

DIN EN ISO 21597-1

ICS 35.240.67; 91.010.01

Einsprüche bis 2018-11-21

Entwurf

**Informationscontainer zur Datenübergabe –
Austausch-Spezifikation –
Teil 1: Container (ISO/DIS 21597-1:2018);
Deutsche und Englische Fassung prEN ISO 21597-1:2018**

Information container for data drop –
Exchange specification –
Part 1: Container (ISO/DIS 21597-1:2018);
German and English version prEN ISO 21597-1:2018

Anwendungswarnvermerk

Dieser Norm-Entwurf mit Erscheinungsdatum 2018-09-21 wird der Öffentlichkeit zur Prüfung und Stellungnahme vorgelegt.

Weil die beabsichtigte Norm von der vorliegenden Fassung abweichen kann, ist die Anwendung dieses Entwurfs besonders zu vereinbaren.

Stellungnahmen werden erbeten

- vorzugsweise online im Norm-Entwurfs-Portal von DIN unter www.din.de/go/entwuerfe bzw. für Norm-Entwürfe der DKE auch im Norm-Entwurfs-Portal der DKE unter www.entwuerfe.normenbibliothek.de, sofern dort wiedergegeben;
- oder als Datei per E-Mail an nabau@din.de möglichst in Form einer Tabelle. Die Vorlage dieser Tabelle kann im Internet unter www.din.de/go/stellungnahmen-norm-entwuerfe oder für Stellungnahmen zu Norm-Entwürfen der DKE unter www.dke.de/stellungnahme abgerufen werden;
- oder in Papierform an den DIN-Normenausschuss Bauwesen (NABau), 10772 Berlin, Burggrafenstr. 6, 10787 Berlin.

Die Empfänger dieses Norm-Entwurfs werden gebeten, mit ihren Kommentaren jegliche relevanten Patentrechte, die sie kennen, mitzuteilen und unterstützende Dokumentationen zur Verfügung zu stellen.

Gesamtumfang 126 Seiten

DIN-Normenausschuss Bauwesen (NABau)

Nationales Vorwort

Dieses Dokument (prEN ISO 21597-1:2018) wurde vom Technischen Komitee ISO/TC 59 „Buildings and civil engineering works“ in Zusammenarbeit mit dem Technischen Komitee CEN/TC 442 „Building Information Modelling (BIM)“ erarbeitet, dessen Sekretariat von SN (Norwegen) gehalten wird.

Das zuständige deutsche Gremium ist der Arbeitsausschuss NA 005-01-39 AA „BIM - Building Information Modeling (SpA zu ISO/TC 59/SC 13 - CEN/TC 442)“ im DIN-Normenausschuss Bauwesen (NABau).

Um Zweifelsfälle in der Übersetzung auszuschließen, ist die englische Originalfassung beigelegt. Die Nutzungsbedingungen für den deutschen Text des Norm-Entwurfes gelten gleichermaßen auch für den englischen Text.

**Informationscontainer zur Datenübergabe — Austausch-Spezifikation —
Teil 1: Container (ISO/DIS 21597-1:2018)**

Information container for data drop — Exchange specification — Part 1: Container (ISO/DIS 21597-1:2018)

ICS:

Deskriptoren:

Dokument-Typ: Europäische Norm
Dokument-Untertyp:
Dokumentstufe: parallele Umfrage
Dokumentsprache: D

STD Version 2.9d

Inhalt

	Seite
Europäisches Vorwort	3
Vorwort	4
Einleitung	5
1 Anwendungsbereich	6
2 Normative Verweisungen	6
3 Begriffe und Abkürzungen	7
3.1 Begriffe	7
3.2 Abkürzungen	9
4 Spezifikationen	10
4.1 Anwendung von RDF-, RDFS- und OWL-Konstrukten	10
4.2 Symbole und Notation	12
4.3 Struktur des Containers	14
4.3.1 Übersicht	14
4.3.2 Ordner „Ontologie-Ressourcen“	16
4.3.3 Ordner „Nutzdaten-Dokumente“	16
4.3.4 Ordner „Nutzdaten-Tripeln“	16
4.4 Ontologien und Datensätze	16
4.4.1 Übersicht	16
4.4.2 Container-Ontologie	16
4.4.3 Linkset-Ontologie	21
4.4.4 Index-Datensatz	25
4.4.5 Link dataset	25
4.5 Versionierung	25
4.6 Zusätzliche Eigenschaften in Datensätzen	27
5 Anforderungen an die Übereinstimmung	27
Anhang A (informativ) Anwendungsfälle	29
A.1 Zusammenhang	29
A.2 Anwendungsfall 1A – Datenübergabe von Dokumenten	29
A.3 Anwendungsfall 1B – Datenübergabe von Dokumenten mit Links	36
Anhang B (informativ) Interoperabilität mit Dublin Core	42
Anhang C (informativ) Wie sich ein ICDD von RDF in XML und zurück konvertieren lässt	43
Anhang D (informativ) Wie die Validierung mit SHACL erfolgt	44
D.1 Schema-Validierung	44
D.1.1 Erkennen zusätzlicher Klassen	44
D.1.2 Erkennen zusätzlicher Prädikate	46
D.2 Daten-Validierung	46
D.2.1 Überprüfungen des Datentyps	46
D.2.2 owl:cardinality als Zwangsbedingung auf der Grundlage der Unique-Naming- Assumption	47
D.2.3 Prädikat owl:unionOf als Zwangsbedingung	47
Anhang E (normativ) Ontologien	48
E.1 Container-Ontologie	48
E.2 Linkset-Ontologie	57

Europäisches Vorwort

Dieses Dokument (prEN ISO 21597-1:2018) wurde vom Technischen Komitee ISO/TC 59 „Buildings and civil engineering works“ in Zusammenarbeit mit dem Technischen Komitee CEN/TC 442 „Building Information Modelling (BIM)“ erarbeitet, dessen Sekretariat von SN gehalten wird.

Dieses Dokument ist derzeit zur parallelen Umfrage vorgelegt.

Anerkennungsnotiz

Der Text von ISO/DIS 21597-1:2018 wurde von CEN als prEN ISO 21597-1:2018 ohne irgendeine Abänderung genehmigt.

Vorwort

ISO (die Internationale Organisation für Normung) ist eine weltweite Vereinigung nationaler Normungsorganisationen (ISO-Mitgliedsorganisationen). Die Erstellung von Internationalen Normen wird üblicherweise von Technischen Komitees von ISO durchgeführt. Jede Mitgliedsorganisation, die Interesse an einem Thema hat, für welches ein Technisches Komitee gegründet wurde, hat das Recht, in diesem Komitee vertreten zu sein. Internationale staatliche und nichtstaatliche Organisationen, die in engem Kontakt mit ISO stehen, nehmen ebenfalls an der Arbeit teil. ISO arbeitet bei allen elektrotechnischen Themen eng mit der Internationalen Elektrotechnischen Kommission (IEC) zusammen.

Die Verfahren, die bei der Entwicklung dieses Dokuments angewendet wurden und die für die weitere Pflege vorgesehen sind, werden in den ISO/IEC-Direktiven, Teil 1 beschrieben. Es sollten insbesondere die unterschiedlichen Annahmekriterien für die verschiedenen ISO-Dokumentenarten beachtet werden. Dieses Dokument wurde in Übereinstimmung mit den Gestaltungsregeln der ISO/IEC-Direktiven, Teil 2 erarbeitet (siehe www.iso.org/directives).

