Extending intelligent content delivery in technical communication by semantics: microdocuments and content services

Wolfgang Ziegler[1*]

[1] Karlsruhe University of Applied Sciences, Faculty of Information Management and Media, 76133 Karlsruhe, Germany

Abstract. We address and develop a new concept for the dynamic delivery of topic-based content created within the domain of technical communication. Corresponding content management environments introduced within the last decades, focused so far on semantically structured and mostly XML-based information models and, more recently, on semantic metadata using taxonomies leading together to concepts of so-called intelligent content. Latest developments attempt to extend these concepts with additional explicit semantic approaches modelled and implemented, for example, by using ontologies and related technologies. In this article, we propose how content users might benefit from these semantic concepts by the delivery of sets of logically connected topics, which can be described as microdocuments (“microDocs”). This generic approach of topic assemblies might also play a role in the provisioning of content by web-services being integrated into different types of content processing and content delivery applications.

1 Introduction

The delivery of so-called intelligent information in technical communication (TC) scenarios is driven by topic-based content enriched by semantics from XML-structuring and corresponding metadata [1, 2]. Content considered in TC is created typically within component content management systems (CCMS). Dynamic user access can then be provided by content delivery portals (CDP) and corresponding interfaces [3]. While these applications allow for metadata-based facetted search, document navigation and full text search, the underlying concepts and technology can vary considerably.

One of the recently discussed enhanced concepts are ontology models. They are well-known in computer science and gained increasing interest in TC to improve model-based content creation processes or effectivity of search and delivery processes. We attempt to expand these considerations by combining topic-based content provisioning, semantics and ontology modelling in a new way. We explore semantic aspects of so-called microdocuments (microDocs) as a use-case driven content provisioning, bridging the gap between single-topic delivery and large-document search. Hereby, the need for context and content amount should be satisfied in a better way.

2 Technological background

In this section, we clarify the technological and methodological background of semantics used so far in the domain of technical communication. In a most general sense, semantic technologies try to define and implement explicitly machine-readable and also human-interpretable representations of data. Following this, we will focus on topic-based content management applications and the corresponding processes. Thus, we will describe and consider content at the level of native intelligence introduced in [4, 5]. In the second subsection, we will summarize a special classification scheme used in TC representing the basis for implementations of higher intelligence levels within the intelligence cascade of content.

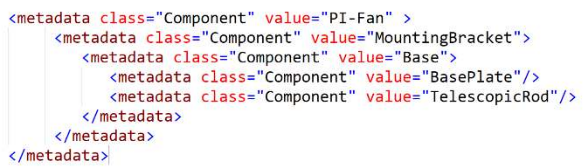
2.1 Semantics of natively intelligent content

Semantics has been introduced in TC early on within content creation processes by structured information models and mostly as DTD/XSD-based XML structures. Subsequently, semantic information became important on a larger scale with the spreading of CCMS. Technical writers apply semantic structuring and tagging to topic-based, reusable content. In turn, structured content should follow and can be controlled by linguistic rules. Moreover, semantically tagged content allows for many data processing scenarios like specific publishing for multiple media, formats, document types, target groups, markets, etc.

The next step in the use of semantics in TC was done by concepts of semantic metadata. The term semantics should be even more clarified in this context: Metadata has been introduced to describe content contained in topics, for retrieval of topics and for process automation. But types, logics and implementations of metadata in

[*] Email: wolfgang.ziegler@hs-karlsruhe.de

CCMS are varying strongly. Beside non-semantic plain-text metadata, metadata are somtimes implemented and processed as XML-attributes included in content object code like *<topic audience="admin" >…<topic>*. Or, even more generic, as XML-encoded label-value pairs separated from the content. By the latter, one can even describe hierarchical metadata modelled as taxonomies, like in the following example of Fig. 1, where labels (here called "class") and values stem from the PI-classification scheme described later.



```
<metadata class="Component" value="PI-Fan" >
     <metadata class="Component" value="MountingBracket">
        <metadata class="Component" value="Base">
           <metadata class="Component" value="BasePlate"/>
           <metadata class="Component" value="TelescopicRod"/>
        </metadata>
     </metadata>
</metadata>
```

Fig. 1. Example of XML metadata definition file for hierarchical metadata. Metadata values are assigned within CCMS authoring process to topics.

