



www.kim-forum.org

© KIM



Metadatenprofile

Schritt für Schritt

Stefanie Rühle

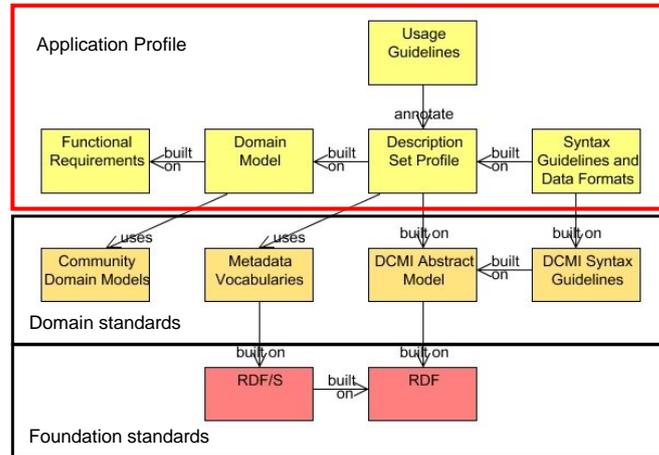
SUB Göttingen

© KIM





Singapore Framework



Überblick

- **Ziele der Anwendung**
- **Funktionelle Anforderungen der Anwendungen**
- **Domain Modell**
- **Beschreibungssatzprofil**
 - Dublin Core Abstract Model
 - Beschreibungstemplates
 - Aussagentemplates
 - Domainspezifische Metadaterme
- **Syntax-Regeln und Datenformate**



Zielsetzung und Geltungsbereich

- **Verpflichtend:**
 - Beschreibe den Kontext in dem das Metadatenprofil verwendet wird oder verwendet werden kann (Anwendungsbereich, Zielgruppen, ...)
- **Optional:**
 - Beschreibe die Personen oder Gruppen, die das Metadatenprofil nutzen sollen
 - Nenne die Organisationen und Personen, die an der Entwicklung des Profils beteiligt waren
 - Nenne Absprachen und Richtlinien bezüglich der zukünftigen Weiterentwicklung und Pflege des Profils



Zielsetzung und Geltungsbereich

- **Dublin Core Collections Application Profile**
<http://dublincore.org/groups/collections/collection-application-profile/index.shtml>

Purpose and Scope

This document describes a DC Application Profile for describing:

- a collection
- a catalogue or index, i.e. an aggregation of metadata that describes a collection

The term "collection" can be applied to any aggregation of physical and/or digital resources. Those resource objects, created objects, "born-digital" items, digital surrogates of physical items, and the catalogues of such aggregation may vary: e.g. by location, by type or form of the items, by provenance of the items, by source and may have varying levels of permanence.

A "collection-level description" provides a description of the collection as a unit: the resource described by its items within that collection. Collection-level descriptions are referred to in Michael Heaney's *An Analytical Model*.

The DC Collections AP specifies how to construct a DC metadata description set that provides a collection level description suitable for a **broad range** of collections, as well as **simple** descriptions of catalogues and index aggregations of the metadata that describe them (catalogues and indices) can be described with similar profiles etc. They are related to each other because the aggregations of physical or digital resources (collections) and indices). In order to distinguish the use of this DCAP either for collections or for catalogues or indices, the



Zielsetzung und Geltungsbereich

- **Scholarly Works Application Profile (SWAP)**

http://www.ukoln.ac.uk/repositories/digirep/index/Eprints_Application_Profile

Objectives

The objectives of the working group are to develop:

- a Dublin Core application profile for eprints (see definition below);
- any implementation / cataloguing rules, that might be necessary to support functionality offered by the search s of the research paper;
- a plan for early community acceptance and take-up, bearing in mind current practice.

In the context of this work an eprint is defined to be a *scientific or scholarly research text* (as defined by the [Budap](#) a preprint, a working paper, a thesis, a book chapter, a report, etc.

The use of the term 'eprints' is not intended to suggest any affiliation with eprints.org software; the application profil

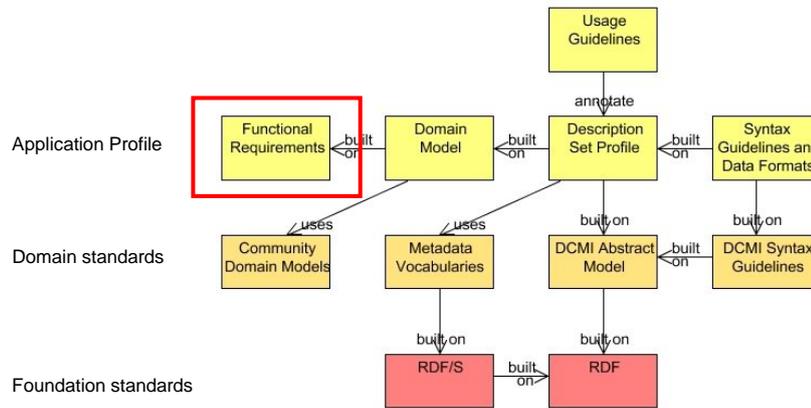


Überblick

- Ziele der Anwendung
- **Funktionelle Anforderungen der Anwendungen**
- Domain Modell
- Beschreibungssatzprofil
 - Dublin Core Abstract Model
 - Beschreibungstemplates
 - Aussagentemplates
 - Domainspezifische Metadaterme
- Syntax-Regeln und Datenformate



Funktionelle Anforderungen



Funktionelle Anforderungen

- **Die funktionellen Anforderungen bilden die Basis des Metadatenprofils. Alle anderen Komponenten müssen diesen Anforderungen entsprechen.**
- **Verpflichtend:**
 - Beschreibt die Funktionen, die das Metadatenprofil unterstützen soll
- **Auf der Basis und im Spannungsfeld**
 - Der Anforderungen der Nutzer
 - Der Eigenschaften der Objekte
 - Der Möglichkeiten des Systems