Es wird auf die Möglichkeit hingewiesen, dass einige Elemente dieses Dokuments Patentrechte berühren können. ISO ist nicht dafür verantwortlich, einige oder alle diesbezüglichen Patentrechte zu identifizieren. Details zu allen während der Entwicklung des Dokuments identifizierten Patentrechten finden sich in der Einleitung und/oder in der ISO-Liste der erhaltenen Patenterklärungen (siehe www.iso.org/patents).

Jeder in diesem Dokument verwendete Handelsname dient nur zur Unterrichtung der Anwender und bedeutet keine Anerkennung.

Eine Erläuterung der Bedeutung ISO-spezifischer Benennungen und Ausdrücke, die sich auf Konformitätsbewertung beziehen, sowie Informationen über die Beachtung der Grundsätze der Welthandelsorganisation (WTO) zu technischen Handelshemmnissen (TBT, en: Technical Barriers to Trade) durch ISO enthält der folgende Link: www.iso.org/iso/foreword.html.

Dieses Dokument wurde vom Technischen Komitee ISO/TC 59, *Buildings and civil engineering works*, Unterkomitee SC 13, *Organization and digitization of information about buildings and civil engineering works, including building information modelling (BIM)*, erarbeitet.

ISO 21597 besteht unter dem allgemeinen Titel *Information Container for Data Drop — Exchange specification* aus den folgenden Teilen:

- *Part 1: Container*
- *Part 2: Dynamic semantics*

Einleitung

Diese Norm wurde als Reaktion auf den Bedarf der Bauindustrie entwickelt, mannigfaltige Dokumente als eine Informationslieferung oder eine Datenübergabe zu behandeln.

Informationslieferungen sind oftmals eine Kombination von Zeichnungen, Informationsmodellen (die gebaute Vermögenswerte oder natürliche Güter in der physischen Welt darstellen), Textdokumenten, Arbeitsblättern usw. Die Fähigkeit, Zusammenhänge zwischen Informationselementen in diesen separaten Dokumenten zu spezifizieren, kann erheblich zum Wert einer Datenübergabe beitragen. Die Zusammensetzung eines solchen Pakets ergibt sich sowohl aus den Anforderungen im Prozess, z. B. Lieferung von Baubestandsinformationen, als auch aus dem spezifischen funktionellen Zweck, z. B. Durchführung einer Mengenermittlung oder Kommunikation über Probleme in 3D-Modellen.

Diese Norm stellt eine Spezifikation für einen Informationscontainer bereit. Sie ermöglicht einen einheitlichen Ansatz hinsichtlich der Art und Weise, wie Informationen in Datenübergaben organisiert werden, indem sie ein Mittel zur Erzeugung semantischer Verknüpfungen zwischen Begriffen in separaten Dokumenten zur Verfügung stellt; außerdem bietet sie auch eine Grundlage für eine zusätzliche Funktionalität, die es ermöglicht, einen Container individuell für einen bestimmten Zweck anzupassen, und unterstützt somit die innovative Entwicklung von Software, die dennoch der Norm entspricht.

Teil 1 dieser Norm enthält eine Spezifikation für einen Container, in dem Dokumente gespeichert werden, zusammen mit einem Mittel zur Verknüpfung ansonsten unzusammenhängender Daten innerhalb dieser Dokumente.

1 Anwendungsbereich

Diese Norm legt ein generisches Containerformat zur Speicherung eines gelieferten Dokumentensatzes fest, einschließlich eines Mittels zur Verknüpfung ansonsten nicht zusammenhängender Daten. In diesem Kontext bezieht sich der Begriff Dokument auf alle digitalen Ressourcen, die Informationen über die gebaute oder natürliche Umwelt liefern, einschließlich, aber nicht beschränkt auf 2D- oder 3D-Darstellungen oder Modelle, Arbeitsblätter sowie bild- oder textbasierte digitale Ressourcen.

Diese Norm ist für alle Parteien geeignet, die sich mit Informationen befassen, die die gebaute Umwelt betreffen, wenn die Notwendigkeit besteht, mannigfaltige Dokumente und ihre Wechselbeziehungen entweder als Teil des Prozesses oder als vertraglich vereinbarte Ergebnisse auszutauschen. Das Format ist dafür vorgesehen, Ressourcen zu verwenden, die entweder im Container enthalten sind (wie z. B. Dokumente) oder die an anderer Stelle in Bezug genommen werden (wie z. B. Web-Ressourcen). Ein Hauptmerkmal besteht darin, dass der Container Informationen über die Beziehungen zwischen den Dokumenten enthalten kann. Maßgebliche Anwendungsfälle spiegeln die Notwendigkeit für den Austausch von Informationen während des gesamten Lebenszyklus eines gebauten Vermögenswertes wider, und dürfen die Übergabe von Folgendem einschließen, sind jedoch nicht allein darauf beschränkt:

- 1) ein veröffentlichtes Ausschreibungspaket;
- 2) in einer bestimmten Projektphase geforderte Projektergebnisse;
- 3) gemeinsame Informationen als Hintergrund oder für die weitere Entwicklung; oder
- 4) veröffentlichte Zulassungspakete.

Das Containerformat umfasst eine Header-Datei und optionale Link-Dateien, welche durch Aufnahme von Verweisungen auf die Dokumente oder auf Elemente innerhalb dieser Beziehungen festlegen. Die Header-Datei identifiziert eindeutig den Container und seine vertragliche oder gemeinschaftliche Intention. Diese Informationen werden unter Anwendung der semantischen Web-Standards RDF und OWL festgelegt.

Die Header-Datei bildet gemeinsam mit allen zusätzlichen RDF/OWL-Dateien oder -Ressourcen eine Suite, die von der Software direkt abgefragt werden darf. Wenn sie Link-Verweise auf den Inhalt von Dokumenten enthält, die genormte Abfragemechanismen nicht unterstützen, darf ihre Auflösung von Interpretern einer Drittpartei abhängen. Alternativ dürfen die Link-Verweise von den Anwendungen des Empfängers ausgewertet oder interaktiv durch den Empfänger eingesehen werden.

Das Format kann auch dazu genutzt werden, mehrere Versionen desselben Dokuments zu liefern mit der Fähigkeit, die bekannten Unterschiede oder eine Priorität zwischen ihnen zu übermitteln.

2 Normative Verweisungen

Die folgenden Dokumente werden im Text in solcher Weise in Bezug genommen, dass einige Teile davon oder ihr gesamter Inhalt Anforderungen des vorliegenden Dokuments darstellen. Bei datierten Verweisungen gilt nur die in Bezug genommene Ausgabe. Bei undatierten Verweisungen gilt die letzte Ausgabe des in Bezug genommenen Dokuments (einschließlich aller Änderungen).

