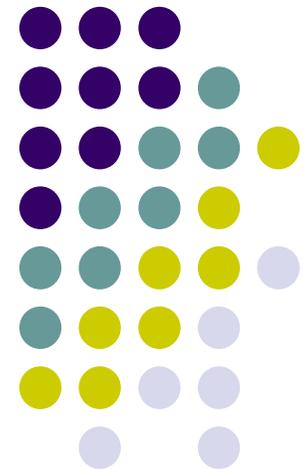


Dublin Core

by Raphael Stür



Agenda



- History of DCMI and Dublin Core
- Dublin Core
 - Simple Dublin Core
 - Qualified Dublin Core
- Das Dump-Down Prinzip
- Encoding Dublin Core
 - Dublin Core und XML
 - Dublin Core und XHTML
- The Warwick Framework
- Fazit

Agenda



- **History of DCMI and Dublin Core**
- Dublin Core
 - Simple Dublin Core
 - Qualified Dublin Core
- Dump-Down Prinzip
- Encoding Dublin Core
 - Dublin Core und XML
 - Dublin Core und XHTML
- The Warwick Framework
- Fazit

DCMI – Dublin Core Metadata Initiative



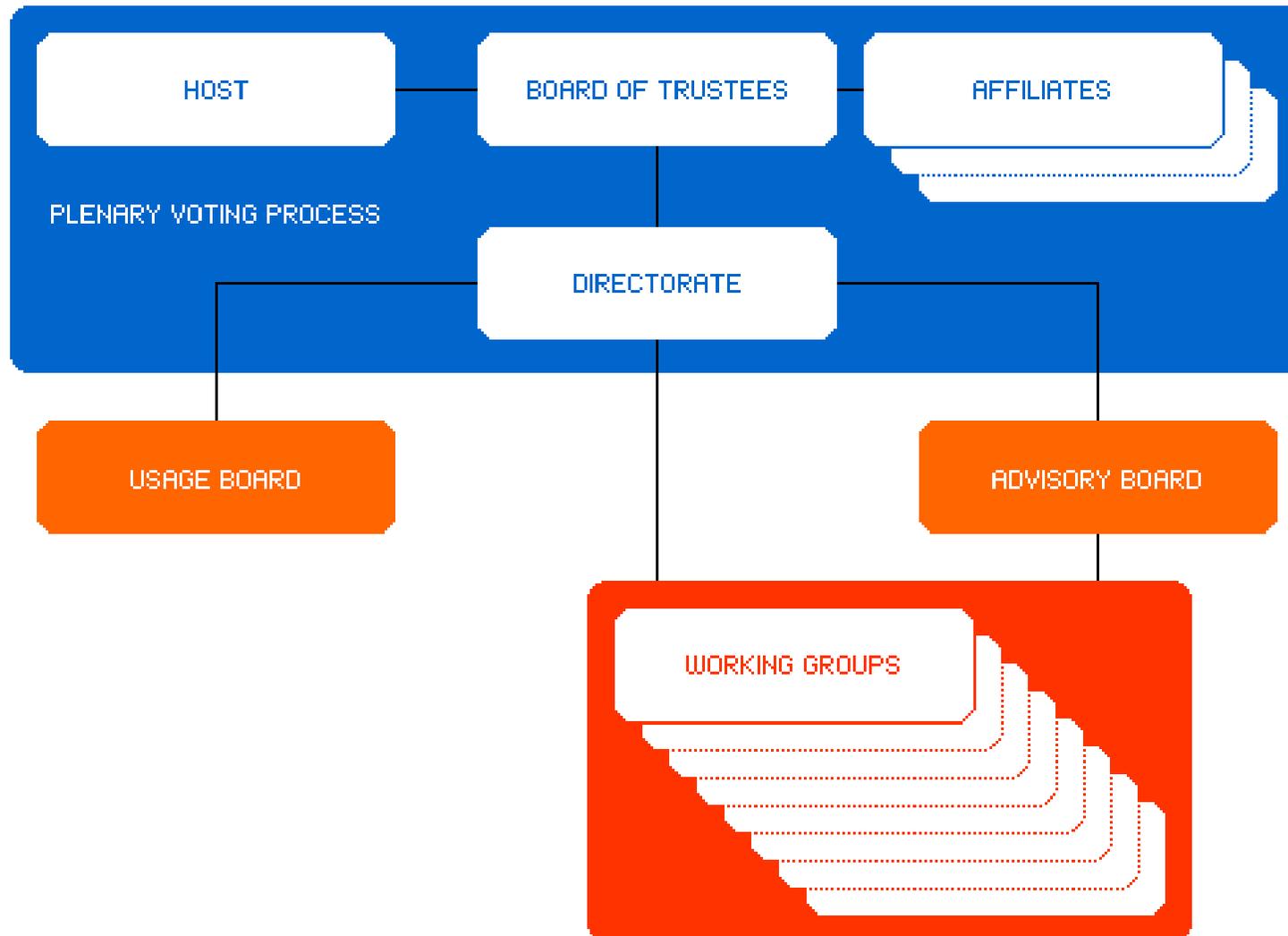
- [3] Ursprünglich ein Workshop über Resource Description
- Wachsendes Interesse führte dann zur Organisation DCMI
- Regelmäßige Workshops und Working Groups für bestimmte Themen