In this regard, metadata are in fact implicitly semantic and machine-processable. But the conceptual interpretation, i.e. meaning and human understanding might be still limited to the domain of the application, for example to system users within a company. The domain range will, of course, extend when using standards like DITA, ATA ISpec 2200 or S1000D. In the latter, not only metadata labels but also metadata values are standardized for military and aviation industries [6]. So far, in summary, CCMS implementations manage and process modularized content within topics and enriched by implicitly semantic metadata. Depending on the CCMS implementation, metadata are stored within topics or, more often, kept separately in databases as data, resp. relations. Separate XML metadata-objects as in Fig. 1 are used, for example, also for data export, exchange or reporting.

2.2 Metadata by PI-classification

One way to add systematically native intelligence to CCMS managed content is the well-established method of PI-classification [2]. Modularized, topic-based content can be assigned to four basic classes of metadata. These classes are organized with respect to products (P) and to information (I):

- intrinsic product classes, stating the physical or virtual product components the content is directly connected to. They can form complex and multi-level taxonomies.
- intrinsic information classes, defining precisely the information contained in the topic. Taxonomies are given by information types (e.g. procedural, descriptive, conceptual, safety). Subclasses thereof can describe more specific information (maintenance, repair, regulations, functions, etc.)
- extrinsic product classes describe the validity of topics with regard to individual products and reflect the portfolio of offered or delivered products.
- extrinsic information classes can be used for the assignment to output media, document types, markets or target groups.

The intrinsic metadata classes define the modular topic concept for authors, while the extrinsic metadata are used for process automation and variant management.

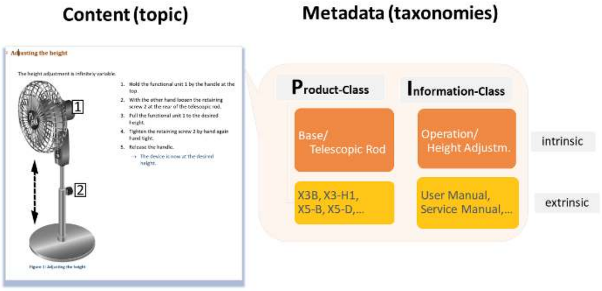

Content (topic)
Metadata (taxonomies)
Adjusting the height
The height adjustment is infinitely variable.
1. Hold the functional unit 1 by the handle at the top.
2. With the other hand loosen the retaining screw 2 at the rear of the telescopic rod.
3. Pull the functional unit 1 to the desired height.
4. Tighten the retaining screw 2 by hand again hand tight.
5. Release the handle.
→ The device is now at the desired height.
1
2
Figure 1: Adjusting the height
Product-Class
Information-Class
Base/ Telescopic Rod
Operation/ Height Adjustm.
intrinsic
X3B, X3-H1, X5-B, X5-D,...
User Manual, Service Manual,...
extrinsic

Fig. 2. Basic PI-Classification example for a topic (left side) of the PI-Fan reference content. The assigned classification values are usually stored in CCMS databases as hierarchical metadata separately from the topic.

This method of a multidimensional information space of metadata has been extended by two additional metadata dimensions: functional metadata and variant properties By functional metadata, delivery functionalities can support IoT-requirements (e.g. error code handling of content), work time calculations or run-time-based functions (e.g. maintenance intervals) [7]. Variant properties, on the other hand, should cover all additional configuration parameters needed for product-specific information delivery [8]. Hence, in the most general description, information architects end up with formally six-dimensional information space depicted in Fig. 3. The classes and values originate here from the reference content of the PI-class method consisting of a virtual product called the PI-Fan [9]. In industrial applications, the abstract dimensions of functional metadata and variant properties split into a manifold of parameters according to delivery use cases and to the specific product domain.