ISO/IEC 21320-1:2015, *Information technology — Document Container File — Part 1: Core*
<http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html>

ISO 6707-1, *Buildings and civil engineering works — Vocabulary — Part 1: General terms*

W3C RDF standard, Resource Description Framework, https://www.w3.org/standards/techs/rdf#w3c_all

RDF Schema 1.1 — W3C Recommendation 25 February 2014, <https://www.w3.org/TR/rdf-schema/>

W3C OWL, *Web Ontology Language*, <https://www.w3.org/TR/2012/REC-owl2-syntax-20121211/>

W3C XML, *Schema Part 2: Datatypes*, <http://www.w3.org/TR/xmlschema-2/>

3 Begriffe und Abkürzungen

3.1 Begriffe

Für die Anwendung dieses Dokuments gelten die folgenden Begriffe.

ISO und IEC stellen terminologische Datenbanken für die Verwendung in der Normung unter den folgenden Adressen bereit:

- ISO Online Browsing Platform: unter <http://www.iso.org/obp>
- IEC Electropedia: unter <http://www.electropedia.org/>.

3.1.1

Klasse

Gruppe von Ressourcen (entweder ein IRI oder ein Literal), die die gleichen Eigenschaften haben

3.1.2

Container

Datei, die dieser Norm entspricht

3.1.3

Container-Ontologie

RDF/OWL-Datei, welche die Objektklassen und Eigenschaften bereitstellt, die anzuwenden sind, um den Inhalt eines Containers festzulegen

3.1.4

Datensatz

RDF/OWL-Datei, die Individuen enthält, die den durch die Ontologien im Ordner „Ontologie-Ressourcen“ festgelegten Klassen entsprechen

3.1.5

Datentyp-Eigenschaft

OWL-Eigenschaft, die Individuen zu Literalen in Beziehung setzen kann (z. B. Strings, Zahlen, Datentypen usw.)

3.1.6

Dokument

feste und strukturierte Menge von Informationen, die gesteuert und als Einheit zwischen Nutzern und Systemen ausgetauscht werden kann

Anmerkung 1 zum Begriff: Diese Einheit wird möglicherweise vom Menschen nicht unbedingt wahrgenommen. Informationen werden üblicherweise auf einem Speichermedium gespeichert.

Anmerkung 2 zum Begriff: Diese Benennung wird in dieser Norm verwendet, um auf alle Dokumente zu verweisen, die einen Teil der Nutzdaten im Container bilden, einschließlich aller 2D- oder 3D-Modelle, die gebaute Vermögenswerte oder natürliche Güter in der physischen Welt darstellen; diese können in einem beliebigen Standardformat oder proprietären Format gehalten werden.

3.1.7

externes Dokument

Dokument, das sich außerhalb des Containers befindet

3.1.8

Index-Datensatz

RDF/OWL-Datei, die ein Verzeichnis des Inhalts des Containers enthält

3.1.9

Individuum

Ressource, die in eine RDF-Klasse als eine Instanz dieser Klasse eingebracht wurde

Anmerkung 1 um Begriff: Wie RDF-Klassen ist jede OWL-Klasse mit einer Gruppe von Individuen verbunden, was als Klassenerweiterung bezeichnet wird; die Individuen in der Klassenerweiterung sind die Instanzen der Klasse.

[QUELLE: <https://www.w3.org/TR/owl-ref/>]

3.1.10

internes Dokument

Dokument, das sich im Container befindet

3.1.11

Link

Beziehung zwischen Dokumenten, einschließlich der Beziehung zwischen Elementen in Dokumenten

3.1.12

Link-Datensatz

RDF/OWL-Datei, die Links nach den Festlegungen in dieser Norm enthält

3.1.13

Linkset-Ontologie

RDF/OWL-Datei, die Objektklassen und Merkmale bereitstellt, die zur Festlegung von Links zwischen Dokumenten in einem Container zu verwenden sind

3.1.14

Namensraum

Gruppe von Bezeichnern für Elemente und Attribute, die gemeinsam so an einen URI gebunden sind, dass ihre Verwendung nicht zu Benennungskonflikten führt

3.1.15

Objekt

Ressource (entweder ein IRI oder ein Literal), die als festgelegte Eigenschaft des Subjekts in einem Tripel zugewiesen ist

Anmerkung 1 zum Begriff: Diese in dieser Norm verwendete Benennung ist Teil des RDF/OWL-Vokabulars, bei dem ein Tripel aus einem Subjekt, einem Prädikat und einem Objekt besteht; eine Menge solcher Tripeln wird als RDF-Graph bezeichnet.

3.1.16

Objekt-Eigenschaft

OWL-Eigenschaft, die Individuen mit anderen Individuen verknüpft

3.1.17

Ontologie

Spezifikation konkreter oder abstrakter Dinge, und die Beziehungen zwischen ihnen, in einer vorgeschriebenen Wissensdomäne

Anmerkung 1 zum Begriff: Die Spezifikation sollte elektronisch auswertbar sein.

[QUELLE: ISO/IEC 19763-3:2010, 3.1.1.1]

3.1.18

Nutzdaten

primäre Informationen (in Form von Dokumenten und Datensätzen), die im Container enthalten sind

Anmerkung 1 zum Begriff: Diese schließen nicht die Header-Datei (Index.rdf) oder die Ontologie-Ressourcen-Dateien ein.

3.1.19

Prädikat

bezeichnet die Beziehung zwischen einem Subjekt und einem Objekt in einem RDF-Tripel, auch als Eigenschaft bezeichnet

3.1.20

Serialisierung

Datei in einem bestimmten Format, die eine Ontologie oder einen Datensatz verschlüsselt

3.1.21

Subjekt

Ressource (entweder ein IRI oder ein Literal), über die eine Aussage in Form eines RDF-Tripels gemacht wird

Anmerkung 1 zum Begriff: Diese in dieser Norm verwendete Benennung ist Teil des RDF/OWL-Vokabulars, bei dem ein Tripel aus einem Subjekt, einem Prädikat und einem Objekt besteht; eine Menge solcher Tripeln wird als RDF-Graph bezeichnet.

3.1.22

Tripel

Aussage in der Form Subjekt-Prädikat-Objekt, die eine Beziehung zwischen zwei Ressourcen angibt

3.2 Abkürzungen

Die folgenden Abkürzungen werden in diesem Dokument verwendet:

DBF	DataBase File
GIS	Geographic Information System
GML	Geography Markup Language
GUID	Globally Unique Identifier
ICDD	Information Container for Data Drop
IFC	Industry Foundation Classes
IRI*	Internationalized Resource Identifier
OWL	Web Ontology Language
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
SHACL	Shapes Constraint Language

SPARQL	SPARQL Protocol And RDF Query Language
SQL	Structured Query Language
UML	Unified Modeling Language
URI*	Uniform Resource Identifier
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
XML	eXtensible Markup Language
XSD	XML Schema Definition
XSLT	Extensible Stylesheet Language Transformations

ANMERKUNG * IRI ist eine Aktualisierung des URI, die 2005 veröffentlicht wurde. URIs sind auf eine Teilmenge des ASCII-Zeichensatzes beschränkt. IRIs können Zeichen aus dem Universal Character Set (Unicode/ISO 10646) enthalten. In diesem Dokument werden URIs und IRIs austauschbar genutzt mit den vorgenannten Einschränkungen.

4 Spezifikationen

4.1 Anwendung von RDF-, RDFS- und OWL-Konstrukten

Die in dieser Norm festgelegten Ontologien verwenden die Sprachen RDF [<https://www.w3.org/RDF/>], RDFS [<https://www.w3.org/TR/rdf-schema/>] und OWL [<https://www.w3.org/OWL/>].

