

Spezifikation REx-Format 1.1

Autoren:

Claudia Oberle, M. Sc. Technische Redaktion, Hochschule Karlsruhe

Prof. Dr. Wolfgang Ziegler, Hochschule Karlsruhe

Dokumentversion 12

Inhaltsverzeichnis

1	Einführung	1
2	REx1.1-Level 1	2
2.1	Version und Level	2
2.2	Angaben zum CMS	2
2.3	Repräsentation der Struktur durch Knoten	4
3	REx1.1-Level 2	8
3.1	Systemweite und gemittelte Kennzahlen	8
3.2	Knotenbezogene Kennzahlen	11

1 Einführung

Das Report Exchange Format (REx-Format) ist ein Kennzahlenaustauschformat, mit dem die Wiederverwendung von Modulen innerhalb eines Content-Management-Systems (CMS) standardisiert protokolliert und analysiert werden kann. Die Struktur des REx-Formats wird in der Version 1.1 über zwei XML-Schemata (XSD-Dateien) definiert. Je nach den enthaltenen Informationen werden zwei Stufen / Levels des REx-Formats unterschieden. In der einfachsten Form soll ein CMS die Daten zur Wiederverwendung in Form eines Exports (Report) auf dem Level 1 liefern. Detaillierte Auswertungen auf dem Niveau von Level 2 werden dann über freie Programme bzw. Skripte ermöglicht. Prinzipiell können CMS auch Reports auf Level 2 liefern. Für die Visualisierung von REx-Daten auf Level 2 werden im Rahmen des REx-Projektes ebenfalls freie Programme entwickelt und zur Verfügung gestellt.

Für jedes REx-Level existiert ein eigenes XML-Schema. Einige Strukturen aus Level 1 werden im Level 2 verwendet, weshalb die XSD-Datei zu Level 1 über den Befehl *xs:include* in die XSD-Datei für Level 2 eingebunden wird. Auf diese Art und Weise kann auf globale Strukturdefinitionen bzw. globale Typen aus Level 1 zugegriffen werden und es können Ableitungen dieser globalen Typen durch Erweiterungen und Einschränkungen erzeugt werden.

Im Folgenden werden Level 1 und 2 des REx-Formats in der Version 1.1 näher beschrieben.

Strukturen aus Level 1, die auch in Level 2 vorkommen, werden nur unter *2 REx1.1-Level 1* erläutert und ggf. Unterschiede zu Level 2 an entsprechender Stelle beschrieben. Pflichtangaben sind als solche gekennzeichnet.

2 REx1.1-Level 1

Das erste Level des REX-Formats enthält den Minimalumfang der Daten zur Wiederverwendung von Objekten (Dokumente, Module oder Medien). Objekte werden teilweise auch als Knoten bezeichnet. Die Rohdaten aus Level 1 stellen den Eingabedatensatz zur Berechnung der Kennzahlen dar.

Folgende Informationen sind im REX-Schema vorgesehen (s. Datei REX_1.1-Level_1.xsd):

2.1 Version und Level

Die Version des REX-Formats und das Level werden angegeben. Das Wurzelement ist je nach Level *REx1* oder *REx2* und enthält die Attribute *rexVersion* mit dem Wert *1.1* und *rexLevel* mit den erlaubten Werten *1* oder *2* je nach Level.

2.2 Angaben zum CMS

Das Element *meta* ist ein Pflichtelement, das verschiedene Informationen zum CMS umfasst. Zur Anonymisierung können alle Ausgabedateien (Level 1 und 2) oder nur die Grafiken ohne Angaben zum CMS ausgewertet werden. Die XSL-Auswerteskripte können mit verschiedenen Parametern gestartet werden.

2.2.1 Unternehmen, das das CMS einsetzt

Pflichtelement: *sysOwner* vom Typ *xs:string*

2.2.2 Hersteller/Version des CMS

Pflichtelement: *sysType* vom Typ *xs:string*

2.2.3 Zeitpunkt, an dem die Daten ermittelt werden

Pflichtelement: *sysTime*

Hinweis: Der Datentyp für dieses Element ist eine Vereinigung der Datentypen *xs:dateTime* und *xs:string* mit der Beschränkung auf *none*. Mögliche Angaben haben daher die Form: *2001-12-18T09:30:47Z* oder *none*.