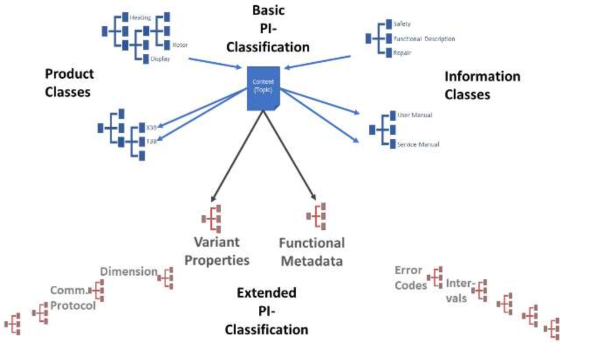


Fig. 3. Metadata information space for topic-based content (center) given by the extended PI-Classification (lower section) and some examples of classes. Basic classification is depicted in the upper section. In/outgoing arrows depict condeptual single-/multi-valued assignments of classes.

It should be noted that in Fig. 1, for simplicity, only the intrinsic base dimension of product components has been included in the code. It also is derived from the PI-Fan reference implementation. In practical applications, all classes (labels) have to be engineered specifically as custom CCMS implementations. Applying this method, semantic PI-classified metadata leads systematically to natively intelligent content and allows for higher CCMS process support and automization. However, the concept range might be limited because the implementations are system- and domain-dependent and therefore, still implicit.

3 Explicit semantics in TC

On the next methodological level of the intelligence cascade, the level of augmented intelligence, ontologies introduce more formalized and therefore explicit semantics. Ontological approaches are not new in the history of philosophy, linguistics, knowledge management [10] and, of course, semantic web technologies. But they gained recent interest in TC because increasing data, content and process complexity urges information architects to seek for more suitable technologies.

In this section, we will first focus on elementary and formal aspects to understand explicitness of semantics given by ontologies. We then give a short overview over applicability of corresponding implementations in the domain of TC.

3.1 Formal representation of semantics

Ontologies consist of definitions of object classes and instances thereof, their properties and relations between each other. There are lightweight and less complex ontology representations, often using the syntax of the resource description framework (RDF) and RDF-schema (RDFS) [11]. Heavy-weight ontologies can include additional rules and restrictions being represented, for example, within the ontology web language OWL [12]. The manifold of objects included in an ontology depends on the considered use case. Metadata classes and topic-based content classes are therefore just a small subset of possible ontological models. Ontologies can be visualized typically as semantic networks.

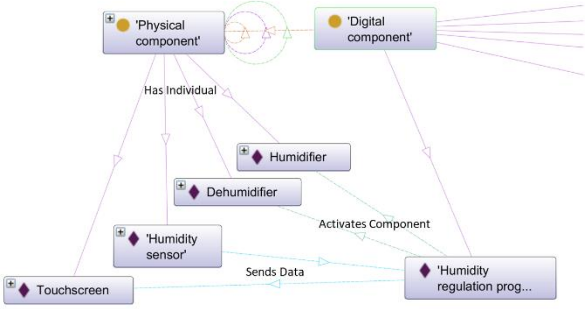
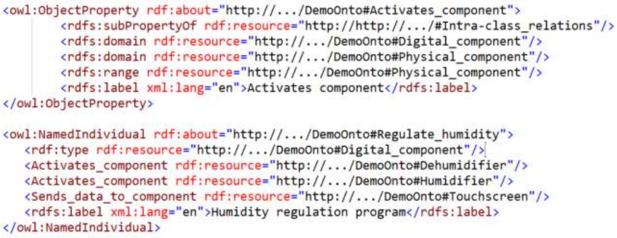


Fig. 4. Ontology detail of classes (circles) and instances (diamonds) defined in the Protégé modelling software [13]. Labelled lines depict different types of explicit relationships.