Funktionelle Anforderungen

- **Functional Requirements of Bibliographic Records (FRBR)**

<http://www.ifla.org/VII/s13/frbr/frbr.htm>

For the purposes of this study the functional requirements for bibliographic records are defined in relation to the following generic tasks that are performed by users when searching and making use of national bibliographies and library catalogues:

- using the data to **find** materials that correspond to the user's stated search criteria (e.g., in the context of a search for all documents on a given subject, or a search for a recording issued under a particular title);
- using the data retrieved to **identify** an entity (e.g., to confirm that the document described in a record corresponds to the document sought by the user, or to distinguish between two texts or recordings that have the same title);
- using the data to **select** an entity that is appropriate to the user's needs (e.g., to select a text in a language the user understands, or to choose a version of a computer program that is compatible with the hardware and operating system available to the user);
- using the data in order to **acquire** or **obtain** access to the entity described (e.g., to place a purchase order for a publication, to submit a request for the loan of a copy of a book in a library's collection, or to access online an electronic document stored on a remote computer).



Funktionelle Anforderungen

- **Dublin Core Collections Application Profile**

<http://dublincore.org/groups/collections/collection-application-profile/index.shtml>

Functional Requirements of the DC Collections AP

The DC Collections AP is intended to provide a means of creating **simple** descriptions of collections and catalogues or primarily to support the discovery and selection of collections, though it may be used to support other functions such as every possible characteristic of every type of collection.

The aim is that the DC Collections AP should support:

- the **discovery** of collections and catalogues or indices of potential interest, by enabling searching on various attributes
 - the name or title of the collection or catalogue or index
 - the subject and coverage of the collection
 - the nature or genre of the items within the collection or catalogue or index
 - the media type or format of the items within the collection or catalogue or index
 - the entity that created the collection or catalogue or index
 - the entity that owns the collection or catalogue or index
- the **identification** of a known collection or catalogue or index and between collections and catalogues
 - relationships between collections, between catalogues or indices and between collections and catalogues
 - the formal identifier(s) of the collection or catalogue or index
 - the name or title of the collection or catalogue or index
 - a textual description of the collection or catalogue or index
- the **selection** of one or more collections or catalogues or indices from amongst a number of discovered collections
 - disclosure of attributes such as:
 - a textual description of the collection or catalogue or index
 - a description of rights held in/over the collection or catalogue or index and conditions of access and use
 - a description of the custodial history of the collection or catalogue or index
 - a description of the way in which items are added to the collection or catalogue or index
- the **identification** of the location of the collection or catalogue or index
- the **identification** of the services that provide access to the collection or catalogue or index



Funktionellen Anforderungen

- **Scholarly Works Application Profile (SWAP)**

<http://www.ukoln.ac.uk/repositories/digirep/index/FunctionalRequirements>

Functional Requirements Specification

Based on the requirements gathering activities indicated above, the *Eprints Application Profile* must support, or make

Richer metadata set

- **Requirement:** Provide a richer set of metadata than is currently possible with simple DC, see [Issues with current](#)
- **Usage scenario:** In current practice, harvesting or cross-searching multiple repositories is faced with a number of Core metadata elements do not offer the level of detail which describing eprints requires. A richer set of metadata metadata and content. ([Richer metadata set](#))
- **Proposed solution:** *ePrints Application Profile* proposes a set of richer metadata.

Consistent metadata

- **Requirement:** Facilitate the creation and sharing of consistent metadata.
- **Usage scenario:** An aggregator is setting up a search service whereby they can provide cross search, browse an wide variety of repositories implemented on different software platforms. The aggregator wants to ensure that the d metadata elements used and the contents of those elements. At present, the use of simple Dublin Core has led to Dublin Core elements is different ways. ([Consistent metadata for aggregator search service](#))
- **Proposed solution:** *ePrints Application Profile* tailored to requirements and will offer guidance.

Preservation metadata approaches

© KIM

Metadatenprofile – Schritt für Schritt



Funktionelle Anforderungen, Zielsetzung und Geltungsbereich

- **Eure Aufgabe:**

- Beschreibt den Kontext, in dem euer Metadatenprofil verwendet werden soll.
- Nennt die Personen, die für die Entwicklung des Profils zuständig sind.
- Beschreibt die Funktionen, die das Profil hat auf der Grundlage der Anforderungen eurer Nutzer.



© KIM

Metadatenprofile – Schritt für Schritt



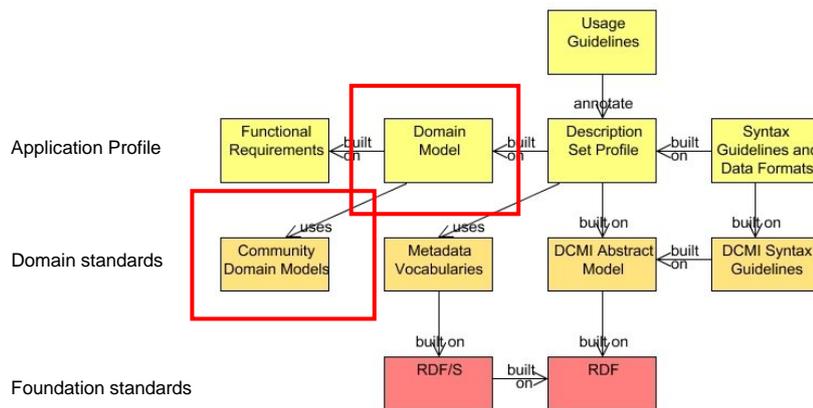


Überblick

- Ziele der Anwendung
- Funktionelle Anforderungen der Anwendungen
- **Domain Modell**
- Beschreibungssatzprofil
 - Dublin Core Abstract Model
 - Beschreibungstemplates
 - Aussagentemplates
 - Domainspezifische Metadaterme
- Syntax-Regeln und Datenformate



Domain Modell





Domain Modell

- **Das Domain Modell beschreibt die Objekte und die Beziehungen zwischen diesen Objekten.**
- **Verpflichtend:**
 - Definiere die Basis-Objekte, die mit Hilfe des Metadatenprofils beschrieben werden sollen.
 - Definiere die Beziehungen, die zwischen diesen Objekten bestehen.
- **Das Domain Modell kann auf einem extern definierten Datenmodell basieren**
 - Nenne das Datenmodell, das deinem Domain Modell zugrunde liegt.
 - Beschreibe Abweichungen von dem Datenmodell, dass du benutzt.

