection history of waste bins by a collector. Keep a record of the collection history by area. Anticipate the next waste bin collection schedule and route plan.
6	**Ontology Requirements**
	Non-functional Requirements The ontology must support a multilingual scenario in the following languages: English, Malay. The ontology must be based on international standards in existence or under development. Functional Requirements How to classify the waste bin? How to anticipate the next collection of a waste bin? How to collect information about damaged waste bins for the replacement of waste bins? How to audit the collection of bins? How to calculate the commission of waste collectors? How to distribute waste bins in different areas? How to label the trucks based on the types of waste? How to repeat each waste bin collection cycle for a particular route or area?

users to browse, inspect, code, and edit ontologies, is called an Ontology Constructing Tool or Ontology Development Editor (ODE). For example, IsaViz, Apollo, SWOOP, and WebProtege [46]. Additionally, WebProtege can be utilized to verify and validate domain ontologies [47]. The most popular languages for representing ontologies include Web Ontology Language (OWL), Knowledge Interchange Format (KIF), Resource Description Framework (RDF), Ontology Inference Layer (DAML+OIL), and DARPA Agent Markup Language [48].

In this phase, the selection of an ODE such as WebProtege and a semantic ontology representation language, such as OWL, was made. WebProtege has been selected for the implementation of the designed OntoWM domain ontology, as it is an extensible, free, and open-source editor.

4) ESTIMATIONS OF HUMAN RESOURCES

In this phase, the estimation of a team with a suitable number of members will be made. The team members should have the required knowledge, suitable experience in their field, and should be interested in designing ontologies.

The interest of the team depends on many factors, such as the working environment, financial benefits, facilities provided, assignment of challenging and achievable targets, timeframe for the completion of a task, suitable supervision of the team, and relevancy of the provided task [49]. The estimation of human resources can be calculated using an ONTOCOM model introduced by Simperl *et al.* [50]. For example, Table 5 presents a sample of estimation of human resources with the required technical skills for the construction of the OntoWM domain ontology.

5) ONTOLOGICAL AND NON-ONTOLOGICAL RESOURCES

This phase will be performed by domain experts and ontology designers after the definition of the domain ontology. In this phase, ontological and non-ontological resources will be selected from reliable sources. These resources will be utilized for the elicitation of domain-knowledge. In this context, the key human resource of domain-knowledge is domain experts. Other resources include books, article publications, technical reports, online or offline surveys, relevant case studies, financial reports, international standards, and reference models [51].

Ontological and non-ontological resources were selected from reliable sources such as the Government Institutions in Malaysia. In this context, all related ontological and non-ontological resources assisted in designing the OntoWM domain ontology. These resources will be utilized later for the instantiation of the OntoWM domain ontology and the construction of knowledge graphs [52].

Table 6 presents some methods used to identify the resources. By interviewing key personnel in the WM sector, and by observing the manual activities involved in waste bin collection and monitoring of this process, this study has identified the resources involved. Specifically, the study focuses on the system that is currently operating in the Kemaman Municipal Council (KMC), in the State of Terengganu, Malaysia.

6) RE-ENGINEERING AND REUSE

This phase will be performed by domain experts and ontology designers. This phase will be performed after the identification of knowledge resources of the domain ontology. At this stage, the resources required for a domain ontology have been identified. However, such resources are not entirely useful as they currently exist; therefore, they should be modified or re-engineered to serve their intended purpose. The content of these resources should be verified, assessed, and compared to determine their suitability. Subsequently, the most suitable ones should be selected for utilization in constructing a domain ontology.

This study has reorganized and harmonized the identified knowledge, extracting candidates to become parts of common knowledge (or shared knowledge) that covers the WM domain. Table 7 illustrates samples of common

TABLE 4. Gantt chart for OntoWM for the WM domain.

Project Activities	2020											
Months	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Stage 1: Pre-conceptualization	■	■	■									
Stage 2: Conceptualization		■	■	■	■	■						
Stage 3 Part A: Post-conceptualization				■	■	■	■	■	■			
Stage 3 Part B: Post-conceptualization						■	■	■	■	■	■	
Stage 3 Part C: Post-conceptualization					■	■	■	■	■	■	■	■

Δ Milestones and Dates

Stage 1: Pre-conceptualization (February– March 2020)

Stage 2: Conceptualization (March– June 2020)

Stage 3: Design WM ontology (April– December 2020)

TABLE 5. A Sample Estimation of Human Resources for OntoWM Ontology for Collection Sub-domain.

No.	Resources	Count	Purpose
1	Project Managers	1	Manage all activities related to ontology design, development, implementation, and maintenance.
2	Domain Experts	2	For knowledge acquisition of the WM domain.
3	Ontology Engineers	2	For the design, development, implementation, and maintenance of the OntoWM domain ontology.
4	Ontology Translators	1	For localization of the OntoWM domain ontology into the Malay Language.
5	Technical Writers	1	Documentation.
6	Back-end Application Developers	1	For the development of the back-end of the SmartBinAnalytics system, an ontology-based web application using Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), and JavaScript (JS).
7	Front-end Application Developers	1	For the development of the front end of the SmartBinAnalytics ontology-based web application using JSP.

TABLE 6. A Sample of Methods for Resources Identification.

No	Method	Sub-domain	Identified resources
1	Interview	Collection and Monitoring	Contractor agreement
2	Interview	Collection and Monitoring	Bins allocation document
3	Document review	Monitoring	Bins inventory
4	Observation	Collection and Monitoring	Collection routes

knowledge that have been identified from the selected knowledge resources. Additionally, this study was unable to find any mature domain ontology related to the WM domain, especially for the waste collection sub-domain. In this way, the re-using of an existing ontology is not exactly feasible.

TABLE 7. A Sample of the Common Knowledge for OntoWM.

No	Source	Sub-domain	Common knowledge
1	Contractor agreement	Collection and Monitoring	#Contractor, #Collector, #Supervisor
2	Contractor agreement	Collection and Monitoring	#Bin, #BinType, #BinGPS, #BinLocation
3	Contractor agreement	Collection and Monitoring	#CollectedBins, #UncollectedBins
4	Collection routes	Collection	#Zone, #Trip

7) SUPPORT FOR INTEROPERABILITY

This phase will be performed by domain experts and ontology designers after identifying the knowledge resources of the domain ontology. Interoperability has four levels: legislative, technical, semantic, and organizational. Over recent years, there has been a developing interest in the utilization of upper-level ontologies for creating domain ontologies, to achieve semantic interoperation of IS. In this phase, this study will focus on semantic interoperability only. Sharing semantic interpretation of knowledge using meta-models is called semantic interpretation. There are many obstacles in sharing knowledge, such as different data formats, the variable meaning of terms [53], area of practice, multiple natural languages [54], different contexts reflected by diverse ontologies, and diverse folksonomies generated in different social media using social tagging [55].

Semantic interoperability between systems requires the same supporting structure or same high-level concepts. The semantic interoperability can be achieved in decision-making systems by a shared understanding of the context, the task, and the capabilities and perspectives of each other [56]. The resulting ontologies will hence have a common global conceptual structure, and it will assist them in sharing their concepts. Semantic interoperability strongly depends on the availability of precise and explicit representations of the conceptualizations of their domains of interest [57].

However, the question that arises is how to provide semantic interoperability by matching ontologies of multiple IS. One way to integrate heterogeneous knowledge of different ontologies is to utilize upper-level ontologies. The utilization of upper-level ontologies provides a common ontological foundation for domain ontologies [58]. Many upper-level ontologies have been developed over the years and utilized in different domains such as Software Engineering, Human Common-sense, Enterprise Modeling, Telecommunications, Bioinformatics, and others. This field is often challenging; it involves the correlation of different actors and various pieces of information. In this context, the Industrial Ontologies Foundry [59] has been trying to reduce the gap between designing formal ontologies and fruitful applications of these ontologies. For providing semantic interoperability, this work has used the Unified Foundational Ontology (UFO) upper-level ontology for modeling the OntoWM domain ontology.

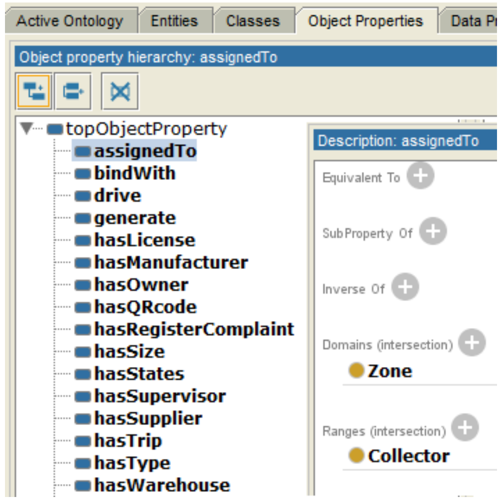


FIGURE 1. A sample list of object properties in OntoWM domain ontology.

8) KNOWLEDGE ACQUISITION

This phase will be performed by domain experts and ontology designers after the identification of knowledge resources of the domain ontology. Any number of methods be they qualitative, quantitative, or mixed methods, can be utilized to identify domain-knowledge. Relevant resources for domains can be identified initially in the previous phase (Ontological and non-ontological resources), which can then be utilized to select sources for ontology development. In this phase, domain-related concepts and their properties will be defined. A final list or glossary of concepts and their properties will be prepared. This glossary will have clear definitions of domain-related concepts. The resulting document contains relevant data regarding bins, collectors, contractors, and other matters (as shown in Table 8).