DCMI – Dublin Core Metadata Initiative



- Ziel: Bereitstellung einfacher Standards um Informationen zu finden, zu teilen und zu verwalten. [1]
 - Entwicklung internationaler Standards
 - Definition von Frameworks zur Kombination von Metadata Sets
 - Erweiterung und Entwicklung von Metadata Sets für einzelne Interessengruppen [8]

DCMI – Dublin Core Metadata Initiative



DCMI – Dublin Core Metadata Initiative



- Host: administrative technische Arbeiten und Personalkosten
- Affiliates: Interessenvertreter, entweder lokale Gruppen oder internationale Organisationen. Schließen zur Mitsprache Rahmenverträge mit der DCMI.
- Board of Trustees: Berät die Direktoren in allen Bereichen, entscheidet endgültig über Änderungen bzgl. des Dublin Core

DCMI – Dublin Core Metadata Initiative



- Directorate: verantwortlich für das Tagesgeschäft
- Usage Board: Verantwortlich für die semantische Entwicklung der Metadata terms
- Advisory Board: erteilt technische und strategische Vorschläge, hält Kontakt zu den Arbeitsgruppen
- Arbeitsgruppen: Werden i.d.R. für 1 Jahr für ein bestimmtes Thema gegründet [1]



Agenda

- History of DCMI and Dublin Core
- **Dublin Core**
 - **Simple Dublin Core**
 - **Qualified Dublin Core**
- Dump-Down Prinzip
- Encoding Dublin Core
 - Dublin Core und XML
 - Dublin Core und XHTML
- The Warwick Framework
- Fazit

Dublin Core



- Unterteilung in Simple Dublin Core und Qualified Dublin Core
- Einfaches Basisset an Elementen für den einfachen domainübergreifenden Datenaustausch
- Unabhängig von Implementierungen und Technik
- Leicht erweiterbar
- auf alle Arten von Ressourcen anwendbar [2]
- Übersetzung in verschiedene Sprachen [6]

Dublin Core



- **Grundregeln:**
 - Jedes Element ist optional.
 - Jedes Element ist beliebig wiederholbar. [2]
 - Der Inhalt eines Elements ist anhand der Elementbeschreibung ohne weitere Hilfen lesbar. [4]



Simple Dublin Core

- 15 Elemente in der Version 1.0 von 1997
- 7 neue Elemente seitdem
- Version 1.1 mit ebenfalls 15 Elementen von 1999 als Standard definiert in:
 - ISO Standard 15836-2003 (February 2003):
<http://www.niso.org/international/SC4/n515.pdf>
 - NISO Standard Z39.85-2001 (September 2001):
<http://www.niso.org/standards/resources/Z39-85.pdf>
 - CEN Workshop Agreement CWA 13874 (March 2000)

Simple Dublin Core - Elemente



- **Title:** Der Name der einer Ressource gegeben wird.
- **Creator:** Eine Entität, die an der Erstellung der Ressource maßgeblich beteiligt war. Dies können z.B. Personen oder auch Unternehmen sein.
- **Subject:** Die Überschrift des Themas oder das übergeordnete Thema.