2.2.4 Spezifische Informationen zum CMS-Anwendungszweck (Produkt, Werk, Abteilung, etc.)

Pflichtelement: *sysVersion* vom Typ *xs:string*

2.2.5 Typ der Analyse

komplexes Pflichtelement: *analysisType*

Es kann zwischen zwei Kindelementen und damit zwei Möglichkeiten der Analyse gewählt werden:

2.2.5.1 Analyse bzw. Auswertung eines oder mehrerer spezifischer Ausgabekanäle

Element: *sysChannel*

Pflichtattribute: *docType*, *lang* und *mediaChannel*

optionales Attribut, das ausschließlich in REX1.1 Level 2 eingesetzt wird: *obsoleteNodes*

Im CMS sollte ausgewählt werden können, welche Ausgabekanäle ausgewertet werden sollen. Die auszuwertenden Kanäle werden in drei Pflichtattributen angegeben, die Angaben zum Dokumenttyp z. B. *Betriebsanleitung*, zur Sprache z. B. *de* und zum Ausgabemedium z. B. *PDF* enthalten.

Hinweis: Für jedes Pflichtattribut von *sysChannel* wird ein Listentyp eingesetzt, der eine durch Leerzeichen getrennte Sequenz von Werten vom Typ *xs:token* enthält. Leerzeichen, die nicht als Trennzeichen gelten sollen, müssen deshalb in den einzelnen Werten der Liste durch „_“ (Unterstriche) ersetzt werden. So werden unbeabsichtigte Trennungen der Werte innerhalb der Liste vermieden. Implizit werden hier keine abhängigen Metadaten erwartet und aus allen drei Attributen wird das Kreuzprodukt gebildet. D.h. wenn mehrere Sprachen und mehrere Medien angegeben werden, wird davon ausgegangen, dass alle Medien in allen Sprachen vorkommen. Die genannten Kanalattribute *docType*, *lang* und *mediaChannel* können ebenfalls für jeden im System enthaltenen Knoten gesetzt werden (s.u.). Sofern eine Auswertung des betreffenden Teilkanals erfolgt, müssen die Werte der Kanalattribute der Knoten in der Auflistung des zugehörigen Kanalattributs von *sysChannel* enthalten sein. Derzeit werden aber keine Auswertungen einzelner Kanäle vorgenommen.

Das optionale Attribut *obsoleteNodes* ist auf die erlaubten Werte *included* oder *not_included* beschränkt und wird ausschließlich in REx1.1 Level 2 eingesetzt. Dieses Attribut gibt an, ob für die Transformation in REx1.1 Level 2 nicht verwendete Objekte berücksichtigt wurden (Wert *included*) oder nicht berücksichtigt wurden (Wert *not_included*).

Hinweis: Der Wert *not_included* bedeutet, dass nicht verwendete Objekte nicht in die Berechnung der Kennzahlen eingeflossen und nicht mehr als *node*-Elemente in der Datei für REx1.1 Level 2 vorhanden sind, also herausgefiltert wurden.

2.2.5.2 Globale Analyse des gesamten Systems

Element: *sysGlobal*

Pflichtattribute: *docType*, *lang* und *mediaChannel*

Dies ist die globale und einfachste Auswertemöglichkeit. Alle im CMS befindlichen Objekte werden erfasst und deren Daten in der REx-Datei ausgegeben.

Hinweis: Es findet (derzeit) keine Auswertung eines einzelnen Kanals statt; die Werte der Pflichtattribute müssen auf *all* gesetzt werden.

2.2.6 Einheit für die Segmentgröße

optionales Element *default_segmentType* in REx1.1 Level 1, Pflichtelement in REx1.1 Level 2 Angabe der Einheit für die Segmentgröße (z. B. *word*), die zur Berechnung der gewichteten Kennzahlen verwendet wird.

Auswahl zwischen folgenden vorgegebenen Werten:

- *word*
Segmentgröße in Wörtern
- *phrase*
Segmentgröße in Sätzen
- *char*
Segmentgröße in Zeichen
- *element*
Segmentgröße als Anzahl von XML-Elementen

- *textElement*
Segmentgröße als Anzahl von XML-Elementen, die „reinen“ Text als Inhalt haben
- *userspecific*
benutzerspezifische Einheit für die Segmentgröße
- *none*
es wird keine Segmentgröße zur Auswertung verwendet

2.3 Repräsentation der Struktur durch Knoten

Für die Rohdaten werden die verschiedenen Knoten eines CMS mit ihren Strukturinformationen dargestellt. Das komplexe Pflichtelement *raw* enthält die Rohdaten und umfasst folgende Informationen:

Jeder Knoten (Dokument, Modul, Fragment, Grafik und weitere Medientypen) wird durch ein eigenes Element *node* repräsentiert. Dabei werden alle enthaltenen Knoten der nächsten Hierarchiestufe in Form eines Verweises über Schlüsselinformationen mit abgebildet. Der Schlüssel besteht aus einem Vektor aus ID, Sprache und Version (Pflichtangaben).