©

um
radaten



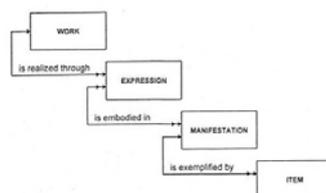
Domain Modell

- **Functional Requirements of Bibliographic Records (FRBR)**

<http://www.ifla.org/VII/s13/frbr/frbr.htm>

The relationships depicted in the diagram indicate that a work may be realized through one or more than one expression (hence the double arrow on the line that links work to expression). An expression, on the other hand, is the realization of one and only one work (hence the single arrow on the reverse direction of that line linking expression to work). An expression may be embodied in one or more than one manifestation; likewise a manifestation may embody one or more than one expression. A manifestation, in turn, may be exemplified by one or more than one item; but an item may exemplify one and only one manifestation.

Figure 3.1: Group 1 Entities and Primary Relationships



© KIM

Metadatenprofile – Schritt für Schritt



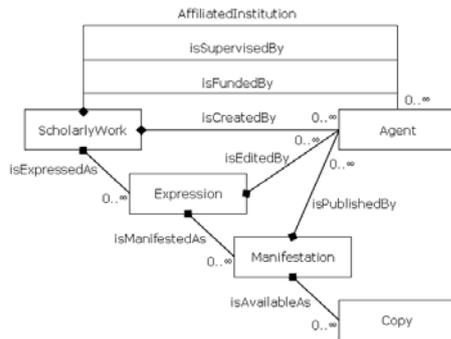


Domain Modell

Scholarly Works Application Profile (SWAP)

<http://www.ukoln.ac.uk/repositories/digirep/index/Model>

Eprint Model



Although FRBR is used as the basis of the model, some of the entity and relationship labels used in dealing with agents:

- a ScholarlyWork is a FRBR Work
- a Copy is a FRBR Item
- an Agent is a FRBR Person or a FRBR Corporate Body
- the isExpressedAs relationship is known as 'is realized through' in FRBR
- the isManifestedAs relationship is known as 'is embodied in' in FRBR
- the isAvailableAs relationship is known as 'is exemplified by' in FRBR
- the isCreatedBy relationship is known as 'is created by' in FRBR
- the isPublishedBy relationship is modelled as the 'publisher' attribute of a Manifestation in FRBR

© KIM

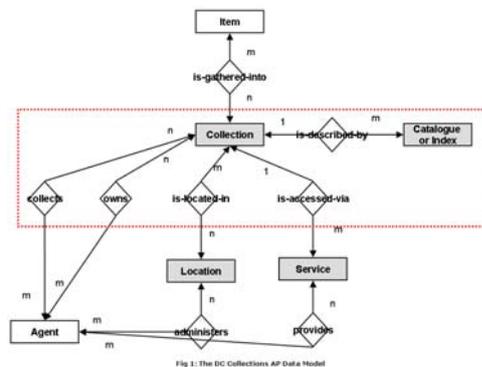
Metadatenprofile – Schritt für Schritt



Domain Modell

Dublin Core Collections Application Profile

<http://dublincore.org/groups/collections/collection-application-profile/index.shtml>



The entity types are:

- **Collection** An aggregation of Items.
- **Item** A physical or digital resource.
- **Location** A place where a Collection is held.
- **Service** A system that provides access to the Items within the Collection.
- **Catalogue or Index** An aggregation of Items which describes a Collection.

The relationship types between entities are:

- an Item is-Gathered-Into one or more Collections
- a Collection is-Located-In zero or more Locations
- a Collection is-Accessed-Via zero or more Services
- a Collection is-Described-By zero or more Catalogues or Indices
- a Collection is-Owned-By one or more Agents
- a Collection is-Collected-By one or more Agents

Fig 1: The DC Collections AP Data Model

© KIM

Metadatenprofile – Schritt für Schritt





Domain Model

- **Eure Aufgabe:**
 - Definiert die Basis-Objekte, die mit eurem Metadatenprofil beschrieben werden sollen (maximal 5)!
 - Definiert die Beziehungen zwischen diesen Objekten



Mein Beispiel

- **Meine Objekte:**



Buch



Figuren



Personen



Organisationen



Mein Beispiel

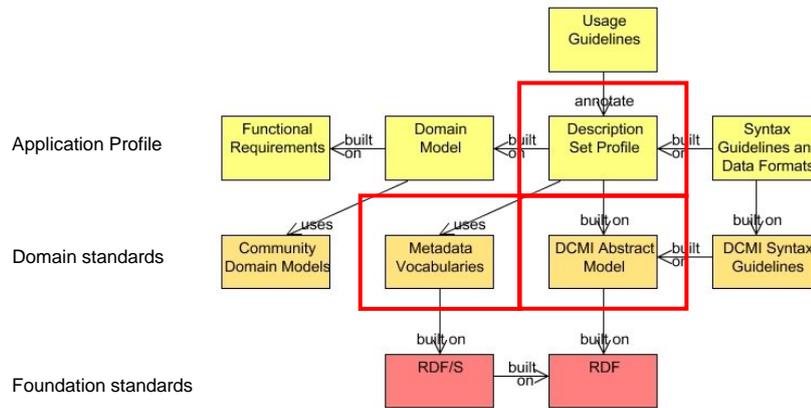


Überblick

- Ziele der Anwendung
- Funktionelle Anforderungen der Anwendungen
- Domain Modell
- **Beschreibungssatzprofil**
 - Dublin Core Abstract Model
 - Beschreibungs Templates
 - Aussagen Templates
 - Domainspezifische Metadaterme
- Syntax-Regeln und Datenformate



