

# Search Ontology Generator (SONG): Ontology-based Specification and Generation of Search Queries

Alexandr UCITELI<sup>a,1</sup>, Stefan KROPF<sup>a,2</sup>, Timo WEILAND<sup>b</sup>, Stefanie MEESE<sup>b</sup>, Klaus GRAEF<sup>b</sup>, Sabrina ROHRER<sup>b</sup>, Marc O. SCHURR<sup>b</sup>, Wolfram BARTUSSEK<sup>c</sup>, Christoph GOLLER<sup>d</sup>, Philipp BLOHM<sup>d</sup>, Robin SEIDEL<sup>e</sup>, Christian BAYER<sup>e</sup>, Manuel KERNENBACH<sup>e</sup>, Wolfgang LAUER<sup>e</sup>, Jörg-Uwe MEYER<sup>f</sup>, Michael WITTE<sup>f</sup> and Heinrich HERRE<sup>a,3</sup>

<sup>a</sup>*Institute for Medical Informatics, Statistics and Epidemiology (IMISE), University of Leipzig, Germany*

<sup>b</sup>*novineon Healthcare Technology Partners GmbH, Germany*

<sup>c</sup>*OntoPort UG, Germany*

<sup>d</sup>*IntraFind Software AG, Germany*

<sup>e</sup>*Federal Institute for Drugs and Medical Devices (BfArM), Germany*

<sup>f</sup>*MT2IT GmbH & Co. KG*

**Abstract.** The observation of medical devices during Post-Market Surveillance (PMS) for identifying safety-relevant incidents is not trivial. A wide range of sources has to be monitored in order to integrate all accessible data about the safety and performance of a medical device. PMS needs to be supported by a clever search strategy and the possibility to create complex search queries by domain experts. Ontologies can support the specification of search queries and can aid the preparation of the document corpus, which contains all relevant documents. In this paper we present the new version of the Search Ontology (SO2), the Excel template based specification of search queries and the Search Ontology Generator (SONG), which is useful for the generation of very complex queries out of the Excel-based specification. Based on our approach a service-oriented architecture was designed, which is able to support domain experts during PMS.

**Keywords.** ontology, information retrieval, search queries, spreadsheet-based ontology specification, ontology generation, Post-Market Surveillance

## 1. Introduction

According to the provisions of the current European Medical Device Directive 93/42/EEC [1,2] and the new European Device Regulation (coming into effect in May 2020) [3], each manufacturer of medical devices has to set up a comprehensive system in order to identify, evaluate and integrate clinical data derived from the field application of a medical device after market access during Post-Market Surveillance (PMS). Small

---

<sup>1</sup> Alexandr Uciteli; E-mail: auciteli@imise.uni-leipzig.de

<sup>2</sup> Stefan Kropf; E-mail: stefan.kropf@imise.uni-leipzig.de

<sup>3</sup> Heinrich Herre; E-mail: heinrich.herre@imise.uni-leipzig.de

and medium-size enterprises in the field of medical devices are in need for operable systems for post-market data retrieval in order to enhance their PMS strategies and to be prepared for the growing requirements of the new European Medical Device Regulation.

A wide range of both internal (own quality management and compliant system) and external (scientific databases, medical congresses, internet-based knowledge & experiences, PMS by competent authorities) sources have to be monitored in order to integrate all accessible data about a medical device's safety and performance. Currently, these detailed, continuous searches are still performed manually with a high input of time and personnel resources, making PMS a daunting task.

Literally, an employee has to type all search queries (e.g., "safety AND coronary stent") into the input fields of a large number of different databases, congress sites and public search engines in order to gain a broad, unspecific hit list, interspersed with single, relevant content. This strategy of data retrieval reaches its limits assuming that several search strings have to be applied in order to monitor a whole range of medical devices, each featured by a variety of decisive search questions.

Additionally, each notable medical database uses its own, inherent syntax to specify search queries. This incompatibility between databases and the amount of different search tasks combined with the manual application to a multitude of databases results in low efficiency and a high potential for human error.

Setting up more complex queries in a simple manner by domain experts enables the definition of the topic of interest in a more specific way and circumvents the problem of retrieving irrelevant content.

In the OntoVigilance project (predecessor project of OntoPMS) we developed the Search Ontology (SO) [4], a promising approach for an ontology-based specification of complex search queries. The modular architecture of the SO enables the re-use of ontology parts in different use cases as well as a quick and easy adaptation and extension of the ontology according to the specific requirements.

The developed SO and its application was evaluated in the OntoVigilance project by domain experts. The SO supports the sets up of a PMS strategy in order to retrieve internet-based information/clinical data on safety and performance of an exemplary medical device, e.g., an endoscopic clipping system. The SO-based, continuous data retrieval was compared to a conventional periodic, manual data search with good success. Relevant content was identified by SO-based data retrieval in a more convenient and comprehensive way.