Es können beliebig viele Knoten abgebildet werden, aber es muss mindestens ein Knoten vorhanden sein. Über Attribute und Kindelemente können zusätzliche Informationen zu einem Knoten in die Rohdaten aufgenommen werden.

komplexes Pflichtelement: *node*

Pflichtattribute: *node_Type*, *id*, *lang*, *version*

optionale Attribute: *mediaChannel*, *docType*, *dateInitial*, *dateFinal*

Kindelemente: *size*, *reuse*

Hinweis: Es muss zu allen Versionen eines Objektes jeweils ein eigenes *node*-Element in einer REX-Ausgabedatei enthalten sein (s. 3.1.3.4 und 3.1.3.7).

2.3.1 Typ des Knotens

Das Pflichtattribut *node_Type* gibt den Typ des Knotens an. Zwischen folgenden vorgegebenen Werten kann gewählt werden:

- *doc*
Dokument auf oberster Ebene
Hinweis: Jedes *node*-Element mit dem Wert *doc* für das Attribut *node_Type* wird als Dokument angesehen, das in dieser Form „wirklich“ publiziert wird und dementsprechend für die Berechnung der Kennzahlen behandelt.
- *module*
Knoten auf untergeordneten Ebenen
- *frag*
Fragment (wiederverwendete Unterstruktur eines Moduls, z. B. aus dem Reuse-Pool)
- *graphic*
Grafik
- *audio*
Audiodatei

- *video*
Videodatei
- *userspecificMedia*
nutzerspezifische Mediendatei; ggf. weitere Medientypen definierbar

2.3.2 ID des Knotens

Das Pflichtattribut *id* gibt die ID des Knotens an und ist vom Typ *xs:string*. Auf den Datentyp *xs:ID* wurde bewusst verzichtet, da dieser Datentyp strenge Vorgaben bzgl. der Schreibweise der ID macht. Hier sollten die eindeutigen Identifikatoren des CMS angegeben werden, mit dem das zugehörige Objekt im CMS wiedergefunden werden kann. Ziel einer späteren REx-Nutzung soll es sein, Objekte im CMS über Links direkt ansprechbar zu machen.

2.3.3 Sprache des Knotens

Das Pflichtattribut *lang* gibt die Sprache des Knotens an. Der Datentyp für dieses Attribut ist eine Vereinigung der Datentypen *xs:language* und *xs:string* mit der Beschränkung auf *none*.

Mögliche Angaben haben daher die Form:

de für Deutsch, *en-US* für amerikanisches Englisch usw., *none*.

Hinweis: Wenn bei der Systemanalyse eine Auswertung dieses Teilkanals erfolgt, muss der Wert, der in diesem Attribut für einen Knoten gesetzt ist, auch in den Metadaten unter dem Element *sysChannel* für das gleichnamige Attribut in der Auflistung der Werte vorkommen.

Bei gemischtsprachigen Dokumenten müssen in diesem Attribut die Sprachangaben aller referenzierten Objekte in einer durch Leerzeichen getrennten Liste genannt werden.

Leerzeichen sind in den einzelnen Werten nicht erlaubt, da es sonst zu unbeabsichtigten Trennungen der Werte der Liste kommt.

2.3.4 Version des Knotens

Das Pflichtattribut *version* gibt die Version des Knotens an und ist vom Typ *xs:double*. Üblicherweise werden hier die Freigabeversionen eines CMS ganzzahlig angegeben. Nicht-numerische Versionsschemata müssen beim REx-Export in numerische Werte umgewandelt werden.

2.3.5 Ausgabeformat/Ausgabemedium des Knotens

Das optionale Attribut *mediaChannel* gibt das Ausgabeformat für Knoten vom Typ *doc* an wie z. B. *PDF*. Hier wird ein Listentyp eingesetzt, der eine durch Leerzeichen getrennte Sequenz von Werten vom Typ *xs:token* enthält, so dass auch mehrere Werte gesetzt werden können. Leerzeichen, die nicht als Trennzeichen gelten sollen, müssen deshalb in den einzelnen Werten der Liste durch „_“ (Unterstriche) ersetzt werden. So werden unbeabsichtigte Trennungen der Werte innerhalb der Liste vermieden.

Hinweis: Wenn bei der Systemanalyse eine Auswertung dieses Teilkanals erfolgt, muss der Wert, der in diesem Attribut für einen Knoten gesetzt ist, auch in den Metadaten unter dem Element *sysChannel* für das gleichnamige Attribut in der Auflistung der Werte vorkommen.