9) DOMAIN ANALYSIS

The domain vocabulary is built by extracting domain-specific concepts from identified ontological and non-ontological resources such as standards, policies, procedures, and reports. A concept symbolizes a set or class of entities within a given domain; for example, "Bin" is a concept in the WM domain. Domain ontologies have classes as key elements, and these classes are derived from domain concepts. Identified classes with their attributes can be represented graphically using UML.

Properties construct the structure of a concept. There are two types of properties: object properties and data properties. Data properties have values, such as a "collectorName" data property which has a value of "Muhammad Ali". Whereas object properties link together instances of different objects. For example, "assignedTo" is an object property that links "Zone" (an instance of Zone class) with "Collector" (an instance of "Collector" class). Fig.1 illustrates a sample list of object properties in the OntoWM domain ontology. Fig.2 illustrates a sample list of concepts or classes in the OntoWM domain ontology.

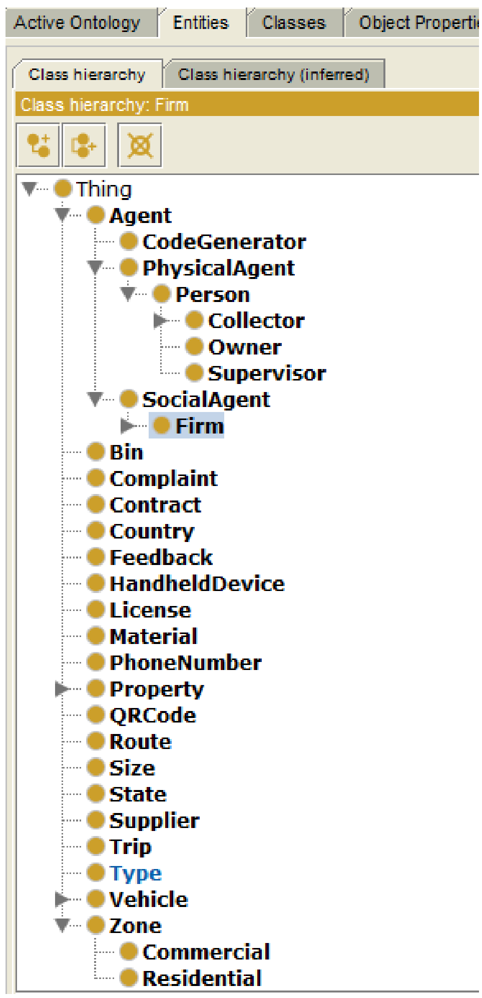


FIGURE 2. A sample list of concepts in OntoWM domain ontology.

10) VALIDATION

This phase is concerned with validating the designed ORSD by putting it to work, or by examining its applicability within a real-world setting. In this context, if the ORSD is not according to the ontology owner or intended users of the designed ontology, then the researchers need to review the ORSD. The ORSD must be according to the needs of the ontology owner or intended users of the designed

TABLE 8. A sample Glossary of Concepts in the WM Domain.

Concept	Description
Country	A nation with its government, occupying a particular territory.
State	A nation or territory considered as an organized political community under one government.
ZoneCategory	Category of a zone; for example, residential area, commercial area, industrial area.
Zone	An area with a particular characteristic, purpose, or use, or subject to particular restriction.
Location	A particular place or position.
Route	A way taken for getting from a commencing point to a destination.
BinManufacturer	A person or company that makes waste bins for sale.
BinSupplier	A person or organization that provides something required, such as waste bins.
BinSize	The overall dimensions or magnitude of a bin; how big the waste bin is. For example, Bin1.5, Bin1.1, Bin240, and so on.
Bin	A container for household refuse, especially one kept outside.
BinCodeGenerator	A combination of hardware and software that generates waste bin codes.
BinRegistration	New waste bin registration to the waste collection authority.
BinMonitoring	Observe and verify the fill level of the bin over some time.
BinCollectionComplaints	A statement that the waste bin collection process is unsatisfactory or unacceptable.
WasteCollectionCost	Charged to a citizen.
BinCollectionPayment	Details of payments.
Collections	The action or process of collecting filled waste bins.
DisposalArea	This means that waste has to be disposed of at a disposal site, which is defined as a site used for the accumulation of waste to dispose of or to treat.
TypeOfWaste	A category of waste having common characteristics. For example, solid wastes, liquid wastes, gas wastes, and so forth.
CategoryOfWaste	Process-able, transportable, recycle-able, store-able, and utilizable are categories of waste.
CompositionOfWaste	Medical waste, production waste, consumer waste, biological waste, and radioactive waste.
WasteManagementMethod	Type of method utilized for WM. For example, dispose of or recycle.
CollectionSupervisor	A person who supervises a waste collection activity.
Feedback	Information about reactions to waste collection.
Waste	Any garbage, refuse, sludge from a wastewater treatment plant, water supply treatment plant, or air pollution control facility and other discarded materials including solid, liquid, semi-solid, or contained gaseous material, resulting from industrial, commercial, mining, and agricultural operations.
EffectOfDisposal	A change that is as a result or consequence of waste. For example, disposal is extremely hazardous, harmless, highly hazardous, low hazardous, or moderately hazardous.
TelecomServiceProvider	Telecommunication companies that provide communication services with the provision of a Subscriber Identification Module (SIM).
Trip	A journey for waste collection.
Collector	Person or organization who collects the waste.
Firm	A firm is a business organization such as a corporation, Limited Liability Company (LLC), or partnership that performs activities regarding WM.
Contractor	A person or firm that undertakes a contract to do all WM activities.
Agent	A person who acts on the behalf of another person or group.
Contract	A written or spoken agreement, concerning WM.
VehicleType	Types of a vehicle for waste collection.
Vehicle	A vehicle is a machine that transports waste from one location to another.
Driver	A person who drives the vehicle.
Smartphone	A phone that performs scanning of barcode on the waste bin.
Property	A building belonging to someone; possessions collectively.
Owner	A person who owns a building.

ontology. Ontology validation ensures that a correct domain ontology is being built [60]. Detailed open-ended questionnaires or interviews with ontology owners and the users of the ontology will be required to validate the specifications of the OntoWM domain ontology. The OntoWM domain ontology is in progress, and validations will be performed after completion of the domain ontology. SmartBinAnalytics, an ontology-based web application, is under construction and will assist in validating the developed OntoWM domain ontology. As such, the developed ontology will be able to assist the owner and the users of ontology in knowledge management, and the results of the validation of the developed ontology will be satisfactory.

B. STAGE 2 – CONCEPTUALIZATION

The activities of this phase are devoted to the construction of domain ontologies. The material things of a domain of discourse, captured in the previous phases, are required to be organized well. The authors of this article have used UFO as an upper-level ontology for providing a skeleton for OntoWM (as shown in Fig.3) [57]. This provides rich semantics and assists in overcoming the situation of ontological deficiencies. These advantages are important for the verification phase (OntoWM evaluation) and are also used to ensure the quality of structural aspects of OntoWM. The utilization of an upper-level ontology will also assist in supporting interoperability among the sub-systems of the WM domain.

Fig.3 presents key knowledge of the WM domain, which is relevant to the development of the SmartBin system. WM, as a single domain, is composed of several other subdomains. Here, the zone is divided into several other sub-zones, while bin locations are actual and original, and are associated with a quality domain composed of quality dimensions including latitude, longitude, and altitude. A class of Location Coordinates, classified as ≪datatype≫, consists of a particular set of location coordinates.

1) CONCEPTUALIZATION OF BOTH ENDURANTS AND PERDURANTS

This phase will be performed by domain experts and ontology designers after domain analysis. It has been observed that most of the researchers focus on endurant ontologies of the WM domain, and put less or no effort into the modeling of perdurant ontologies. The state of a thing is a collection of property values at a point in time. Fullness is a state of a waste bin. Emptiness is also a state of a waste bin. A change in the state is called an event. Changing the status of fullness to emptiness of a waste bin is called an event or action or speech act. All the emptiness and fullness events over a timespan are called the history of a particular waste bin. The filling of a waste bin is a speech act that will create or destroy the fullness and emptiness of institutional facts for a waste bin in the WM domain. In this context, speech acts or actions are the central things to model in domain ontologies [13], and [61].

Unified Modeling Language (UML) includes several models that allow the representation of events, in particular, the Activity Model [62]. These activities and actions are perdurant event types. For example, every CollectBin activity will have an instance of a perdurant event type. The conditions under which these events can occur are represented by the Object Constraint Language (OCL) predicates [63]. The outcome of an action is specified by the local post-condition(s).

Finally, the selected upper-level ontology should be implementable in widely used powerful semantic languages such as OWL and RDF. Additionally, the endurant concepts such as kind, sub-kind, phase, role, and relator have been proposed by Falbo [64], which can be represented as stereotypes of the UML meta-class Class.

An endurant (an institutional fact) is an entity that exists timelessly. All of its parts exist at the same time. When a waste bin is registered in the WM system, an instance of a waste bin is created, which is not an instance of any of the subclasses. The event instance is the appearance of the instance of the waste bin. When the collections are managed, that event instance is the appearance of instances of the subclasses Collector, Route, and CheckFullness. This is the point at which the activity CollectBin occurs (see Fig. 4).

In this context, for modeling modular domain ontologies, and to solve the problem of semantic interoperability, a suitable upper-level ontology is required. The use of upper-level ontologies such as UFO provides rich semantics, ensures the quality of structural aspects, and assists in overcoming the situation of ontological deficiencies. To ensure that the selected upper-level ontology can be extended to the required domain ontology, it should be universal.