Beschreibungssatzprofil



Beschreibungssatzprofil

- **“The Dublin Core Description Set Profile model is designed to offer a simple constraint language for Dublin Core metadata, based on the DCMi Abstract Model. A DSP constrains the resources that may be described by descriptions in a description set conforming to the application profile, the properties that may be used, and the ways a value may be referenced.”**

Siehe:

- DCMi Abstract Model (<http://dublincore.org/documents/abstract-model/index.shtml>)
- Description Set Profiles: A constraint language for Dublin Core Application Profiles (<http://dublincore.org/documents/dc-dsp/index.shtml>)



Überblick

- Ziele der Anwendung
- Funktionelle Anforderungen der Anwendungen
- Domain Modell
- Beschreibungssatzprofil
 - Dublin Core Abstract Model
 - Beschreibungstemplates
 - Aussagentemplates
 - Domainspezifische Metadaterme
- Syntax-Regeln und Datenformate



DCMI Abstract Model

- **Basis ist das RDF-Modell**
- **Definiert, was notwendig ist, um Informationsstrukturen zu beschreiben**
 - Was für Komponenten benötigt eine Informationsstruktur?
 - Wie müssen diese Komponenten miteinander kombiniert werden?
- **Drei „Informationsmodelle“**
 - Ressourcenmodell
 - Beschreibungssatzmodell
 - Modell der Vokabulare



DCMI Abstract Model – Ressourcen Modell

- **Ressourcenmodell**

“Das Abstraktmodell für die zu beschreibenden Ressourcen/Objekte sieht folgendermaßen aus:

- Jede **Ressource** wird mit einem oder mehreren Paaren aus Eigenschaften und Werten (property-value pair) beschrieben.
- Jedes dieser Paare besteht aus einer **Eigenschaft (property)** und einem **Wert (value)**
- **Jeder Wert ist eine Ressource** – eine physische, digitale oder begriffliche Entität ...



DCMI Abstract Modell – Ressourcen Modell

- **Im Ressourcenmodell geht es um:**

- **die zu beschreibenden Objekte**

Ein zu beschreibendes Objekt (oder Entität) kann:

- ein Buch, eine Person, ein Konzept, einfach alles sein

- **und die Beziehungen** zwischen diesen Objekten:

1. X steht in irgendeiner Beziehung zu Y **= X ist das zu beschreibende Objekt, Y ist der Wert**
2. Y steht in irgendeiner Beziehung zu Z **= Y ist das zu beschreibende Objekt, Z ist der Wert**
3. ...

- **Verwendete Terminologie**

- Zu beschreibende Ressource/Objekt/Entität
- Eigenschaft = Art der Beziehung
- Wert = eine andere(s) Ressource/Objekt/Entität



Mein Beispiel



DCMI Abstract Model – Ressourcen Modell

- **Ressourcen Modell**
 - ... – eine physische, digitale oder begriffliche Entität oder ein **Literal** (literal), das mit einer Eigenschaft (property) assoziiert wird, wenn ein Paar aus einer Eigenschaft und einem Wert (property-value pair) benutzt wird, um eine Ressource zu beschreiben. Demgemäß ist jeder Wert entweder ein **literaler Wert** (literal value) oder **kein literaler Wert** (non-literal value).”



DCMI Abstract Model – Ressourcen Modell

- **Ressourcen Modell**
 - Ein **literaler Wert** (literal value) ist ein Wert, der ein **Literal** (literal) ist.
 - Ein **nicht-literaler Wert** (non-literal value) ist ein Wert, der eine physische, digitale oder begriffliche Entität ist.



DCMI Abstract Model – Ressourcen Modell

- **Literal**
 - In sich geschlossen
 - Kann/darf nicht weiter beschrieben werden
 - Der Wert, der verwendet wird, ist ein **“String”**

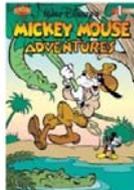
“Ein nach RDF definiertes Literal ist eine Entität, die, um eine Ressource zu bezeichnen, eine Unicode-Zeichenkette sowie optional die Sprache und/oder den Datentyp verwendet.”
- **Nicht-Literal**
 - Verweist auf etwas Anderes
 - Kann/muss weiter beschrieben werden
 - Der Wert, der verwendet wird, ist ein **“Ding”**

“A value which is a physical, digital or conceptual entity.”



Mein Beispiel

- **Meine Basis-Eigenschaften**



Hat einen Titel
Hat eine Sprache
Hat einen Verleger
Hat Figuren



Hat einen Namen
Hat eine Adresse
Hat Freunde
Hat einen Schöpfer
Hat ein Geschlecht



Hat einen Namen



Hat einen Nachnamen
Hat einen Vornamen
Hat ein Geburtsdatum
Hat ein Geburtsland



Hat einen Gründer
Hat ein Gründungsdatum



DCMI Abstract Model – Ressourcen Modell

- **Wenn ich ein Domain Modell erstelle, muss ich entscheiden: Welches sind die „Dinge“?**
- **“things” can be described, “strings” can't. (Pete Johnston)**
- **Ich definiere Werte als „Ding“, weil ich sie aufgrund meiner funktionellen Anforderungen**
 - näher beschreiben muss
 - oder anderen die Möglichkeit geben will, sie zu beschreiben
 - oder die Beschreibungen anderer nachnutzen will
- **Ich definiere Werte als “String”, weil ich sie aufgrund meiner funktionellen Anforderungen**
 - nicht weiter beschreiben will
 - und auch keine Beschreibungen anderer nachnutzen kann/will



Mein Beispiel



© KIM

Metadatenprofile – Schritt für Schritt



Ressourcen Modell

• Eure Aufgabe:

- Macht eine Liste der Basis-Eigenschaften für die Beschreibung eurer Objekte – nicht mehr als fünf für jedes Objekt.
- Entscheidet, ob es sich bei den Werten, die für die Eigenschaften stehen, um Literale oder Nicht-Literale Werte handelt. Beachtet dabei die funktionellen Anforderungen eures Profils und euer Domain Modell.



