UDC 004.91
DOI https://doi.org/10.32782/IT/2023-1-9

Mariia LYASHKEVYCH
Assistant Professor at the System Design Department, Ivan Franko National University of Lviv, 50 Drahomanova str., Lviv, Ukraine, 79005, Ukraine, mariia.liashkevych@lnu.edu.ua
ORCID: *0000-0002-9655-036X*
Scopus Author ID*: 56007435700*

Vasyl LYASHKEVYCH
Candidate of Technical Sciences, Associate Professor of the System Design Department, Ivan Franko National University of Lviv, 50 Drahomanova str., Lviv, Ukraine, 79005, vasyl.liashkevych@lnu.edu.ua
ORCID: *0000-0003-2810-6061*
Scopus Author ID*: 35317759200*

Roman SHUVAR
Candidate of Physical and Mathematical Sciences, Associate Professor, Chairperson of the System Design Department, Ivan Franko National University of Lviv, 50 Drahomanova str., Lviv, Ukraine, 79005, roman.shuvar@lnu.edu.ua
ORCID: *0000-0001-6768-4695*
Scopus Author ID*: 8521299500*

To cite this article: Liashkevych, M., Liashkevych, V., Shuvar, R. (2023). Proiektuvannia ontolohii rozmitky dlia system keruvannia kontentom [Markup ontology design for a content management system]. *Information Technology: Computer Science, Software Engineering and Cyber Security,* 1, 67–73, doi: https://doi.org/10.32782/IT/2023-1-9

MARKUP ONTOLOGY DESIGN FOR A CONTENT MANAGEMENT SYSTEM

Low standards of written content and poor standard of search engine optimization (SEO) are the main problems in the content management area, especially, when we are talking about huge sitemaps with millions of pages. For each webpage, the same content can be represented for two targets: for search robots which work with special markup schemas and for users. As usual, the information for search robots is limited by keywords which cannot describe clearly the meaning of textual messages on the webpage thanks for different reasons. The problem here is content meaning synchronization for both users and for search robots across all the millions of pages. The different components of the webpage have their own style and could contain something meaningful context which we are going to synchronize with page meaning too.

The most famous markup schema was created by leaders of search engines in the market, namely: Google, Yahoo, Microsoft and others (Schema.org, 2022). It helps the search engine robots better understand the content of web pages. The schema can be extended by demand using a well-documented extension model including vocabularies which describe entities and relationships between them.

We are going to resolve the problems of content synchronization and its single representation form for users and for search robots, therefore, the text generation techniques are not considered in the paper. Sure, a matter of control of the content through millions of pages is so hard to resolve even using modern content management systems (CMS) systems. Thus, we resolve the content management problems by installing semantics for webpage markup schemas, webpages and their content using a single knowledge representation in markup ontology. The proposed ontology-based approach is able to synchronize the meaning between content for users and for search robots of webpages and could be implemented as an extra plugin for CMS.

Key words: *content management system, ontology, markup, search robots.*

Марія ЛЯШКЕВИЧ
асистент кафедри системного програмування, Львівський національний університет імені Івана Франка, вул. Драгоманова, 50, м. Львів, Україна, 79005
***ORCID:** 0000-0002-9655-036X*
***Scopus Author ID**: 56007435700*

Василь ЛЯШКЕВИЧ
кандидат технічних наук, доцент, доцент кафедри системного проєктування, Львівський національний університет імені Івана Франка, вул. Драгоманова, 50, м. Львів, Україна, 79005, vasyl.liashkevych@lnu.edu.ua
***ORCID:** 0000-0003-2810-6061*
***Scopus Author ID**: 35317759200*

Роман ШУВАР
кандидат фізико-математичних наук, доцент, завідуючий кафедрою системного проєктування, Львівський національний університет імені Івана Франка, вул. Драгоманова, 50, м. Львів, Україна, 79005, roman.shuvar@lnu.edu.ua
***ORCID:** 0000-0001-6768-4695*
***Scopus Author ID**: 8521299500*

Бібліографічний опис статті: Ляшкевич, М. Ляшкевич, В., Шувар, Р. (2023). Проєктування онтології розмітки для систем керування контентом. *Information Technology: Computer Science, Software Engineering and Cyber Security,* 1, 67–73, doi: https://doi.org/10.32782/IT/2023-1-9