In Fig. 4, we depict a cut out of product details as a possible extension of the PI-Fan reference model. Classes and their instances of digital and physical components show relationships among each other. As a simple example, the model includes a digital software component (humidity regulation program) activating a humidifier hardware component.

Semantics of relations and classes, resp. instances is expressed explicitly by the above-mentioned ontology representations like OWL/RDF(S). In Fig. XY some corresponding code excerpts of the model are illustrated. One finds the definition of a relation "Activates_component" between digital or physical components (domain) activating physical components (range). The individual relation is then given between specific instances as introduced above.

Similarly, one can also define relationships between metadata as outlined implicitly in the previous section. Using ontology languages, one can express and subsequently even visualize explicitly, for example, common taxonomic (is subclass of) or partitive (is part of) relationships or any other system of logically ordered metadata values.



```
<owl:ObjectProperty rdf:about="http://.../DemoOnto#Activates_component">
        <rdfs:subPropertyOf rdf:resource="http://http://.../#Intra-class_relations"/>
        <rdfs:domain rdf:resource="http://.../DemoOnto#Digital_component"/>
        <rdfs:domain rdf:resource="http://.../DemoOnto#Physical_component"/>
        <rdfs:range rdf:resource="http://.../DemoOnto#Physical_component"/>
        <rdfs:label xml:lang="en">Activates component</rdfs:label>
</owl:ObjectProperty>

<owl:NamedIndividual rdf:about="http://.../DemoOnto#Regulate_humidity">
   <rdf:type rdf:resource="http://.../DemoOnto#Digital_component"/>
   <Activates_component rdf:resource="http://.../DemoOnto#Dehumidifier"/>
   <Activates_component rdf:resource="http://.../DemoOnto#Humidifier"/>
   <Sends_data_to_component rdf:resource="http://.../DemoOnto#Touchscreen"/>
   <rdfs:label xml:lang="en">Humidity regulation program</rdfs:label>
</owl:NamedIndividual>
```

Fig. 5. Code excerpt of the example ontology of Fig. 4: Definition of the relation described in the text and the individual relation between component instances in OWL/RDF representation. Class definitions of components have been omitted for simplicity.

But still, the domain range and the widely understanding and interpretation of relationships is an open question. It is usually answered by standardization and, necessarily, a broad acceptance of standards. As an example, the intelligent information request and delivery standard (iiRDS) has been developed basically for the exchange of modularized technical information [14]. Therefore, one intends to standardize content exchange focusing first on the exchange between CCMS and CDP. The core metadata model for topic classification is based on the PI-classification method introduced earlier. iiRDS defines a packaging mechanism on-top and uses RDFS for the representation of taxonomic metadata. More complex cross-relations would have to be modelled as extensions, thereby, potentially reducing the domain range again.

3.2 Ontology applications in TC

Recently, information and TC system architects started to take advantage of the semantical richness of ontologies. One can find several application types and corresponding process steps where ontologies can be applied by:

- Type I support model-based content engineering aligned with product-modelling. Modular content creation, variant and configuration management is modelled with the aid of partitive, functional or other relations given by product engineering and corresponding component classes. Semantic modelling software visualizing complex product dependencies might also interact by interfaces with CCMS to reduce or control metadata complexity in the authoring environment [15].

• Type II improve search and retrieval of information using CDP or other search environments. Here, semantic relations are modelled to optimize relevance and precision of search results. Relations are modelled along the use cases and business cases of content delivery. Also, content from other sources then CCMS can be included.

• Type III connect multiple data sources: Semantic software and contained models can act as a middleware between data sources and content consuming applications. Semantic networks are used to map data, various content types and documents along with their specific metadata environments. This is going beyond the scope of TC and therefore, CDP are just one of the possible delivery channels.