© KIM

Metadatenprofile – Schritt für Schritt





Überblick

- Ziele der Anwendung
- Funktionelle Anforderungen der Anwendungen
- Domain Modell
- **Beschreibungssatzprofil**
 - Dublin Core Abstract Model
 - Beschreibungstemplates
 - Aussagentemplates
 - Domainspezifische Metadaterme
- Syntax-Regeln und Datenformate



Beschreibungssatzprofil

- **“The Dublin Core Description Set Profile model is designed to offer a simple constraint language for Dublin Core metadata, based on the DCMI Abstract Model. A DSP constrains the resources that may be described by descriptions in a description set conforming to the application profile, the properties that may be used, and the ways a value may be referenced.”**

Siehe:

- DCMI Abstract Model (<http://dublincore.org/documents/abstract-model/index.shtml>)
- Description Set Profiles: A constraint language for Dublin Core Application Profiles (<http://dublincore.org/documents/dc-dsp/index.shtml>)



DCMI Abstract Model - Beschreibungssatzmodell

- „A Description Set Profile specifies how to build a metadata record in terms of templates and constraints“.
 - Beschreibungs-Template
 - Aussagen-Template

Das Beschreibungssatzmodell beschreibt dies folgendermaßen:

- „Ein **Beschreibungssatz** (description set) besteht aus einer oder mehreren Beschreibungen, von denen jede eine einzige Ressource beschreibt.“
- „Eine Beschreibung besteht aus einer oder mehreren **Aussagen** (statements) über eine und nur eine Ressource und Null oder einem **URI der beschriebenen Ressource** (described resource URI), der die beschriebene Ressource identifiziert.“



Beschreibungssatzprofil

- Das Beschreibungssatzprofil dokumentiert, welche Metadaterme innerhalb meiner Anwendung verwendet werden sollen mit allen dazugehörigen Regeln
- Jede Objektklasse, die im Domain-Modell genannt wurde, erhält ein eigenes Beschreibungstemplate. Das Beschreibungstemplate definiert die Klasse, zu der das Objekt gehört und deklariert, was für Aussagen über diese Klasse gemacht werden.
- Jedes Beschreibungstemplate setzt sich aus einer oder mehreren Aussagentemplates zusammen, die die Eigenschaften des Objekts beschreiben.



Beschreibungssatzprofil

- **Bevor ich das Beschreibungssatzprofil erstelle, muss ich prüfen:**
 - Welche der von mir bisher genannten Eigenschaften brauche ich wirklich?
 - Welcher Metadatenstandard ist für meine Anwendung angemessen?
 - Ist es notwendig, Terme aus verschiedenen Standards zu verwenden?
 - Ist es notwendig, eigene Terme zu deklarieren?
- **Dabei ist zu beachten**
 - Die verwendeten Terme sollten DCAM-konform sein



DCAM-Konformität

- Ein syntaxunabhängiges Metamodell
- Basierend auf dem RDF-Modell
- Dient der syntaxübergreifenden Konvertierung von Metadaten
- DCAM unterscheidet vier „Arten“ von Metadatermen:
 - Klassen (classes)
 - Eigenschaften (properties)
 - Syntaktische Kodierungsschemas (syntax encoding scheme)
 - Semantische Kodierungsschemas (vocabulary encoding scheme)
- **Minimale Anforderung für die DCAM-Konformität ist:**
 - Der Term muss explizit einem dieser Metadaten-Typen zugewiesen werden.
 - Der Term muss über einen URI identifizierbar sein.



Beschreibungssatzprofil (BSP)

Metadatenprofil	spezifiziert	Metadaten-anwendung
Beschreibungssatzprofil	spezifiziert	Beschreibungssatz
Beschreibungstemplate	spezifiziert	Beschreibung
Aussagentemplate	spezifiziert	Aussage



DCMI Abstract Model - Beschreibungssatzmodell

- **Beschreibungstemplate**
 - „Das Beschreibungstemplate spezifiziert, welche **Aussagen** über die zu beschreibende Ressource gemacht werden und welche **Einschränkungen** innerhalb der Beschreibung gelten.“
 - „Wenn laut dem Domain Modell mehr als eine Ressource beschrieben werden soll, so enthält das **Beschreibungssatzprofil** für jede Ressource ein separates **Beschreibungstemplate**, wobei die verschiedenen Templates deutlich voneinander abzugrenzen sind.“



Beschreibungstemplate

- **Verpflichtend**
- **Der Header des Beschreibungstemplate muss folgende Informationen beinhalten**
 - Die Klasse/Klassen (class/classes) deren Instanz/Instanzen die beschriebene(n) Ressource(n) sind.
- **Optional**
- **Der Header des Beschreibungstemplate sollte die folgenden Informationen beinhalten**
 - Die Angabe, ob Beschreibungen, die auf dem Template beruhen, allein stehen können.
 - Wie häufig die Beschreibung, die auf dem Template beruht, minimal und maximal in einem Metadatensatz erscheinen darf.



Beschreibungstemplate

- **Klasse**
 - “A group containing members that have attributes, behaviours, relationships or semantics in common; a kind of category”

<http://dublincore.org/documents/abstract-model/#sect-7>
- **Scholarly Works Application Profile (SWAP)**

http://www.ukoln.ac.uk/repositories/digirep/index/Eprints_EntityType_Vocabulary_Encoding_Scheme#ScholarlyWork

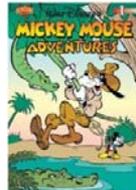
The vocabulary consists of seven classes (listed below) where each of the classes corresponds to one of the key entities in the Eprints Model.

