A Marriage for Life?

DITA and iiRDS as a Power Couple for Content Delivery
DITA and iiRDS as a Power Couple for Content Delivery
Content Delivery
Content Delivery Powered by DITA

Content Delivery needs:
• Classification
• Taxonomies
• Terminology
• Granular information
• Content chunking
• Information types
• Structured content formats

DITA offers:
• Metadata
• Subject schemes
• Topics, maps
• Topic types
• Transformations via DITA Open Toolkit
Intelligent Information Request and Delivery Standard

- Technical communication will be part of Industry 4.0 and smart factories.
- Standardized metadata facilitates data exchange and data retrieval.
- Information from different manufacturers and sources (e.g. CCMS) needs to be combined in Content Delivery Portals.
Intelligent Information Request and Delivery Standard

- Packaging of content and metadata for exchange
- Extensible terminology for technical communication
- Metadata based on RDF
- Content in any format
Process Overview

Authoring
- Authoring
  - Topics
  - Maps
  - Subject schemes

Publication
- Transform
- Import
  - iiRDS Package
    - HTML
    - RDF

Delivery
  - Use

Authoring Pipeline:
1. Author
2. Classify

Tools:
- DITA-OT
- CDP
Authoring

Authoring:
- Topics
- Maps
- Subject schemes

Authoring Process:
1. Author
2. Classify

Publication:
- PDF
- HTML
- RDF

iiRDS Package:
- HTML
- RDF

Delivery:
- Use

Tools:
- DITA-OT
- CDP
DITA Solution

DITA files + iiRDS

Specialized elements and attributes, e.g. `<document-type>`

Standard elements, e.g. `<prodname>`, `<othermeta>`
DITA iiRDS Unspecialized

- Extract iiRDS metadata from standard DITA metadata elements. `<prodname>`, `<component>`

- Extract metadata from generic DITA metadata elements. `<othermeta name="..." value="...">`

Pros:
- Generic approach (<`othermeta`>)
- No adaption to content model needed

Cons:
- Author needs to know names and values
- Error-prone

```xml
<task>
  <title>Mounting the rotor</title>
  <prolog>
    <metadata>
      <prodinfo>
        <prodname>T5-DH2</prodname>
        <vrmlist>
          <vrm version=""/>
        </vrmlist>
        <component>Rotor</component>
      </prodinfo>
      <othermeta name="ProductLifecyclePhase" content="Assembly"/>
    </metadata>
    <prolog>
    <taskbody>
      ....
    </taskbody>
  </task>
```
Plugin for Specialized iiRDS Content

• Relax NG schemas for DITA 1.3
• iiRDS domain for iiRDS elements specialized from <data>
• Attributes specialized based on @props
• Duplication of some existing elements/attributes: prodname/@product > iirds-product, component > iirds-component

```xml
<prolog><metadata>
  <iirds-productmeta>
    <iirds-product>X5-D</iirds-product>
  </iirds-productmeta>
  <iirds-othermeta>
    <information-subject>Technical data</information-subject>
    <document-type>Maintenance instructions</document-type>
    <document-type>Quick Quide</document-type>
    <document-type>Repair-instructions</document-type>
    <document-type>Operation instructions</document-type>
  </iirds-othermeta>
</metadata></prolog>
```
Current State of iiRDS Metadata
iiRDS Domain and Attributes

iiRDS domain: elements reflect iiRDS classes and subclasses

iiRDS attributes: classes and subclasses provided via subject scheme map
Subject Scheme Support

- DITA:
  - Subject scheme maps to control attribute values
  - Taxonomies for metadata attributes
  - Separation of ID and naming of concepts
  - Multi lingual

```xml
<subjectdef keys="document-typeClass">
  <subjectdef keys="administrator-guide">
    <topicmeta>
      <navtitle>Administrator guide</navtitle>
    </topicmeta>
  </subjectdef>
  <subjectdef keys="operation-instructions">
    <topicmeta>
      <navtitle>Operating instructions</navtitle>
    </topicmeta>
  </subjectdef>
</subjectdef>
```
Subject Scheme Example

DITA and iiRDS as a Power Couple for Content Delivery

Subject Scheme Example

<subjectScheme>
  <subjectHead><subjectHeadMeta><navtitle>Skill level</navtitle></subjectHeadMeta></subjectHead>