This previous study has proven that the SO is a suitable method for modeling complex queries, but it should be optimized to face all requirements of domain experts. On the one hand, the structure of the SO can be simplified in order to improve the usability by domain experts. On the other hand, certain extensions are necessary to model all relevant query types. Based on these findings, we further developed and optimized the SO in the OntoPMS project. The improved version of SO is called **SO2** in this paper. The SO2 is generic and can be used in any domain. For the application of the SO2 in a particular domain, it has to be extended by a domain ontology (in this paper a domain ontology for the PMS domain), so that the classes of the domain ontology are subclasses of the SO2 classes. We call such domain ontologies Domain-specific Search Ontologies (**DSO**). In addition, we developed an Excel template to specify the information required to create a DSO, which significantly simplifies the ontology development by domain experts. For the automatic generation of a DSO from the Excel-based specification, we implemented the Search Ontology Generator (SONG). In contrast to the OntoQueryBuilder (OQB) [4], the SONG generates the complete DSO, including all

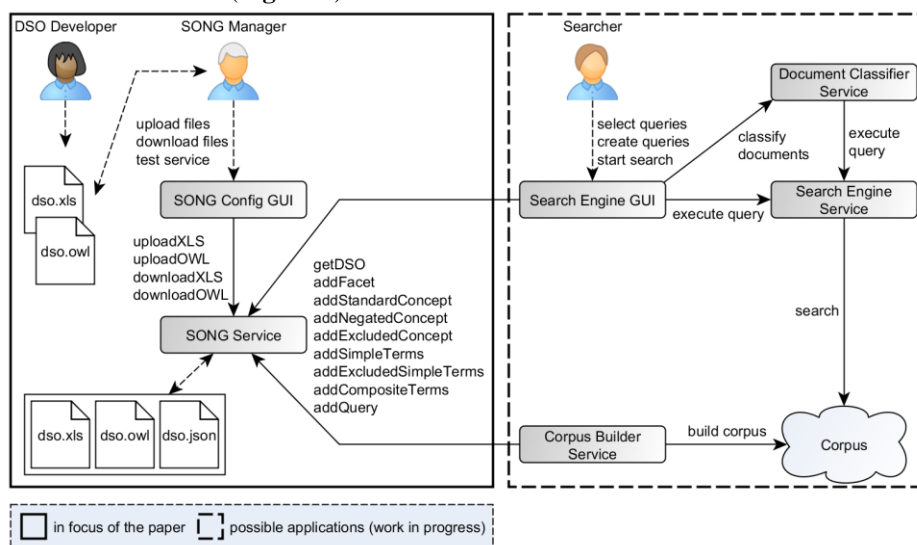
specified queries in the correct query syntax from the Excel template and provides it for external tools (e.g., the search engine). In this way, the search engine can get the complete DSO by accessing the SONG service without any requests or generating queries at search time.

## 2. Methods

Focus of this paper is the improvement of the SO and development of the SONG (see Results) to support the search pipeline at various points:

1. *Creating a corpus.* The SONG provides special queries for building a corpus of relevant documents for a domain of interest.
2. *Searching for relevant documents.* The SONG generates the DSO, including complex search queries modeled by domain experts. The DSO can be displayed on the GUI of the search engine; the queries are selected by the user and are executed by the search engine.
3. *Classifying the search results.* The SONG queries can also be used to classify the search results.

For a better understanding of our approach, this section introduces possible applications of the SONG service (**Figure 1**).



**Figure 1.** SONG and its applications

The DSO Developer specifies the DSO in Excel or optimizes the generated OWL version. The SONG Manager uploads/downloads files and tests the service. The SONG Service is responsible for the generation of OWL and JSON from Excel, as well as JSON from OWL and offers different methods for possible applications (see SONG section). The Searcher uses the Search Engine GUI, which represents the DSO, and can select desired concepts or queries for the execution by the search engine.

### 2.1. Search Engine

The SONG can be used with any Lucene-based search engine for the generation of queries in the Lucene query syntax out of the Excel-based specification. In addition, new

expressive query operators were implemented in the OntoPMS project, which significantly extend the Lucene query syntax.

To identify risks or complications (for PMS) in unstructured documents, very complex patterns have to be detected. Such patterns go beyond the standard capabilities of state-of-the-art search engines like Elasticsearch [5]. Therefore, we extended these capabilities by creating our own search plugin, providing the required functionalities and improving search quality. The extension was realized as an Elasticsearch plugin and contains, among others, the following additional features: improved tokenization, lemmatization and word decomposition; build-in support for several normal forms / term types; improved quality for ambiguous searches; named entity, date and measurement recognition; additional search modes; NEAR operators.