ПРОЄКТУВАННЯ ОНТОЛОГІЇ РОЗМІТКИ ДЛЯ СИСТЕМ КЕРУВАННЯ КОНТЕНТОМ

Низькі стандарти написання контенту та низькі стандарти пошукової оптимізації (SEO) є головними проблемами в області управління контентом, особливо коли ми говоримо про величезні карти сайту з мільйонами сторінок. Для кожної веб-сторінки однаковий контент може бути представлений для двох цілей: для пошукових роботів, які працюють зі спеціальними схемами розмітки, і для користувачів. Зазвичай, інформація для пошукових роботів обмежена ключовими словами, які з різних причин не можуть чітко описати зміст текстових повідомлень на веб-сторінці. Проблема полягає в синхронізації вмісту як для користувачів, так і для пошукових роботів на всіх мільйонах сторінок. Різні компоненти веб-сторінки мають свій власний стиль і можуть містити де-який контент, що ми також будемо узгоджувати зі змістом сторінки.

Найвідоміша схема розмітки була створена лідерами пошукових систем на ринку, а саме: Google, Yahoo, Microsoft та іншими (Schema.org, 2022). Це допомагає роботам пошукових систем краще розуміти контент веб-сторінок. Схему можна розширити за потребою, використовуючи добре задокументовану модель розмітки, включаючи словники, які описують сутності та зв'язки між ними.

Ми збираємося вирішити проблеми синхронізації контенту та єдиної форми його представлення для користувачів і для пошукових роботів, тому методи генерації тексту в даній роботі не розглядаються. Звичайно, питання контролю вмісту мільйонів сторінок дуже важко вирішити навіть за допомогою сучасних систем керування контентом (CMS). Таким чином, ми вирішуємо проблеми керування контентом, встановлюючи семантику для схем розмітки веб-сторінок, компонентів веб-сторінок та їхнього контенту за допомогою єдиного представлення знань в онтології розмітки. Запропонований підхід на основі онтології здатний синхронізувати смисл контенту для користувачів і пошукових роботів веб-сторінок і може бути реалізований як додатковий плагін для CMS.

***Ключові слова**: система керування контентом, онтології, розмітка, пошукові роботи.*

Webpage as an object of content representation. The webpage (fig. 1) is the main place for content representation and doesn't matter how big the site is. There are a lot of different approaches how to choose the right style in making the UI/UX design and as a consequence, all the pages of the site followed it. Looking at a webpage as an object of content representation we are aiming to define the webpage's structural components first and then inspect the semantic relationships between them and the texts which should be located there.

Generally, manual content management is a big problem when we work with quite huge sites which have millions of pages. Even content management systems (CMS) are not able to do this efficiently using their special tools because they required the control from content manager side. Let's look at CMS deeply (Barker, 2016):

– Some of the CMS have open-source options and website builders capabilities.

– CMS helps users create, format, edit, and publish content. This may include support for media, written content, or drop quotes based on the CMS, but the core idea is that you can make and publish some sort of content. The best content management system is the one that makes users comfortable when publishing.

– CMS stores the content in a database, so, the content always gets logged inside a database.

– One CMS may have unique user permissions, while another might allow for specific editor, author, and admin roles.

– CMS presents the content. This usually happens on the front end of a live website, but some content management systems allow for private or even offline publication.

There are some challenges in designing the ontology-based tool for a content management system, below.

As mentioned before, the common CMS has its own taxonomies and dictionaries for content management purposes so we should represent the markup ontology in that format or prepare the right plugins which can do that automatically.

The module of a webpage as an object of content representation (fig. 1) definition must define a webpage structure based on analysis of the CSS styles, JS functions, HTML blocks and other present components including DOM.

Content and keywords definition could be resolved by machine learning algorithms which extract the keywords from the content of each webpage. Meanwhile, we should consider the ability to manage a set of webpage keywords.

Based on the webpage structure and content definition we have to build the data model (Lyashkevych, 2009) which are describing the feature of web page structure and content specification. It's a very important part of the job because we have to connect web page structure and content meaning (Lyashkevych, 2013) with pictured information as well (Zhang, 2022; Yuan, 2022).

