Specification REx Format 1.1

Authors:

Prof. Dr. Wolfgang Ziegler, University of Applied Sciences Karlsruhe

Claudia Oberle, M. Sc. Technical Communication, University of Applied Sciences Karlsruhe

Document version 12

Table of Contents

1 Introduction 1

2 REx 1.1 Level 1 2

2.1 Version and Level 2

2.2 Information on the CCMS 2

2.3 Representing the Objects in the CCMS as Nodes 4

1 Introduction

This specification defines the Report Exchange format (REx format). It is used to measure, record and analyze the reuse of modular content in Component Content Management Systems (CCMS) in a standardized way. The REx format distinguishes between two levels of the format depending on the information contained in the report file. Level 1 only includes basic data on the reuse of modular content and is exported directly from a CCMS. External scripts provide features for further, detailed analyses that belong to the information contained in level 2 of the format. In principle, it is also possible that a CCMS generates a report in level 2 of the REx format.
External scripts providing features for dashboard-like visualizations of the data are also part of the REx project.

Two XML schemas (XSD files) define the structure, content and semantics of the current REx format version 1.1. There is one XML schema for each level of the REx format. Some structures contained in level 1 are required for level 2 as well. That is why the XSD file for level 1 is included in the XSD file for level 2 using *xs:include*. In doing so globally-defined elements and types from level 1 can be used in level 2 and derived by extension and restriction.

The following chapter describes level 1 of the REx format 1.1 in detail. Structures that are used in both level 1 and level 2 are only explained in chapter *2 REx 1.1 Level 1*. Differences concerning these structures in level 2 are mentioned, where necessary, in this chapter as well. The description of level 2 is not included in this document.
Mandatory elements and attributes are marked as “required”.

2 REx 1.1 Level 1

Level 1 of the REx format contains the basic data on the reuse of modular content. Considered objects in the CCMS are documents, modules, (module-)fragments or media objects. These objects are also referred to as “nodes”. The basic data from level 1 is used as input for calculating the various quantities.

The following information is defined by the REx XML schema (see file REx_1.1-Level_1.xsd):

2.1 Version and Level

The version and level of the REx format is stated. The root element is *REx1* (or *REx 2* for a level 2 file) and has the attributes *rexVersion* with the restricted value *1.1* and *rexLevel* with the restricted value *1* (or *2* for a level 2 file).

2.2 Information on the CCMS

The complex required element *meta* contains information on the CCMS. To make the report anonymous, all REx files (level 1 and/or 2 or only graphics) can be generated without information on the CCMS. For example various parameters can be passed to the external XSL scripts in order to anonymize the reports.

2.2.1 Company that Uses the CCMS

Required element: *sysOwner*, datatype: *xs:string*

2.2.2 Provider/Version of the CCMS

Required element: *sysType*, datatype: *xs:string*

2.2.3 Timestamp for Report

Required element: *sysTime*

Note: The datatype of this element is derived by union of the datatypes *xs:dateTime* and *xs:string* with the restriction to the value *none*. Examples are:
2001-12-18T09:30:47Z or *none*.

2.2.4 Specific Information on the Intended Use of the CCMS (Product, Production Sites, Department, etc.)

Required element: *sysVersion*, datatype: *xs:string*

2.2.5 Type of Analysis

Complex required element: *analysisType*
The REx format distinguishes two types of analysis and therefore two child elements of the element *anaysisType*:

2.2.5.1 Analysis of Specific Output Channel(s)

Element: *sysChannel*
Required attributes: *docType*, *lang* and *mediaChannel*
Optional attribute that is only used in REx 1.1 level 2: *obsoleteNodes*

The user chooses in the CCMS the output channel(s) to be considered for the export of a REx file. Information on these channels is given in the three required attributes of the element *sysChannel* that specify the document type (e. g. *operating manual*), the language (e. g. *en*) and the media output channel (e. g. *PDF*).

Note: For each required attribute of *sysChannel* a list datatype is used that may contain a sequence of space-seperated items belonging to the datatype *xs:token*.
That is why whitespaces that should not be recognized as separator of items have to be replaced in the list of values by "_" (underscore). By this means the values are separated correctly.
Implicitly no interdependent meta data is expected and therefore the cross product of the attribute values is built. For example, if various languages and media output channels are given, all given media output channels are considered to possibly occur in all given languages.
The attributes *docType*, *lang* and *mediaChannel* will be used for the element *node* (representing an object in the CCMS) as well (see chapter 2.3 Representing the Objects in the CCMS as Nodes).
If the export is an analysis of one or more output channels, the values of the attributes *docType*, *lang* and/or *mediaChannel* belonging to the element *node* have to be included in the list of values in the corresponding attributes of the element *sysChannel*.
Currently, individual output channels are not analyzed.