Simple Dublin Core - Elemente



- **Description:** Ein Überblick über den Inhalt der Ressource.
- **Publisher:** Eine Entität, die für die Veröffentlichung verantwortlich ist.
- **Contributor:** Eine an der Ressource mitwirkende Entität.
- **Date:** Ein beliebiges Datum, das im Zusammenhang mit der Ressource steht.

Simple Dublin Core - Elemente



- **Type:** Die Natur bzw. Art der Ressource.
- **Format:** Die Art des physikalischen oder digitalen Vorhandenseins der Ressource.
- **Identifizier:** Eine eindeutige Referenz auf die Ressource. Wie der Identifizier aussieht ist nicht festgelegt im Simple Dublin Core, da dieser vom Kontext abhängig ist.

Simple Dublin Core - Elemente



- **Source:** Eine Quellenangabe, auf was sich die Ressource bezieht bzw. von was die Ressource abstammt.
- **Language:** Die Sprache in der die Ressource verfasst ist. Hier ist keine Programmiersprache oder Ähnliches gemeint!
- **Relation:** Verweis auf eine ähnliche oder weiterführende Ressource.

Simple Dublin Core - Elemente



- **Coverage:** Andere durch die Ressource abgedeckte Bereiche, meist regional oder zeitlich verwendet.
- **Rights:** Sowohl Rechte im Sinne von geistigem Eigentum als auch Zugriffsrechte können in diesem Element angegeben werden. [1]

Zusammenfassung: Simple Dublin Core



- Allgemeine Elemente mit definierter Bedeutung, aber ohne spezielle Aussagekraft.
 - Vorteile:
 - Implementierungen zum Simple Dublin Core sind meist einfach.
 - Vollkommen unabhängig von sprachlichen, regionalen oder fachgebietsbezogenen Besonderheiten in der Interpretation
 - Nachteil:
 - Für die weitere Verarbeitung sind die Informationen in den Elementen i.d.R. zu unspezifisch.

Qualified Dublin Core



- Der Qualified Dublin Core spezifiziert den Simple Dublin Core in zweierlei Hinsicht:
 - Es kann eine genauere Aussage zur Semantik des Inhalts eines Simple Dublin Core Elements gemacht werden. (refinement)
 - Es kann eine Angabe zu dem Format gemacht werden, in dem der Inhalt des Elements verfasst ist. (encoding)
- Das besondere daran ist: Aus dem Qualified kann leicht der Simple Dublin Core gewonnen werden!

Qualified Dublin Core – Auszüge aus den Refinements



- **Abstract:** Eine Zusammenfassung des Inhalts einer Ressource. Dies ist eine inhaltliche Spezifizierung des Elements Description.
- **AccessRights:** Gibt an, dass der Inhalt des Rights Elements die Zugriffsrechte auf diese Ressource beschreibt.
- **Available:** Erscheinungsdatum als ein besonderes Datum im Zusammenhang mit der Ressource.

Qualified Dublin Core – Auszüge aus den Encodings



- Identifier oder Source können als **URI** angelegt sein, womit klar ist, dass es sich dem eindeutigen Verweis um String im Format eines URI handelt.
- Ein Datum könnte als **W3CDTF** definiert sein. Damit wäre klar auf welche Weise die Zahlangaben im String gespeichert sind und ausgewertet werden müssen. [1]

Agenda



- History of DCMI and Dublin Core
- Dublin Core
 - Simple Dublin Core
 - Qualified Dublin Core
- **Dump-Down Prinzip**
- Encoding Dublin Core
 - Dublin Core und XML
 - Dublin Core und XHTML
- The Warwick Framework
- Fazit



Das Dump-Down Prinzip

- Methode, um aus dem Qualified den Simple Dublin Core zu gewinnen:
- Es werden alle Attribute eines Elements einfach ignoriert, die nicht zum Simple Dublin Core gehören.
- Damit verbleibt nur noch das Simple Dublin Core Element und der Inhalt. Deswegen auch die Forderung bei den Regeln, dass der Inhalt ohne Spezialkenntnisse lesbar sein muss. [4]