Data management tools assignments. Generally, the data management tools on the high level of the architecture definition can be seen in Fig. 2.

Data interpretation and classification task of content. The classification task has to guarantee the best webpage structure according to the meaning of the proposed content but we still have a question regards to styling here. The webpage style creates a special atmosphere and focuses on the user and these sentiments we can extract by computer vision (CV) way if we represent the page as an image. Having the defined architecture of the neural network, we can choose the comfortable interpretation of the data. This task is very essential because we need to interpret the web page structure and its content in a comfortable form for the further learning process.

Data and sets of entities design. The management tools can update the content, styles and other components including ontology. Let's consider a set of data models for markup ontology.

The schema.org proposed a quite big data model which includes already described sets of entities and relationships between them. Using the existing data model we have built the webpage model which can be seen in Fig. 3.

On par with the data interpretation, we are considering the data integration task. The difference is that the task will make an analysis of data and will propose the best place to append the needed data. This analysis has to be provided by abstract, semantic and structural levels of the ontology. At the time, we have to have a description of new data on a conceptual level because we want to manage it automatically. This description we can consider as a meaning vector too.

Generally, regardless of the used techniques, we will end up creating RDF graphs that use concepts and properties from the ontology, that describe part of the entities of interest we need for the system. For instance, a small RDF graph for each data model (tree/graph) we have. Possibly, we may

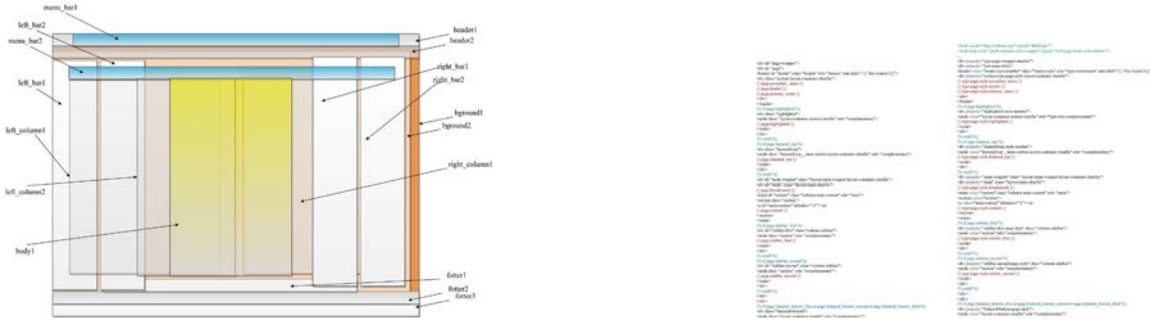

Fig. 1. Common webpage structure: a) visible components on the webpage; b) common HTML tags of the webpage