Value URI	Label	Definition
http://purl.org/eprint/entityType/ScholarlyWork	Scholarly Work	A distinct intellectual or artistic creation. A ScholarlyWork equates to a FRBR Work.
http://purl.org/eprint/entityType/Expression	Expression	The intellectual or artistic realization of a ScholarlyWork. A ScholarlyWork may be expressed as several different revisions, translations or other versions. An Expression equates to a FRBR Expression.
http://purl.org/eprint/entityType/Manifestation	Manifestation	The physical or digital embodiment of an expression of a work. A Manifestation equates to a FRBR Manifestation.
http://purl.org/eprint/entityType/Copy	Copy	A single exemplar of a manifestation. A Copy is a concrete entity. A Copy equates to a FRBR Item.
http://purl.org/eprint/entityType/Person	Person	A single individual, living or deceased. Person equates to a FRBR Person.
http://purl.org/eprint/entityType/Organization	Organization	An organization or group of individuals and/or organizations acting as a unit. An Organization equates to a FRBR Corporate Body.
http://purl.org/eprint/entityType/Agent	Agent	A Person or Organization.



Mein Beispiel

- meine Objekte = meine Klassen



Bücher



Figuren



Organisationen



Personen



Mein Beispiel

- **Mein Template**

Beschreibungssatzprofil
Beschreibungstemplate Label: Buch URI: http://example.org/myExample/entityType/Buch Definition: Type of Term: Class
Beschreibungstemplate Label: Organisation URI: http://example.org/myExample/entityType/Organisation Definition: Type of Term: Class
Beschreibungstemplate Label: Person URI: http://example.org/myExample/entityType/Person Definition: Type of Term: Class
Beschreibungstemplate Label: Figur URI: http://example.org/myExample/entityType/Protagonist Definition: Eine fiktive Persönlichkeit Type of Term: Class



Beschreibungssatzprofil

- **Eure Aufgabe:**
 - Wählt zwei eurer Ressourcen aus und entwickelt für diese ein Beschreibungstemplate mit den folgenden Informationen:
 - Label, URI, Definition, Type of Term



Überblick

- Ziele der Anwendung
- Funktionelle Anforderungen der Anwendungen
- Domain Modell
- **Beschreibungssatzprofil**
 - Dublin Core Abstract Model
 - Beschreibungstemplates
 - Aussagentemplates
 - Domainspezifische Metadaterme
- Syntax-Regeln und Datenformate



DCMI Abstract Model

- **Beschreibungssatzmodell**
 - Eine Beschreibung besteht aus einer oder mehreren **Aussagen** (statements) über eine und nur eine Ressource und Null oder einem **URI der beschriebenen Ressource** (described resource URI), der die beschriebene Ressource identifiziert.
 - Jede Aussage (statement) realisiert ein **Paar aus einer Eigenschaft und einem Wert** (property-value pair) und besteht aus dem **URI der Eigenschaft** (property URI), der die Eigenschaft (property) identifiziert und einem Platzhalter für den **Wert** (value surrogate).



Beschreibungssatzprofil – Aussagentemplate

- **Das Aussagen-Template beschreibt die Einschränkungen, die in einer bestimmten Aussage gelten.**
- **Für jeden Metadaten-Term gibt es ein Aussagen-Template. Dies kann die folgenden Informationen beinhalten:**
 - Label des Terms = dient der menschenlesbaren Bezeichnung des Terms
 - Term URI = identifiziert den Term in Form eines URI
 - Definition = beschreibt das Konzept und die Art des Terms
 - Kommentar = zusätzlichen Informationen über den Term oder seine Verwendung
 - Art des Terms = Property, Class, Vocabulary Encoding Scheme, Syntax Encoding Scheme
 - Verpflichtungsgrad = “mandatory”, “optional”, “recommended”
 - Häufigkeit = Angaben zur Wiederholbarkeit
 - Literal / Non-Literal
 - Kodierungsschema (Vocabulary Encoding Scheme oder Syntax Encoding Scheme)
 - Informationen zur Sprache
 - ...



Aussagentemplate

- **Eigenschaft**
 - “A specific aspect, characteristic, attribute, or relation used to describe resources.”

<http://dublincore.org/documents/abstract-model/#sect-7>
- **Scholarly Works Application Profile (SWAP)**

<http://dublincore.org/scholarwiki/SWAPDSP>

Description of the eprint as a ScholarlyWork	
Entity type	
Property	http://purl.org/dc/elements/1.1/type
Min occurrence	1
Max occurrence	1
Literal?	No
Definition	The type nature or genre of the content of the resource.
Eprint-specific recommendation	Each of the entity descriptions in the description sets conforming with this application profile will need to be explicitly typed. This is done using a dc:type statement with one of the following value URIs taken from the @Eprints.Entity.Type Vocabulary Encoding Scheme corresponding to the entity being described.
Value URI Constraint:	
Occurrence	mandatory
Choose from:	http://purl.org/dc/terms/EntityType/ScholarlyWork/
Vocabulary Encoding Scheme Constraint:	
Occurrence	mandatory
Choose from:	http://purl.org/dc/terms/EntityType/
Value String Constraint:	
Max occurrence	0



Beschreibungssatzprofil

- **Singapore Framework ermöglicht die Kombination**
 - verschiedener Metadatenstandards
 - standardisierter Terme mit domain-spezifischen Termen

So lassen sich z.B. kombinieren:

- **DCMI Metadata Terms**
 - <http://dublincore.org/documents/dcmi-terms/>
- **FOAF Vocabulary Specification 0.91**
 - <http://xmlns.com/foaf/spec/>
- **und/oder eigene Terme**



Mein Beispiel – DCMI Metadata Terms

- **Mein Template**

Beschreibungssatzprofil
Beschreibungstemplate Label: Buch URI: http://example.org/myExample/entityType/Buch Definition: Type of Term: Class
Aussagentemplate Label: Titel URI: http://purl.org/dc/terms/title Definition: A name given to the resource Type of Term: property Obligation: mandatory Occurrence: 1/1 Literal / Non-Literal: Literal
Aussagentemplate Label: Sprache URI: http://purl.org/dc/terms/language Definition: A language of the resource Comment: Recommended best practice is to use a controlled vocabulary such as RFC 4646 Type of Term: property ...