In particular, the search modes and new search operators are extensively used in our DSO to produce more precise queries. The search modes correspond to the different types of terms (exact[E], diacritics-normalized[D], lemmatized baseforms[B], compounds-parts[C]) that we use in our index. For example, with the query “MODE/E(SafeSet)” we only search for SafeSet with two upper case S. Words within a NEAR/n query must not have a token-distance greater than n. With NEAR/S and NEAR/P words must occur within one sentence or one paragraph. These new NEAR-Operators can be combined and nested with other queries in an arbitrary way.

The search engine GUI accesses the SONG service und receives the DSO (getDSO). The search concepts and multiple concept queries are displayed on the GUI as a tree with checkable nodes. The user can select desired concepts or multiple concept queries by checking the corresponding checkboxes. After pressing the submit button, selected queries are transmitted to the search engine service and the received search results are displayed. Optionally, the user can classify documents (e.g., new incident reports, see Classification of search results) using the Document Classifier Service. The Document Classifier Service stepwise sends all single concept queries to the Search Engine Service and assigns the search results to the corresponding concepts. The user can then click on a concept in the tree and get the documents classified by the concept.

Additionally, the GUI supports the creation of new queries respectively query parts (search terms, search concepts, etc.). For example, for including a new multiple concept query to the ontology the user has to select the desired concepts and has to press the “create query” button. Then, the GUI sends this information to the SONG service (using add methods like addQuery). After that, the SONG adds the new entities to the ontology und transmits the updated DSO back to the search engine GUI.

## *2.2. Corpus Builder*

PMS requires information from several types of sources including proprietary manufacturer data, and information from the web. Getting information from the web requires some knowledge how to look for it, how to access it, and which parts to extract. Additionally, following the links found on a page identified by a given URL quickly turned out to be a potential trap, since the pages may be completely out of scope. Therefore, we concluded that an ontology would help to define the scope.

For data acquisition, we developed the Corpus Builder. Its input component, the Prospector, metaphorically speaking, “roams” the internet in order to identify suitable data to “feed” the OntoPMS Corpus. To achieve this, it uses a special corpus query, which is part of the DSO. Currently, the Prospector delivers its documents to the NLP pipeline, which analyses the contents to identify documents that are important to the

respective projects and rejects (i.e. blacklist) documents that shall not be included into the OntoPMS corpus. This processing is done by a kind of control circuit. The “plant” of the feedback loop is controlled by a seed list, produced by the prospector and by the feedback component. It does the crawling and gathering of new URLs by following forward and backward links and reading the contents identified by the URLs. Then, the output is checked by the “sensor”. The sensor is controlled by the corpus query and allows a deep analysis of the content. Then, questionable content is fed back to a splitter component which sorts out garbage (blacklist) and boosts domains with a high amount of documents we want to include (whitelist). URLs, based on those white listed domains are then included if not yet part of the seed list. If the output contains unwanted documents, we have to improve the corpus query. Hence, we have a semi-supervised learning component, where the manual part of supervision is made on the abstract level of ontologies. This enables us to change the behavior of the corpus builder without changing the software.

### *2.3. Classification of search results*

From the regulatory perspective of the German competent authority for risk assessment for medical devices, the Federal Institute for Drugs and Medical Devices (BfArM), there is an increasing demand to have the risk assessment process of critical incidents supported by an intelligent IT solution. With respect to the exponential increase in reported incidents with medical devices in Germany, specific DSOs for different aspects of the incident must be developed. These specific DSOs will allow the identification of similar incidents and thereby an automatic recommendation of classification of new incidents will become possible. The aspects currently in focus include DSOs for the resulting health problem, device problem, root cause and components, all of which are at present being developed on the basis of the FDA coding system [6] by the IMDRF working group on Adverse Event Terminology and Coding [7]. Using the SONG approach allows domain experts to easily create and modify the subsequent search concepts for each of the IMDRF Codes. Our approach accelerates the risk classification, which in turn allows to create individual views of device specific problems as well as to monitor the performance of different manufacturers within certain devices groups such as hip implants, cardiac pacemakers or insulin pumps.

## **3. Results**

### *3.1. Improved Search Ontology (SO2)*

This section presents the optimized Search Ontology (SO2), which contains some improvements compared to the SO. One of the advantages of the new version is that the SO2-based DSOs can be specified in a specially developed Excel template rather than with an ontology editor. The Excel template allows only the specification of such query types, which have been determined as relevant in the above-mentioned study, while in the previous version any combination and nesting of the query parts was allowed. In addition, the keywords for the search (search terms) had to be defined as instances of the search term classes (with different labels) and linked to the search concepts using the property restrictions (based on the property *described\_by*). In the new version, the search terms are directly associated with search concepts as annotations. Both aspects simplify

the structure of the ontology. Other extensions include negated concepts and direct storage of queries in the ontology.