<subjectdef keys="qualification-roleClass">
  <subjectdef keys="service-technician">
    <topicmeta><navtitle>Service technician</navtitle></topicmeta>
  </subjectdef>

<subjectdef keys="operator">
  <topicmeta><navtitle>Operator</navtitle></topicmeta>
</subjectdef>

</subjectScheme>

<row information-subject="diagnostics" qualification-role="serviceTechnician">
  <entry>The rotor does not turn even though the device is switched on.</entry>
  <entry>The gear unit is damaged.</entry>
  <entry>Have the gear unit replaced.</entry>
</row>

<iirds:requires-qualification>
  <iirds:Role>
    <rdfs:label>Service technician</rdfs:label>
  </iirds:Role>
</iirds:requires-qualification>
Publication

Authoring

Topics
Maps
Subject schemes

Authoring

Publication

Publication

iiRDS Package

• HTML
• RDF

Transform

Import

CDP

Use

DITA-OT

26 October 2017
DITA Transformation Process
DITA-OT Transformation Pipeline

Image source: http://www.dita-ot.org/2.4/dev_ref/processing-structure.html
DITA-OT iiRDS Pipeline

DITA Map & Topics

Common Preprocessing Stage

Preprocessed map & topics

Filter attributes

Extract metadata

HTML-based formats

Transform Topics to HTML

XHTML / HTML5

Copy assets (CSS, images, etc.)

Eclipse

HTMLHelp

JavaHelp

aa5c

aa5c

aa5c

aa5c

aa5c

aa5c

aa5c

aa5c

aa5c

aa5c

aa5c

aa5c

aa5c

aa5c

iiRDS RDF

iiRDS packaging
Overall Usage Scenario

CMS → <diita> → DITA Open Toolkit → CDP → Browser

XML editor
The Full Process

Authoring
- Topics
- Maps
- Subject schemes

Publication
- iiRDS Package
  - HTML
  - RDF

Delivery

Author → Classify

Transform → Import

DITA-OT → CDP

26 October 2017

DITA and iiRDS as a Power Couple for Content Delivery
Lessons Learned

Integrating DITA and iiRDS works.

BUT: Open questions and issues remain…
iiRDS Transformation Scope and Limitations

Transformation of one DITA root map results in:

- One iiRDS document (without content)
- Multiple iiRDS topics (in HTML/or restricted XHTL format) according to chunking settings in DITA map
- Navigation
- Based on iiRDS RFC (October 20, 2017)

Some limitations still exist.
Metadata vs. Filtering

- Dilemma: Separation of metadata and filtering in DITA
- Metadata elements: no filtering
  
  <iirds-productmeta>
  <iirds-product>T3-H1</iirds-product>
  <iirds-component>Rotor</iirds-component>

- Filtering attributes: not part of generated output
  
  <topic iirds-product="T3-H1" iirds-product="rotor"/>

What is needed?
Semantic Overlap between iiRDS and DITA

Some elements are inherent to DITA, e.g. topic type.

Solution:
- Evaluate semantic structure, already done for topic types.
- Expand to more elements like `<hardardstatement>`.

Some metadata elements or attributes already exist in DITA, e.g. `<prodname>`.

Solution:
- Decision to duplicate elements with iiRDS prefix, e.g. `iirds-product`.
- Allows to group elements in iiRDS metadata classes.
- Clear separation of iiRDS and DITA metadata.
The Full Process

Authoring

Topics
Maps
Subject schemes

Author ➔ Classify

Publication

iiRDS Package
- HTML
- RDF

Transform ➔ Import

Delivery

Use

DITA-OT

CDP
To Do’s and Open Questions

• Update DITA plugin to fully support iiRDS RFC (leftover tasks).
• Convert Relax NG schemas to DTD.
• Provide preprocessing to use metadata elements for filtering?
• Provide data typing for some metadata elements, e.g. Planning Time?
• Make metadata extraction and mapping from terms to concepts configurable.
• Generate subject scheme maps from RDF files.
Next Steps: Going Open Source

- Provide updated plugins and sample content via GitHub.
- Testing and feedback from interested parties.
- Discuss concept and issues with DITA community.
Thank you!
Questions?

Marion Knebel
Senior Technical Communicator
Parson AG
marion.knebel@parson-europe.com
www.parson-europe.com

Dr. Martin Kreutzer
Principal Solution Engineer
Empolis Information Management GmbH
martin.kreutzer@empolis.com
www.empolis.com