CollectBin (Activity)

- Select Collector (Action)
- Select Route (Action)
- Check Fullness (Action)
- Collect Bin (Action)

The action SelectCollector assigns a pool of waste bins to a collector. The collector performs the SelectRoute action for collecting the particular waste bin on a particular route. The collector checks the status of the waste bin by the CheckFullness action. And finally, the collector collects the filled bin and performs the Collect action. These activities and actions are perdurant event types. Every CollectBin activity has an instance of each of these types. The outcome of an action is specified by the local post-condition. The pre-condition of one action is the post-condition of the preceding action. If an activity is a sub-activity of a larger activity, then the pre-conditions in the sub-activity are not shown. Its pre-condition is specified by a local post-condition in a preceding action. These pre-conditions are necessary for the invocation of the concerned action. Fig.5 shows a fragment of the endurant ontology associated with the perdurant ontology of Fig. 4. By registering a CollectBin P-act, a new institutional fact is created in the WM domain. The CollectBin perdurant ontology includes the actions SelectCollector, SelectRoute, CheckFullness and Collect.

C. STAGE 3 PART A – Post-Conceptualization

1) COLLABORATIVE CONSTRUCTION

Designing domain ontologies requires strong methodological support for collaborative construction. Creating and maintaining domain ontologies is not an individual task or activity. In this context, the interaction of domain experts, knowledge engineers, ontology engineers, technical writers, and other stakeholders during the Ontology Development Life Cycle (ODLC) is essential. In this phase, the configuration of a collaborative environment for ontology construction will be

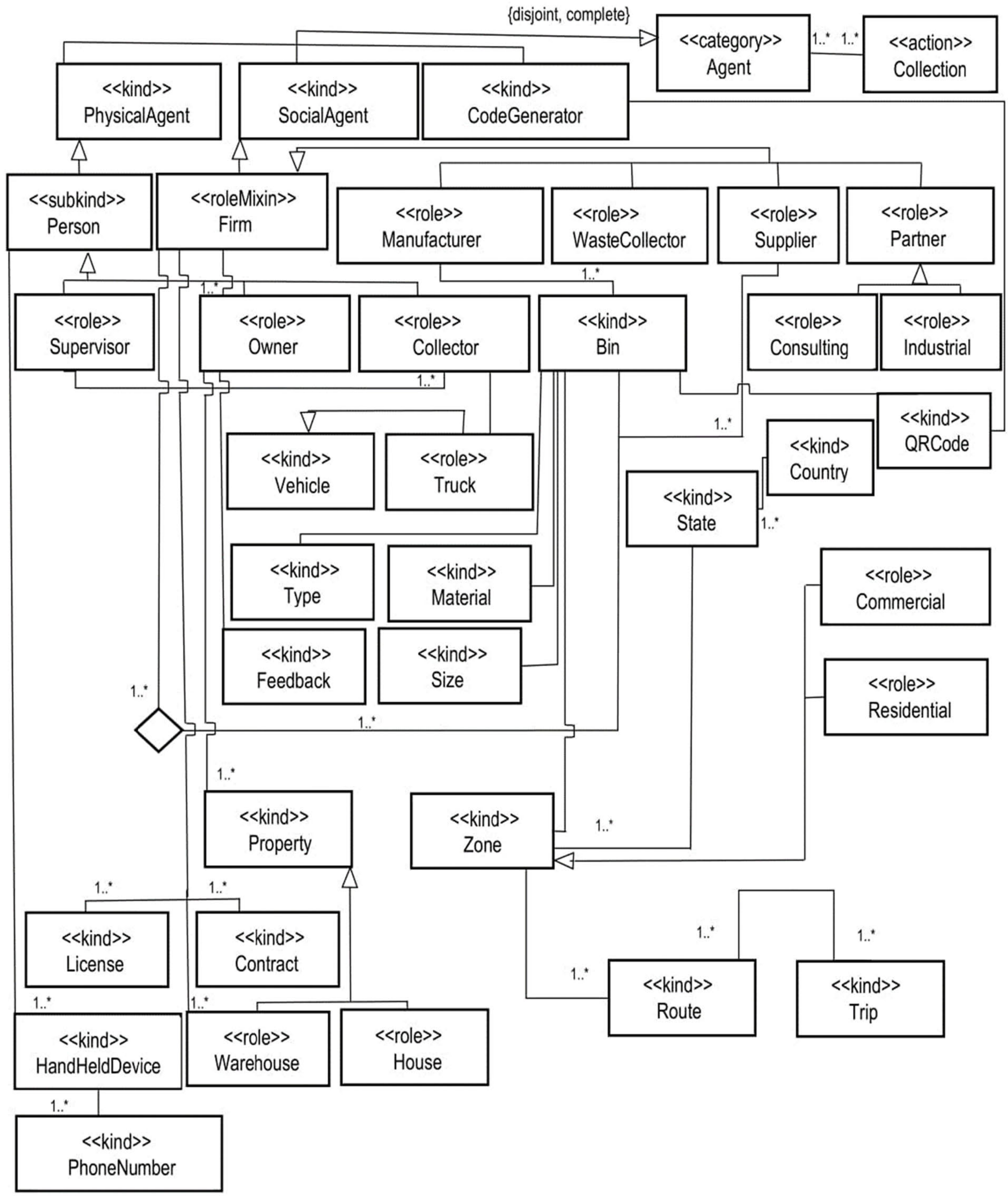


FIGURE 3. **A portion of OntoWM domain ontology.**

established. This collaborative environment will assist the stakeholders in sharing documentation, making consensus on conflicts through argumentation, and other ontology development challenges.

In this context, the selection of a dedicated software tool is required for supporting activities like reusing design patterns, reusing existing ontologies, re-engineering of a database, lexica, and thesauri schemas. For example, the latest version of the Kali-ma NeOn Toolkit plugin with additional required plugins can be utilized for the aforementioned purposes [65]. Another ODE is WebProtege; it can be utilized for the collaborative ontology construction of domain ontologies. WebProtege is a collaborative tool that is currently hosting 68,000 OWL ontology projects and has accounts of 50,000

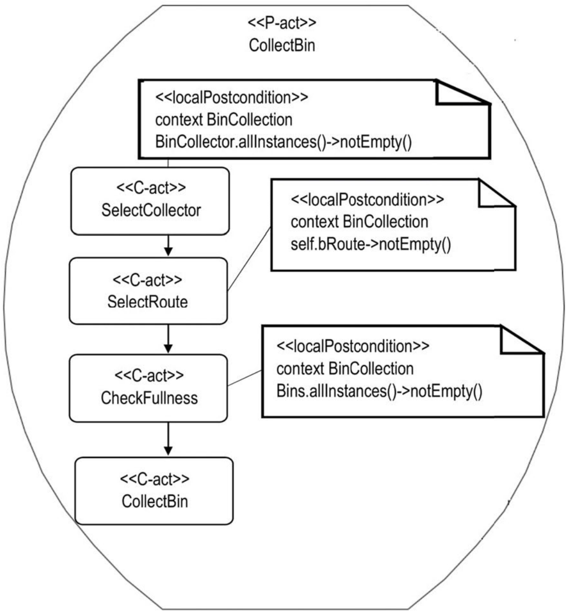


FIGURE 4. CollectBin perdurant ontology.

TABLE 9. Tools for Collaborative Ontology Construction.

Tool	Collaborative construction
OntoStudio	Yes
Protégé	No
WebProtege	Yes
Swoop	Yes
TopBraid Composer (FE)	Yes

worldwide registered users [66]. The WebProtege ODE is a web-based version of Protégé ODE. Table 9 represents a few tools for ontology construction adopted from Horridge *et al.* [66]. The selected ODE will assist the ontology developers in the ontology development process. This study has used WebProtege as a semantic ontology tool for the collaborative construction of the OntoWM domain ontology. The WebProtege ODE allows different participants of the OntoWM domain ontology project to work from remote physical locations. This feature of a web based WebProtege ODE makes the task of ontology construction easy.

2) LOCALIZATION

This phase will be performed by domain experts and ontology translators. This phase will be performed (if required) after the conceptualization of the domain ontology. Adaption of domain ontologies to a culture and a local natural language is called the localization of ontology [20]. The success of the Semantic Web is subject to the availability of easily accessible high-quality content in a local natural language. According to I. W. Statistics [67], the distribution of internet users by languages is as follows: English 25.9%, Chinese 9.4%, Spanish 7.9%, Arabic 5.2%, Japanese 5.2%, Portuguese 3.7%, French 3.3%, Russian 2.5%, German 2.0%, and other languages 23.1%. In this context, domain ontologies should be translated into multiple languages.

For example, one mechanism for localization of domain ontologies is to provide a translation file that contains a translation of all terms of concepts, properties, and relationships. In this context, there will be a separate translation file for each natural language. For instance, there are different types of waste generators, one of them is a Household type. For localization of a concept "House" in different languages such as Malay, Urdu, Chinese, and Arabic, this concept should have a translation in a translation file (.po) using a POEDIT translation editor [68]. The translation of the "House" concept in four different natural languages is illustrated in Fig.6. However, additional algorithms, constructs of implementation language, and a file structure will be required in software applications for the integration and processing of this translated file (.po). A sample translation file for a Malay Language instance is illustrated in Fig.7.