ANMERKUNG Es wird erwartet, dass RDF/OWL in den kommenden Jahrzehnten eine wichtige Technologie und allgemeine Plattform für Ontologien sein wird. Proprietäre Systeme werden RDF/OWL zunehmend übernehmen.

Bei Anwendung im Zusammenhang mit dem World Wide Web nutzen diese Sprachen im Allgemeinen die folgenden Grundsätze, um die logische Argumentation zu unterstützen:

- Open-World-Assumption – die Wahrheit einer Aussage ist unabhängig davon, ob sie bekannt ist. Mit anderen Worten impliziert die Unkenntnis, ob eine Aussage explizit wahr ist nicht, dass die Aussage falsch ist;
- No-Unique-Names-Assumption – sofern nicht explizit anders angegeben, kann nicht angenommen werden, dass Ressourcen, die von verschiedenen URIs identifiziert werden, verschieden sind.

Die Datensätze, die mit den in dieser Norm festgelegten Ontologien übereinstimmen, müssen die folgende Interpretation von RDF, RDFS und OWL anwenden:

- Closed-World-Assumption – von einer Aussage, die wahr ist, ist auch bekannt, dass sie wahr ist; daher ist im Umkehrschluss alles, was in einem Container nicht formell als wahr festgelegt ist, falsch;
- Unique-Naming-Assumption – Ressourcen in einem Container, die mit verschiedenen URIs identifiziert werden, werden als verschieden betrachtet, sofern sie nicht explizit für gleich erklärt werden (durch Verwendung des Prädikats *owl:sameAs*).

In Tabelle 1 sind die RDF/OWL-Konstrukte aufgeführt, die in dieser Norm verwendet werden, sowie die Interpretation, die für die Validierung des Inhalts eines Containers anzuwenden ist. Es wird darauf hingewiesen, dass die Daten, sobald der Inhalt des Containers validiert wurde, in einem Open-World-Kontext verwendet werden können.

Tabelle 1 — Auflistung von Konstrukten, die in dieser Norm verwendet werden, und ihre Interpretation

Construct	Interpretation
<i>owl:Class</i>	In einem Datensatz innerhalb eines Containers muss die Klassenzugehörigkeit für jedes Individuum explizit erklärt werden, es sei denn, sie wird durch Prädikate wie z. B. <i>rdfs:subClassOf</i> oder <i>owl:equivalentClass</i> implizit abgeleitet.
<i>rdfs:subClassOf</i> <i>rdfs:subPropertyOf</i>	Diese Norm weicht nicht von den W3C-Definitionen ab (https://www.w3.org/TR/rdf-schema/). Aussagen, die aufgrund von <i>rdfs:subClassOf</i> - oder <i>rdfs:subPropertyOf</i> -Aussagen geschlussfolgert werden können, sind selbst dann als wahr zu betrachten, wenn dies nicht explizit erklärt wird. ANMERKUNG Aussagen, in denen eine Klasse erwähnt wird, sind auch für all ihre Unterklassen wahr. Gleichmaßen sind Aussagen, in denen eine Eigenschaft erwähnt wird, auch für all ihre Untereigenschaften wahr.
<i>rdfs:subClassOf</i> [<i>rdf:type owl:Class</i> ; <i>owl:unionOf</i> (: <i>X</i> : <i>Y</i>)];]	Im Zusammenhang mit einem Container wird dieses Konstrukt als Restriktion interpretiert, indem diese Klasse abstrakt gemacht wird, was bedeutet, dass jedes Individuum dieser Klasse mindestens einer der im Prädikat <i>owl:unionOf</i> aufgeführten Klassen angehören muss. Gegebenenfalls wird dies in Verbindung mit dem Prädikat <i>owl:disjointWith</i> angewendet, um zu erzwingen, dass das Individuum lediglich einer der im Prädikat <i>owl:unionOf</i> aufgeführten Klassen angehört.
<i>owl:FunctionalProperty</i>	Diese Norm interpretiert <i>owl:FunctionalProperty</i> als Eigenschaft mit einer maximalen Kardinalität von 1.
<i>owl:equivalentClass</i>	Diese Norm weicht nicht von den W3C-Definitionen ab (https://www.w3.org/TR/owl-ref/). Aussagen, die aufgrund von <i>owl:equivalentClass</i> -Aussagen abgeleitet werden können, sind selbst dann als wahr zu betrachten, wenn dies nicht explizit erklärt wird.
<i>owl:disjointWith</i>	Diese Norm weicht nicht von der Originaldefinition ab. Da die Eigenschaft symmetrisch ist, ist sie in beiden Richtungen gültig, selbst wenn nur eine Richtung in einer Ontologie erklärt wird. Ein Individuum, das mehreren Klassen angehört, die disjunkt sind, ist ungültig.
<i>rdfs:range</i> , <i>rdfs:domain</i>	Diese Aussagen müssen als Restriktionen interpretiert werden. Es ist ungültig, ein Subjekt oder Objekt einer Aussage (Tripel) in einem Datensatz zu haben, in dem dieses Individuum einer Klasse angehört, die nicht mit den Deklarationen <i>rdfs:range</i> oder <i>rdfs:domain</i> der entsprechenden <i>owl:ObjectProperty</i> oder <i>owl:DatatypeProperty</i> übereinstimmt.
<i>owl:restriction</i> <i>owl:onProperty</i> <i>owl:allValuesFrom</i> <i>owl:someValuesFrom</i> <i>owl:hasValue</i> <i>owl:cardinality</i> <i>owl:minCardinality</i> <i>owl:maxCardinality</i>	Diese Aussagen müssen als Restriktionen interpretiert werden. Jede Abweichung von der festgelegten Restriktion innerhalb eines einzelnen Containers wird als ungültig betrachtet. ANMERKUNG Beispielsweise ist ein Datensatz, der nicht genau 2 Ereignisse enthält, nicht gültig, wenn <i>owl:cardinality</i> als 2 festgelegt ist.
<i>owl:inverseOf</i>	Diese Norm weicht nicht von der Originaldefinition ab. Es wird empfohlen, dass inverse Eigenschaften für Individuen in einem Datensatz nicht erklärt werden. Wenn sie erklärt werden, dürfen sie den Erklärungen, die in der entgegengesetzten Richtung gemacht wurden, nicht widersprechen.

4.2 Symbole und Notation

In dieser Norm wird der Aufbau der Ontologien durchweg mittels einer UML-Notation veranschaulicht. Der Zweck dieses Abschnitts besteht darin, die Notation und Bedeutung der verwendeten Begriffe und Symbole zu beschreiben.

In Tabelle 2 und Tabelle 3 werden die Namensräume und die zugehörigen Präfixe aufgeführt, die in dieser Norm verwendet werden.

