iiRDS03 Tutorial

iiRDS for Technical Writers – Introduction to the Metadata

Prof. Wolfgang Ziegler, Karlsruhe University of Applied Sciences
Ulrike Parson, parson AG, Hamburg
Jürgen Sapara, tecteam Gesellschaft für Technische Kommunikation mbH, Dortmund

Dynamic content delivery has recently drawn much attention. There is a variety of use cases connected to topic-based information architectures. Addressing technical writers and information managers, we discuss highly relevant applications of content delivery and the upcoming iiRDS metadata standard. Starting with metadata concepts for structured authoring and content management, we show how to enhance these concepts by adding contextual, linked content with iiRDS.

Theoretical Background of iiRDS

iiRDS is based on established standards and methodologies for technical documentation and metadata. In the following, we briefly summarize how the underlying technologies integrate into the upcoming iiRDS standard.

The PI classification method

The PI classification method has been applied for several years and is a systematic approach to enrich modular information units with metadata. Content management systems (CMS) can use the method to automate several processes within the CMS. PI classification helps information architects to develop a concise concept for creating information, such as topics. The classification is used to derive, for example, the type of content contained in the topics, the size of the topic, or the content delimitation. In addition, it can be combined with writing guidelines for specific information types. In this sense, the PI classification is a natural extension also to semantic information models describing and enforcing the content structures within topics.

In more detail, the PI classification method defines semantic metadata according to four basic classes building up the dimensions of an information space. These classes are organized with respect to products (P) and to information (I):

- **Intrinsic product classes**
  They state the physical or virtual product components that the information is directly connected to. For complex products, they can build a large taxonomy.
- **Intrinsic information classes**
  They define precisely the type of information contained in the modular topic.
  Usually they build a taxonomy of classes like procedural, descriptive, conceptual, safety information, and all corresponding subclasses.
- **Extrinsic product classes**
  They are derived from the individual products of the individual company that uses the CMS. They usually build a hierarchy of all considered product groups.
- **Extrinsic information classes**
  They describe the relevant document types and output media, but can also cover properties like target groups or markets.

As described above, the classification method can be used to automate processes. In a CMS, the focus lies on document creation, which can cover content aggregation by intrinsic and extrinsic metadata. In many cases, the method is used for variant management, for example by filtering master documents or document maps using mostly extrinsic metadata. More complexity is added by additional variant characteristics that describe product variants with additional parameters, such as control types, material, coating, voltage and plugs, or parts numbers. In this sense, they can act as additional filters.

The technology of content delivery can then take advantage of the (PI) classification. Semantic metadata defined in a CMS can be transferred and used in delivery systems. There, they can support users to narrow down content in traditional document views. Alternatively, they show up as search facets within the structured search for topics. Finally, as content delivery should also cover automated request processes in Industry 4.0 and IoT scenarios, they allow web services to retrieve exactly the topics that are relevant.

The Semantic Web

Semantic web technologies have gained increasing importance over the last years. The semantic web extends the normal web by enriching data with additional information (metadata). Humans usually derive such information from the context; machines require explicit statements in machine-readable formats. Statements can be formulated and stored in semantic web standards like RDF (https://de.wikipedia.org/wiki/Resource_Description_Framework).

Technical documentation that is published as web resources, such as HTML pages, can also be enriched by machine-readable metadata. This enables applications, such as content delivery portals or service apps, to query, filter, and search technical documentation based on metadata. Also, applications can use the metadata to link technical documentation with information from other sources, e.g. product information, training material, or service bulletins.

With iiRDS, tekom aims at providing a common vocabulary for metadata describing technical documentation. This is especially important for connected industry scenarios that link products and components from different suppliers in a cyber-physical system. An aggregated technical documentation consisting of information from different suppliers can only be searched and filtered correctly, if it uses a common vocabulary.

iiRDS is published as RDF schema that you can download from https://iirds.tekom.de/ and use under a Creative Commons license.

Use Cases for Intelligent Information

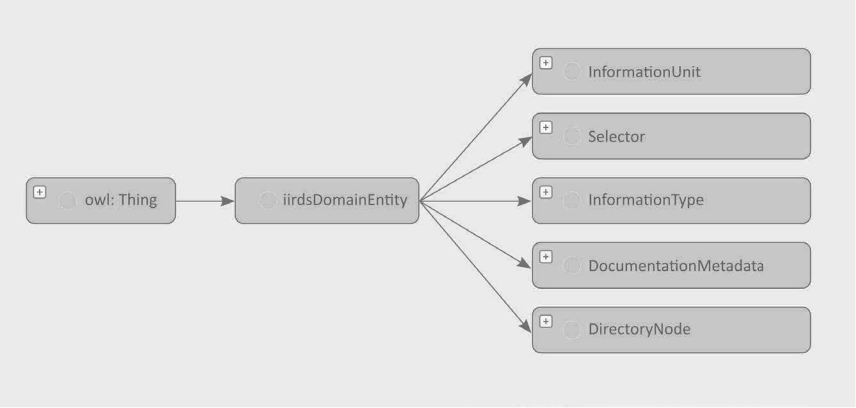
The first work package of the tekom working group “Information 4.0” includes the definition of requirements and use cases for iiRDS. The following table shows some sample requirements which we use in the tutorial to demonstrate the application and use of the iiRDS standard.

Epic	User story
Matching the information to persons, roles, context, and task	**Person recognition:** As anyone I would like to be recognized as a person by the plant so that I can work in a precise and efficient manner, do exactly what is required, and only carry out tasks for which I am currently qualified.
	Fault messages: As a service engineer/user, I would like to be shown error or warning messages in the event of malfunctioning so that I can react to the situation appropriately.
	Filtering service information: As a service engineer/user, I would like to be shown information required for maintaining the plant only, so that I only receive information required for the servicing work.
Information for scheduled servicing	**Spare part information for scheduled servicing:** As a machine, I would like to compile a list of the required spare parts from the product information for scheduled servicing work so that I can order spare parts at the right time.
	Timely servicing information: As a service engineer/user, I would like to receive timely servicing information for all installed modules/components so that I can schedule servicing times and reduce downtimes (scheduled servicing).
Information corresponding to design	**Information corresponds to design:** As a user, I would like to only receive information which corresponds to the current design of my system, including supplied components, so that I can orient myself quickly.
Media-compatible presentation	**Augmented reality coordinated with information on a different terminal device:** As a user/service engineer, when I see information and possible work steps displayed in my data glasses, I would like to be able to select one work step on a different terminal device. That means, I want to be able to use the information on different devices.

The iiRDS Data Model

iiRDS focuses on the common properties of technical documentation. As a standard, it must not contain product- or company-specific metadata. For this reason, the iiRDS vocabulary mainly represents the I classes from the PI classification and functional metadata for applications using intelligent information.

iiRDS defines five main classes for different object types.



The main classes of iiRDS

- InformationUnit: Central class for documentation units, such as topics, documents, or ranges in topics/documents. Most relations to other iiRDS metadata have the InformationUnit as their source (subject of the triple).
- Selector: Assigns the InformationUnit to a physical source file or a range within a file.
- InformationType: Describes the type of content in an InformationUnit. Examples: document type, topic type and subject of the documentation.
- DocumentationMetadata: Parent class for functional metadata (such as required tools, parts, or events) and metadata that describes products, components, and features.
- DirectoryStructure: Provides a navigation structure for intelligent information. Example: table of contents, reading sequence, etc.

Contact information:
wolfgang.ziegler@i4icm.de
ulrike.parson@parson-europe.com
j.sapara@tecteam.de