The SO2 models three types of entities: search concepts, search terms, and search queries (Figure 2). The search concepts are concepts (in the sense of General Formal Ontology, GFO [8,9]), whose descriptions or designations have to be found in texts. The other two entities are symbolic structures (gfo:Symbolic\_Structure) and serve to model single keywords or phrases of the concept description as well as queries.

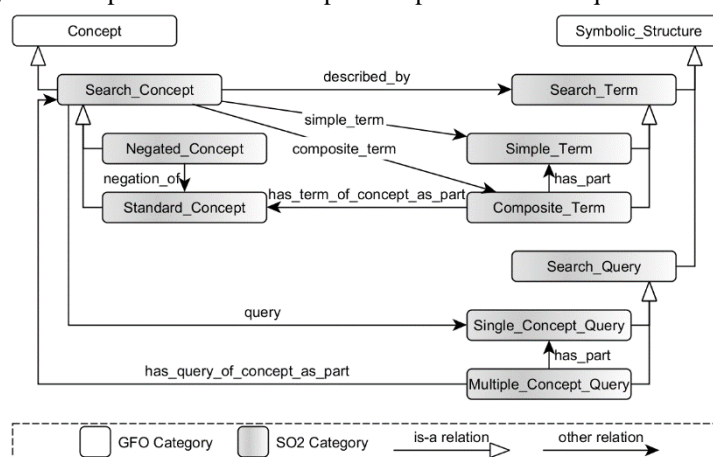


Figure 2. SO2

**Search Terms.** The search terms are descriptions or designations of search concepts. A distinction is made between simple and composite terms. The simple terms are either single words (e.g., “clip”, “Nitinol”) or fixed (defined by the user) phrases (e.g., “endoscopic clipping system”). The composite terms are combinations (*has\_part*) of simple terms of two search concepts (*has\_terms\_of\_concept\_as\_part*). They are defined by the user by choosing the two concepts (e.g., Unexpected and Complication) and are generated by the generator as an AND-connection of the OR-linked simple terms of the selected concepts (e.g., “(unexpected OR unforeseeable OR unknown) AND (complication OR failure OR incident)”).

**Search Concepts.** The search concepts are described or designated by search terms (*described\_by*, *simple\_term*, *composite\_term*). We distinguish between standard (e.g., Complication) and negated (e.g., No\_Preclinical) concepts. While the terms of the standard concepts (e.g., “complication”, “failure”, “incident”) have to be contained in the resulting documents, the terms of the negated concepts (e.g., “animal”, “study”, “preclinical”) have to be excluded/negated. Each concept is additionally associated with a single concept query (see Search Queries), which is used for the search for descriptions of the concept.

**Search Queries.** The single concept query is an OR-connection of all terms (simple and composite) for a standard concept or a negated OR-connection of all terms for a negated concept. The query can additionally contain specified search operators and brackets. The multiple concept query is an AND-connection of single concept queries of selected concepts (*has\_query\_of\_concept\_as\_part*). The query is specified by the user through a selection of desired concepts and generated by the generator.

### 3.2. Specification of the DSO

Market access of a medical device is granted after passing an extensive series of tests, risks analyses and evaluations of clinical data on the medical device's safety and efficacy. Nevertheless, the behavior of a medical device over time in broad application can be investigated a priori only in a limited manner. Thus, PMS strategies are set up in order to summarize application data of, e.g., medical implants in order to identify residual risks. For example, reports on unfavorable interactions between implant material and patient's tissue are identified and evaluated in order to a) control residual and/or unexpected risk, b) to identify vulnerable patient subpopulations and c) to improve the respective medical implant or material, respectively.

For a better understanding, a practically relevant search query was specified (as part of the DSO for the PMS domain) focusing on unexpected side effects of the metal alloy Nitinol used for construction of endoscopic clipping systems. The search query was constructed by defining concepts covering the different aspects of the PMS question like "unexpected complication", "type of medical device" (endoscopic clipping system) and "used material" (Nitinol), as well as the associated search terms within the easy applicable Excel template. **Figure 3** illustrates this part of the DSO by screenshots of the several sheets.

Sheet: Material		Simple Terms (en)	
Material			
	Nitinol		nitinol, "nickel titanium", NiTi

Sheet: Medical_Device		Simple Terms (en)	
Medical_Device			
	Clip		clip
	Endoscopic_Clipping_System		endoscopic clipping system, endoscopic clipping device, endoclip, endoscopic clip, over-the-scope-clip, through-the-scope-clip

Sheet: Negated_Concept		Excluded Concept	Excluded Simple Terms (en)
No_Study			
	No_Preclinical		animal, study, benchtest, benchmark, preclinical, cadaver

Sheet: Incident		Simple Terms (en)	
Incident			
	Complication		complication, damage, emergency, failure, incident^5, injury, malfunction, risk, adverse event, safety
	Unexpected_Complication		