Tabelle 2 — Namensräume und Präfixe, die in den in dieser Norm festgelegten Ontologien verwendet werden

Ontologie	Präfix	Namensraum
Container ontology	ct	http://www.iso-icdd.org/part1/2019/Container
Linkset ontology	ls	http://www.iso-icdd.org/part1/2019/Linkset

Tabelle 3 — Namensräume und Präfixe, die in Ontologien verwendet werden, die in dieser Norm in Bezug genommen werden

Ontologie	Präfix	Namensraum
XML Schema	xsd	https://www.w3.org/2001/XMLSchema
Resource Description Framework	rdf	https://www.w3.org/1999/02/22-rdf-syntax-ns
RDF Schema	rdfs	https://www.w3.org/2000/01/rdf-schema
Web Ontology Language	owl	https://www.w3.org/2002/07/owl

In Bild 1 sind die in dieser Norm verwendeten UML-Notationen für Klassen und Eigenschaften dargestellt.

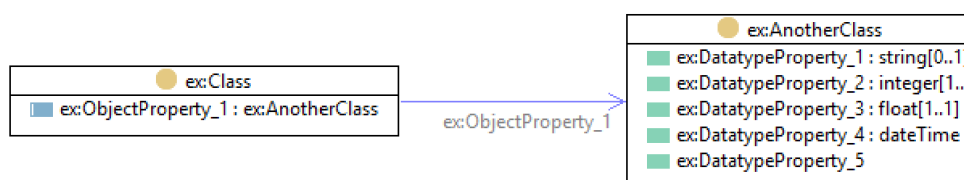


Bild 1 — UML-Notation für Klassen und Eigenschaften

Eine Klasse (*owl:Class*) wird als rechteckige Box mit zwei Feldern dargestellt, wie in Bild 1 gezeigt. Im oberen Feld wird die Klassenbezeichnung (im vorstehenden Bild "*ex:Class*") angezeigt. Es wird angemerkt, dass die Klassenbezeichnung nach dem Muster "*prefix:ClassName*" dargestellt wird, wobei der Präfix ("ex" in dem Beispiel) den Namensraum der Ontologie bezeichnet und "ClassName" der Name der Klasse ist. Die tatsächlich in dieser Norm verwendeten Präfixe sind in der vorstehenden Tabelle 2 festgelegt.

Das untere Feld zeigt die festgelegten Eigenschaften für diese Klasse. Es gibt zwei allgemeine Arten von Eigenschaften:

- Datentyp-Eigenschaften sind diejenigen Eigenschaften, für die der Wert ein Daten-Literal ist, wie für *ex:AnotherClass* in Bild 1 dargestellt; und
- Objekt-Eigenschaften, für die der Wert ein Individuum ist; z. B. *ex:Class* in Bild 1, in dem die Eigenschaft *ex:ObjectProperty_1* ein Individuum der Klasse *ex:AnotherClass* in Bezug nimmt.

Die Definitionen von Eigenschaften werden nach dem Muster “*prefix:propertyName : range[cardinality]*” dargestellt. Der Wertebereich einer Datentyp-Eigenschaft ist gewöhnlich einer der im XML-Schema vordefinierten Datentypen (<https://www.w3.org/2001/XMLSchema-datatypes>). Der Wertebereich einer Objekt-Eigenschaft ist gewöhnlich eine der Klassen, die in der Ontologie vorkommen, darf sich aber auch auf eine Klasse in einer anderen Ontologie beziehen.

Wenn Klassen auf beiden Seiten (Domäne und Wertebereich) einer Objekt-Eigenschaft in einem Diagramm sichtbar sind, darf die Objekt-Eigenschaft auch mit einem (blauen) Pfeil zwischen den Klassen dargestellt werden, der von der Domänenklasse zur Wertebereichsklasse zeigt. Der Name der Objekt-Eigenschaft wird entlang des Pfeils sowie im Eigenschaftsfeld der Klassen-Box wie vorstehend erklärt angezeigt.

Alle Kardinalitätsrestriktionen werden in eckigen Klammern unter Anwendung der folgenden Notation dargestellt: *[minCardinality..maxCardinality]*, wobei *minCardinality* die Mindestanzahl zulässiger Ereignisse und *maxCardinality* die maximale Anzahl zulässiger Ereignisse festlegt.

Die Kardinalitätsrestriktionen müssen auf folgende Weise interpretiert werden:

- weggelassen – es sind keine Kardinalitätsrestriktionen vorhanden, d. h. jede beliebige Anzahl von Ereignissen von Null bis viele ist zulässig;
- *maxCardinality* weggelassen (z. B. [0..], [1..] usw.) - die maximale Kardinalität ist unbeschränkt.

Wenn zwei Klassen zueinander in Beziehung stehen, indem ein *rdfs:subClassOf*-Prädikat verwendet wird, wird dies durch einen Pfeil umgesetzt, wie in Bild 2 gezeigt. In diesem Diagramm wird durch Verwendung eines *rdfs:subClassOf*-Prädikats dargestellt, dass *ex:SubClass* und *ex:Class* zueinander in Beziehung stehen.

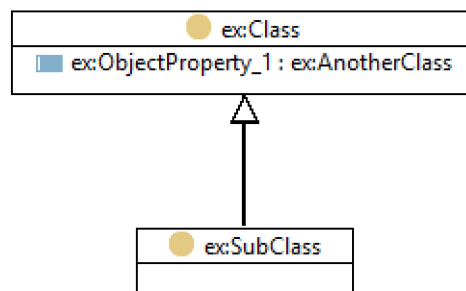


Bild 2 — Darstellung einer Unterklassen-Beziehung

Vereinigte Klassen und disjunkte Klassen werden in Bild 3 dargestellt.

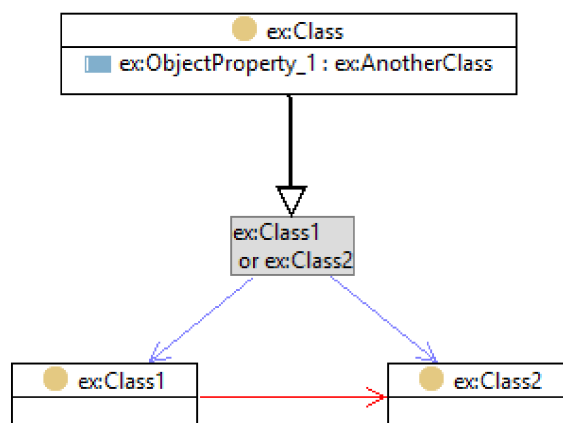


Bild 3 — Darstellung von vereinigten Klassen und disjunkten Klassen

Der rote Pfeil, der von *ex:Class1* zu *ex:Class2* zeigt, gibt an, dass die Klassen disjunkt sind, was bedeutet, dass es nicht zulässig ist, dass eine Instanz sowohl *ex:Class1* als auch *ex:Class2* angehört. Dies wird mit einer *owl:disjointWith*-Aussage (*ex:Class1 owl:disjointWith ex:Class2*) erklärt. Die Eigenschaft *owl:disjointWith* ist symmetrisch, was bedeutet, dass, wenn *Class1* von *Class2* disjunkt ist, dann auch *Class2* von *Class1* disjunkt ist.

Die vereinigte Klasse (Vereinigung von *ex:Class1* und *ex:Class2*) wird durch die graue Box mit nur einem Feld dargestellt, in dem die Namen der Klassen aufgeführt sind, die der Vereinigung angehören. Das Schlüsselwort „oder“ gibt an, dass es sich tatsächlich um eine Vereinigung handelt. Wenn die vereinigte Klasse und die Klassen, die der Vereinigung angehören, im selben Diagramm dargestellt werden, darf das Diagramm auch Pfeile von der vereinigten Klasse zu den ihr angehörenden Klassen enthalten. Die in Bild 3 festgelegte Gesamtstruktur gibt im Wesentlichen an, dass *ex:Class* eine abstrakte Klasse ist, und dass jede Instanz eine Instanz von entweder *ex:Class1* oder *ex:Class2* sein muss.

