

Ontologies and use case based planning of content delivery

Daniela Burkhardt^{1*} and Stefanie Clesle^{1**}

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

Abstract. This paper's objective was the development of an ontology-based Content Delivery Portal (CDP) in combination with use cases. The aim was to find out how to enrich content in general through an ontology and to investigate the added value of ontologies. In particular, it has been researched how content can be delivered in CDP via an ontological context. The focus of the research team was on the knowledge builder of the ontology system manufacturer *i-views* for modelling the ontology and embedding content. Furthermore, the research team planned a CDP and implemented the model in the CDP *i-views content*. To accomplish this, use cases for users in the context of smart homes were developed to derive which information users want and when and how to grant access to the desired content. The implementation in the CDP was fully based on the ontology which was developed over the course of the project. Although the research team presumes that part of the implementation could have been realised with metadata from a CMS, the ontology can be used to provide context information that goes beyond the metadata that is traditionally assigned to individual topics in a CMS.

1 Introduction

The volume and complexity of products has risen sharply in recent years – also as a result of industrial digitalization and globalization. Traditional print manuals are often no longer sufficient to reflect the complexity of products and product variants. Users also consume products now differently than before and have increased expectations of companies in terms of information procurement [1].

In order to create and manage this increasingly large amount of information, there are different ways of content management and delivery to support both users and companies with this process.

The aim of the hereby presented paper was to research intelligent information in the context of smart homes with a focus on exploring the value that ontologies could add in connection with content delivery.

In the following report, the research team gives an overview of the main results and aspects of ontologies and use case based planning of content delivery.

The authors of this paper worked with the content management system Smart Media Creator, provided by Expert Communication Systems GmbH, with the ontology software *i-views* knowledge builder and with the content delivery portal *i-views content*, provided by intelligent views GmbH.

2 How to plan a CDP based on ontology

2.1 Definitions

First, important terms from the field of intelligent information are explained for better comprehensibility.

2.1.1 Content management systems

A content management system (CMS) is a tool for technical editors to create, manage and publish content. When writing technical documents, technical editors are constantly faced with the challenges of handling multiple content variants, writing in multiple languages and publishing to multiple media. To tackle these challenges, topic-oriented documentation, standardization, single sourcing and the use of metadata are employed as basic principles.

Content is divided into topics (also called “modules”). A topic is a self-contained information unit that can be reused in multiple contexts. Modules are created as XML files according to a defined information model, typically in the form of a DTD (document type definition) or XSD (XML schema definition). Further methods can be applied to achieve standardized content. [2, p. 379ff]. It is crucial to prevent a copy-paste workflow, but rather to reuse individual modules. This allows to trace them back to their single source, thus eliminating duplicates. [2, p. 293].

In a CMS, modules are annotated with metadata. This includes metadata that comes from content lifecycle,

* Daniela Burkhardt: buda1031@hs-karlsruhe.de; daniela.burkhardt1991@gmail.com

** Stefanie Clesle: clst1012@hs-karlsruhe.de; stefanie.clesle@gmail.com

which is often generated automatically by the software, and descriptive metadata that comes from the organization’s individual metadata concept. An often used concept for metadata is the PI-Classification, where classification considers intrinsic and extrinsic metadata, both product-related and information-related. Further metadata can be used to enable efficient variant management and automation processes. [2, p. 362ff]

2.1.2 Content delivery portals

In recent years, content delivery portals (CDP) have gained popularity in technical communication. Before CDP, technical information was aggregated and published from CMS and delivered as a static document to users.

In contrast to this procedure, CDP give users access to modular content that can be retrieved and detached from the document structure it might be embedded into. Other information types, like the metadata, can be provided as well. Users can retrieve content via structured search using facets and filters, classic document structures, text search, automated retrieval mechanisms, or by means of a mixture of the above. CDPs are usually web-based and offer access by various end devices. [3, p. 49]

2.1.3 Ontologies

An ontology describes and represents a field of knowledge by defining concepts and the relationships among them. In order to make the knowledge that is represented by the ontology machine-readable, ontology languages are used, such as RDF or OWL. [4]

Software manufacturers that offer tools for modelling an ontology don’t always use consistent terminology, hence in the following report, definitions for the terminology used by i-views are given which will be applied throughout this paper [5, 6]:

- **Semantic network** (also called “domain”): The semantic network contains knowledge of a part of the world. Objects within the semantic network are modelled within the context of the corresponding area of knowledge.
- **Instance** (also called “object”, “individual”) and **type** (also called “class”, “concept”): Instances are classified into types; types can therefore be more abstract. An instance inherits properties of its corresponding type. In i-views, types can contain subtypes, building a hierarchy based on *is a subtype of* relations.
- **Relation** and **relation type** (also called “relationship”): Relations are classified into relation types. The relation types defines how a relation can be used, the relation itself is the manifestation of a relation type that expresses the relation between instances or types.
- **Attribute** and **attribute type**: Attributes are classified into attribute types. Attributes express properties of other objects in the ontology.

2.2 Key steps

The research team identified the key steps shown in Fig. 1 to plan a CDP based on an ontology in combination with use cases. These steps do not need to be executed in a chronological order but can instead overlap and influence each other.

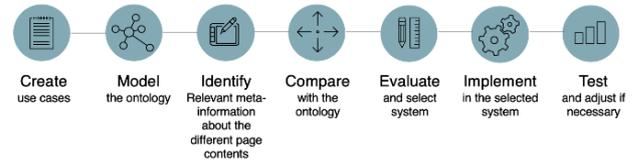


Fig. 1. Overview over the key steps of CDP planning.

2.2.1 Create use cases

The possibilities for building an ontology and a CDP are virtually unlimited. In order to identify what is relevant for users, use cases should be defined at the beginning of a project. [1]

A use case is a tool to describe how users will perform within the functionality of a system that already exists or that is being planned. Therefore the interaction between the actor and the system has to be documented for each use case as a sequence of different steps. The added value consists in reducing the costs and complexity of planning and maintaining the system. Furthermore use cases are also a source for identifying requirements and typical scenarios. [7]

Initially it was determined that the research team would not approach the project component-oriented. Instead, the research team wanted to provide information that enable users to explore the topic of smart homes in general and to link together knowledge.

The modelled use cases had to be constantly adapted and checked throughout the entire project. A tabular form for the best possible development of use cases is recommended. See the following Table 1 for an elaborated example of a use case:

Table 1. Example of use case planning.

Use case	How can I save energy in the kitchen? Anne cooks every day. She needs a lot of energy. She wants to know if there are some smart home devices, which could help her to save energy and costs in the kitchen.
Situation (actor, knowledge level, implementation scale)	<ul style="list-style-type: none"> · private person · knowledge level (in terms of smart home technologies): beginner
Intent	<ul style="list-style-type: none"> · uses a lot of energy in the kitchen; accordingly, wants so save energy
Output	<ul style="list-style-type: none"> · devices that fulfil the purpose energy efficiency and/or cost efficiency · general description of the corresponding devices
Input	<ul style="list-style-type: none"> · purpose: energy efficiency; cost efficiency

	· room: kitchen
Scenario	<ol style="list-style-type: none"> 1. User opens CDP. 2. CDP shows facets. 3. User choses facet “kitchen”. 4. CDP filters content. 5. User choses facet “energy efficiency” and “cost efficiency”. 6. CDP filters content. 7. User choses facet “general description”.

2.2.2 Model the ontology

In this step, the necessary information is modelled in the ontology. System designers can now create suited types for the instances and model the instances accordingly. Instances can inherit information from the according type and if the system allows it, types can be structured in supertypes and subtypes. Attributes can be an option to record information for types or instances. To create an information network, relations are created that connect either a type to a type or an instance to an instance. Mixtures should be avoided. Depending on the software, developers can define which types a relation can connect and restrict the model, e. g. the number of a certain relation for a certain type. [5]

The content that will be provided to users is embedded into the ontology. This could be realised via an automated import or via manual processes. For more information on this, refer to the article “Approaches for deployment and access of content in delivery portals” (Rauffman and Wolf) in this volume. Relations are used to connect the modules’ respective information units to the types and instances of the ontology. For more information on ontology modelling, refer to the article “Ontology based modelling for content delivery systems” (Bürkel and Padeken) in this volume.