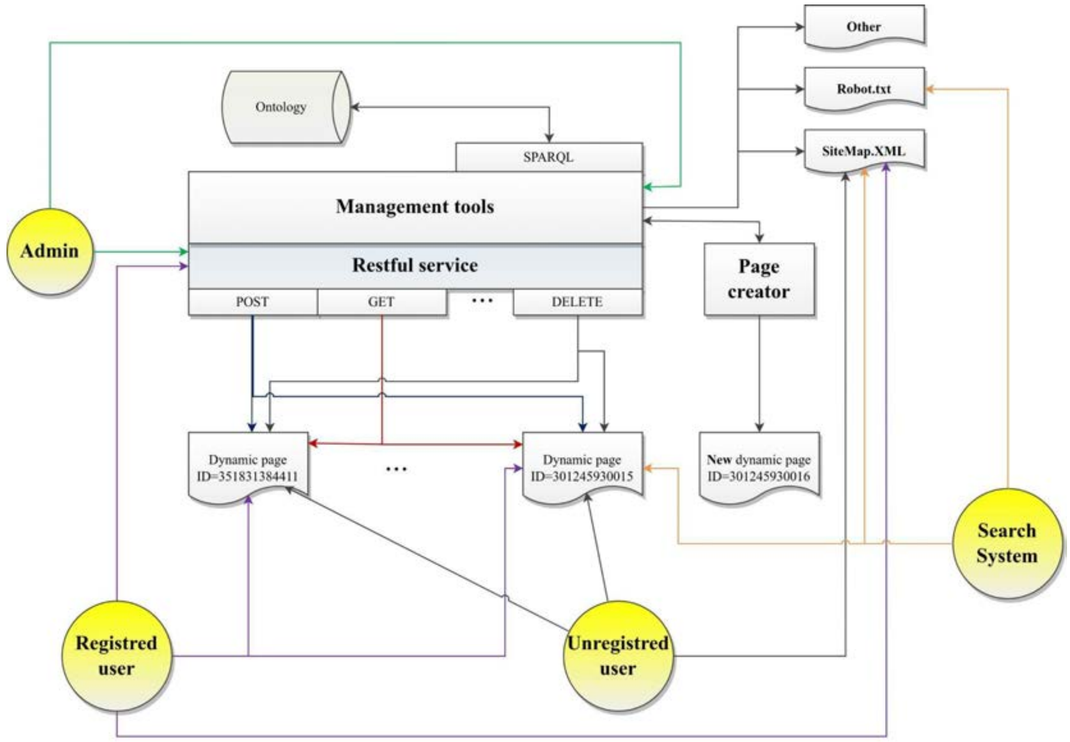

Fig. 2. High-level data management tools architecture

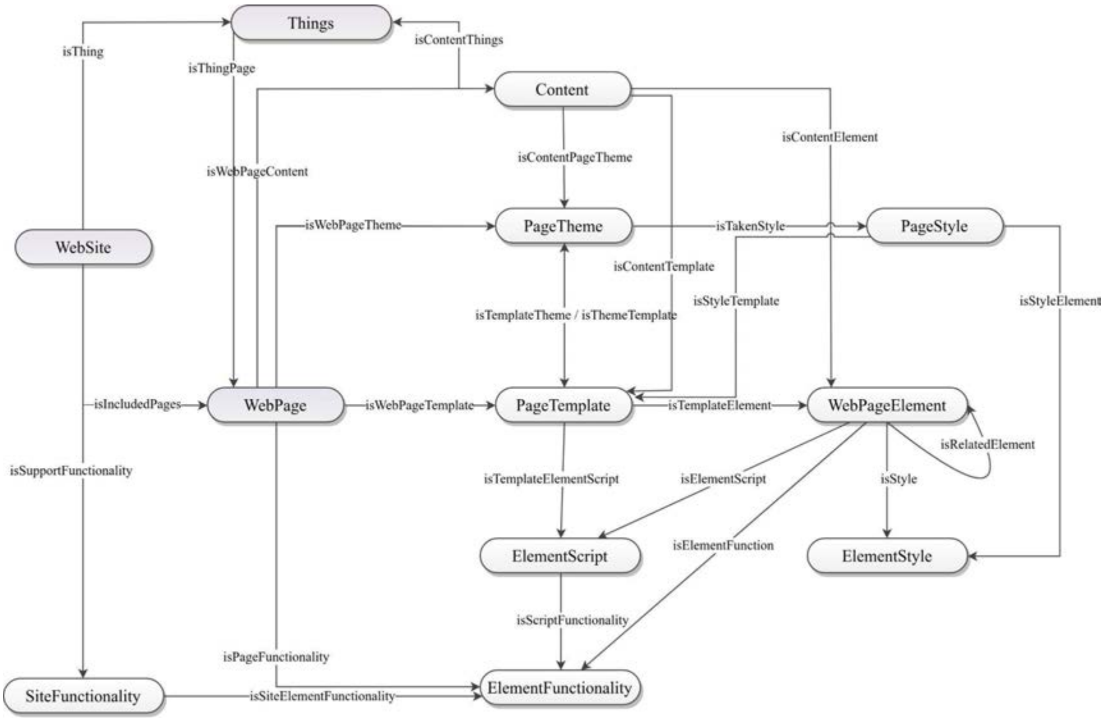

Fig. 3. Webpage data model

have to add information about entities that are not directly the topic of interest of the application, but that may be useful for managing your system, such as the organisation that takes care of some trees. So we may need to use ontologies for organisation, people, agriculture, health care, etc. We may need to devise new ontological terms that are only useful for your application or system. Then, we should make changes in the reasoner or inference engine to get the right knowledge from the ontology.

The knowledge we keep in the semantics in the markup ontology but data instances we save in the databases (fig. 4). We can easily get the right data instance by an appropriate link from RDF storage.