Eine Schnittmengen-Klasse wird gleichermaßen durch eine graue Box dargestellt, wobei in diesem Fall jedoch das Schlüsselwort „oder“ durch das Schlüsselwort „und“ ersetzt wird.

Zwei Klassen, die durch die Verwendung von *owl:equivalentClass* (z. B. *ex:Class3 owl:equivalentClass ex:Class4*) für gleichwertig erklärt werden, werden wie in Bild 4 gezeigt dargestellt.

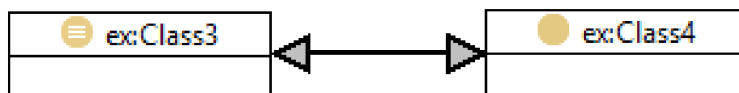


Bild 4 — Darstellung gleichwertiger Klassen

Schließlich darf eine Klasse sowohl die *Domäne* als auch der *Wertebereich* für eine bestimmte *Objekt-Eigenschaft* sein. Eine solche Beziehung wird wie in Bild 5 gezeigt umgesetzt, d. h. mit dem kleinen Pfeil, der sich auf der unteren Linie der Box für die Klasse befindet, ohne dass an dem Pfeil eine Beschriftung angebracht wird.

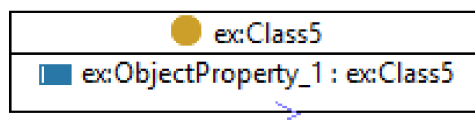


Bild 5 — Darstellung einer Objekt-Eigenschaft, die durch ein Individuum derselben Klasse festgelegt ist

4.3 Struktur des Containers

4.3.1 Übersicht

Ein Container ist eine Datei, die die Erweiterung „icdd“ haben und mit ISO 21320-1, auch bekannt als ZIP64, übereinstimmen muss.

Ein Container enthält eine Header-Datei im Ordner der obersten Ebene; diese Datei entspricht der RDF-Norm und ist in RDF/XML serialisiert (<https://www.w3.org/TR/rdf-syntax-grammar/>). Der Name dieser Header-Datei ist *Index.rdf*.

Ein Container muss mindestens drei Ordner haben, wie in Bild 6 dargestellt. Der Zweck dieser Ordner der obersten Ebene wird in den folgenden Abschnitten erklärt. Die Ordner „Nutzdaten-Dokumente“ und „Nutzdaten-Tripeln“ dürfen eingeschachtelte Ordner enthalten, um zu ermöglichen, dass Gruppen zusammengehöriger digitaler Ressourcen zusammengehalten und als eine Gruppe in Bezug genommen werden (z. B. Bauwerksdatenmodell mit seinen zugehörigen Referenzdateien oder eine Gruppe verknüpfter Arbeitsblatt-Dokumente).

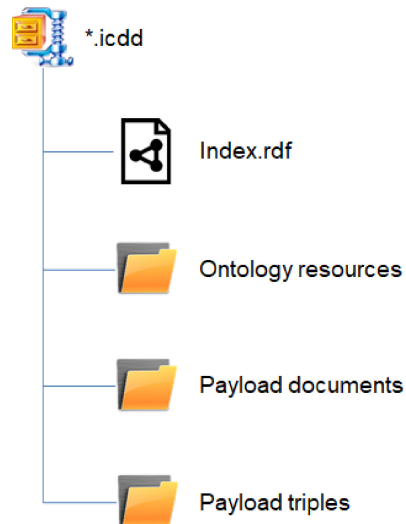


Bild 6 — Mindestgliederung des Hauptverzeichnisses eines Containers

Bild 7 zeigt die Hierarchie der Ordner und Dateien innerhalb eines Containers.

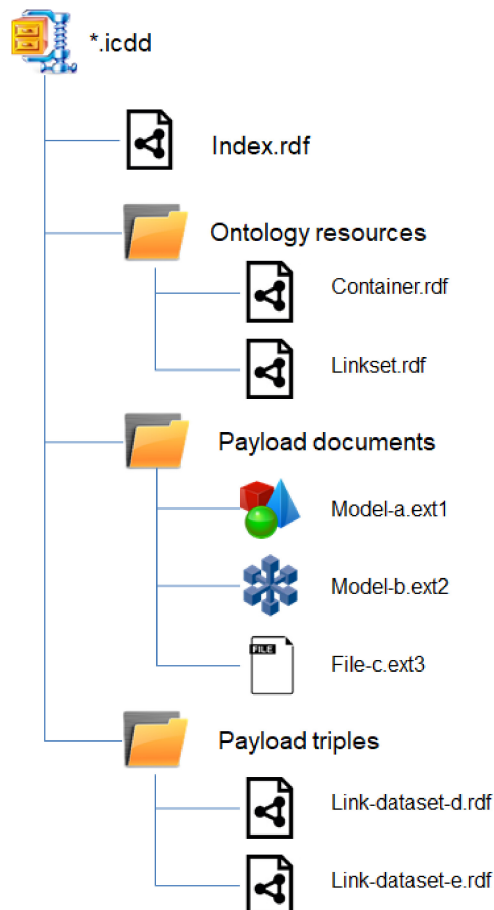


Bild 7 — Hierarchie von Ordnern und Dateien in einem Container

4.3.2 Ordner „Ontologie-Ressourcen“

Der Ordner „Ontologie-Ressourcen“ darf zur Speicherung der Linkset.rdf- und Container.rdf-Ontologien verwendet werden, die gemeinsam die Objektklassen und Eigenschaften bereitstellen, die verwendet werden müssen, um den Inhalt von und die Links zwischen den Dokumenten innerhalb des Containers festzulegen. Diese Ontologien müssen im RDF/XML-Format vorliegen. Da sie beide als Online-Ressourcen verfügbar sind, ist es nicht vorgeschrieben, sie hier einzubeziehen, wenn sie jedoch einbezogen werden, müssen die Dateien in diesem Ordner Vorrang haben.

4.3.3 Ordner „Nutzdaten-Dokumente“

Der Ordner „Nutzdaten-Dokumente“ muss zur Speicherung aller Dokumente verwendet werden, die im Container enthalten sind (und als Nutzdaten bezeichnet werden). Unterordner sind zulässig.

4.3.4 Ordner „Nutzdaten-Tripeln“

Der Ordner „Nutzdaten-Tripeln“ muss für die Speicherung von Datensatz-Dateien verwendet werden und darf Unterordner enthalten.

4.4 Ontologien und Datensätze

4.4.1 Übersicht

Unter Verwendung der RDF/OWL-Technologie werden die Objektklassen und Eigenschaften, die zur Festlegung des Inhalts von und der Links zwischen den Dokumenten innerhalb des Containers verwendet werden, im Container unter Nutzung der folgenden beiden Ontologien festgelegt:

- Container-Ontologie, online verfügbar unter www.iso-icdd.org/part1/2019/Container.
- Linkset-Ontologie, online verfügbar unter www.iso-icdd.org/part1/2019/Linkset.