Agenda

- History of DCMI and Dublin Core
- Dublin Core
 - Simple Dublin Core
 - Qualified Dublin Core
- Dump-Down Prinzip
- **Encoding Dublin Core**
 - **Dublin Core und XML**
 - **Dublin Core und XHTML**
- The Warwick Framework
- Fazit



Encoding Dublin Core

- Die DCMI gibt vor, wie der Dublin Core implementiert werden sollte um die Ziele für die er entwickelt wurde nicht zu untergraben.
- Alle Elementnamen sollten klein geschrieben werden, auch wenn XML nicht case-sensitive ist.
- Spezifikationen für: [1]
 - XML
 - XHTML
 - RDF

Simple Dublin Core und XML



```
<?xml version="1.0"?>
```

```
<metadata
```

```
  xmlns:dc="http://purl.org/dc/elements/1.1/">
```

```
  <dc:title>Dublin Core</dc:title>
```

```
  <dc:creator>me</dc:creator>
```

```
  <dc:creator>myself</dc:creator>
```

```
  <dc:creator>I</dc:creator>
```

```
</metadata>
```

Simple Dublin Core und XML



```
<?xml version="1.0"?>
```

```
<metadata
```

```
  xmlns:dc="http://purl.org/dc/elements/1.1/">
```

```
  <dc:title>Dublin Core</dc:title>
```

```
  <dc:description>Referat zum Thema Dublin  
  Core</dc:description>
```

```
  <dc:date>09.05.2006</dc:date>
```

```
  <dc:creator>Raphael Stür</dc:creator>
```

```
</metadata>
```

Qualified Dublin Core und XML



```
<?xml version="1.0"?>
<metadata
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dcterms="http://purl.org/dc/terms/">
  <dc:title>Dublin Core</dc:title>
  <dc:description xml:lang="de">Referat zum Thema Dublin
    Core</dc:description>
  <dcterms:dateSubmitted xsi:type="dcterms:W3CDTF">
    2006-05-09</dcterms:dateSubmitted>
  <dc:creator>Raphael Stürer</dc:creator>
</metadata>
```

Simple Dublin Core und HTML/XHTML



```
<HTML>
  <HEAD>
    <link rel="schema.DC"
      href="http://purl.org/dc/elements/1.1/" />
    <meta name="DC.title" content="Dublin Core" />
    <meta name="DC.description"
      content="Referat zum Thema Dublin Core" />
    <meta name="DC.date" content="2006-05-09" />
    <meta name="DC.creator" content="Raphael Stüer" />
  </HEAD>
  ....
```

Qualified Dublin Core und HTML/XHTML



```
<HTML>
  <HEAD>
    <link rel="schema.DC"
      href="http://purl.org/dc/elements/1.1/" />
    <link rel="schema.DCTERMS"
      href="http://purl.org/dc/terms/" />
    <meta name="DC.title" content="Dublin Core" />
    <meta name="DC.description" lang="de"
      content="Referat zum Thema Dublin Core" />
    <meta name="DCTERMS.dateSubmitted"
      scheme="DCTERMS.W3CDTF" content="2006-05-09" />
    <meta name="DC.creator" content="Raphael Stür" />
  </HEAD>
  ....
```

Agenda



- History of DCMI and Dublin Core
- Dublin Core
 - Simple Dublin Core
 - Qualified Dublin Core
- Dump-Down Prinzip
- Encoding Dublin Core
 - Dublin Core und XML
 - Dublin Core und XHTML
- **The Warwick Framework**
- Fazit

The Warwick Framework – Idee



- Grundlegendes Problem:
Was tun, wenn mehrere Namespaces und Metadata Sets für z.B. verschiedene Interessengruppen mit zum Teil überlappenden Elementbezeichnern existieren?
- Die Idee:
Man betrachtet die Metadata Sets als Container, in dem mehrere Packages enthalten sein können.