SmartBinAnalytics, an ontology-based web application, is currently under construction. A module will be developed in the SmartBinAnalytics application for the localization of the OntoWM domain ontology.

3) IMPLEMENTATION

In this phase, a formal model of a designed ontology will be transformed or translated into a formal language such as OWL, OWL-S, RDF, LINGO, and others. The translation from the visual-based ontology specification, to any formal-based ontology specification, usually takes place for many purposes, principally for computational or machine-readable purposes. In this context, automated tools such as WebProtege can assist in converting a formal model to a formal language [66].

The selection of a formal language for implementation depends on the purpose and scope of the domain ontologies. In this context, the selection will be made by the ontology developers. Sometimes, a semantic formal language does not meet the requirements of a designed formal model. For example, a case where the selected semantic language has no construct for the implementation of perdurants. In such a situation, ontology developers, as ontology experts, will change their selection of a formal semantic language. In this study, a formal model of a designed OntoWM domain ontology is transformed into a formal language such as OWL. OWL has been selected because of its powerful representation and expressiveness.

4) INSTANTIATION

This phase will be performed by ontology developers after the implementation of the domain ontology. In this phase, instances and individuals of classes will be created by assigning data values of relevant data types such as integer, real, string, and time. Additionally, these instances will serve as building blocks for an ontology-based IS. This will be done with the support of the relevant feature offered by the selected

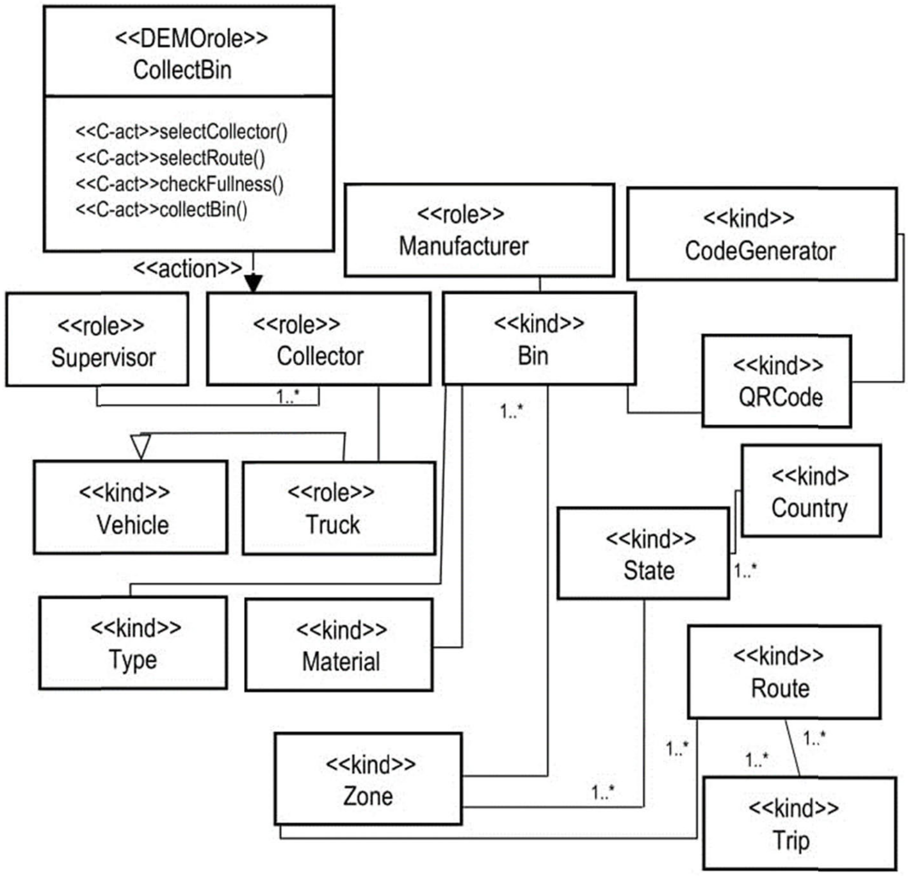


FIGURE 5. CollectBin endurant ontology.

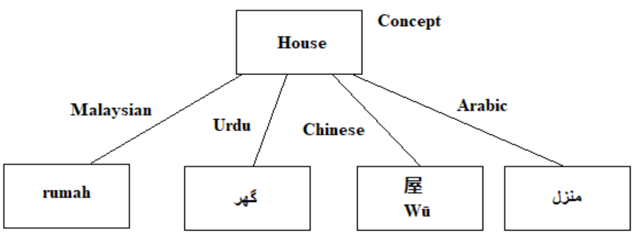


FIGURE 6. Localization of a concept.

Source text - English	Malaysian Language
House	Rumah
Truck	Lorri
Solid waste	sisa pepejal

FIGURE 7. A translation (.po) file for the Malay-language.

ODE such as WebProtege. For example, instances of the Bin concept are shown in Table 10.

Fig.8 graphically presents an individual of a waste collecting Firm (a concept) in the OntoWM domain ontology. These individuals are stored using triples in the OWL ontology. A triple contains a subject, predicate, and an object. For example, "Ali Brother Pvt. signs a contract" is a triple, here

TABLE 10. Instances of Bin Concept.

Concept	Instance
Bin	LB1/K01/660/1
Bin	LB1/K01/240/1
Bin	LB1/K01/60/1
Bin	LB1/K03/240/1
Bin	LB1/K02/240/1
Bin	LB1/K03/60/1
Bin	LB1/K03/660/1
Bin	LB1/K03/240/1

"Ali Brother Pvt." is a subject, "signs" is a predicate, and "contract", is an object. The predicate "signs" associates the firm with a contract and shows how the subject and object are connected.

5) VERIFICATION

This phase will be performed by domain experts and ontology developers after the ontology population and is concerned with verifying the developed domain ontology against ORSD. This document has details about CQs, functional requirements, and non-functional requirements. In other words, it ensures that the structure of the developed domain ontology is intact. Additionally, this phase ensures that a targeted domain

FIGURE 8. Localization of a concept.

ontology is being built correctly [60]. In this context, if the developed ontology is different from the ORSD, then necessary corrective measures will be taken.

6) VERSIONING

This phase will be performed by ontology developers after verification of the domain ontology. Ontology versioning primarily manages the management of changes in ontologies. In this phase, a record of different versions of an ontology will be kept. In a collaborative ontology construction environment, the evolution of the ontology should be managed and monitored carefully. The OWL ontology has supportive primitives for maintaining different versions of a domain ontology. The ODE plays a key role in maintaining ontology versions. The initial version of the OntoWM domain ontology has been designed and developed in this study. In the future, more versions of OntoWM will be developed, and this will require the management of different versions, especially when improved versions of the OntoWM domain ontology will be developed by different ontologists.

7) INTEGRATION AND MERGING

This phase will be performed by domain experts, integration experts, and ontology developers. Ontology integration is a process of constructing a new integrated ontology by the integration of two or more ontologies [69]. Ontology integration is different from merging in that, in ontology integration, source ontologies have different domains from the domain of the integrated resulting ontology [70]. For example, ontology integration has been applied in the domain Ontology for Energy Management Applications (OEMA), which is composed of eight interconnected domain ontologies [69]. In ontology merging, sub-ontologies of a domain are merged for constructing a comprehensive ontology [70]. For example, the Intelligent System proposed by Petrov *et al.* [71], that merged anatomical sub-ontologies based on heuristics algorithms and directed acyclic graph models. Both ontology integration and merging are key aspects of collaborative ontology construction. Ontology integration is required when researchers desire to re-use an existing ontology, and ontology merging is utilized when ontology designers desire to merge sub-ontologies of the same domain.

Additionally, in the example of the OntoWM domain ontology, UFO has been integrated as an upper-level ontology for modeling endurants, and in the future, this will assist in merging other sub-ontologies in the WM domain such as Regulation, Monitoring, Transport, Treatment, and Disposal. The initial version of the OntoWM domain ontology has been designed and developed. As future work, after the development of sub-ontologies of other sub-domains, the integration process will be implemented.

D. STAGE 3 PART B – Post-Conceptualization

1) EVALUATION

This phase will be performed by domain experts, ontology designers, ontology developers, and application developers. In this phase, the technical quality of a domain ontology will be verified [44]. The primary goal of this phase is to verify the correctness and quality of a domain ontology. There are a few important goals of ontology evaluation. Firstly, domain coverage will focus on the coverage of a domain of discourse in a domain ontology construction project. Secondly, the quality of the design and development process of a domain ontology will ensure its reliability. Thirdly, the suitability of developed ontologies for ontology-based applications will be evaluated [62], and [72]. Finally, ontology utilization and adoption; how other ontology designers rate the developed domain ontology, and how they re-use this ontology in their ontologies will be explored [73]. The developed ontology should be evaluated based on the following ontology characteristics [21]:

- Competency Questions (CQs): Developed ontology can be verified against defined CQs.
- Ontology consistency: Consistency of a domain ontology can be verified with the assistance of ontology reasoners such as Racer or FaCT++ [74].
- Ontology content: Completeness and correctness are two important criteria for evaluating the developed domain ontology [75].
- Maintenance: The changes in a domain of discourse are required to be incorporated in a reliable domain ontology [76].

Over the years, researchers have proposed a variety of approaches for evaluating ontologies [60]. The selection of the approach for the evaluation of domain ontologies will be decided by the domain experts, ontology owners, and the other stakeholders.