Wie bereits zuvor angemerkt, können diese Ontologien im Container in den Ordner „Ontologie-Ressourcen“ aufgenommen werden, um einen in sich abgeschlossenen Container zu erzeugen, der offline oder für Archivierungszwecke genutzt werden darf. Da diese Ontologien der vorliegenden Norm entsprechen, dürfen sie nicht in irgendeiner Form geändert werden.

Darüber hinaus muss der Container RDF/OWL-Datensätze enthalten, die den Inhalt des Containers und die Links zwischen diesen Dokumenten beschreiben. Es wird zwischen zwei Arten von Datensätzen unterschieden. Erstens muss der Container einen Index-Datensatz enthalten, der als Index.rdf bezeichnet und im Hauptverzeichnis des Containers gehalten wird; er wird zur Beschreibung des Containers und zur Festlegung des Inhalts der Dokumente verwendet. Zweitens darf der Container null oder mehrere Link-Datensätze enthalten, die zur Festlegung der Link-Beziehungen zwischen Dokumenten verwendet werden. Alle Datensätze müssen die Container- und Linkset-Ontologien in Bezug nehmen.

Die Container- und Linkset-Ontologien werden in den folgenden beiden Abschnitten beschrieben, wobei die RDF/OWL-Quelle in Anhang E enthalten ist. Die beiden Arten von Datensätzen werden dann in 4.2.4 bzw. 4.2.5 beschrieben.

4.4.2 Container-Ontologie

Die Container-Ontologie ist eine RDF/OW-Datei, die Definitionen der Klassen und Eigenschaften enthält, welche in einem Index-Datensatz verwendet werden, der Metadaten über den Container zur Verfügung stellt. Der Index-Datensatz ermöglicht die Festlegung:

- der Version der ICDD-Norm über den Import der Referenz-Ontologie;

- einer Liste externer Dokumente mit den Metadaten;
 - vorgeschriebener Dateiname (einschließlich eines URI oder IRI);
 - freigestelltes Format;
 - freigestellte Beschreibung;
- einer Liste interner Dokumente mit den Metadaten;
 - vorgeschriebener Dateiname (einschließlich seines Pfads innerhalb der Ordnerstruktur des Containers);
 - freigestelltes Format;
 - freigestellte Beschreibung;
- einer Verweisung auf Link-Datensätze.

In den Tabellen 4, 5 und 6 sind die Objekte, Datentyp-Eigenschaften bzw. Objekteigenschaften aufgeführt, die in der Container-Ontologie verwendet werden, wobei für alle jeweils Kurzbeschreibungen angegeben sind.

Tabelle 4 — In der Container-Ontologie festgelegte Klassen

Objektname	Beschreibung
ct:ContainerDescription	eine Beschreibung für einen Container, in der alle Dokumente aufgeführt werden und in der Link-Datensätze aufgefunden werden können
ct:EncryptedDocument	eine Verweisung auf ein verschlüsseltes Dokument
ct:ExternalDocument	eine Verweisung auf ein Dokument außerhalb eines Containers
ct:InternalDocument	eine Verweisung auf ein Dokument innerhalb eines Containers
ct:Linkset	eine Verweisung auf eine RDF/OWL-Datei, die Links enthält
ct:Document	eine abstrakte Klasse für Verweisungen auf ein Dokument; ein Individuum muss <i>ct:ExternalDocument</i> oder <i>ct:InternalDocument</i> angehören; und wahlweise dürfen Individuen auch anderen Untertypen von <i>ct:Document</i> angehören, wie z. B. <i>ct:SecuredDocument</i> und/oder <i>ct:EncryptedDocument</i>
ct:SecuredDocument	ein Dokument, das durch einen Prüfsummen-Algorithmus geschützt ist (siehe auch Eigenschaften <i>ct:checksum</i> und <i>ct:checksumAlgorithm</i>)
ct:FolderDocument	ein Dokument, das mehrere Dateien umfasst, die sich in einem Ordner befinden, wie z. B. ein GIS-Datensatz, der aus SHP-Dateien mit zugehörigen DBF-Dateien besteht
ct:Party	eine abstrakte Klasse (verursacht durch die <i>owl:UnionClass</i> , die als Bedingung interpretiert wird) für <i>ct:Organisation</i> und/oder <i>ct:Person</i> ; Entitäten können sich auf ein Individuum einer Unterklasse von <i>ct:Party</i> über die Objekteigenschaften <i>ct:creator</i> oder <i>ct:modifier</i> beziehen
ct:Person	eine Klasse, die eine Person für Herkunftszwecke darstellt
ct:Organisation	eine Klasse, die eine Organisation für Herkunftszwecke darstellt

Tabelle 5 — In der Container-Ontologie verwendete Datentyp-Eigenschaften

Bezeichnung des Datentyps	Beschreibung
ct:checksum	eine Prüfsumme für die Verweisung auf Dokumente; der Prüfsummen-Algorithmus wird durch die Eigenschaft <i>ct:checksumAlgorithm</i> festgelegt
ct:checksumAlgorithm	der Algorithmus, der zur Erzeugung der Prüfsumme verwendet wird
ct:conformanceIndicator	ein stringbasierter Indikator für <i>ct:ContainerDescription</i> um zu zeigen, welchem Teil der Norm dieser Container entspricht: bei einem Container nach Teil 1 sollte der Wert auf "ICDD-Part1-Container" gestellt werden; der Wertebereich ist nicht eingeschränkt, um andere Indikatorwerte zuzulassen
ct:creationDate	das Erstellungsdatum als <i>xsd:dateTime</i>
ct:description	eine allgemeine Beschreibung
ct:encryptionAlgorithm	freigestellter String, der die Verschlüsselung beschreibt
ct:filename	der Dateiname eines <i>ct:Linkset</i> oder <i>ct:InternalDocument</i> ; das Hauptverzeichnis stimmt mit dem Ordner „Nutzdaten-Dokumente“ des ICDD-Containers überein; das Zeichen Schrägstrich ("/") muss als Trennzeichen vom Ordner verwendet werden
ct:foldername	ein Ordnername zur Festlegung eines Ordners, in dem ein Dokument aus mehreren Dateien aufgefunden werden kann; das Hauptverzeichnis stimmt mit dem Ordner „Nutzdaten-Dokumente“ des ICDD-Containers überein; das Zeichen Schrägstrich ("/") muss als Trennzeichen vom Ordner verwendet werden
ct:filetype	ein String, der den Dateityp festlegt, wie z. B. "GML", "IFC", "shp", "xlsx", "pdf", "rvt"; bei dem String darf es sich um einen zusammengesetzten String handeln, um Version und Datenformat anzugeben (z. B. "ifc-4-xml-zip")
ct:format	der Medientyp eines Dokuments nach der Spezifikation durch die Internet Assigned Numbers Authority (https://www.iana.org/assignments/media-types/media-types.xhtml); Beispiele sind „application/pdf“ und „audio/mpeg“
ct:modificationDate	das Änderungsdatum als <i>xsd:dateTime</i>
ct:name	ein Name für ein Dokument
ct:pending	ein Boole'scher Wert zur Angabe, ob ein Dokument erforderlich ist oder nicht; wenn diese Eigenschaft nicht festgelegt ist, kann der Wert als „falsch“ interpretiert werden
ct:url	der URL, unter dem das externe Dokument gefunden werden kann
ct:userID	der vom Nutzer festgelegte Bezeichner
ct:versionDescription	eine freigestellte Zeichenfolge, die verwendet werden darf, um eine Beschreibung einer Version der entsprechenden Ressource bereitzustellen
ct:versionID	eine freigestellte Zeichenfolge, die verwendet werden darf, um eine Version der entsprechenden Ressource zu identifizieren