The optional attribute *obsoleteNodes* is restricted to the values *included* or *not_included* and is only used in REx 1.1 level 2. Obsolete nodes are objects that are not used in any document in the CCMS. The attribute *obsoleteNodes* describes whether obsolete nodes were included when generating the REx 1.1 level 2 file or not.

Note: The value *not_included* indicates that obsolete nodes were not considered for calculating the various quantities and that the corresponding elements *node* have been filtered out. Therefore these elements are not included in the REx1.1 level 2 file.

2.2.5.2 Global Analysis of the CCMS

Element: *sysGlobal*
Required attributes: *docType*, *lang* and *mediaChannel*

This option is the simplest analysis including data of all objects in the CCMS for the export as a REx file.

Note: No output channel is analyzed that is why the value of all required attributes is restricted to *all*.

2.2.6 Unit of Segment Size

Optional element *default_segmentType* in REx1.1 level 1, required element in REx1.1 level 2
This element indicates the unit of the segment sizes (e. g. *word*) that is used for calculating the various content-weighted quantities.
The value of this element is restricted to the following enumerated values:

- *word*
- *phrase*
- *char*
- *element*
  XML elements
- *textElement*
  XML elements that contain only text (e. g. no mixed content)
- *userspecific*
  userspecific unit for the segment size
- *none*

2.3 Representing the Objects in the CCMS as Nodes

The basic data provides information on the various nodes of a CCMS and their structure. The complex required element *raw* contains this basic data and therefore the following information:

Each document, module, fragment or media object is represented by an individual element *node*. In addition all directly (re-)used nodes (belonging to the next lower level of the reuse hierarchy) are represented by a reference consisting of a unique key. This key is made up of the ID, language and version of the referenced node.

The number of nodes is not restricted but there has to be at least one element *node*. There are additional attributes and child elements to add more information to a node.

Complex required element: *node*
Required attributes: *node_Type, id, lang, version*
Optional attributes: *mediaChannel, docType, dateInitial, dateFinal*
Child elements: *size*, *reuse*

Note: The REx file has to contain one element *node* for each version of an object.

2.3.1 Type of Node

The required attribute *node_Type* indicates the type of a node. The value of this attribute is restricted to the following enumerated values:

- *doc*
  document at top level of reuse hierarchy
  Note: Each element *node* with the value *doc* for the attribute *node_Type* is considered and treated for calculation as an aggregation of modular objects that is published (e. g. no master document).
- *module*
  modular objects directly including textual content or reference other modules, media objects and fragments
- *frag*
  fragment (smaller content chunk reused inside a module, e. g. from the reuse pool)
- *graphic*
- *audio*
  audio file

- *video*
  video file
- *userspecificMedia*
  userspecific media object; additional media types can be specified

2.3.2 ID of Node

The required attribute *id* states the ID of a node. The datatype *xs:string* is assigned to this attribute. The datatype *xs:ID* is not used because of its strict restrictions. The value of the attribute represents the unique identifier of the object in the CCMS. For future use and development of REx one goal is to retrieve directly the objects in the CCMS using links and these unique identifiers.

2.3.3 Language of Node

The required attribute *lang* states the language of a node.

The datatype of this attribute is derived by union of the datatypes *xs:language* and *xs:string* with the restriction to the value *none*. Examples are:
en-US for American English, *de* for German, etc., *none*.

Note: If the export is an analysis of output channels including this output channel, the values of the attributes *lang* belonging to the elements *node* have to be included in the list of values in the corresponding attribute *lang* of the element *sysChannel*.
If the element *node* represents a document or module that references other objects, the attribute *lang* has to list all distinct values of the attributes *lang* of the referenced objects in a space-seperated sequence. That is why whitespaces that should not be recognized as separator of items have to be replaced in the list of values by "_" (underscore). By this means the values are separated correctly.

2.3.4 Version of Node

The required attribute *version* states the version of a node. The datatype *xs:double* is assigned to this attribute. In general this attribute's value represents the release version of an object in the CCMS as an integer value. Non-numeric version values have to be converted during the export of the REx file.