© KIM

Metadatenprofile – Schritt für Schritt



Mein Beispiel – FOAF

- **Mein Template**

Beschreibungssatzprofil
Beschreibungstemplate Label: Person URI: http://example.org/myExample/entityType/Person Definition: Type of Term: Class
Aussagentemplate Label: Vorname URI: http://xmlns.com/foaf/spec/#term_firstName Definition: The first name of a person Type of Term: property Obligation: mandatory Occurrence: 1/n Literal / Non-Literal: Literal
Aussagentemplate Label: Nachname URI: http://xmlns.com/foaf/spec/#term_surname Definition: The surname of some person Type of Term: property Obligation: mandatory Occurrence: 1/n

© KIM

Metadatenprofile – Schritt für Schritt





Beschreibungssatzprofil

- **Eure Aufgabe:**
 - Erstellt innerhalb eines eurer Beschreibungstemplates die Aussagentemplates unter Berücksichtigung der folgenden Informationen:
 - Label, URI, Definition, Comment, Type of Term, Obligation, Occurrence, Literal/Non-Literal



Überblick

- Ziele der Anwendung
- Funktionelle Anforderungen der Anwendungen
- Domain Modell
- Beschreibungssatzprofil
 - Dublin Core Abstract Model
 - Beschreibungstemplates
 - **Aussagentemplates**
 - Domainspezifische Metadaterme
- Syntax-Regeln und Datenformate



DCMI Abstract Model

- **Modell der Vokabulare**
 - Spezifiziert die Art der Terme, die im Metadatenkontext verwendet werden
 - Klassen = Classes
 - Eigenschaften = Properties
 - Semantisches Kodierungsschema = Vocabulary Encoding Schemes (VES)
 - Syntaktisches Kodierungsschema = Syntax Encoding Schemes (SES)

und ihre Beziehungen



DCMI Abstract Model – VES und SES

- **Beschreibungssatzmodell**
 - Der Wert ist entweder durch einen **literalen** Platzhalter (literal value surrogate) oder einen **nicht-literalen** Platzhalter (non-literal value surrogate) vertreten.
 - Ein **literaler** Platzhalter (literal value surrogate) ist der Platzhalter für einen literalen Wert (literal value) und besteht aus genau einer Zeichenkette (value string). Die Zeichenkette (value string) ist ein Literal (literal), das den literalen Wert (literal value) codiert.
 - Eine **Zeichenkette** (value string) ist entweder eine **einfache Zeichenkette** (plain value string) oder eine **typisierte Zeichenkette** (typed value string).



DCMI Abstract Model – Beschreibungssatzmodell

- **Beschreibungssatzmodell**
 - Eine einfache Zeichenkette (plain value string) kann verwendet werden mit einer Angabe zur Sprache der Zeichenkette (value string language). Dabei handelt es sich um eine ISO-normierte Sprachangabe (z. B. „en-GB“). Einfache Zeichenketten (plain value string) sollen von Menschen lesbar sein.
 - Eine typisierte Zeichenkette (typed value string) ist verbunden mit dem **URI des syntaktischen Kodierungsschemas** (syntax encoding scheme URI), der das syntaktische Kodierungsschema (syntax encoding scheme) identifiziert.



DCMI Abstract Model – syntaktisches Kodierungsschema

- **Wenn eine typisierte Zeichenkette (typed value string) verwendet werden soll, dann ist ein syntaktisches Kodierungsschema (Syntax Encoding Scheme) erforderlich.**
 - „Syntax Encoding Schemes indicate that the value is a string formatted in accordance with a formal notation, such as "2000-01-01" as the standard expression of a date.“

<http://dublincore.org/documents/usageguide/glossary.shtml>



Aussagen-Template

- **Scholarly Works Application Profile (SWAP)**

<http://dublincore.org/scholarwiki/SWAPDSP>

Eigenschaft mit einem syntaktischen Kodierungsschema

Identifizier							
Property	http://purl.org/dc/elements/1.1/identifier						
Min occurrence	1						
Literal?	Yes						
Definition	An unambiguous reference to the resource within a given context.						
Eprint-specific recommendation	A URI for the eprint.						
Value (Literal)	<table border="1"><thead><tr><th colspan="2">Syntax Encoding Scheme:</th></tr></thead><tbody><tr><td>Occurrence</td><td>mandatory</td></tr><tr><td>Choose from</td><td>http://purl.org/dc/terms/URI</td></tr></tbody></table>	Syntax Encoding Scheme:		Occurrence	mandatory	Choose from	http://purl.org/dc/terms/URI
Syntax Encoding Scheme:							
Occurrence	mandatory						
Choose from	http://purl.org/dc/terms/URI						



DCMI Abstract Model – Beschreibungssatzmodell

- **Beschreibungssatzmodell**

- Ein nicht-literaler Platzhalter (non-literal value surrogate) steht für einen nicht-literalen Wert (non-literal value) und besteht aus Null oder dem **URI des Werts** (value URI), (einem URI, der den nicht-literalen Wert, der mit der Eigenschaft assoziiert ist, identifiziert), Null oder dem **URI des semantischen Kodierungsschemas** (vocabulary encoding scheme URI), (einem URI, der das semantische Kodierungsschema, dem der nicht-literale Wert angehört, identifiziert) und Null oder mehreren **Zeichenketten** (value strings). **Jede Zeichenkette (value string) ist ein Literal (literal), das den nicht-literalen Wert (non-literal value) repräsentiert.**