2) THE GOLD STANDARD EVALUATION

This approach compares the ontology to high-level and ‘golden’ standards, which can be the ontology itself. Concerning the gold standard evaluation, in some scenarios, access to such standards (or ontology) or its provision may not be possible.

3) DATA-DRIVEN EVALUATION

This assessment method compares the ontology with a source of data such as a corpus. If the terms utilized in the ontology are not present in the corpus or vice versa, the ontology is penalized.

4) EVALUATION BY HUMANS

This approach utilizes a set of pre-defined criteria. This allows the users to measure the suitability of existing ontologies considering the requirements of a system. The measurements are based on the five main dimensions of tools, language, content, methodology, and cost.

5) APPLICATION-BASED EVALUATION

The application-based evaluation approach first utilizes the ontologies in applications, and then evaluates the results. This approach is useful to assess the capabilities of the developed ontology to meet its objectives, for example, decision support or knowledge management. This study has applied the application-based utilization of the OntoWM domain ontology.

6) MAINTENANCE

In this phase, the changes in terms of errors or missing knowledge of a domain of discourse are required to be incorporated in a domain ontology [76]. Additionally, improved versions of the domain ontology will be developed with proper versioning control. For future work, with new requirements or any modifications in the previous requirements, these requirements are required to be implemented, and will also be implemented in a future version of the OntoWM domain ontology. Finally, improved versions of the OntoWM domain ontology will be developed with proper versioning.

E. STAGE 3 PART C – Post-Conceptualization

1) DOCUMENTATION

This phase will be performed by domain experts, ontology designers, and technical writers. This phase will be performed throughout the ODLC because the processes and activities of each phase will be documented. Every activity of ontology development should be descriptive, and well-documented [64]. Documentation assists in gaining an understanding of the ontology, its use, reuse, and revisions. Ontology owners should make the documentation available online. It will be useful for researchers or users of a developed ontology, to get an understanding of the different aspects of this ontology. Ontology documentation includes human-readable contents of an ontology, machine-readable metadata of documentation, and availability of a web-based version of the ontology documentation [77]. Every statement in the ontology, the purpose of the ontology, re-usability of other ontologies, modeling criteria, and theories on which the ontology is based, should be explained in detail in the documentation [70].

Every activity of the development process of the OntoWM domain ontology has been documented. The ontology documentation has been done in each stage and phase discussed above. Each stage of the ontology development process has been described in detail, giving detailed information on how the ontology has been developed.

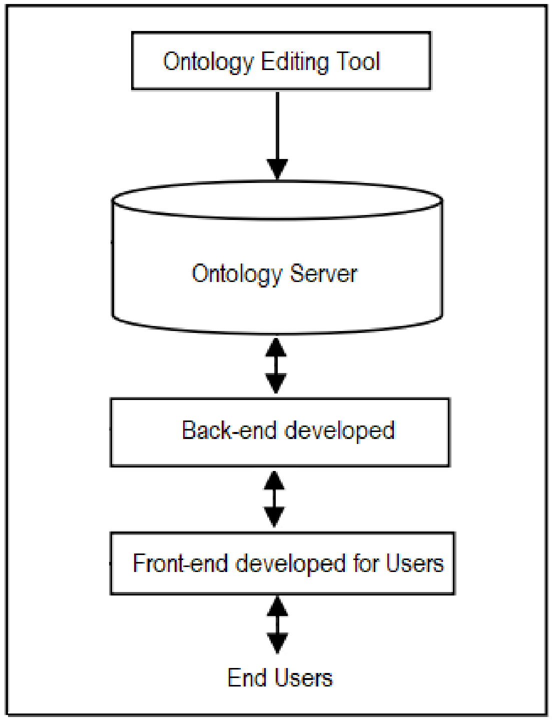


FIGURE 9. An ontology-based knowledge retrieval model.

IV. UTILIZATION OF THE DESIGNED OntoWM DOMAIN ONTOLOGY USING SmartBinAnalytics

The primary task of Information Systems is knowledge management and sharing. These knowledge management and sharing activities are provided for the stakeholders of the domain of discourse, to meet their requirements of knowledge and information in an acceptable time. To achieve the aforementioned knowledge management and sharing goals, knowledge management deals with representation, organization, storage, and access to information or knowledge items. Due to the invention of intelligent software agents, stakeholders need knowledge in human as well as machine-readable formats. For providing easy access of knowledge to stakeholders, IS provide an easy-to-use Graphical User Interface (GUI) for accessing the required knowledge. The GUI sends the knowledge request in the form of SPARQL to the base ontology. Finally, the generated output is displayed backed to the GUI for users. Fig.9 illustrates the ontology-based knowledge retrieval model. The results obtained illustrate the feasibility of the proposed ontology-based web application, which can significantly improve the knowledge management and sharing for the waste collection sub-domain of the WM domain.

A prototype ontology-based web application, namely, SmartBinAnalytics has been designed and developed to demonstrate; how the proposed OntoWM domain ontology is applied to capture and store the knowledge relevant to the waste collection sub-domain of the WM domain. SmartBinAnalytics is an ontology-based web application developed using JSP. A Java platform has been selected for developing

SmartBinAnalytics because it allows for portable applications, and it has open-source libraries such as Jena for ontology development and deployment. In technical architecture, a web application processes Hypertext Transfer Protocol (HTTP) requests of users from their respective web browsers. This web application reads the developed OWL ontology using a Java framework for building semantic web applications, namely, the Jena API [78]. The helpful provision of a programmatic environment by Jena API makes the life of application developers easy. The Jena API assists in developing ontology-based applications based on ontologies developed in RDF and OWL formats. The availability of open-source ODE assists in graphically designing ontologies such as WebProtege. This study has used WebProtege for the development of the OntoWM domain ontology.

The prototype described in this study has been evaluated by the application developers. Its purpose is to be used as a proof of concept to demonstrate the utilization of the OntoWM domain ontology. It serves as a working prototype that has been used in discussions regarding subsequent work that will involve on-site operational systems. This prototype may help to comprehend how practical ontology-driven applications can be used to support the waste collection process. The following sub-sections describe the different usage scenarios of the SmartBinAnalytics ontology-based web application.

A. OntoWM USAGE SCENARIO 1: TRANSPORT DEPARTMENT REQUIRES COLLECTION HISTORY OF COLLECTED WASTE BINS

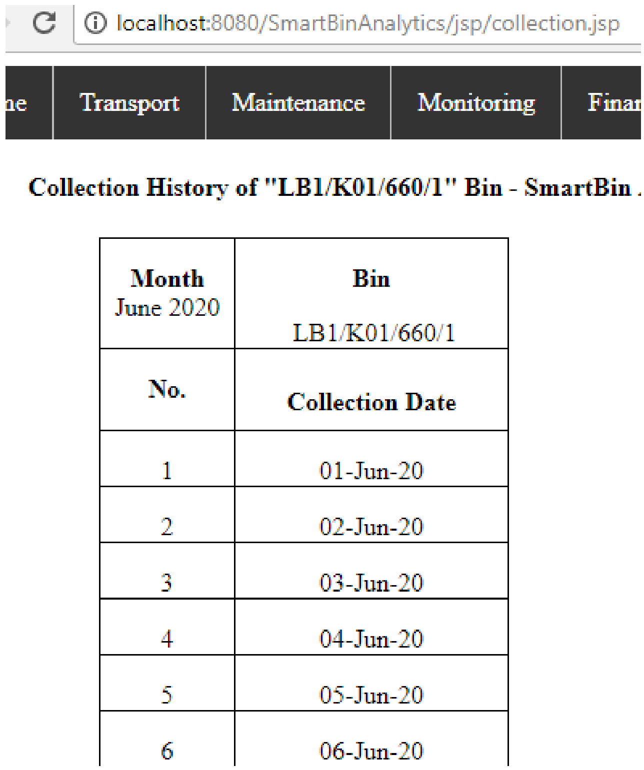
The primary function of the Collection Department is the collection of filled waste bins. The Collection Department requires a history of collected waste bins, and the SmartBinAnalytics application assists the Collection Department in gathering this information. Fig.10 illustrates the report generated from the SmartBinAnalytics web application about the collection history of a waste bin with code LB1/K01/660/1. It contains values of collectionDate, and binCode datatype properties. The value of collectionDate datatype property consists of collection date and collection time.

B. OntoWM USAGE SCENARIO 2: MAINTENANCE DEPARTMENT REQUIRES INFORMATION ABOUT UNUSABLE WASTE BINS

The primary function of the Maintenance Department is to ensure the availability of waste bins and to verify this availability after a specific period. The Maintenance Department requires information on unusable and usable waste bins, and the SmartBinAnalytics web application assists them in gathering this information. Fig.11 illustrates the report generated from the SmartBinAnalytics web application showing unusable and usable waste bins.