2.3.6 Verwendungsart des Knotens

Das optionale Attribut *docType* gibt die Verwendungsart bzw. Textsorte für Knoten vom Typ *doc* an wie z. B. *Benutzerhandbuch*. Hier wird ein Listentyp eingesetzt, der eine durch Leerzeichen getrennte Sequenz von Werten vom Typ *xs:token* enthält, so dass auch mehrere Werte gesetzt werden können.

Leerzeichen, die nicht als Trennzeichen gelten sollen, müssen deshalb in den einzelnen Werten der Liste durch „_“ (Unterstriche) ersetzt werden. So werden unbeabsichtigte Trennungen der Werte innerhalb der Liste vermieden.

Hinweis: Wenn bei der Systemanalyse eine Auswertung dieses Teilkanals erfolgt, muss der Wert, der in diesem Attribut für einen Knoten gesetzt ist, auch in den Metadaten unter dem Element *sysChannel* für das gleichnamige Attribut in der Auflistung der Werte vorkommen.

2.3.7 Datum der Erstellung des Knotens

Das optionale Attribut *dateInitial* gibt das Datum der ersten Erstellung des Knotens in der jeweiligen freigegebenen Version an und ist vom Typ *xs:dateTime*.

Hinweis: Mögliche Angaben haben die Form *2001-12-18T09:30:47Z*. *none* ist als Wert für dieses Attribut nicht zugelassen. Da es sich dabei aber um ein optionales Attribut handelt, muss es nicht gesetzt werden, wenn kein Datum der Erstellung angegeben werden kann.

2.3.8 Datum der letzten Änderung des Knotens

Das optionale Attribut *dateFinal* gibt das Datum der Freigabe des Knotens an und ist vom Typ *xs:dateTime*.

Hinweis: Mögliche Angaben haben die Form *2001-12-18T09:30:47Z*. *none* ist als Wert für dieses Attribut nicht zugelassen. Da es sich dabei aber um ein optionales Attribut handelt, muss es nicht gesetzt werden, wenn kein Datum der letzten Änderung angegeben werden kann.

2.3.9 Segmentgröße des Knotens

Das komplexe Pflichtelement *size* enthält die Segmentgröße des Knotens. Es sind parallele Angaben in mehreren Einheiten möglich.

Für den Fall, dass die Segmentgröße nicht bekannt ist, kann das Element *<none>1</none>* gewählt werden. Falls die Segmentgröße bekannt ist, kann zwischen folgenden Einheiten/Kindelementen gewählt werden:

- *word*
Segmentgröße in Wörtern (Inhalt vom Typ *xs:double*)
- *phrase*
Segmentgröße in Sätzen (Inhalt vom Typ *xs:double*)
- *char*
Segmentgröße in Zeichen (Inhalt vom Typ *xs:double*)
- *element*
Segmentgröße in XML-Elementen (Inhalt vom Typ *xs:double*)
- *textElement*
Segmentgröße in XML-Elementen, die „reinen“ Text als Inhalt haben (Inhalt vom Typ *xs:double*)
- *userspecific*
Segmentgröße in benutzerspezifischer Einheit (Inhalt vom Typ *xs:double*)

Hinweis: Die Segmentgröße muss nicht in allen möglichen Einheiten angegeben werden. In der Regel wird nur eine Einheit angegeben, idealerweise *word*. Wenn mehrere Einheiten ausgewählt werden, muss die Reihenfolge der Kindelemente eingehalten werden wie sie oben aufgelistet ist (*word, phrase, char, element, textElement, userspecific*).

Sofern nicht für alle Knoten die Segmentgröße in der Einheit angegeben ist, die unter *default_segmentType* festgelegt wurde, können bestimmte Kennzahlen nicht ausgewertet werden.

Für Dokumente und Module, die weitere Knoten referenzieren, gibt der Inhalt des *size*-Elements die Segmentgröße an, die das Dokument / das Modul zusätzlich zu den referenzierten Knoten hat. Die gesamte Segmentgröße des Knotens, inklusive der wiederverwendeten Knoten, wird erst im Level 2 berechnet.

2.3.10 Wiederverwendete Knoten in tieferen Hierarchiestufen

Das komplexe Element *reuse* stellt einen Verweis auf einen Knoten der nächsten Hierarchiestufe dar, der in dem betreffenden Knoten wiederverwendet wird. Dabei steht je ein *reuse*-Element für einen referenzierten Knoten. Es können keine bis beliebig viele Verweise auf weitere Knoten dargestellt werden. Bei dem Verweis handelt es sich um einen dreiteiligen Schlüssel, der aus den drei Pflichtattributen *id*, *lang* und *version* des *reuse*-Elements aufgebaut ist.