It should be pointed out that the ontology must not cover all possible aspects of a domain. In this project, the research teams experienced that it is, in fact, likely to represent “too much” information in the model, including aspects that are not necessary later on. Over the course of the project, the research team reduced and summarized the ontology multiple times, discarding elements that were not required in the use cases. It is highly recommended to refer to the use cases repeatedly during this step.

The following statement applies as a guidance principle: the model must enable the recipient (or a computer) to “read” the ontology, so that the recipient (or a computer) can deduce the information that is required in the use case.

In this project for example, for the use case *How can I save energy in the kitchen*, the model must represent the statement: “A smart refrigerator serves the purpose energy efficiency”. To model this information in an ontology, the type *purpose* with its instance *energy efficiency* is required, along with the type *smart home device* with its instance *refrigerator*. The two instances are connected by a relation that can be evaluated in both directions *serves* resp. *is fulfilled by*.

2.2.3 Identify relevant meta-information

In the next step, system designers identify the meta-information that should be provided in the CDP in different pages and decide how it will be presented.

Helpful key questions during this step are:

- What information can help users access the information they need?
- What context information do users need?

Metadata to help users accessing the required information can be suited to implement as facets or quick access options. Typically, this is information that users know and can therefore enter or select whilst using the CDP.

Metadata that provides context can be suited to implement as visible meta-information for a topic, document or information object. Since the goal of this project was to enable users to explore the topic of smart homes in general and to link together knowledge, the meta-information displayed for a specific topic enables users to put the topic into context and also serve as a starting point to access more related content. Depending on the system, a section of the ontology can be displayed to users either as text or interactive graph display.

In this project, the research team decided to present quick access links for expected input, such as the purpose or the specific smart home device. To allow users to narrow down the hit list, metadata should be presented as facets. When reading a specific topic, the document structure of the source document should be provided, along with descriptive metadata such as the smart home device or the location of the device.

It was also decided to explore more applications of the ontology and enhance the user experience. For this, synonyms are maintained as attributes in the ontology and are considered in search mechanisms. Furthermore, the metadata presented with the content of a specific topic is linked to a visualization of a section of the ontology. Users can start browsing the ontology, getting an insight of the knowledge network and accessing related topics. The research team also planned an interactive search with drop-down-lists to cover typical user intents that should be presented on the start page of the CDP.

2.2.4 Compare with the ontology

After having a clear idea on what will be presented in the CDP, the requirements are compared with the ontology. The purpose of this step is to ensure that the ontology as data basis is complete and efficient for the application in the CDP. System designers check whether the requested information can be derived from the model or if the model needs to be adjusted, e. g. specified to get more precise results. They should also consider if the ontology allows an efficient retrieval of the information or if it can be simplified for easier implementation.

Over the course of the project, the research team introduced a relation as supertype of the three relations *is located at area*, *is located at item* and *is located at environment*. The subtypes were required to indicate the exact location of the device, but it made queries more

complex when all subtypes were meant to be included in the query results. By using the superordinate relation in the hierarchy, the research team was able to define queries more easily and efficiently.

2.2.5 Evaluate and select system

Content delivery systems are extremely costly and vary greatly in their function. In order to select the system most suitable for the individual needs of the respective company, an evaluation should be carried out, which requires a software requirements specification software requirements specification. The decisions made in previous steps result in requirements that need to be met by the system. [8]

The research group used the ontology system i-views knowledge builder and the CDP i-views content. Since the systems had already been selected at the beginning of this project, this step has not been executed.

2.2.6 Implement

After the software has been selected and set up, the implementation can begin. In this step, content and ontology are implemented in the CDP as planned in previous steps.

Depending on the software being used, this can require data transformations via XML and XSLT [9, p.26ff]. The ontology system and the CDP that were used by the research group have the advantage that both systems are interrelated and thus do not require special data transformations. However, a thorough understanding of the functionality as well as the modelled ontology are required.

In both cases, system designers might want to get support from developers during this step. Detailed information on the implementation in this project is given in Chapter 3.

2.2.7 Test

The implementation should be tested before it is released, preferably by a person that is part of the target group. All functionalities should be tested as well as the user experience. Special focus is given to the question of whether the use cases can be fulfilled. Test results should be documented and evaluated to decide whether adjustments are necessary.