Sheet: Additional_Concept		Simple Terms (en)	
Additional_Concept			
	Adjective		
	Unexpected		unexpected, unforeseeable, unknown, unprecedented, unlikely

Sheet: Composite_Term				
Part1	Part2	MODE	NEAR	Concept
Unexpected	Complication		NEAR/S	Unexpected_Complication

Sheet: Multiple_Concept_Query						
Query	MODE	NEAR	Concept	Concept	Concept	Concept
Nitinol_Complication_Query			Unexpected_Complication	Nitinol	Endoscopic_Clipping_System	No_Preclinical

**Figure 3.** Specification of the DSO (excerpt)

Our developed Excel template consists, on the one hand, of the predefined data sheets (Negated\_Concept, Composite\_Term and Multiple\_Concept\_Query) and, on the other hand, of the user-defined sheets (facet sheets) for the specification and

classification of the search concepts and simple terms. We consider the facets, similarly to the Colon Classification [10], as multiple dimensions for the categorization of the Universe of Subjects. From an ontological point of view, these facets are top-level categories that have a defined subclass hierarchy. Our template represents these top-level categories by different facet sheets, which are evolving through time and which are freely definable by the domain expert. For the observation of medical products during PMS, we introduced among others the following facets: Medical\_Device, Medical\_Area, Medical\_Problem, Incident, Material and Risk. In our example, the different facets of the search are represented by the Excel sheets Material, Medical\_Device and Incident. In every such facet a subclass structure represents a categorization of the knowledge within the appropriate area. For instance, we subdivided medical devices in clip, stent, occluder and implant; moreover, these devices categories can be subdivided into special types, e.g., endoscopic clipping system, PFO occluder, PDA occluder, and so on. Next to the nodes of the subclass hierarchy, the domain expert can enter simple terms; two columns are used for enabling the separation of English and German simple terms. Several types of query operators, like the wildcard or boost (e.g., incident^5) can be applied to simple terms in order to refine the search query.

For excluding documents that contain descriptions of certain concepts (e.g., complications in preclinical tests) the Negated\_Concept sheet is used. The concept is specified (e.g., No\_Preclinical) and described by simple terms to be excluded (e.g., “animal”, “study”, “preclinical”). The Composite\_Term sheet is used for the specification of terms based on other concepts. The composite terms for describing the search concept Unexpected\_Complication, are combined from the simple terms of Unexpected (part 1) and Complication (part 2). For the creation of complex (multiple concept) queries, which are based on the conjunction of queries of multiple search concepts (e.g., Unexpected\_Complication, Nitinol, Endoscopic\_Clippling\_System and No\_Preclinical), the Multiple\_Concept\_Query sheet is used. The specification of the improved MODE or NEAR (e.g., NEAR/S) operators is possible for both, composite terms and multiple concept queries.

Out of these spreadsheet-based specifications, it is possible to generate the DSO in different formats like OWL or JSON using the SONG.

### 3.3. Search Ontology Generator (SONG)

The SONG consists of a web service that provides various methods for the external tools and a simple web app (SONG Config App) that is used to upload/download the files as well as for testing the service (**Figure 1**).

The SONG service generates the ontology in OWL and JSON format out of the Excel-based specification, allows adding new entities to the ontology (add methods like addSimpleTerms or addQuery), and provides the generated DSO for external tools, especially for search apps (getDSO). The OWL format is used for a possible optimization of the generated ontology with an ontology editor or for an integration of external ontologies, while JSON is utilized for communication with external tools. The SONG manages the generation of OWL and JSON from Excel, as well as JSON from OWL. After each file upload (Excel or OWL) or after an adding of a new entity, the new ontology (OWL and JSON) is generated.

By the generation of the ontology, the model presented in **Figure 2** is not applied one-to-one, but rather simplified. For example, simple terms and single concept queries are defined as annotations of the search concept classes.