Hinweis: Damit eine XML-Instanz bzgl. der Schemata für REx 1.1 Level 1 und 2 valide ist, muss für jeden Knoten, der innerhalb des *reuse*-Elements referenziert wird, auch ein *node*-Element mit diesem dreiteiligen Schlüssel vorhanden sein. Genauer gesagt muss für jedes *reuse*-Element ein *node*-Element mit denselben Werten für die Pflichtattribute *id*, *lang* und *version* existieren. Diese Beschränkung wurde mithilfe der Befehle *xs:key* und *xs:keyref* umgesetzt, die zwischen den Pflichtattributen *id*, *lang* und *version* der Elemente *node* und *reuse* einen Bezug herstellen, der vergleichbar zu der Beziehung zwischen Primär- und Fremdschlüssel ist. Falls die *id* des CMS bereits die Sprache und die Version des Objektes beinhaltet, müssen die Attribute *lang* und *version* dennoch nochmals separat mit den jeweiligen Werten befüllt werden.

3 REx1.1-Level 2

Die zweite Stufe des REx-Formats enthält den Maximalumfang der CMS-Auswertung, d. h. die Rohdaten und die daraus berechneten Kennzahlen. System- und medienspezifische Kennzahlen werden unter dem komplexen Element *calc* eingefügt. Kennzahlen, die sich auf je einen Knoten (bzw. auf die höchste Version eines Knotens) beziehen, werden in dem jeweiligen *node*-Element als Attribute ergänzt. Folgende Informationen und Kennzahlen sind im REx-Schema vorgesehen (s. Datei REx_1.1-Level_2.xsd):

3.1 Systemweite und gemittelte Kennzahlen

Alle system- und medienspezifische Kennzahlen werden unter dem komplexen Element *calc* aufgeführt. Dabei werden folgende Kennzahlen von freien REx-Skripten errechnet oder direkt aus dem System exportiert:

3.1.1 Systembasierte Kennzahlen

Die systembasierten Kennzahlen werden sowohl bei der Auswertung einzelner Publikationskanäle, als auch für ein Gesamtsystem berechnet. Das Element *system* umfasst folgende systembasierte Kennzahlen, die aus unterschiedlichen Knotentypen berechnet und über Attribute abgebildet werden:

3.1.1.1 Gesamtzahl der Segmente im System

Attribut *agg_segmentCount* vom Typ *xs:double*

Es werden die aktuellsten Knoten mit der höchsten Version berücksichtigt.

Zur Berechnung dieser Kennzahl sind Angaben zu den Segmentgrößen aller Knoten in der unter *default_segmentSize* gesetzten Einheit erforderlich.

3.1.1.2 Anzahl nicht verwendeter Segmente im System

Attribut *agg_segmentsObsolete* vom Typ *xs:double*

Sobald eine Version eines Objektes verwendet ist, gelten alle Versionen des Objektes implizit als verwendet und werden nicht mehr in dieser Kennzahl berücksichtigt. Die Kennzahl sollte nur für Systemauswertungen (*all*) einen von Null verschiedenen Wert liefern können. Auswertungen über definierte Publikationskanäle verwenden immer alle enthaltenen Knoten.

Zur Berechnung dieser Kennzahl sind Angaben zu Segmentgrößen in der unter *default_segmentSize* gesetzten Einheit erforderlich.

3.1.1.3 Anzahl verwendeter Segmente im System

Attribut *agg_segmentsUsed* vom Typ *xs:double*

Zur Berechnung dieser Kennzahl sind Angaben zu Segmentgrößen in der unter *default_segmentSize* gesetzten Einheit erforderlich.

3.1.1.4 Mittleres Inhaltsgewicht der Inhalte

Attribut *avg_contentWeight* vom Typ *xs:double*

Das mittlere Gewicht der Inhalte ist die durchschnittliche Anzahl von Segmenten pro Knoten (für alle Knotentypen). Es geht jeweils die aktuelle (höchste) Version eines Knotens ein. Es werden aber nur solche mit eigenem Inhalt berücksichtigt und nur solche, die verwendet werden (keine obsoleten).

Zur Berechnung dieser Kennzahl sind Angaben zu Segmentgrößen in der unter *default_segmentSize* gesetzten Einheit erforderlich.

3.1.1.5 Summe über alle Wiederverwendungen von Modulen und Fragmenten

Attribut *agg_reuses* vom Typ *xs:double*

Die Summe aller Wiederverwendungen von Modulen und Fragmenten ist die Summe über alle Verwendungszahlen von Modulen und Fragmenten, wobei eine Verwendung von jeder Verwendungszahl subtrahiert wird ($Z - 1$), da nur die Wiederverwendungen einbezogen werden sollen. Es geht jeweils die aktuelle (höchste) Version eines Moduls / Fragments ein und nur solche, die verwendet werden (keine obsoleten).