Due to the limited time and resources in this project, this step has not been executed.

3 Implementation using software

3.1 Introduction i-views

i-views (intelligent views GmbH) is a company based in Darmstadt, Germany, that manufactures software. Their software solutions enable customers to use semantic technologies to manage data and link it, building knowledge networks. [10]

3.2 Software architecture

As a tool to develop ontologies, i-views offers the i-views knowledge builder. Developers can work with tree structures as well as a graph editor.

The content delivery portal is called i-views content that is based on the data modelled in i-views knowledge builder. Hence, it is not only a CDP that is able to integrate ontologies, but is fully based on ontologies.

As an interface between the two, i-views offers the built-in tool ViewConfig-Mapper (VCM). All panels that are visible in the CDP represent an object in the VCM and can be edited in terms of graphic display, content and functionality. It is not necessary to write code to do so, since the VCM offers a GUI for the developer. The possibility to create queries with an in-built editor should be emphasized here. These queries allow developers to search the knowledge network and evaluate it according to individual requirements. For more advanced actions, extensions with JavaScript can be used. [5, 6]

3.3 Implementation of i-views content

The start page of the CDP offers a search bar within which users can search for keywords. Per default, it is a full-text search mechanism. The research team enriched the full-text search mechanism by the use of synonyms that are maintained in the ontology. In addition, there are also pre-searched questions below to assist users even more effectively with specific use-case-based questions. As described in chapter 2.2.3, these were planned to be implemented as interactive search phrases with drop-down-options, but due to the current state of the technology, this would have required some development effort that could not be realized within the given time frame. Instead, a more simple variation has been implemented.

Via clicking a button on the start page, users get to the overview page of the CDP. Here, one has various quick access options. Based on the use cases, the research team implemented quick access tiles for expected input, such as a specific smart home device and a specific purpose. Also, basic introductions are provided, along with user-defined bookmarks.

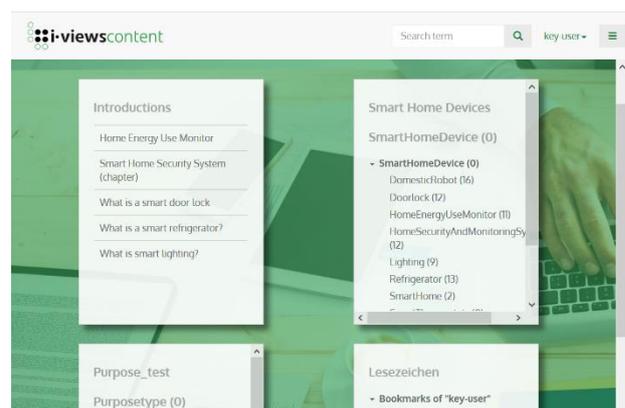


Fig. 2. Overview page of i-views content

The respective search can be narrowed down further by using the provided facets. Topics that match the search parameters are listed with the title, a text excerpt, the

document and chapter they are contained in, the topic version and language. In this project, only one language and no different versions were maintained, but the research team considers this metadata helpful in a real scenario. When viewing a topic, the document structure is provided, from which users can click through all topics available in the respective document. The content of the selected topic is shown together with meta-information, providing context (see Fig. 6).

If users then select one of the facets, they can browse through the ontology. Users can display the connected knowledge network in a graph view or as a list and by selecting different relations, they can identify connections and thus get contextual information.



Fig. 3. Display of the ontology in i-views content

3.4 Show case applying VCM

As described in chapter 3.2, the VCM is used to transform the ontology of the knowledge builder into i-views content. In the following report, a concrete example is presented to illustrate how this works.

According to the use cases in this project, users might be interested in the location of a smart home device. In the knowledge network, the instances of the type smart home device resp. its subtypes. The information that needs to be modelled includes, for example, the statements “The smart device smoke detector is located in the bedroom / entry / hall” and “The smart device doorlock is located at a door, which is located at the entry / balcony / parking lot”. Their representation is shown in Fig. 4.