In this paper, we focus on the technical side of information systems. But, of course, TC-relevant ontologies are also well-known within linguistic sciences and terminology management, which is strongly related to the metadata development described so far.

For the first two application types of TC mentioned above, a schematic picture can be found in Fig. 6. Content transfer is achieved by packaging of content and metadata, which is done in proprietary formats or, for example, according to DITA-conform topic assemblies (maps). Another standardized approach would be the use of the above-mentioned iiRDS packaging format including RDFS syntax for taxonomic metadata.

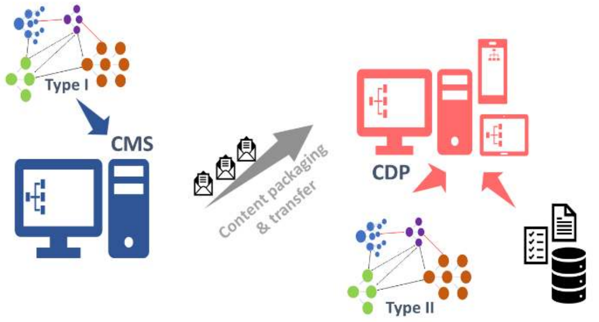


Fig. 6. Ontologies used in TC depicted as networks can support content creation (application type I on CCMS side) and search and delivery processes (application type II on CDP side).

4 Extending content delivery

In this section, we will combine the above investigated concepts of semantics with CDP functionalities. Starting point is the search and delivery of topic-based content from CCMS applications.

4.1 Dynamic deliverables as micro-documents

Delivery applications are understood in many cases as search interfaces adapted to the TC domain. This means, users can search manually for content using facets, direct text search or by navigation within document structures. Facets are usually derived from taxonomic metadata stemming from CCMS or, as described above, from classes and instances of a semantic model (application type II). The search results, i.e. the deliverables consumed by users consist either in single topics or in documents. The latter can be monolithic or assemblies of topics following traditional table-of-content (toc) structures. We now claim that even if topics are developed self-contained and therefore follow intrinsic classification, there is, in many cases, a lack of context or required correlated content. On the other hand, in traditional documents, there is potentially an abundance of content because documents are developed typically as a normatively or product-driven most complete set of information. But users in a specific use case and in a specific role need a relevant, precise and reduced, but sufficient amount of information. This has been the initial idea of content delivery in TC.

Preciseness of primary search results in CDP should be ensured by metadata quality from taxonomic classification. Then, in addition, the relevance of contextual content and the sufficiency of delivered topics should be expressible by rules. These rules can, in general, also depend on use cases and user roles. So, the user can finally define corresponding dynamic deliverables received upon request by users as micro-documents (microDocs): A microDoc is a (sub-)set of topics required by predefined use cases and connected by a logical concept as a dynamic publication in search media. [16]

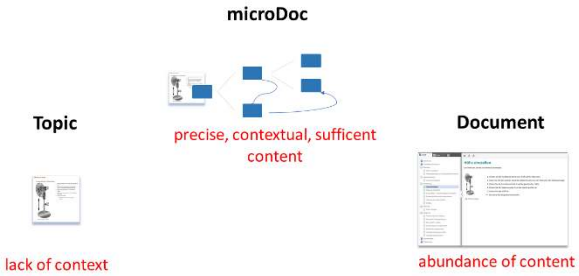


Fig. 7. microDocs as a rules-based content search and delivery optimization method with respect to required context and content amount.

It is now a question of involved technologies, i.e. the prevailing intelligence cascade level, how the respective logical concept can be implemented as rules for correlating content. Consequently, there will be several implementation levels of microDocs with increasing complexity:

a) Static documents predefined as topic assemblies on CCMS-level for selected and potentially most relevant use cases.

b) Dynamic topic arrangement within CDP and within the retrieval process. The arrangement is based on rules, applied to taxonomic metadata of topics.

c) Arrangement of content following rules predefined by semantic relations, properties of ontology classes and their instances.