3.1.1.6 Summe über alle gewichteten Wiederverwendungen von Modulen und Fragmenten

Attribut *agg_weightedReuses* vom Typ *xs:double*

Die Summe aller gewichteten Wiederverwendungen von Modulen und Fragmenten ist die Summe über alle Verwendungszahlen von Modulen und Fragmenten, wobei eine Verwendung von jeder Verwendungszahl subtrahiert wird ($Z - 1$), da nur die Wiederverwendungen einbezogen werden sollen, gewichtet mit der jeweiligen Segmentgröße des Moduls / Fragments. Es geht jeweils die aktuelle (höchste) Version eines Moduls / Fragments ein und nur solche, die verwendet werden (keine obsoleten).

Zur Berechnung dieser Kennzahl sind Angaben zu Segmentgrößen in der unter *default_segmentSize* gesetzten Einheit erforderlich.

3.1.2 Kennzahlen, die über alle Dokumente berechnet werden

Das komplexe Element *document* umfasst folgende Kennzahlen, die über alle Dokumente im System aufsummiert werden und über Attribute abgebildet werden:

3.1.2.1 Anzahl Dokumente im System

Attribut *agg_documents* vom Typ *xs:double*

3.1.2.2 Durchschnitt der modulbezogenen Wiederverwendungsrate

Attribut *avg_ratioReused* vom Typ *xs:double*

Mittelwert der modulbezogenen Wiederverwendungsraten (Redundanzgrad) aller Dokumente. Der Redundanzgrad gibt an, welcher Anteil von Modulen oder Fragmenten eines Dokumentes bereits in anderen Dokumenten verwendet wird.

3.1.2.3 Durchschnitt der inhaltsbezogenen Wiederverwendungsrate

Attribut *avg_ratioReusedContent* vom Typ *xs:double*

Mittelwert der Wiederverwendungsraten (Redundanzgrad) aller Dokumente, wobei die Zahl der (Standard-)Segmente als Gewichtungsfaktor genutzt wird.

Zur Berechnung dieser Kennzahl sind Angaben zu Segmentgrößen in der unter *default_segmentSize* gesetzten Einheit erforderlich.

3.1.2.4 Durchschnitt Anzahl Module pro Dokument

Attribut *avg_moduleCount* vom Typ *xs:double*

Hier werden nur Module und Fragmente in beliebigen Hierarchietiefen berücksichtigt (keine Grafiken).

3.1.3 Kennzahlen, die über alle Module berechnet werden

Das komplexe Element *module* umfasst folgende Kennzahlen, die über alle Module im System aufsummiert werden und über Attribute abgebildet werden:

3.1.3.1 Anzahl Module im System

Attribut *agg_modules* vom Typ *xs:double*

3.1.3.2 Anzahl nicht verwendeter Module im System

Attribut *agg_obsoletes* vom Typ *xs:double*

Hinweis: Nicht verwendete Module sollten nur vorkommen, wenn kein Kanal ausgewertet wurde. Zudem werden Versionen berücksichtigt: Sobald eine (beliebige) Version eines Moduls wiederverwendet wird, wird dieses Modul nicht als *obsolete* angesehen.

3.1.3.3 Anzahl verwendeter Module im System

Attribut *agg_modulesUsed* vom Typ *xs:double*

3.1.3.4 Mittlere Verwendungszahl der Module

Attribut *avg_uses* vom Typ *xs:double*

Mittelwert der Verwendungszahlen aller Module im System (oder Kanal).

3.1.3.5 Mittlere gewichtete Verwendungszahl

Attribut *avg_weightedUses* vom Typ *xs:double*

Mittelwert der Verwendungszahlen aller Module im System (oder Kanal), wobei die Zahl der (Standard-)Segmente als Gewichtungsfaktor genutzt wird.

Zur Berechnung dieser Kennzahl sind Angaben zu Segmentgrößen in der unter *default_segmentSize* gesetzten Einheit erforderlich.

3.1.3.6 Durchschnitt der Wiederverwendungsrate

Attribut *avg_ratioReused* vom Typ *xs:double*

Mittelwert der Wiederverwendungsraten (Redundanzgrad) aller Module. Der Redundanzgrad gibt an, welcher Anteil von Modulen oder Fragmenten eines Moduls bereits in anderen Modulen verwendet wird.

3.1.3.7 Durchschnitt der inhaltsbezogenen Wiederverwendungsrate

Attribut *avg_ratioReusedContent* vom Typ *xs:double*

Mittelwert der Wiederverwendungsraten (Redundanzgrad) aller Module, wobei die Zahl der (Standard-)Segmente als Gewichtungsfaktor genutzt wird.