The Warwick Framework – Container



- Der Container kennt nur eine Operation: Nacheinander alle enthaltenen Packages (Metadata Sets) auszugeben.
- Der Empfänger des Containers kann dann entscheiden welche Sets er einfach ignoriert und mit welchen er weiter arbeitet.
- Die Größe jedes enthaltenen Packages muss bekannt sein. ([DCTERMS] extent)

The Warwick Framework – Packages



- Ein Package kann entweder ein Metadata Set, ein indirekter Verweis oder ein weiterer Container sein.
- Ein Verweis zeigt auf ein anderes Informationsobjekt was wiederum zugehörige Metadaten haben oder auch wieder ein Container sein kann. [7]

The Warwick Framework – Vorteile



- Es können z.B. ein Simple Dublin Core Metadata Set und ein sehr spezielles für die eigenen Anwendungen im Container enthalten sein. Dann versteht jeder, der die Metadaten liest, das Simple Dublin Core Metadata Set und kann das spezielle Package einfach ignorieren.

The Warwick Framework – Vorteile



- Die Metadaten können einfach erweitert werden, indem ein Package zu dem Container hinzugefügt wird. Dadurch muss in den bestehenden Abläufen nichts geändert werden und auch verstehen alle anderen Agenten noch die gleichen Packages wie vorher.

The Warwick Framework – Nachteile



- Der Schachtelung der Container ineinander ist keine Grenze gesetzt, also können auch keine Performance Vorhersagen beim Auslesen eines Containers getroffen werden.
- Dieser Effekt wird noch durch die Verweis-Packages verstärkt, da die Ressource, auf die verwiesen wird, ja irgendwo liegen kann und somit evtl. sehr langsame Verbindungen mit sehr hohen Antwortzeiten beteiligt sind.

Agenda



- History of DCMI and Dublin Core
- Dublin Core
 - Simple Dublin Core
 - Qualified Dublin Core
- Dump-Down Prinzip
- Encoding Dublin Core
 - Dublin Core und XML
 - Dublin Core und XHTML
- The Warwick Framework
- **Fazit**



Fazit

- Sinnvoller Standard, der einige, für Ressourcen jeglichen Typs, sinnvolle Metadaten Elemente und deren Semantik definiert.
- Mit dem Qualified Dublin Core wird mit den Encodings sogar noch die Möglichkeit zur maschinellen Weiterverarbeitung der Inhalte gegeben.

Fazit – Vorschlag eines Basisschemas für die Referate



```
<?xml version="1.0"?>
<metadata
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dcterms="http://purl.org/dc/terms/">
  <dc:title>Titel des Referats</dc:title>
  <dcterms:abstract>Inhaltsangabe</dcterms:abstract>
  <dc:language>Sprache</dc:language>
  <dcterms:dateSubmitted xsi:type="dcterms:W3CDTF">
    2006-05-09</dcterms:dateSubmitted>
  <dc:creator>Referent 1</dc:creator>
  <dc:creator>Referent 2</dc:creator>
  <dc:format>z.B. PowerPoint oder PDF</dc:format>
  <dcterms:extent>Größenangabe</dcterms:extend>
</metadata>
```

Quellenangaben



- [1] www.dublincore.org
- [2] Shigeo Sugimoto, Thomas Baker, Stuart L. Weibel. Dublin Core: Process and Principles.
- [3] Stuart Weibel. The State of the Dublin Core Metadata Initiative. D-Lib Magazine, April 1999, Volume 5 Number 4, ISSN 1082-9873.
- [4] Baker, Thomas. A Grammar of Dublin Core. D-Lib Magazine, October 2000, Volume 6 Number 10, ISSN 1082-9873.



Quellenangaben

- [5] Carl Lagoze. Keeping Dublin Core Simple. D-Lib Magazine, January 2001, Volume 7 Number 1, ISSN 1082-9873.
- [6] Thomas Baker. Languages for Dublin Core. D-Lib Magazine, December 1998, ISSN 1082-9873.
- [7] Lagoze, Carl. The Warwick Framework: A Container Architecture for Diverse Sets of Metadata. D-Lib Magazine, July/August 1996, ISSN 1082-9873.
- [8] Stuart L. Weibel. Dublin Core: Mission, Current Activities, and Future Directions. D-Lib Magazine, December 2000, Volume 6 Number 12, ISSN 1082-9873.