V. DISCUSSION

Most of the methodologies studied provide domain analysis, conceptualization, implementation, evaluation, instantiation,



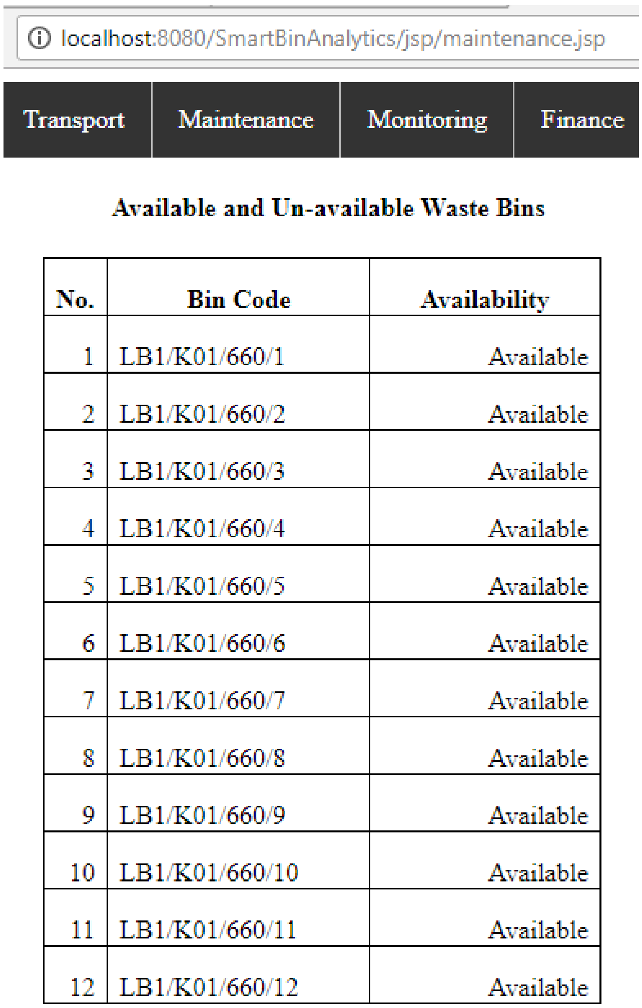
Month June 2020	Bin LB1/K01/660/1
No.	Collection Date
1	01-Jun-20
2	02-Jun-20
3	03-Jun-20
4	04-Jun-20
5	05-Jun-20
6	06-Jun-20

FIGURE 10. Collection history of a waste bin LB1/K01/660/1 generated from SmartBinAnalytics.

and provide an example of domain ontology. Although most of them have not provided all the details of the techniques and activities involved. Less details make it difficult to follow a methodology for designing ontologies for a specific domain. Some exceptions provide details, in particular [17]–[20]. Additionally, most of the methodologies studied (as shown in Table 1) do not offer support for maintenance, documentation, construction of collaborative ontologies, support for reuse, support for integration, support for interoperability, ontological localization, and human resources estimation (for example, ontology experts).

The development of methodologies remains a complicated and tedious task. Current literature shows that most of the methodologies are based on the experience of one or more projects, which is not sufficient to fully validate their effectiveness. The analysis highlights gaps in the system of methodologies. Based on the discussion and analysis carried out in the previous sections, the following points, therefore, converge to conclude this study.

- None of the methodologies are completely mature, if analyzed and compared based on the established criteria.
- Most of the methodologies discussed in the study do not provide sufficient details on the techniques and activities used therein. There are some exceptions, mainly NeOn [35].
- Some methodologies support the concept of reusability, localization, integration, interoperability, estimation


localhost:8080/SmartBinAnalytics/jsp/maintenance.jsp

Transport | Maintenance | Monitoring | Finance

Available and Un-available Waste Bins

No.	Bin Code	Availability
1	LB1/K01/660/1	Available
2	LB1/K01/660/2	Available
3	LB1/K01/660/3	Available
4	LB1/K01/660/4	Available
5	LB1/K01/660/5	Available
6	LB1/K01/660/6	Available
7	LB1/K01/660/7	Available
8	LB1/K01/660/8	Available
9	LB1/K01/660/9	Available
10	LB1/K01/660/10	Available
11	LB1/K01/660/11	Available
12	LB1/K01/660/12	Available

FIGURE 11. A sample list of available and un-available waste bins generated from SmartBinAnalytics.

of human resources, and reengineering. However, only some of them have this kind of support.

- Collaborative construction is an important aspect of ontological engineering; however, little attention has been paid to this aspect.
- Most methodologies opt for conventional strategies to identify the concepts of ontology, integration, and domain analysis. Other new methods such as Colomb and Ahmad [14], and techniques should be explored to make this process more efficient and practical for ontology engineers, as it plays an essential role in the ontology design phase.

The methodology proposed in previous work, Ahmad *et al.* [16], was an initial attempt to design a methodology for designing a domain ontology for the WM domain. The improved methodology has included several important phases such as the estimation of human resources, re-engineering and re-using of resources, collaborative ontology construction, the conceptualization of both endurants and perdurants, localization of an ontology, ontology integration and merging, support for interoperability, versioning of an ontology, and ontology population.

The experience from developing the OntoWM ontology exhibited that using the proposed methodology with DSR principles in the ontology development process produced the following assistances: reducing the difficulty of ontology development activities such as conceptualization, improving communication with the assistance of collaborative construction between ontology engineers and domain experts, and the continuous assessment of the project status, keeping domain experts involved during ontology development, focusing on the most important requirements, and the ability to respond to change knowledge rapidly. The proposed methodology will assist in the estimation of human resources, re-engineering and re-using of resources, collaborative ontology construction, ontology construction of both endurants and perdurants, localization of an ontology, ontology integration and merging, support for interoperability, versioning of an ontology, and ontology population.

VI. CONCLUSION

The development of an ontology is a complex, tedious, and time-consuming task, and this task requires a well-designed methodology. In this article, an ontology development methodology, rooted in DSR has been proposed. A comparative evaluation with existing ontology engineering methodologies based on the C1-C16 criteria has been presented. Results showed that the proposed methodology is rooted in DSR and will assist in the better conceptualization of IWs domains. The strength of this methodology lies in its ability to be customized to fit several factors, including ontology complexity, a domain of interest, and ontology size. The proposed methodology will assist in the estimation of human resources, re-engineering and re-using of resources, collaborative ontology construction, ontology construction of both endurants and perdurants, localization of an ontology, ontology integration and merging, support for interoperability, versioning of an ontology, and ontology population. The proposed methodology was applied to design an OntoWM domain ontology for the WM domain. The experience from building OntoWM ontology indicated its applicability and a high degree of acceptance by ontology engineers. Moreover, it showed that the utilization of DSR methodology simplifies the implementation of the ontology development activities. Finally, the validation of the proposed methodology using SmartBinAnalytics, an ontology-based web application, is presented. It is recommended as future work to adopt the proposed methodology in up-coming ontology development and ontology-based applications, to obtain additional validation cases.

ACKNOWLEDGMENT

The authors would like to thank the Ministry of Higher Education (MOHE) and Universiti Kebangsaan Malaysia (UKM).

REFERENCES

[1] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowl. Acquisition*, vol. 5, no. 2, pp. 199–220, Jun. 1993, doi: 10.1006/knac.1993.1008.

[2] S.-H. Liao, C.-C. Chen, and D.-C. Hu, "The role of knowledge sharing and LMX to enhance employee creativity in theme park work team: A case study of taiwan," *Int. J. Contemp. Hospitality Manage.*, vol. 30, no. 5, pp. 2343–2359, May 2018, doi: 10.1108/IJCHM-09-2016-0522.

[3] M. N. Ahmad, N. H. Zakaria, and D. Sedera, "Ontology-based knowledge management for enterprise systems," *Int. J. Enterprise Inf. Syst.*, vol. 7, no. 4, pp. 64–90, Oct. 2011, doi: 10.4018/jeis.2011100104.

[4] T. Tudorache, "Ontology engineering: Current state, challenges, and future directions," *Semantic Web*, vol. 11, no. 1, pp. 1–14, 2019, doi: 10.3233/SW-190382.

[5] A. Sattar, "Learning ontology from relational database: An automatic ontology construction approach," 2013.

[6] M. Pasha and A. Sattar, "Building domain ontologies from relational database using mapping rules," *Int. J. Intell. Eng. Syst.*, vol. 5, no. 1, pp. 20–27, Mar. 2012, doi: 10.22266/ijies2012.0331.03.

[7] M. Kultsova, R. Rudnev, A. Anikin, and I. Zhukova, "An ontology-based approach to intelligent support of decision making in waste management," in *Proc. 7th Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, Jul. 2016, pp. 1–6, doi: 10.1109/IISA.2016.7785401.

[8] I.-H. Chou and C.-F. Fan, "An agent-based national radioactive waste management framework design," *Prog. Nucl. Energy*, vol. 52, no. 5, pp. 470–480, Jul. 2010, doi: 10.1016/j.pnucene.2009.10.003.

[9] E. Sanfilippo, Y. Kitamura, and R. Young, "Formal ontologies in manufacturing," *Appl. Ontol.*, vol. 14, no. 2, pp. 1–7, 2019, doi: 10.3233/AO-190209.

[10] S. Khantong and M. N. Ahmad, "An ontology for sharing and managing information in disaster response: In flood response usage scenarios," *J. Data Semantics*, vol. 9, no. 1, pp. 39–52, Mar. 2020, doi: 10.1007/s13740-019-00110-6.

[11] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock, "Gene ontology: Tool for the unification of biology," *Nature Genet.*, vol. 25, no. 1, pp. 25–29, May 2000, doi: 10.1038/75556.

[12] M. Pasha and A. Sattar, "Comparative analysis of ontology construction approaches from relational databases," in *Proc. 4th Int. Conf. Intell. Netw. Intell. Syst. (ICINIS)*, Nov. 2011, pp. 25–28, doi: 10.1109/ICINIS.2011.14.

[13] R. M. Colomb and M. N. Ahmad, "A perdurant ontology for interoperating information systems based on interlocking institutional worlds," *Appl. Ontol.*, vol. 5, no. 1, pp. 47–77, 2010.