Zur Berechnung dieser Kennzahl sind Angaben zu Segmentgrößen in der unter *default_segmentSize* gesetzten Einheit erforderlich.

3.1.3.8 Durchschnitt Segmente pro Modul

Attribut *avg_segmentCount* vom Typ *xs:double*

Zur Berechnung dieser Kennzahl sind Angaben zu Segmentgrößen in der unter *default_segmentSize* gesetzten Einheit erforderlich.

3.1.4 Kennzahlen, die über alle Knoten eines Medientyps berechnet werden

Das komplexe Element *media* kann für jeden Medientyp einmal vorkommen. Über Attribute werden zusätzliche Informationen und Kennzahlen zu Knoten des jeweiligen Medientyps abgebildet.

3.1.4.1 Medientyp

Das Pflichtattribut *mediaType* gibt den Medientyp an, für den die Kennzahlen bestimmt wurden. Zwischen folgenden vorgegebenen Werten kann gewählt werden:

- *graphic*
Grafik
- *audio*
Audiodatei
- *video*
Videodatei
- *userspecificMedia*
nutzerspezifische Mediendatei; ggf. weitere Medientypen definierbar

3.1.4.2 Anzahl Knoten des Medientyps

Attribut *agg_media* vom Typ *xs:double*

Gesamtzahl der Medien des jeweiligen Typs.

3.1.4.3 Anzahl nicht verwendeter Knoten des Medientyps

Attribut *agg_obsoletes* vom Typ *xs:double*

3.1.4.4 Anzahl verwendete Knoten des Medientyps

Attribut *agg_mediaUsed* vom Typ *xs:double*

3.1.4.5 Mittlere Verwendungszahl eines Medientyps

Attribut *avg_uses* vom Typ *xs:double*

Mittelwert der Verwendungszahlen eines Medientyps im System (oder Kanal).

3.1.4.6 Durchschnittliche Verwendungen des Medientyps pro Knoten

Attribut *avg_count* vom Typ *xs:double*

Es wird gemittelt, wie viele Medien pro Knoten (Dokument, Modul oder Fragment) verwendet werden. Knoten, die Medien enthalten, müssen selbst auch verwendet werden, um angerechnet zu werden. Analysiert wird die höchste Knotenversion.

3.1.4.7 Summe über alle Wiederverwendungen dieses Medientyps

Attribut *agg_reuses* vom Typ *xs:double*

Die Summe aller Wiederverwendungen von Knoten des jeweiligen Medientyps ist die Summe über alle Verwendungszahlen dieser Knoten, wobei eine Verwendung von jeder Verwendungszahl subtrahiert wird ($Z - 1$), da nur die Wiederverwendungen einbezogen werden sollen. Es geht jeweils die aktuelle (höchste) Version ein und nur solche, die verwendet werden (keine obsoleten).

3.2 Knotenbezogene Kennzahlen

Die Kennzahlen, die sich auf je einen Knoten (bzw. auf die höchste Version eines Knotens) beziehen, werden in dem jeweiligen *node*-Element als Attribute abgebildet. Für diesen Zweck wurde der komplexe, globale Typ *nodeType* aus dem XML-Schema für REx1.1 Level 1 um Attribute erweitert, die die entsprechenden Kennzahlen repräsentieren. Folgende Kennzahlen gehören zu den knotenbezogenen Kennzahlen:

3.2.1 Neueste im System vorkommende Version

Attribut *calc_maxVersion* vom Typ *xs:double*

Bei Auswertung eines Kanals steht dieses Attribut für die neueste im Kanal vorkommende Version. Diese Kennzahl wird für alle Knotentypen und alle Versionen von Knoten berechnet.

3.2.2 Anzahl der Verwendungen im System

Attribut *agg_uses* vom Typ *xs:double*

Abundanz, d.h. absolute Zahl der Wiederverwendungen. Diese Kennzahl wird für alle Knoten berechnet, die nicht vom Typ *doc* sind.

3.2.3 Gewichtete Verwendungszahl

Attribut *calc_weightedUses* vom Typ *xs:double*

Verwendungszahlen des Knotens, wobei die Zahl der (Standard-)Segmente als Gewichtungsfaktor genutzt wird. Diese Kennzahl wird für alle Knoten berechnet, die nicht vom Typ *doc* sind.

Zur Berechnung dieser Kennzahl sind Angaben zu Segmentgrößen in der unter *default_segmentSize* gesetzten Einheit erforderlich.