2.3.5 Media Output Channel of Node

The optional attribute *mediaChannel* states the output format like for example *PDF* for nodes representing documents (value *doc* for the attribute *node_Type*).
For the attribute *mediaChannel* a list datatype is used that may contain a sequence of space-seperated items belonging to the datatype *xs:token*.
That is why whitespaces that should not be recognized as separator of items have to be replaced in the list of values by "_" (underscore). By this means the values are separated correctly.

Note: If the export is an analysis of output channels including this output channel, the values of the attributes *mediaChannel* belonging to the elements *node* have to be included in the list of values in the corresponding attribute *mediaChannel* of the element *sysChannel*.

2.3.6 Document Type of Node

The optional attribute *docType* states the document type like for example *operating manual* for nodes representing documents (value *doc* for the attribute *node_Type*).

For the attribute *docType* a list datatype is used that may contain a sequence of space-seperated items belonging to the datatype *xs:token*.
That is why whitespaces that should not be recognized as separator of items have to be replaced in the list of values by "_" (underscore). By this means the values are separated correctly.

Note: If the export is an analysis of output channels including this output channel, the values of the attributes *docType* belonging to the elements *node* have to be included in the list of values in the corresponding attribute *docType* of the element *sysChannel*.

2.3.7 Creation date of Node

The optional attribute *dateInitial* states the creation date of a node in a particular release version. The datatype *xs:dateTime* is assigned to this attribute.

Note: An example for the value of this attribute is *2001-12-18T09:30:47Z.* The value *none* is not allowed for this attribute as the attribute itself is optional.

2.3.8 Release Date of Node

The optional attribute *dateFinal* states the release date of a node. The datatype *xs:dateTime* is assigned to this attribute.

Note: An example for the value of this attribute is *2001-12-18T09:30:47Z.* The value *none* is not allowed for this attribute as the attribute itself is optional.

2.3.9 Segment Size of Node

The complex required element *size* states the segment size of a node. It is possible to specify the segment size in various units at the same time.

If it is not possible to state the segment size of a node, the element <*none*>*1*</*none*> has to be used.

The following sequence of child elements is available to specify the segment size in various units:

- *word*
  segment size in words (datatype *xs:double*)
- *phrase*
  segment size in phrases (datatype *xs:double*)
- *char*
  segment size in characters (datatype *xs:double*)
- *element*
  segment size in XML elements (datatype *xs:double*)
- *textElement*
  segment size in XML elements that contain only text, e. g. no mixed content (datatype *xs:double*)
- *userspecific*
  segment size in a userspecific unit (datatype *xs:double*)

Note: The segment size does not have to be specified in each unit. In general only the unit *word* or *char* (character) is used.
If more than one unit is choosen, the child elements need to appear in the right order listed above (*word, phrase, char, element, textElement, userspecific*).
If the segment size of a node is not available in the unit specified in the element *default_segmentType,* various content-weighted quantities cannot be calculated.

In case of nodes representing documents or modules that reference other objects, the element *size* only specifies the segment size of the node itself without the additional segment sizes of any reused nodes.
The overall segment size including reused nodes is only calculated for level 2 of the REx format.

2.3.10 Reused Nodes

The complex element *reuse* is a reference to a reused node belonging to the next lower level of hierarchy.

Each element *reuse* represents one referenced node. The number of these elements is unbounded. If a node does not reference any other nodes, no element *reuse* is used.

The reference is a unique key made up of the attributes *id*, *lang* and *version* of the element *reuse*.

Note: For an XML instance to be valid according to the XML schemas for REx 1.1 level 1 and 2 each element *reuse* needs to have a corresponding element *node* with the same values for the attributes *id*, *lang* and *version*. This restriction is implemented using *xs:key* and *xs:keyref* that link the elements *node* and *reuse* to each other with some sort of primary and secondary key.
Even if the attribute *id* is already made up of the language and version of a node, the values still have to be repeated in the attributes *lang* and *version* of that node.

The basic concepts of the reuse metrics considered for the REx format are explained in the online publication: “Metrische Untersuchung der Wiederverwendung im Content Management – Statische Kennzahlen in der Technischen Redaktion“ (Reuse Metrics in Content Management – Static Key Performance Indicators for Technical Communication) by Prof. Dr. Ziegler (www.home.hs-karlsruhe.de/~ziwo0001/PUB/CMS-Metrik_Ziegler.pdf)