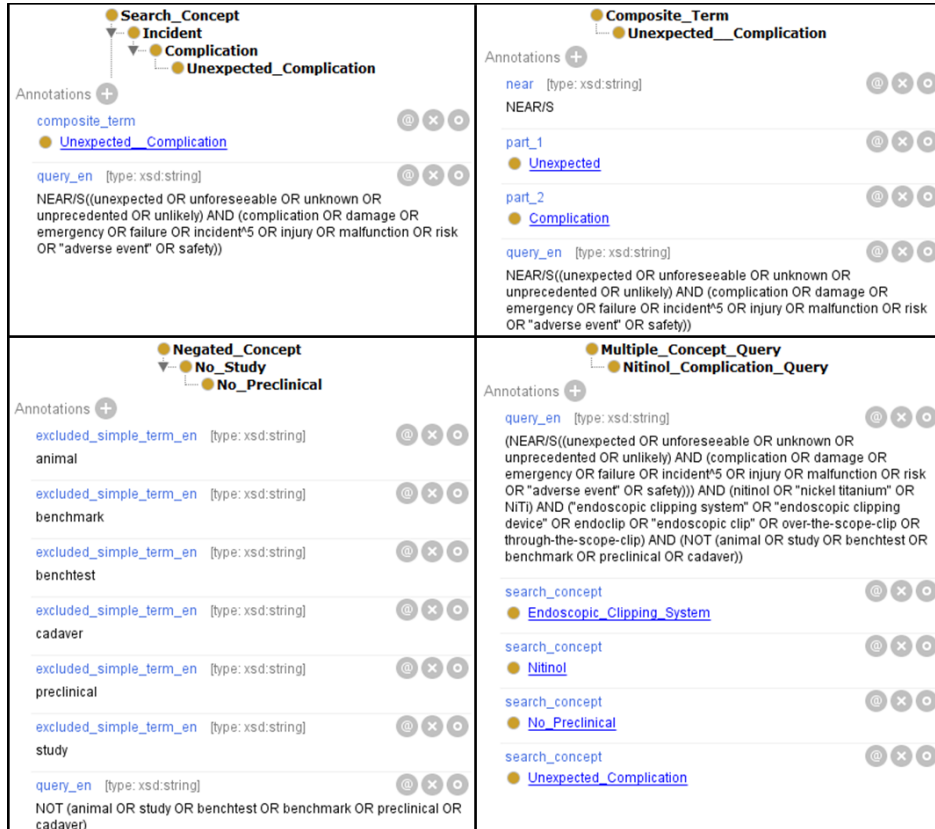


Figure 4. Parts of the DSO in Protégé

Firstly, the SONG generates the class hierarchy. The search concept trees from the user-defined sheets (facet sheets) are placed under *Search\_Concept* and the negated concept tree under *Negated\_Concept*. Next, the simple terms are linked to their concepts using the annotation property *simple\_term*.

The composite term classes are generated as subclasses of *Composite\_Term*. The annotation properties *has\_term\_of\_concept\_as\_part\_1* and *has\_term\_of\_concept\_as\_part\_2* (shortly: *part\_1* and *part\_2*) are used to specify the two search concepts whose simple terms have to be put together. In addition, the specified MODE or NEAR operators are generated as annotations. Then, for each composite term class, the corresponding term is generated as an AND-connection of the OR-linked simple terms of the selected concepts, and is associated to the composite term class using the annotation property *query*. The possibly specified query operators for composite terms are taken into account in the correct syntax. Then, the composite term classes are referenced in the search concept classes by the annotation property *composite\_term*.

After that, the single concept queries of all search concepts are generated as an OR-connection for standard concepts or a negated OR-connection for negated concepts from all their terms, and are associated with the respective concept using the annotation property *query*. For negated concepts, the standard concepts can also be specified, whose terms have to be excluded (*excluded\_concept*).

The multiple concept queries specified on the Multiple\_Concept\_Query sheet are generated as subclasses of Multiple\_Concept\_Query. Then, the multiple concept query classes are associated with the concepts whose single concept queries are to be combined using the annotation property *has\_query\_of\_concept\_as\_part* (shortly: *search\_concept*). Then, the single concept queries are AND-linked and stored using the annotation property *query*. Similar to composite terms, the possibly specified operators for queries are taken into account during the generation process.

In the **Figure 4**, some parts of the generated DSO are illustrated. The upper part shows the search concept Unexpected\_Complication, which are described by composite term Unexpected\_\_Complication (with two underscore characters). In the lower left corner the negated concept (No\_Preclinical) is described, which is used for the exclusion of several simple terms. In the right lower corner, the complete (multiple concept) query is illustrated, which consists of different search concepts. In the annotation property *query* is the generated query, which can be executed by a search engine.

## 4. Related work

### 4.1. Ontology-based information retrieval

Since finding meaningful and intelligent information is difficult, there are different ontology information retrieval techniques methods available [11]. In the wide world of semantic searches the approach of this paper can be classified as Research Search [12], because we denoted search queries by concepts. Semantic searches are usually executed not on plain documents but on ontologies, which requires expensive manual annotation or natural language processing steps (NLP) for extracting semantic data out of the documents. After that step the information of the documents is stored in a semantic knowledge base [13] or in a semantically enriched enhanced document index [14], on which semantic searches can be applied by using semantic retrieval languages like SPARQL [15] or SeRQL [16]. The early TAMBIS project [17] provides a foresighted semantic search approach for accessing multiple bioinformatics databases, using a complex biological concept model for query formulation. Despite of semantic knowledge bases or structured data sources the approach of this paper builds up on indexed documents which can be retrieved by complex Boolean expressions, which are difficult to construct [18]. Using ontologies as navigation tree structure in form of a Concept-based Information Retrieval Interface (CIRI) seems to be more effective than a direct interface (input field) [19].