[14] R. M. Colomb and M. N. Ahmad, "Merging ontologies requires interlocking institutional worlds," *Appl. Ontol.*, vol. 2, no. 1, pp. 1–12, 2007.

[15] Y. Sure-Vetter, S. Staab, and R. Studer, "Ontology engineering methodology," in *Handbook on Ontologies*, S. Steffen and S. Rudi, Eds. Berlin, Germany: Springer, 2009, pp. 135–152.

[16] M. N. Ahmad, K. B. A. Badr, E. Salwana, N. H. Zakaria, Z. Tahar, and A. Sattar, "An ontology for the waste management domain," in *Proc. 22nd Pacific Asia Conf. Inf. Syst.*, 2018, p. 12.

[17] B. Fawei, J. Pan, M. Kollingbaum, and A. Wyner, "A methodology for a criminal law and procedure ontology for legal question answering," in *Proc. 8th Joint Int. Conf. JIST*, 2018, pp. 198–214, doi: 10.1007/978-3-030-04284-4_14.

[18] S. John, N. Shah, and C. Stewart, "Toward a software centric approach for ontology development: Novel methodology and its application," in *Proc. Int. Conf. e-Bus. Eng. (ICEBE)*, 2018, pp. 139–146, doi: 10.1109/ICEBE.2018.00030.

[19] J. Li and S. Alian, "Design and development of a biocultural ontology for personalized diabetes self-management of American indians," in *Proc. IEEE 20th Int. Conf. e-Health Netw., Appl. Services (Healthcom)*, Sep. 2018, pp. 1–7, doi: 10.1109/HealthCom.2018.8531107.

[20] M. C. Suárez-Figueroa, A. Gómez-Pérez, and M. Fernández-López, "The NeOn methodology framework: A scenario-based methodology for ontology development," *Appl. Ontol.*, vol. 10, no. 2, pp. 107–145, Sep. 2015, doi: 10.3233/AO-150145.

[21] A. Abdelghany, N. Darwish, and H. Hefni, "An agile methodology for ontology development," *Int. J. Intell. Eng. Syst.*, vol. 12, no. 2, pp. 170–181, Apr. 2019, doi: 10.22266/ijies2019.0430.17.

[22] A. A. Alsanad, A. Chikh, and A. Mirza, "A domain ontology for software requirements change management in global software development environment," *IEEE Access*, vol. 7, pp. 49352–49361, 2019, doi: 10.1109/ACCESS.2019.2909839.

[23] K. Jacksi, "Design and implementation of E-campus ontology with a hybrid software engineering methodology," *Sci. J. Univ. Zakho*, vol. 7, no. 3, pp. 95–100, Sep. 2019, doi: 10.25271/sjuoz.2019.7.3.613.

[24] A. Yunianta, A. Hoirul, A. Satria, A. Bramantoro, I. Syamsuddin, N. Yusof, A. Omran, and K. Alsubhi, "OntoDI: The methodology for ontology development on data integration," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 1, pp. 160–168, 2019, doi: 10.14569/IJACSA.2019.0100121.

[25] G. Brusa, M. Laura Caliusco, and O. Chiotti, "Towards ontological engineering: A process for building a domain ontology from scratch in public administration," *Expert Syst.*, vol. 25, no. 5, pp. 484–503, Oct. 2008.

[26] N. Trokanas, L. Koo, and F. Cecelja, "Toward a methodology for reusable ontology engineering: Application to the process engineering domain," in *Computer Aided Chemical Engineering*, vol. 43, A. Friedl, J. J. Klemeš, S. Radl, P. S. Varbanov, and T. Wallek, Eds. Amsterdam, The Netherlands: Elsevier, 2018, pp. 471–476.

[27] L. Zhang, H. Chen, Y. Zhang, and T. Chen, "Fine construction of HIV protein ontology," in *Proc. 3rd Int. Conf. Inf. Syst. Eng. (ICISE)*, May 2018, pp. 82–88, doi: 10.1109/ICISE.2018.00023.

[28] A. Saad and S. Shaharin, "The methodology for ontology development in lesson plan domain," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 4, pp. 1–16, 2016, doi: 10.14569/IJACSA.2016.070472.

[29] A. De Nicola and M. Missikoff, "A lightweight methodology for rapid ontology engineering," *Commun. ACM*, vol. 59, no. 3, pp. 79–86, Feb. 2016, doi: 10.1145/2818359.

[30] B. Dutta, U. Chatterjee, and D. P. Madalli, "YAMO: Yet another methodology for large-scale faceted ontology construction," *J. Knowl. Manage.*, vol. 19, no. 1, pp. 6–24, Feb. 2015, doi: 10.1108/JKM-10-2014-0439.

[31] M. R. Bautista-Zambrana, "Methodologies to build ontologies for terminological purposes," *Procedia-Social Behav. Sci.*, vol. 173, pp. 264–269, Feb. 2015, doi: 10.1016/j.sbspro.2015.02.063.

[32] A. Sattar, E. Salwana, M. Nazir, M. Ahmad, and A. Kamil, "Comparative analysis of methodologies for domain ontology development: A systematic review," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 5, pp. 99–108, 2020, doi: 10.14569/IJACSA.2020.0110515.

[33] A. Sattar, M. N. Ahmad, E. Salwana, A. K. Mahmood, and M. I. M. Ismail, "Issues in designing ontology for waste management: A systematic review," *Test Eng. Manage.*, vol. 82, pp. 11889–11897, Jan./Feb. 2020.

[34] E. Blomqvist, V. Presutti, E. Daga, and A. Gangemi, "Experimenting with extreme design," in *Proc. Int. Conf. Knowl. Eng. Knowl. Manage.*, 2010, pp. 120–134.

[35] M. C. Suárez-Figueroa, A. Gómez-Pérez, and M. Fernández-López, "The NeOn methodology for ontology engineering," in *Ontology Engineering in a Networked World*, M. C. Suárez-Figueroa, A. Gómez-Pérez, E. Motta, and A. Gangemi, Eds. Berlin, Germany: Springer, 2012, pp. 9–34.

[36] E. Simperl, C. Tempich, and Y. Sure-Vetter, "A cost estimation model for ontology engineering," in *Proc. Int. Semantic Web Conf.*, vol. 4273, 2006, pp. 625–639.

[37] A. Gomez-Perez, M. Fernández-López, and O. Corcho, *Ontological Engineering: With Examples From the Areas of Knowledge Management, e-Commerce and the Semantic Web*, 1st ed. London, U.K.: Springer-Verlag, 2004.

[38] M. Gawich, A. Badr, A. Hegazy, and H. Ismail, "A methodology for ontology building," *Int. J. Comput. Appl.*, vol. 56, no. 2, pp. 39–45, Oct. 2012, doi: 10.5120/8867-2834.

[39] K. I. Kotis, G. A. Vouros, and D. Spiliotopoulos, "Ontology engineering methodologies for the evolution of living and reused ontologies: Status, trends, findings and recommendations," *Knowl. Eng. Rev.*, vol. 35, pp. 1–34, Jan. 2020, doi: 10.1017/S0269888920000065.

[40] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A design science research methodology for information systems research," *J. Manage. Inf. Syst.*, vol. 24, no. 3, pp. 45–77, Dec. 2007.

[41] M. N. Ahmad, K. B. A. Badr, R. M. Colomb, and R. Ibrahim, "Ontology-based applications in information systems research: Through the lens of design science research methodology," in *Proc. Pacific Asia Conf. Inf. Syst. (PACIS)*, 2012, p. 177.

[42] A. Hevner, S. T. March, J. Park, and S. Ram, ''Design science in information systems research,'' *Manage. Inf. Syst. Quart.*, vol. 28, no. 1, pp. 75–105, 2004.

[43] G. L. Geerts, ''A design science research methodology and its application to accounting information systems research,'' *Int. J. Accounting Inf. Syst.*, vol. 12, no. 2, pp. 142–151, Jun. 2011, doi: 10.1016/j.accinf.2011.02.004.

[44] M. Suárez-Figueroa, A. Gomez-Perez, and B. V. Terrazas, ''How to write and use the ontology requirements specification document,'' in *Proc. Int. Conf. Move Meaningful Internet Syst.*, vol. 1, 2009, pp. 966–982, doi: 10.1007/978-3-642-05151-7_16.

[45] M. Uschold and M. L. King, ''Toward a methodology for building ontologies,'' in *Innovations in E-Learning, Instruction Technology, Assessment, and Engineering Education*, M. Iskander, Ed. Dordrecht, The Netherlands: Springer, 1995, pp. 205–210.

[46] N. Rastogi, D. P. Verma, and D. P. Kumar, ''Analyzing ontology editing tools for effective semantic information retrieval,'' *Int. J. Eng. Sci. Res. Technol.*, vol. 6, no. 5, pp. 40–47, 2017.

[47] H. Knublauch *et al.*, ''The protege OWL experience,'' in *OWL: Experiences and Directions*. 2005.

[48] T. Slimani, ''Ontology development: A comparing study on tools, languages and formalisms,'' *Indian J. Sci. Technol.*, vol. 8, no. 24, pp. 1–12, Sep. 2015, doi: 10.17485/ijst/2015/v8i1/54249.

[49] H. Yazici, L. Zidek, and H. St Hill, ''A study of critical thinking and cross-disciplinary teamwork in engineering education,'' in *Women in Industrial and Systems Engineering*, A. Smith, Ed. Cham, Switzerland: Springer, 2020, pp. 185–196.