Usually, the ontology is built following the situational approach which involves the expert analysis of the subject domain. We used the sheme.org data model for keywords extraction and represented them in our ontology which is shown in Fig. 5. The keywords are prepared for saving in RDF format. The use case of the task is when we want to change one logic part of ontology by another don't append, only change. So, we cannot do it so easily because taxonomy reflects the

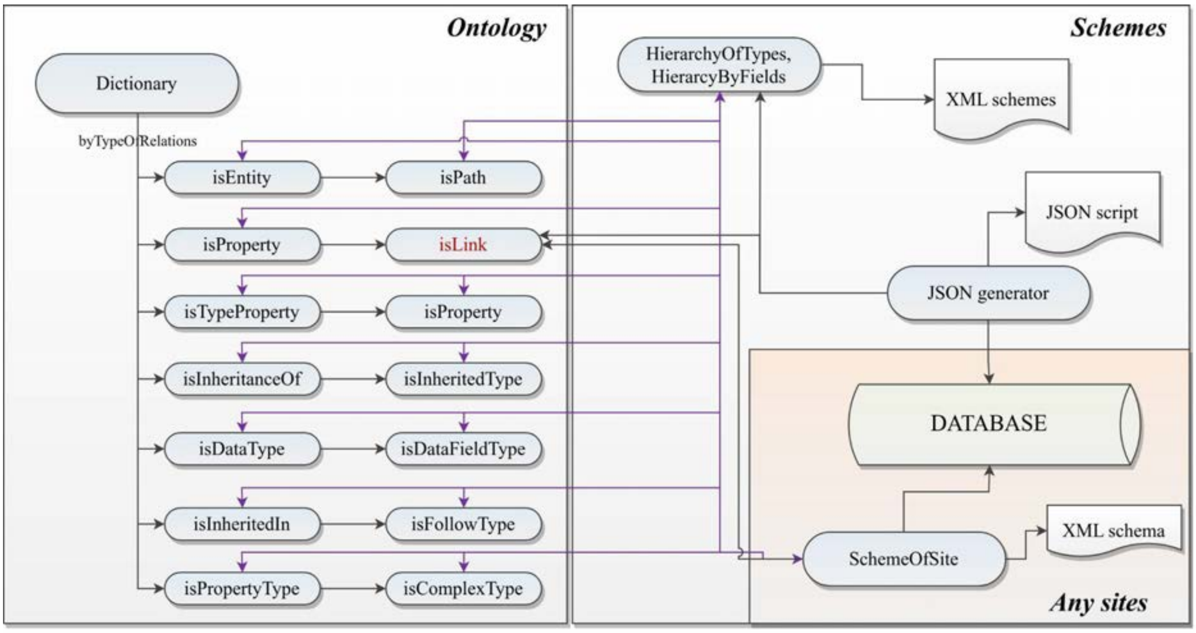

Fig. 4. Ontology with extra database

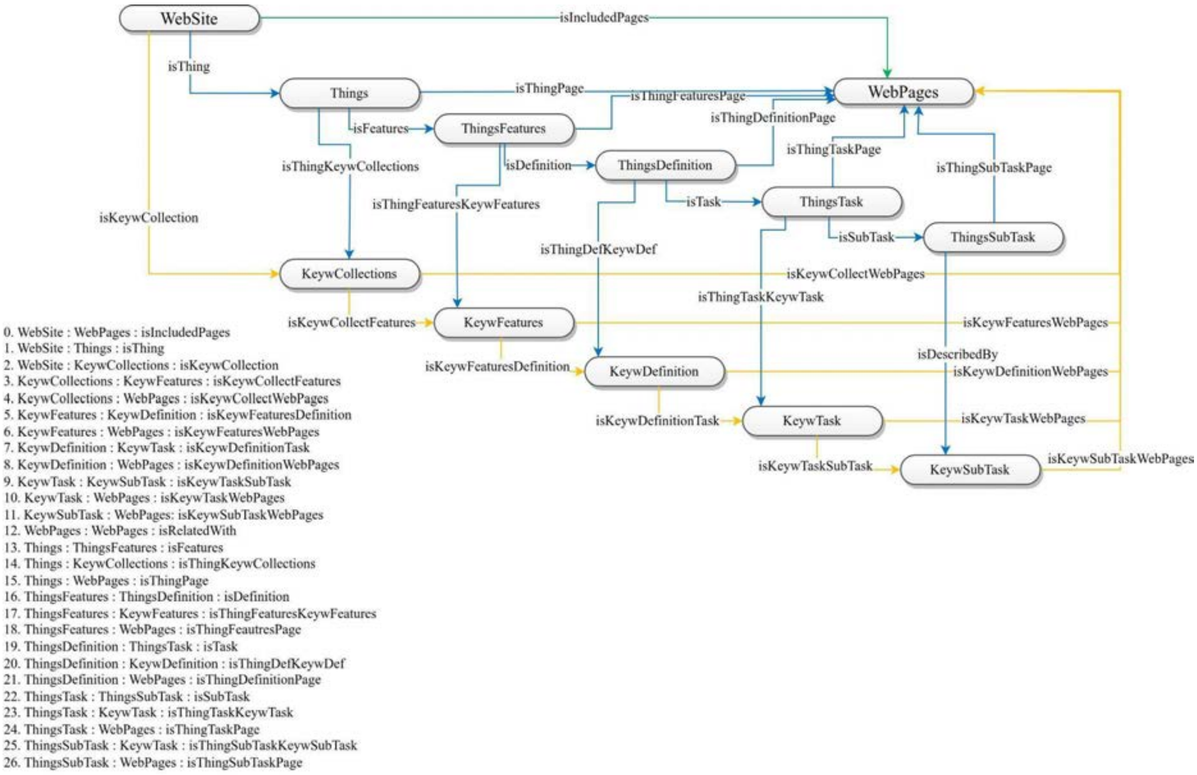
WebSite
isIncludedPages
isThing
Things
isThingPage
isThingFeaturesPage
WebPages
isFeatures
ThingsFeatures
isThingDefinitionPage
isThingTaskPage
isThingKeywCollections
isDefinition
ThingsDefinition
isThingSubTaskPage
isThingFeaturesKeywFeatures
isTask
ThingsTask
isKeywCollection
isSubTask
ThingsSubTask
isThingDefKeywDef
KeywCollections
isKeywCollectWebPages
isThingTaskKeywTask
isKeywCollectFeatures
KeywFeatures
isKeywFeaturesWebPages
isDescribedBy
isKeywFeaturesDefinition
KeywDefinition
isKeywDefinitionWebPages
isKeywDefinitionTask
KeywTask
isKeywTaskWebPages
isKeywTaskSubTask
KeywSubTask
isKeywSubTaskWebPages
0. WebSite : WebPages : isIncludedPages
1. WebSite : Things : isThing
2. WebSite : KeywCollections : isKeywCollection