GoPubMed [20] uses the Gene Ontology for search on PubMed. In contrast to our approach, the user is not able to increase the precision of the search by simply developing and using his own DSO, exactly tailored to his needs.

Textpresso [21] is a text-mining system for scientific literature. It implements categories of terms (an ontology) which can be used for a search on a database of articles. Regular expressions have to be created for each category to match the corresponding terms in the text and the documents have to be labeled according to the lexicon of the ontology. The documents are then indexed with respect to labels and words. Our solution does not use any in the ontology contained information for pre-processing or indexing the documents. The ontology is constantly under development and is adapted by the domain experts to meet their current needs. Our approach does not require any additional

pre-processing steps (e.g., labeling) as well as re-indexing the document collection when ontology changes.

#### *4.2. Excel-based ontology development*

Since ontology engineering is difficult for non-ontologists, there is a need for a rapid and collaborative ontology engineering methodology and easy to use tools [22]. The transformation of spreadsheets in OWL is already used in life science projects [23,24], tools and plugins enable the population of OWL ontologies out of spreadsheet templates [25]. The template we developed differs in that way, that it is not intended for the ontology development in general based on modelling certain OWL constructs. Instead, our template is exactly tailored to the SO2-based development of DSOs in order to make its use by domain experts as intuitive as possible.

### **5. Conclusion and future work**

The specification of complex search queries is a recurring task in different domains. The search for complications in the usage of medical devices within the Post-Market Surveillance or the classification of the incident reports are only two examples in this area.

We presented the improved Search Ontology (SO2), a promising domain-independent approach to specify complex search queries. Our solution allows advanced search for relevant documents in different domains using suitable DSOs and supports an automatic classification of search results. The second version of the Search Ontology include enhancements like the inclusion of new search operators, negated concepts and direct storage of generated queries in the ontology. For easier handling by non-ontologists, we developed an Excel template, which facilitates a SO2-based specification of DSOs without the usage of ontology editors or knowing the query syntax. A service-oriented architecture was introduced; in the core of the architecture stands the Search Ontology Generator (SONG), which provides methods for an access by search engines as well as DSO administration methods. By the enhancement of the SO, the spreadsheet-based specification and the service-oriented architecture, we improved the access to the Search Ontology for domain experts and external tools.

The service-oriented architecture of the SONG is already intended for the access of the DSO by external tools. The future work includes the further development of the described software components like search engine GUI and Document Classifier Service. After the definition of a sufficiently extensive knowledge base in form of DSOs, ontology learning can be exploited for supporting a semi-automatic query creation. In addition, applications will be developed to modify external terminologies/ontologies for their usage in or as DSOs.

### **Acknowledgement**

This work has been funded by the German Federal Ministry of Education and Research (BMBF) in the “KMU-innovativ” funding program as part of the “OntoPMS” project (reference number 01IS15056B).