[50] E. Simperl, I. Popov, and T. Bürger, ''ONTOCOM revisited: Toward accurate cost predictions for ontology development projects,'' in *Proc. Eur. Semantic Web Conf.*, vol. 5554, 2009, pp. 248–262, doi: 10.1007/978-3-642-02121-3_21.

[51] A. L. A. Menolli, H. S. Pinto, S. S. Reinehr, and A. Malucelli, ''An incremental and iterative process for ontology building,'' in *Proc. ONTOBRAS-Ontol. Res. Brazil*, 2013, pp. 215–220.

[52] I. Y.-P. Chen, M. Agrawal, S. Horng, and D. A. Sontag, ''Robustly extracting medical knowledge from EHRs: A case study of learning a health knowledge graph,'' in *Proc. Pacific Symp. Biocomput.*, vol. 25, 2020, pp. 19–30.

[53] T. Gruber, ''Collective knowledge systems: Where the social Web meets the semantic Web,'' *J. Web Semantics*, vol. 6, no. 1, pp. 4–13, Feb. 2008, doi: 10.1016/j.websem.2007.11.011.

[54] P. Lévy, ''From social computing to reflexive collective intelligence: The IEML research program,'' *Inf. Sci.*, vol. 180, no. 1, pp. 71–94, Jan. 2010, doi: 10.1016/j.ins.2009.08.001.

[55] H. Halpin, V. Robu, and H. Shepherd, ''The complex dynamics of collaborative tagging,'' in *Proc. Int. Conf. World Wide Web*, 2007, pp. 211–220, doi: 10.1145/1242572.1242602.

[56] K. Bosch and A. Bronkhorst, ''Human-AI cooperation to benefit military decision making,'' in *Proc. NATO IST Spec. Meeting Big Data Artif. Intell. Mil. Decis. Making*, 2018.

[57] G. Guizzardi, G. Wagner, J. P. A. Almeida, and R. S. S. Guizzardi, ''Towards ontological foundations for conceptual modeling: The unified foundational ontology (UFO) story,'' *Appl. Ontol.*, vol. 10, nos. 3–4, pp. 259–271, Dec. 2015, doi: 10.3233/AO-150157.

[58] W. Degen, B. Heller, H. Herre, and B. Smith, ''GOL: Toward an axiomatized upper-level ontology,'' in *Proc. FOIS*, Ogunquit, ME, USA, 2001, pp. 34–46.

[59] E. Wallace, D. Kiritsis, B. Smith, and C. Will, ''The industrial ontologies foundry proof-of-concept project,'' in *Proc. Int. Conf. Adv. Prod. Manage. Syst. (APMS)*, vol. 536, 2018, pp. 402–409.

[60] J. Raad and C. Cruz, ''A survey on ontology evaluation methods,'' in *Proc. 7th Int. Joint Conf. Knowl. Discovery, Knowl. Eng. Knowl. Manage.*, 2015, pp. 179–186, doi: 10.5220/0005591001790186.

[61] G. Guizzardi, N. Guarino, and J. Almeida, ''Ontological considerations about the representation of events and endurants in business models,'' in *Proc. Int. Conf. Bus. Process Manage.*, 2016, pp. 20–36, doi: 10.1007/978-3-319-45348-4_2.

[62] M. N. Ahmad, R. M. Colomb, and S. Sadiq, ''UML-based DEMO profiles as metaconcepts for interlocking institutional worlds,'' in *Proc. Pacific Asia Conf. Inf. Syst.*, 2008, p. 4.

[63] R. Villarroel, E. Fernández-Medina, M. Piattini, and J. Trujillo, ''A UML 2.0/OCL extension for designing secure data warehouses,'' *J. Res. Pract. Inf. Technol.*, vol. 38, no. 1, pp. 31–43, 2006.

[64] R. A. de Falbo, ''SABiO: Systematic approach for building ontologies,'' in *Proc. 1st Joint Workshop ONTO.COM/ODISE Ontologies Conceptual Modeling Inf. Syst. Eng.*, 2014, pp. 1–14.

[65] A. Adamou and V. Presutti, ''Customizing your interaction with Kali-ma,'' in *Ontology Engineering in a Networked World*, M. C. Suárez-Figueroa, A. Gómez-Pérez, E. Motta, and A. Gangemi, Eds. Berlin, Germany: Springer, 2012, pp. 319–342.

[66] M. Horridge, R. S. Gonçalves, C. I. Nyulas, T. Tudorache, and M. A. Musen, ''WebProtégé: A cloud-based ontology editor,'' in *Proc. World Wide Web Conf.*, 2019, pp. 686–689, doi: 10.1145/3308560.3317707.

[67] Internet World Statistics. (2020). *Number of Internet Users by Language*. Accessed: Mar. 30, 2020. [Online]. Available: https://www.internetworldstats.com/stats7.htm

[68] E. Hau and M. Aparício, ''Software internationalization and localization in Web based ERP,'' in *Proc. 26th Annu. ACM Int. Conf. Design Commun. (SIGDOC)*, 2008, pp. 175–180, doi: 10.1145/1456536.1456570.

[69] J. Ariza, F. Larrinaga, and E. Curry, ''A unified semantic ontology for energy management applications,'' in *Proc. 2nd Int. Workshop Ontol. Modularity, Contextuality, Evol.*, 2017, pp. 86–97.

[70] H. S. Pinto and J. P. Martins, ''A methodology for ontology integration,'' in *Proc. Int. Conf. Knowl. Capture (K-CAP)*, 2001, pp. 131–138, doi: 10.1145/500737.500759.

[71] P. Petrov, M. Krachunov, E. Todorovska, and D. Vassilev, ''An intelligent system approach for integrating anatomical ontologies,'' *Biotechnol. Biotechnol. Equip.*, vol. 26, no. 4, pp. 3173–3181, Jan. 2012.

[72] M. Fernandez, C. Overbeeke, M. Sabou, and E. Motta, ''What makes a good ontology? A case-study in fine-grained knowledge reuse,'' in *Proc. 4th Asian Conf. Semantic Web*, 2009, pp. 61–75, doi: 10.1007/978-3-642-10871-6_5.

[73] I. Cantador, M. Fernandez, and P. Castells, ''Improving ontology recommendation and reuse in WebCORE by collaborative assessments,'' in *Proc. Workshop Social Collaborative Construct. Structured Knowl. 16th Int. World Wide Web Conf.*, 2007, pp. 1–11.

[74] V. Jain, ''Evaluation and validation of ontology using Protege tool,'' *Int. J. Res. Eng. Technol.*, vol. 4, no. 4, pp. 2347–4599, 2019.

[75] H. Zhu, D. Liu, I. Bayley, A. Aldea, Y. Yang, and Y. Chen, ''Quality model and metrics of ontology for semantic descriptions of Web services,'' *Tsinghua Sci. Technol.*, vol. 22, no. 3, pp. 254–272, Jun. 2017, doi: 10.23919/TST.2017.7914198.

[76] D. Looser, H. Ma, and K.-D. Schewe, ''Using formal concept analysis for ontology maintenance in human resource recruitment,'' in *Proc. Asia–Pacific Conf. Conceptual Modelling*, 2013, pp. 61–68.

[77] D. Garijo, ''WIDOCO: A wizard for documenting ontologies,'' in *Proc. Int. Semantic Web Conf.*, 2017, pp. 94–102.

[78] J. J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, and K. Wilkinson, ''Jena: Implementing the semantic Web recommendations,'' in *Proc. 13th Int. World Wide Web Conf. Alternate Track Papers Posters (WWW Alternate)*, 2004, pp. 836–854, doi: 10.1145/1013367.1013381.

ABDUL SATTAR received the bachelor's degree in information technology and the M.S. degree in computer science from Bahauddin Zakariya University, Pakistan, in 2005 and 2012, respectively. He has authored seven publications/conferences. He is currently pursuing the Ph.D. degree, under the supervision of Dr. Mohammad Nazir Ahmad, with Universiti Kebangsaan Malaysia (UKM), Malaysia. He is currently working as a Visiting Lecturer at the Khwaja Fareed University of Engineering & Information Technology, Rahim Yar Khan, Pakistan. He has served as a Visiting Lecturer at Bahauddin Zakariya University from December 2017 to June 2012. He also served as a Lecturer at the Government Post Graduate College Rajanpur, Pakistan.

MOHAMMAD NAZIR AHMAD received the bachelor's degree in information technology and the M.Sc. degree in information systems in industrial computing from Universiti Kebangsaan Malaysia (UKM), in 1998 and 2002, respectively, and the Ph.D. degree in the field of information technology from the University of Queensland (UQ), Brisbane, Australia, in 2009. He is currently an Associate Professor at UKM.

ELY SALWANA MAT SURIN received the B.Sc. (Hons.) in management information system and the master's degree in information technology from the Universiti Teknologi Malaysia (UTM), Malaysia, and the Ph.D. degree in computer science from the University of Malaya (UM), Malaysia. She is currently a Senior Fellow at Universiti Kebangsaan Malaysia (UKM).

AHMAD KAMIL MAHMOOD received the B.Sc. (Hons.) degree in actuarial science and the M.Sc. degree in actuarial science and statistics from the University of Iowa, in 1986 and 1988, respectively, and the Ph.D. degree from Salford University, U.K., in 2005. He has over 20 years of experience in various fields of information systems. He has taught undergraduate and postgraduate programs and managed numerous R&D and consultancy projects successfully.