3. KeywCollections : KeywFeatures : isKeywCollectFeatures
4. KeywCollections : WebPages : isKeywCollectWebPages
5. KeywFeatures : KeywDefinition : isKeywFeaturesDefinition
6. KeywFeatures : WebPages : isKeywFeaturesWebPages
7. KeywDefinition : KeywTask : isKeywDefinitionTask
8. KeywDefinition : WebPages : isKeywDefinitionWebPages
9. KeywTask : KeywSubTask : isKeywTaskSubTask
10. KeywTask : WebPages : isKeywTaskWebPages
11. KeywSubTask : WebPages: isKeywSubTaskWebPages
12. WebPages : WebPages : isRelatedWith
13. Things : ThingsFeatures : isFeatures
14. Things : KeywCollections : isThingKeywCollections
15. Things : WebPages : isThingPage
16. ThingsFeatures : ThingsDefinition : isDefinition
17. ThingsFeatures : KeywFeatures : isThingFeaturesKeywFeatures
18. ThingsFeatures : WebPages : isThingFeautresPage
19. ThingsDefinition : ThingsTask : isTask
20. ThingsDefinition : KeywDefinition : isThingDefKeywDef
21. ThingsDefinition : WebPages : isThingDefinitionPage
22. ThingsTask : ThingsSubTask : isSubTask
23. ThingsTask : KeywTask : isThingTaskKeywTask
24. ThingsTask : WebPages : isThingTaskPage
25. ThingsSubTask : KeywTask : isThingSubTaskKeywSubTask
26. ThingsSubTask : WebPages : isThingSubTaskPage

Fig. 5. Keywords model in the RDF storage

logical relations into ontology and we have to do it safely without any data and relationships losing.