## References

- [1] Council Directive 93/42/EEC of 14 June 1993 concerning medical devices, *OJ. L* (1993) 1–43.
- [2] Council Directive 90/385/EEC of 20 June 1990 on the approximation of the laws of the Member States relating to active implantable medical devices, *OJ. L* (1990) 17–36.
- [3] Regulation (EU) 2017/745 of the European Parliament and of the Council of 5 April 2017 on medical devices, *OJ. L* (2017) 1–175.
- [4] A. Uciteli, C. Goller, P. Burek, S. Siemoleit, B. Faria, H. Galanzina, T. Weiland, D. Drechsler-Hake, W. Bartussek, and H. Herre, Search Ontology, a new approach towards Semantic Search, in: E. Plödereder, L. Grunske, E. Schneider, and D. Ull (Eds.), *FoRESEE: Future Search Engines 2014 - 44. Annual Meeting of the GI, Stuttgart - GI Edition Proceedings LNI, Köllen, Bonn, 2014*: pp. 667–672.
- [5] Elasticsearch, (n.d.). <https://www.elastic.co/products/elasticsearch>.
- [6] IMDRF Presentation: Adverse Event Terminology and Coding Working Group, (2017). <http://www.imdrf.org/docs/imdrf/final/meetings/imdrf-meet-170314-canada-presentation-wg-aet.pdf>.
- [7] MEDWATCH Medical Device Reporting Code Instructions, (n.d.). <https://www.fda.gov/MedicalDevices/ucm106737.htm>.
- [8] H. Herre, General Formal Ontology (GFO): A Foundational Ontology for Conceptual Modelling, in: R. Poli, M. Healy, and A. Kameas (Eds.), *Theory and Applications of Ontology: Computer Applications*, Springer, Netherlands, 2010: pp. 297–345.
- [9] H. Herre, B. Heller, P. Burek, R. Hoehndorf, F. Loebe, and H. Michalek, General Formal Ontology (GFO): A Foundational Ontology Integrating Objects and Processes. Part I: Basic Principles (Version 1.0), Research Group Ontologies in Medicine (Onto-Med), University of Leipzig, 2006.
- [10] S.R. Ranganathan, *Colon Classification*, Ed. 7 (1971): A Preview, Sarada Ranganathan Endowment for Library Science, 1969.
- [11] S.K. Sahu, D.P. Mahapatra, and R.C. Balabantaray, Analytical study on intelligent information retrieval system using semantic network, in: 2016 International Conference on Computing, Communication and Automation (ICCCA), 2016: pp. 704–710. doi:10.1109/CCAA.2016.7813813.
- [12] R. Guha, R. McCool, and E. Miller, Semantic Search, in: *Proceedings of the 12th International Conference on World Wide Web*, ACM, New York, NY, USA, 2003: pp. 700–709. doi:10.1145/775152.775250.
- [13] D. Vallet, M. Fernández, and P. Castells, An Ontology-Based Information Retrieval Model, in: *The Semantic Web: Research and Applications*, Springer, Berlin, Heidelberg, 2005: pp. 455–470. [https://link.springer.com/chapter/10.1007/11431053\\_31](https://link.springer.com/chapter/10.1007/11431053_31).
- [14] G.B. Marchisio, K. Koperski, J. Liang, T. Nguyen, C. Tusk, N.S. Dhillon, L. Pochman, and M.E. Brown, Method and system for extending keyword searching to syntactically and semantically annotated data, US7526425 B2, 2009. <http://www.google.ch/patents/US7526425>.
- [15] R. Chauhan, R. Goudar, R. Sharma, and A. Chauhan, Domain ontology based semantic search for efficient information retrieval through automatic query expansion, in: 2013 International Conference on Intelligent Systems and Signal Processing (ISSP), 2013: pp. 397–402. doi:10.1109/ISSP.2013.6526942.
- [16] Y. Lei, V. Uren, and E. Motta, SemSearch: A Search Engine for the Semantic Web, in: *Managing Knowledge in a World of Networks*, Springer, Berlin, Heidelberg, 2006: pp. 238–245. [https://link.springer.com/chapter/10.1007/11891451\\_22](https://link.springer.com/chapter/10.1007/11891451_22).
- [17] P.G. Baker, A. Brass, S. Bechhofer, C. Goble, N. Paton, and R. Stevens, TAMBIS--Transparent Access to Multiple Bioinformatics Information Sources, *Proc Int Conf Intell Syst Mol Biol.* **6** (1998) 25–34.
- [18] W.B. Croft, Boolean queries and term dependencies in probabilistic retrieval models, *J. Am. Soc. Inf. Sci.* **37** (1986) 71–77. doi:10.1002/(SICI)1097-4571(198603)37:2<71::AID-ASI3>3.0.CO;2-4.
- [19] S. Suomela, and J. Kekäläinen, Ontology as a Search-Tool: A Study of Real Users' Query Formulation With and Without Conceptual Support, in: *Advances in Information Retrieval*, Springer, Berlin, Heidelberg, 2005: pp. 315–329. [https://link.springer.com/chapter/10.1007/978-3-540-31865-1\\_23](https://link.springer.com/chapter/10.1007/978-3-540-31865-1_23).
- [20] A. Doms, and M. Schroeder, GoPubMed: exploring PubMed with the Gene Ontology, *Nucleic Acids Res.* **33** (2005) W783–786. doi:10.1093/nar/gki470.
- [21] H.-M. Müller, E.E. Kenny, and P.W. Sternberg, Textpresso: an ontology-based information retrieval and extraction system for biological literature, *PLoS Biol.* **2** (2004) e309. doi:10.1371/journal.pbio.0020309.
- [22] A. De Nicola, and M. Missikoff, A Lightweight Methodology for Rapid Ontology Engineering, *Commun. ACM.* **59** (2016) 79–86. doi:10.1145/2818359.
- [23] K. Tahar, M. Schaaf, F. Jahn, C. Kücherer, B. Paech, H. Herre, and A. Winter, An Approach to Support Collaborative Ontology Construction, *Stud Health Technol Inform.* **228** (2016) 369–373.
- [24] A. Blfgeh, J.D. Warrender, C.M.U. Hilken, and P. Lord, A document-centric approach for developing the tolAPC Ontology, in: *Proceedings of the 7th Workshop on Ontologies and Data in Life Sciences*, CEUR Workshop Proceedings, Halle (Saale), Germany, 2016: pp. B.1–6.
- [25] S. Jupp, M. Horridge, L. Iannone, J. Klein, S. Owen, J. Schanstra, K. Wolstencroft, and R. Stevens, Populous: a tool for building OWL ontologies from templates, *BMC Bioinformatics.* **13** (2012) S5. doi:10.1186/1471-2105-13-S1-S5.