Within the CDP, the location of a device will be presented as context information when the user is currently viewing a topic about said device. To implement this into the CDP, the corresponding object in the VCM needs to be edited, as displayed in Fig. 5. i-views content offers a plug-in to easily identify the object. [5]

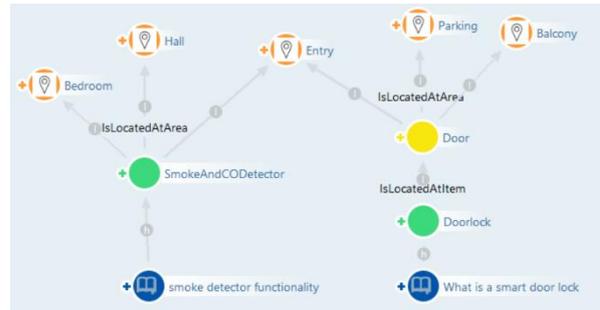


Fig. 4. Relevant section of the ontology

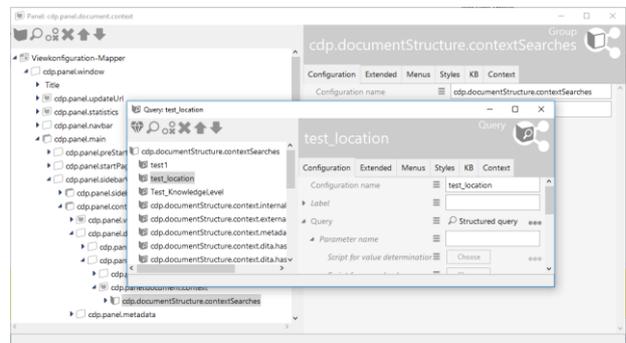


Fig. 5. Relevant object in VCM

A structured query can be defined within the VCM that evaluates the ontology. In this case, the query must cover two different logics. A: A device is located in an area; query will return the area. B: A device is located at an item, the item is located in an area; query will return the area. The query shown in Fig. 7 meets these demands.

The result in the CDP is shown in Fig. 6. All context panels, facets, quick access panels etc. are defined in this manner. In principle, the software i-views can be used to create queries of any complexity that evaluate the ontology.

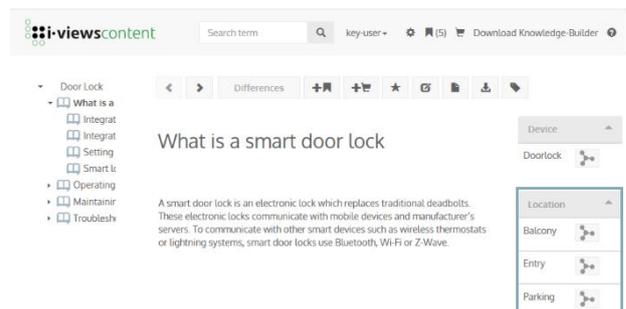


Fig. 6. Topic with context information location

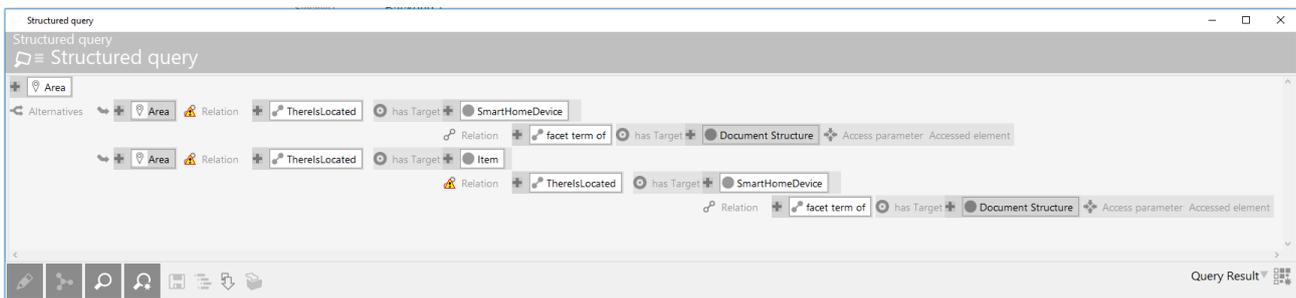


Fig. 7. Structured query in VCM

4 Discussion and conclusion

We have shown the key steps for planning a CDP based on an ontology and use cases. The process of this must be seen as iterative and we want to encourage system designers to check and adapt their decisions throughout the project.