Query processing is hard due to the recursive function. The query, the same as a reasoner, returns us the logical graph schema with knowledge and links to the data instances. Actually, we query the concepts (fig. 6) in the dictionary on the structural level of ontology and meta-ontology.

On the conceptual level, ontology is represented by entities, their object and data properties with semantic relationships between them and other concepts.

Website entity design. In defining a site entity we cover the site structure, its components and content. We can define the structure of the site on an abstract level of definitions that we can combine together the site functionality with its components. In general, the website entity looks like a graph schema (fig. 7). The different leaves of a graph have to describe a different part of the functionality. Here we will use the already existing results from the data modelling task.

A website entity is not so detailed and can be enhanced per demands. The main idea is controlling the content type, webpage components, keywords and content as well. As can be seen in Fig. 7, we are going to control the webpages by grouping them thematically.

Conclusions. The proposed ontology-based solution can describe the webpage's styles, content, and media content on conceptual and semantic levels. Also, ontology allows us to integrate the results from deep learning solutions.

The solution modelling shows that we can efficiently manage the content of the sites with millions of pages. Even if we use text generation techniques in the future it is possible to lose control due to differences of actual and generated content.

The proposed features of CMS as an ontology-based plugin can significantly increase the

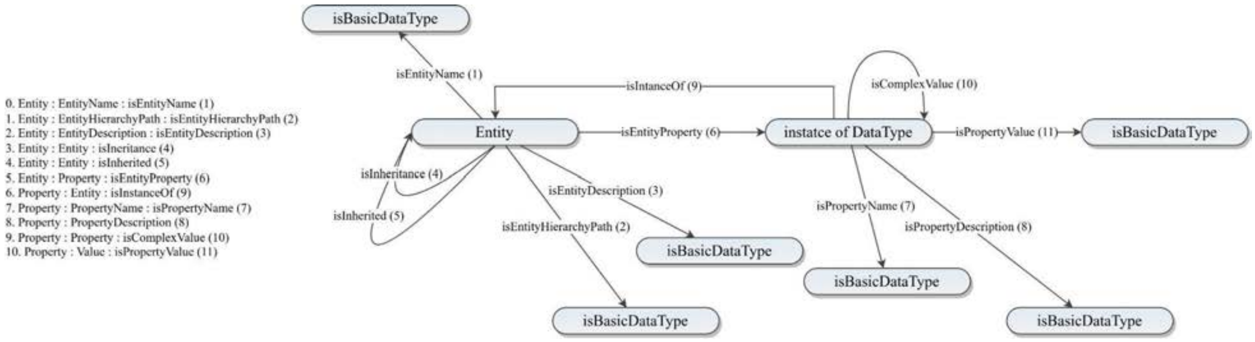
0. Entity : EntityName : isEntityName (1)
1. Entity : EntityHierarchyPath : isEntityHierarchyPath (2)
2. Entity : EntityDescription : isEntityDescription (3)
3. Entity : Entity : isIneritance (4)
4. Entity : Entity : isInherited (5)
5. Entity : Property : isEntityProperty (6)
6. Property : Entity : isInstanceOf (9)
7. Property : PropertyName : isPropertyName (7)
8. Property : PropertyDescription (8)
9. Property : Property : isComplexValue (10)
10. Property : Value : isPropertyValue (11)
isBasicDataType
isEntityName (1)
isIntanceOf (9)
isComplexValue (10)
Entity
isEntityProperty (6)
instatce of DataType
isPropertyValue (11)
isBasicDataType
isInheritance (4)
isInherited (5)
isEntityDescription (3)
isEntityHierarchyPath (2)
isPropertyName (7)
isPropertyDescription (8)
isBasicDataType
isBasicDataType
isBasicDataType
isBasicDataType