DCMI Abstract Model – Semantisches Kodierungsschema

- **Im Falle eines Semantischen Kodierungsschemas muss ein nicht-literaler Wert verwendet werden**
 - „Vocabulary Encoding Schemes indicate that the value is a term from a controlled vocabulary, such as the value "China - History" from the Library of Congress Subject Headings.“

<http://dublincore.org/documents/usageguide/glossary.shtml>



Aussagen-Template

- **Scholarly Works Application Profile (SWAP)**

<http://dublincore.org/scholarwiki/SWAPDSP>

Eigenschaft mit einem semantischen Kodierungsschema

Subject	
Property	http://purl.org/dc/elements/1.1/subject
Literal?	No
Definition	The topic of the content of the resource. The topic of the print. In general, choose the most significant and unique words for keywords, avoiding those too general to describe a particular print. For free-text keywords use multiple statements, one for each term. There are no requirements regarding the capitalization of keywords though internal (within archive) consistency is recommended.
Eprint-specific recommendation	Where terms are taken from a standard classification scheme encode each term in a separate statement. Use a vocabulary encoding scheme URI to indicate the classification scheme in use. Encode the complete subject descriptor according to the relevant scheme. Use the capitalisation and punctuation used in the original scheme. Where subject terms are taken from LCSH, the subfields of the subject heading should be separated by double dash (-) and spaces should be omitted. If the subject of the print is a person or an organization, provide their name as a value string according to a note about the form of personal and organizational names used in value strings .
Value (Non-Literal)	Value URI Constraint: Occurrence: optional
	Vocabulary Encoding Scheme Constraint Occurrence: optional
	Value String Constraint: Max occurrence: 1
	Syntax Encoding Syntax Constraint: Occurrence: disallowed
	Language Constraint: Occurrence: optional



Sprachangaben

- „Ein nach RDF definiertes Literal ist eine Entität, die, um eine Ressource zu bezeichnen, eine Unicode-Zeichenkette sowie optional die Sprache und/oder den Datentyp verwendet.“



Mein Beispiel – Sprachangaben

- Jede Zeichenkette kann durch eine Sprachangabe spezifiziert werden

Beschreibungssatzprofil
Beschreibungstemplate Label: Buch URI: http://example.org/myExample/entityType/Buch Definition: Type of Term: Class
Aussagentemplate Label: Titel URI: http://purl.org/dc/terms/title Definition: A name given to the resource Type of Term: property Language constraint: yes Occurrence: 1/1 Literal / Non-Literal: Literal
Aussagentemplate Label: Sprache URI: http://purl.org/dc/terms/language Definition: A language of the resource Type of Term: property Language constraint: yes Literal / Non-Literal: Literal



Beschreibungssatzprofil

- **Eure Aufgabe:**
 - Erweitert euer Aussagen-Template um die folgenden Informationen:
 - Has Syntax Encoding Scheme, has Vocabulary Encoding Scheme, has Language Constraint



Überblick

- Ziele der Anwendung
- Funktionelle Anforderungen der Anwendungen
- Domain Modell
- **Beschreibungssatzprofil**
 - Dublin Core Abstract Model
 - Beschreibungstemplates
 - Aussagentemplates
 - Domainspezifische Metadaterme
- Syntax-Regeln und Datenformate



DCAM-Konformität

- Ein syntaxunabhängiges Metamodell
- Basierend auf dem RDF-Modell
- Dient der syntaxübergreifenden Konvertierung von Metadaten
- **DCAM unterscheidet vier „Arten“ von Metadatermen:**
 - Klassen (classes)
 - Eigenschaften (properties)
 - Syntaktische Kodierungsschemas (syntax encoding scheme)
 - Semantische Kodierungsschemas (vocabulary encoding scheme)
- **Minimale Anforderung für die DCAM-Konformität ist:**
 - Der Term muss explizit einem dieser Metadaten-Typen zugewiesen werden.
 - Der Term muss über einen URI identifizierbar sein.



Domainspezifischer Term

- **Scholarly Works Application Profile (SWAP)**

<http://dublincore.org/scholarwiki/SWAPDSP>

Domainspezifischer Term

Version Number or String

Property	http://purl.org/eprint/terms/version
Literal?	Yes
Definition	A version number or version string associated with the resource.
Eprint-specific recommendation	A version number or version string associated with the described expression of the eprint.



Mein Beispiel – domainspezifischer Term

- **Domainspezifischer Term**

Beschreibungssatzprofil

Beschreibungstemplate
Label: Buch
URI: <http://example.org/myExample/entityType/Buch>
Definition:
Type of Term: Class

Aussagentemplate
Label: Protagonist
URI: <http://example.org/myExample/protagonist>
Definition: Eine fiktive Persönlichkeit, die in einem literarischen Werk eine Rolle spielt
Type of Term: property
Occurrence: 1/n
Literal / Non-Literal: Non-Literal
Refines: <http://purl.org/dc/terms/subject>
Has Vocabulary Encoding Scheme: <http://example.org/myExample/entityType/Figur>



Description Set Profile

- **Eure Aufgabe:**

- Entwerft eine domainspezifische Aussage für eure Objekte, indem ihr im Aussagentemplate folgende Informationen verwendet:
 - Label, URI, Definition, Comment, Type of Term, Obligation, Occurrence, Literal/Non-Literal, has Vocabulary Encoding Scheme, has Syntax Encoding Scheme, Language Constraint





Was wir bisher gelernt haben

- **Ziele der Anwendung**
- **Funktionelle Anforderungen der Anwendungen**
- **Domain Modell**
- **Beschreibungssatzprofil**
 - Dublin Core Abstract Model
 - Beschreibungs Templates
 - Statement Templates
 - Metadaten Terme
- **Syntax-Regeln und Datenformate**



Syntax-Regeln und Datenformate

Gleich geht's weiter ...

Stefanie Rühle
SUB Göttingen
Sruehle(at)sub.uni-goettingen.de