Using ontologies for planning and implementing a CDP offers many possibilities.

It enables system designers to provide different retrieval mechanisms that are far more individual and specific than static documents. Users benefit from modular content and annotated metadata that is presented within the CDP. In this project, the metadata from the CMS could not be imported in the used software due to technical reasons, so the CDP was fully based on the developed ontology. Although we presume that part of the implementation could have been realised with metadata from a CMS, the possibilities of using an ontology go beyond the metadata that is traditionally assigned to individual modules in a CMS. In the ontology itself, developers are not restricted to hierarchical metadata, which is usually the case in a CMS. Relations specifically created for the requested use cases allow to represent connections that are far more complex. While tracing the ontology e. g. by using queries, system designers can benefit from evaluating indirect relations, so implicit knowledge can be gleaned.

Furthermore, the knowledge network itself can be presented to users and, depending on the system being used, allow them to navigate through the ontology, thus recognizing the correlations and connections and easily accessing related information. The CDP obtains an explorative character and supports users to familiarise themselves with a subject autonomously.

Due to the current status of the technology, data cannot be exchanged smoothly between different systems. The lack of cross-tool-standardization led to additional manual effort, which would presumably be eliminated in future versions. The research team estimates that software manufacturers will implement standards such as iiRDS (intelligent information Request and Delivery Standard) developed by the *tekom working group information 4.0* to address this challenge [11].

We would like to express our gratitude to the International Office at Karlsruhe University of Applied Sciences, the University of Aizu, the Faculty of Information Management and Media, and the Master's program of Communication and Media Management at Karlsruhe University of Applied Sciences for the generous financial support that made this project and exchange possible. Furthermore, we would like to thank Prof. Dr. Roy for organizing and coordinating the ACM Chapter Conference. We would also like to thank the Intelligent Views GmbH for providing their software for us to work with, and especially Dr. Steinacker for helping them to tackle technical questions about the knowledge builder and CDP. Last but not least, we would like to thank Mrs. Newcomb Wölflé for her constructive criticism and for proofreading this paper.

- [1] W. Ziegler, H. Beier: *Content delivery portals: The future of modular content*. In: tcworld e-magazine 02/2015 (2015) Retrieved January 19, 2020 from <http://www.tcworld.info/e-magazine/content-strategies/article/content-delivery-portals-the-future-of-modular-content/>
- [2] P. Drewer, W. Ziegler: *Technische Dokumentation: Übersetzungsgerechte Texterstellung und Content-Management*. 2nd Edition. Vogel, Würzburg, Germany. (2014)
- [3] W. Ziegler: *Drivers of Digital Information Services: Intelligent Information Architectures in Technical Communication*. In: ACM Chapter Proceedings on Educational Technology, Language & Technical Communication, Volume 1 (2019)
- [4] W3C: *W3C Semantic Web Frequently Asked Questions*. (2009) Retrieved February 13, 2020 from <https://www.w3.org/2001/sw/SW-FAQ>
- [5] intelligent views gmbh: *i-views 5.2*. (n.d) Retrieved January 10, 2020 from <http://documentation.i-views.com/manual/5.2/a97dbe1b-5931-499d-8dde-034a6a6fe106>
- [6] intelligent views gmbh: *Einführung in die Modellierung*. (n.d.) Training document
- [7] Usability.gov: *Use Cases* (n.d.) Retrieved March 5, 2020 from <https://www.usability.gov/how-to-and-tools/methods/use-cases.html>
- [8] IEEE: *830-1984 – IEEE Guide for Software Requirements Specifications* (02/10, 1984) Retrieved March 4, 2020 from <https://ieeexplore.ieee.org/document/278253>
- [9] K. Hanczaryk: *XML Content Transformation and Import Interfaces for Content Delivery Portals*. In: ACM Chapter Proceedings on Educational Technology, Language & Technical Communication, Volume 1 (2019)
- [10] intelligent views gmbh: *About us*. (n.d.) Retrieved March 2, 2020 from <https://i-views.com/en/about-us/>
- [11] tekom e.V.: *What Is iiRDS, Exactly? A Short Introduction* (n.d.) Retrieved March 10, 2020 from <https://iirds.org/iirds-a-short-introduction/>