3.2.4 Anzahl Module des Knotens

Attribut *agg_moduleCount* vom Typ *xs:double*

Diese Kennzahl wird für alle Knoten (Dokumente, Module, Fragmente) berechnet, die mindestens ein reuse-Element vom Typ *module* oder *frag* beinhalten (mindestens ein Module oder Fragment) verwenden. Für Knoten (Module, Fragmente), die keine reuse-Elemente vom Typ *module* oder *frag* beinhalten, wird dieser Wert standardmäßig auf Null gesetzt. Es werden alle Hierarchiestufen berücksichtigt.

3.2.5 Anzahl wiederverwendete Module des Knotens

Attribut *agg_reusedModuleCount* vom Typ *xs:double*

Diese Kennzahl wird für alle Knoten (Dokumente, Module, Fragmente) berechnet, die mindestens ein reuse-Element vom Typ *module* oder *frag* beinhalten (mindestens ein Module oder Fragment) verwenden. Für Knoten (Module, Fragmente), die keine reuse-Elemente vom Typ *module* oder *frag* beinhalten, wird dieser Wert standardmäßig auf Null gesetzt. Es werden alle Hierarchiestufen berücksichtigt. Es werden aber nur solche Knoten berücksichtigt, die mehrfach verwendet werden.

3.2.6 Anzahl Segmente des Knotens

Attribut *agg_segments* vom Typ *xs:double*

Wie 3.2.4 für Segmente: Diese Kennzahl wird für alle Knoten berechnet, die mindestens ein reuse-Element beinhalten (mindestens einen Knoten) verwenden, es werden alle Hierarchiestufen berücksichtigt.

Zur Berechnung dieser Kennzahl sind Angaben zu Segmentgrößen in der unter *default_segmentSize* gesetzten Einheit erforderlich.

Hinweis: Im Gegensatz zu dem *size*-Element wird hier die gesamte Segmentgröße des Knotens berechnet, bestehend aus den eigenen Segmenten des Knotens und der Summe der Segmente aus den wiederverwendeten Knoten.

3.2.7 Modulbezogene Wiederverwendungsrate

Attribut *calc_ratioReusedModules* vom Typ *xs:double*

Relative Berechnung 3.2.5 / 3.2.4:

Relativer Anteil der verwendeten Knoten (reuse-Elemente vom Typ *module* oder *frag*) eines Knotens, die auch in anderen Knoten verwendet werden.

Diese Kennzahl wird für alle Knoten berechnet, die mindestens ein reuse-Element vom Typ *module* oder *frag* beinhalten / mindestens ein Modul / Fragment verwenden. Für Knoten (Module, Fragmente), die keine reuse-Elemente vom Typ *module* oder *frag* beinhalten, wird dieser Wert standardmäßig auf Null gesetzt.

3.2.8 Inhaltsbezogene Wiederverwendungsrate

Attribut *calc_ratioReusedContent* vom Typ *xs:decimal*

Relativer Anteil der Segmente des Knotens, die auch in anderen Knoten verwendet werden.

Diese Kennzahl wird für alle Knoten berechnet, die mindestens ein reuse-Element vom Typ *module* oder *frag* beinhalten / mindestens ein Modul / Fragment verwenden. Für Knoten (Module, Fragmente), die keine reuse-Elemente vom Typ *module* oder *frag* beinhalten, wird dieser Wert standardmäßig auf Null gesetzt.

Zur Berechnung dieser Kennzahl sind Angaben zu Segmentgrößen in der unter *default_segmentSize* gesetzten Einheit erforderlich.

3.2.9 Bewertete Verwendungszahl eines Dokumentes

Attribut *calc_reuseAmount* vom Typ *xs:double*

Summe der Verwendungszahlen aller Module des Dokumentes (im System), dividiert durch Modulzahl des Dokumentes. Diese Kennzahl wird für alle Knoten vom Typ *doc* berechnet.

3.2.10 Bewertete und mit dem Inhalt gewichtete Verwendungszahl

Attribut *calc_weightedReuseAmount* vom Typ *xs:double*

Summe des Produkts von Segmentzahl und Verwendungszahl aller Module des Dokumentes, dividiert durch Segmentzahl des Dokumentes.

Diese Kennzahl wird für alle Knoten vom Typ *doc* berechnet.

Zur Berechnung dieser Kennzahl sind Angaben zu Segmentgrößen in der unter *default_segmentSize* gesetzten Einheit erforderlich.

Die Grundlagen zu den im REx-Format enthaltenen Kennzahlen sind in der Online-Publikation dargestellt: „Metrische Untersuchung der Wiederverwendung im Content Management – Statische Kennzahlen in der Technischen Redaktion“ von Prof. Dr. Ziegler:
www.home.hs-karlsruhe.de/~ziwo0001/PUB/CMS-Metrik_Ziegler.pdf