Note, that the arrangement of topics and content objects in more general search systems might be realized in very different ways in the front-end applications. Examples might be aggregations in document-style or as topic clouds, reduced semantic networks, generated links, filtering of documents etc. And of course, there can be transitions and mixtures between the above introduced levels.

It is now still an open question, from where the afore-mentioned rules can be logically derived. As a starting point, TC information architects and writers can define

the most important rules, for example, according to already known user feedback, problem analysis, didactics or implied task logics and required prior knowledge. Once delivery is initiated, one can consider two additional levels:
d) Improvement of rules by web-analytics. Hereby, search behavior as well as quality of search results and other indirect or even direct feedback can be analyzed. Analytics is done at this level by humans and implications will change rules and even semantics.
e) Improvement of rules by artificial intelligence. Machine-learning algorithms can analyze search interactions and can derive, express and improve rules automatically. This would match scenarios of so-called predictive content.

These two levels d), e) could also be understood as quality assurance measures of a)-c) and not as independent levels themselves. Moreover, if further advanced deep-learning technologies like neural networks were introduced to the process, this next level would then not even impose the existences of rules set or explicitly understood by humans.

4.2 Content services

The information architecture derived so far, has especially on delivery side further consequences on the architecture of involved systems. In the architectural picture drawn for CDP, the user front-end is part of the delivery-application. But recent developments also tend to an architecture where CDP can act as a content service application (CSA), also known as headless CDP, resp. headless CMS.

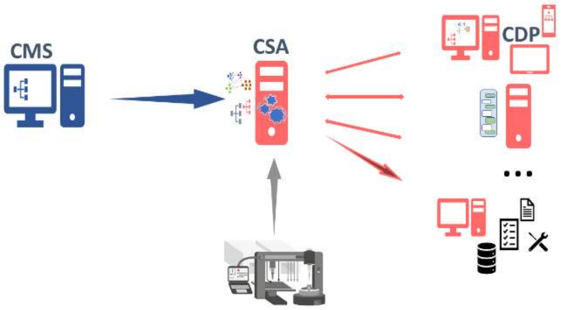


Fig. 8. Content service application CSA in the center providing content-based web-services upon request to a variety of content consuming applications like front-end CDP or chatbot applications. MicroDocs can be generated by processing rules applied to semantic relations within CSA. In the lower part, a push scenario is sketched.

The advantage of such a web-service architecture shown in Fig. 8. is, that a larger variety of content consuming applications can request, process and deliver content according to their specific use case and independently from the internal logic of the intermediate CSA server. Traditional search interfaces and facetted CDP search are only some of the possible content consumers. Others might be chatbots, social media platforms, service management applications, sales applications, help-desk support applications and many more. In this sense, CSA resemble partially to the application type III of ontology applications as an information hub. The difference would be the more limited side of content sources focusing CCMS-based content or, at least, information objects classified according to a common metadata concept.

In Fig. 8 the visualization of the corresponding architectural concept can be found. Combining the theoretical considerations of microDocs and the recent approaches of service-oriented architectures, one can understand microDocs as dynamic microservices supporting predefined front-end applications and use cases by dynamically optimized content assemblies. They could be transferred to front-end CDP upon request as standardized iiRDS packages, whereas other applications might request different packaging formats or just simple web-formats. The logic, i.e. the rules defined in the CSA, can act on taxonomies or ontologies, depending again on the intelligence level of involved technologies. But in such a case, rules don’t have to be implemented in the front-end CDP. So far, we considered pull, technologies delivering information upon request. But of course, event-driven push technologies can also be supported as shown also in Fig. 8. There, a software-detected event within a machine can trigger the automized delivery of topics or microDocs to a service application without an active request.