Fig. 6. Conceptual level of knowledge representation in the ontology

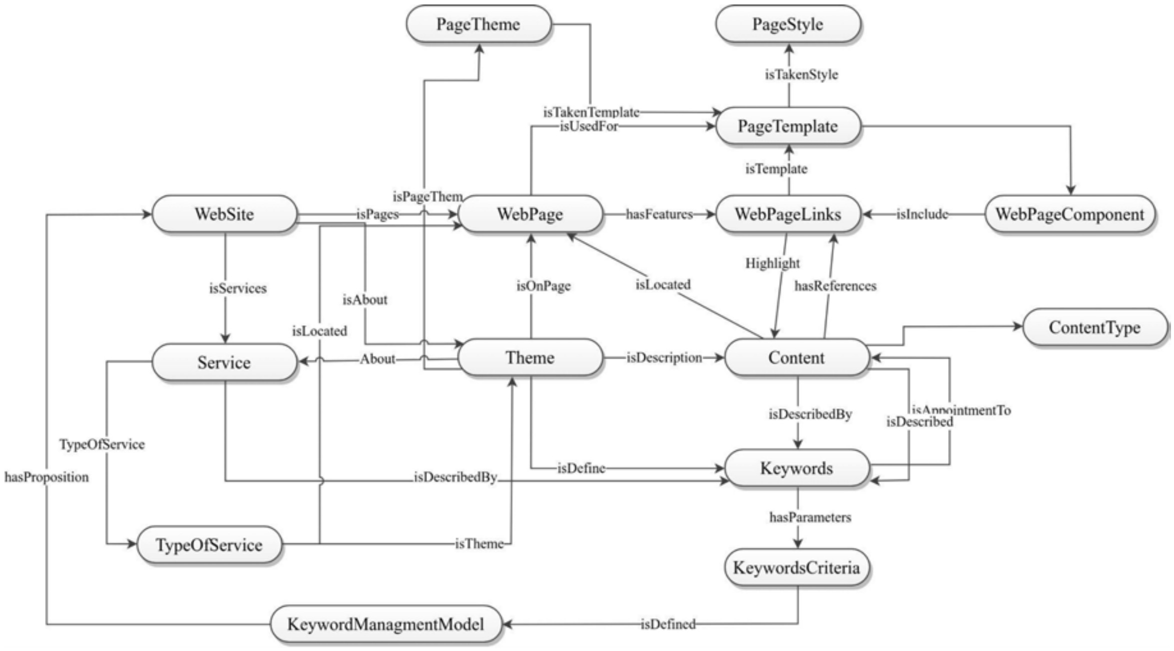

Fig. 7. A website data model for markup ontology

ability to control and manage the content on the site. Assuming, we have a branch of webpages thematically grouped and we want to add a new webpage there, for example, for a new type of product. So, we are going to generate a new webpage and put the content about a new type of product there. As a result, when we are so far from the root webpage of the site, the similarity of the generated content might be less but we keep the similarity on the same level using the markup ontology.

Using different data models in ontology, we can easily represent the same content to different needs based on an appropriate set of product rules for inference.

Content generation techniques are one of the hottest topics for our future investigation and it might be the next step for proceeding.

REFERENCES:

1. Welcome to Schema.org. 2022. URL: https://schema.org/

2. Deane Barker. Web Content Management. Systems, Features, and Best Practices: O'Reilly Media, 2016. p. 550.

3. Lyashkevych V., Oksana O., Mirosh O. Logic-Textual and Neuronet Approach to Search Information. Proceedings of the 5th IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications 21-23 September 2009, Rende (Cosenza), Italy. – P.539-543. - DOI: https://doi.org/10.1109/IDAACS.2009.5342920

4. Lyashkevych V., Olar O., Liashkevych M. Software Ontology Subject Domain Intelligence Diagnostics of Computer Means. The 7th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 12-14 September 2013, Berlin, Germany. V.2. pp. 601-606. DOI: https://doi.org/10.1109/IDAACS.2013.6662995

5. Generating news image captions with semantic discourse extraction and contrastive style-coherent learning / Zhang Z., Zhang H., Wang J., Sun Z., Yang Z. Computers and Electrical Engineering. 2022. V. 104. Part A. DOI: https://doi.org/10.1016/j.compeleceng.2022.108429

6. Syntax Customized Video Captioning by Imitating Exemplar Sentences / Yuan, Yitian, Ma, Lin, Zhu, Wenwu. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2022. V. 44, I. 12. pp. 10209-10221. DOI: https://doi.org/10.1109/TPAMI.2021.3131618