5 Summary

In this article, we gave a basic introduction into the recent situation of semantics and how it is introduced via intelligent content in the domain of technical communication. We focused first on the information architecture of topic-based content used in CCMS and outlined the transition from natively intelligent content of metadata taxonomies to more explicit semantics of ontologies and so-called extended intelligence. Thereby, we used the PI-classification and its formulation as OWL/RDF representation to clarify explicitness of ontological approaches. Standards like iiRDS based on RDF and PI-classification with a normatively given vocabulary can then increase the domain range and acceptance of such technologies. iiRDS packaging, is thereby intended for data transfer between CCMS and delivery systems.

Furthermore, we introduced a new concept of microdocuments to optimize content relevance of requested information via CDP. This is done by bridging the gap between topic-based and document-based delivery by building a new type of rules-based and use-case-dependent dynamic topic assemblies. The logical concept of these microDocs, i.e. the relevant context and the amount of required content, should be derived at different levels from semantic rules and models. These models then can be either taxonomies or ontologies.

Finally, we gave an outlook on implications for recent system architectures of CDP. We focused on content service applications as a middleware between CCMS and a variety of front-end applications. Semantic rules are processed within CSA to generate and deliver microDocs in push and pull scenarios. Therefore, microDocs can be understood as microservices providing more contextual rich content than just single topics in CDP scenarios.

The author acknowledges support with ontology modelling of a modified and extended PI-Fan product model by A. Ahmadpour.

References

1. A. Rockley, *Managing Enterprise Content. A Unified Content Strategy*. New Riders. (2012)
2. W. Ziegler, *Basic Concepts Which Support the Management and Delivery of Intelligent Content*, Frontier, Official Journal of Japan Technical Communicators Association JTCA, p. 84-91, (2017)
3. W. Ziegler, H. Beier, *Content delivery portals: The future of modular content*, tcworld e-magazine 02/2015
4. W. Ziegler, *The Evolution of Content Management towards Intelligent Delivery Systems for Technical Communication*, Frontier, Official Journal of Japan Technical Communicators Association JTCA, p. 68-75, (2017)
5. W. Ziegler, *Drivers of Digital Information Services: Intelligent Information Architectures in Technical Communication,* ACM Chapter Proceedings on Educational Technology, Language and Technical Communication ETLTC, ISSN: 2434 – 9526, p. 48-52, **1** (2019)
6. ASD, *S1000D International Specification for Technical Publications,* http://public.s1000d.org/Pages/Home.aspx
7. W. Ziegler, *Intelligente Informationen*, Schriften zur Technischen Kommunikation, Vol. 22, p. 51-66, Ed. Hennig J., Tjarks-Sobhani M., tcworld GmbH, (2017)
8. W. Ziegler, U. Parson, J. Sapara, *iiRDS for technical writers – Introduction to the Metadata*, Tutorial Presentation & Proceedings tcworld conference, p. 225-228 (2017)
9. W. Ziegler, *Content management systems tested: Introducing the PI-Fan*, tcworld e-magazine 10/2016
10. N. Guarino, *Formal Ontology, Conceptual Analysis and Knowledge Representation*. Int. J. Human-Computer Studies 43, (1995)
11. W3C, *RDF Schema 1.1,* https://www.w3.org/TR/rdf-schema/ (2014)
12. W3C, *Web Ontology Language (OWL)*, https://www.w3.org/OWL/ (2012)
13. *Protege Desktop User Documentation*, https://protegewiki.stanford.edu/wiki/ Protege4UserDocs, (2019)
14. iiRDS – The International Standard for intelligent information Request and Delivery https://iirds.org/material-downloads/iirds-version-1-0-1/
15. C. Deschner, W. Ziegler, *Informationsmanagement und Ontologien – entwicklungsnahe Produktdokumentation in der Medizintechnik*, Presentation & Proceedings tcworld conference, p. 98-99 (2018)
16. W. Ziegler, *Alles hängt mit allem zusammen: Content vom Anfang bis zum Ende gedacht*, Presentation & Proceedings tcworld conference, (2019)