Tabelle 6 — In der Container-Ontologie verwendete Objekt-Eigenschaften

Bezeichnung der Objekt-Eigenschaft	Beschreibung
ct:creator	eine Verweisung auf einen Erzeuger dieser Instanz, der nur eine <i>ct:Party</i> (oder eine Unterklasse von <i>ct:Party</i>) sein kann
ct:modifier	eine Verweisung auf einen Modifikator dieser Instanz, der nur eine <i>ct:Party</i> (oder eine Unterklasse von <i>ct:Party</i>) sein kann
ct:publisher	die Partei, die dafür verantwortlich ist, dass der Container verfügbar ist
ct:alternativeDocument	eine Eigenschaft zur Verknüpfung eines Dokuments mit einer alternativen Version dieses Dokuments
ct:belongsToContainer	eine OWL-Eigenschaft, die die Beziehung zwischen einer Dokumentverweisung und einem Container festlegt
ct:containsLinkset	eine Beziehung einer <i>ct:ContainerDescription</i> zu einer <i>ct:Linkset</i> -Verweisung. Mehrere Linkset-Verweisungen sind zulässig.
ct:containsDocument	eine Beziehung einer <i>ct:ContainerDescription</i> zu einer Dokumentverweisung. Beziehungen zu mehreren Dokumentverweisungen sind zulässig.
ct:priorVersion	eine freigestellte Verweisung auf die vorherige Version dieser Ressource

Bild 8 stellt den Zusammenhang der Container-Ontologie dar, indem die Struktur eines Containers und die damit verbundenen Metainformationen dargestellt werden. Die Klasse *ct:ContainerDescription* ist eine Unterklasse der Klasse *owl:Ontology* (mehr Informationen zu *owl:Ontology* können aus 8.2 von <https://www.w3.org/TR/2012/REC-owl2-primer-20121211/> entnommen werden). Jede Ontologie ist ein Individuum, das der Klasse *owl:Ontology* angehört.

Der Index-Datensatz eines Containers gibt ein Individuum an, das ein Typ von *ct:ContainerDescription* ist und somit auch der Klasse *owl:Ontology* angehört. Der IRI oder URI dieses Individuums wird als Bezeichner für den Index-Datensatz verwendet und kann folglich zur Identifizierung des Containers verwendet werden. Der Index-Datensatz importiert die Container-Ontologie über die Eigenschaft *owl:import*, so dass das Individuum eine Eigenschaft *ct:description* und eine vorgeschriebene Objekteigenschaft *ct:publisher* übernimmt, die auf *ct:Party* verweisen. Er verweist auch auf Individuen der Klassen *ct:Linkset* und *ct:Document* über die Objekteigenschaften *ct:containsLinkset* bzw. *ct:containsDocument*, wodurch die Struktur des Containers festgelegt wird.

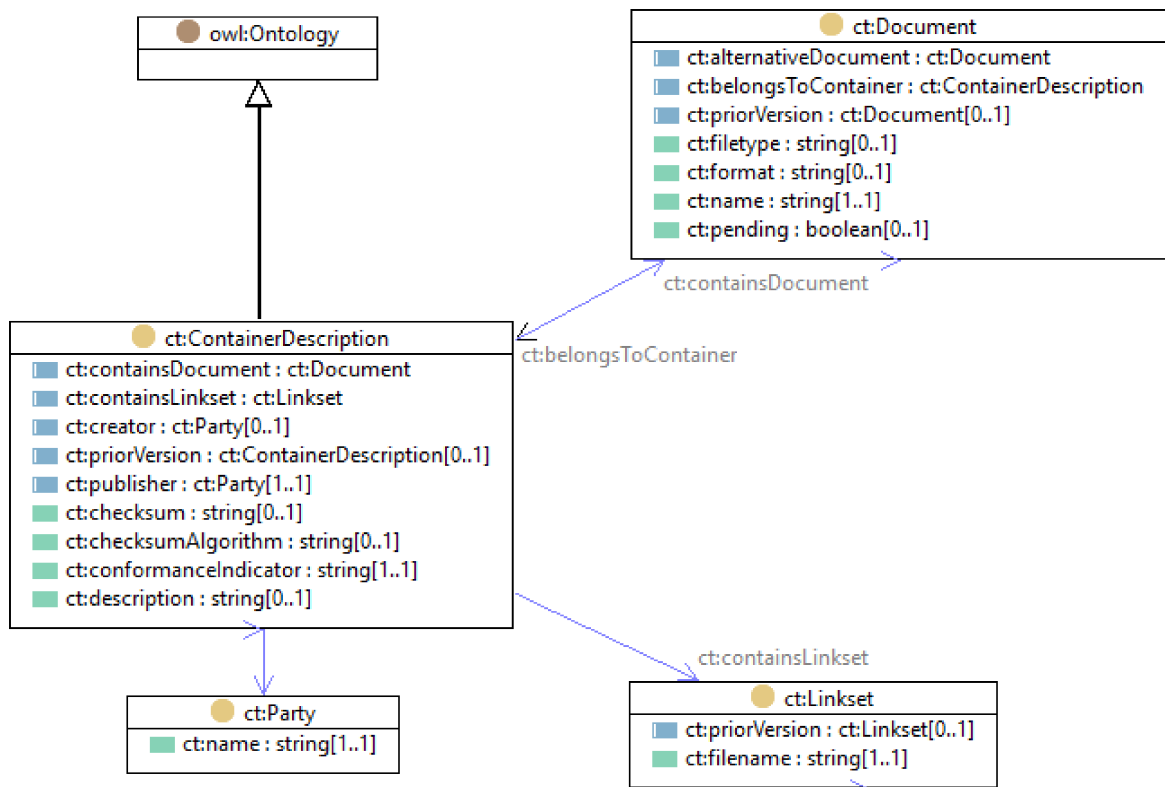


Bild 8 — Zusammenhang von *ct:ContainerDescription*

Bild 9 stellt die Eigenschaften und Unterklassen von Dokumenten dar, die in einem Container, der dieser Norm entspricht, unterstützt werden. Alle Dokumente müssen über *ct:name* und freigestellt über die Eigenschaften *ct:filetype*, *ct:format* und *ct:pending* verfügen (letztere ist ein Boole'scher Wert zur Angabe, dass das Dokument vom Empfänger des Containers angefordert wird). Individuen der Klasse *ct:Document* müssen entweder als *ct:InternalDocument* oder *ct:ExternalDocument* eingegeben werden (erzwungen durch eine *owl:unionOf*-Aussage, die als Restriktion interpretiert wird). Das Bild zeigt zusätzliche Unterklassen von *ct:Document*, die verwendet werden dürfen, um verschlüsselte, geschützte oder Ordner-Dokumente anzugeben.

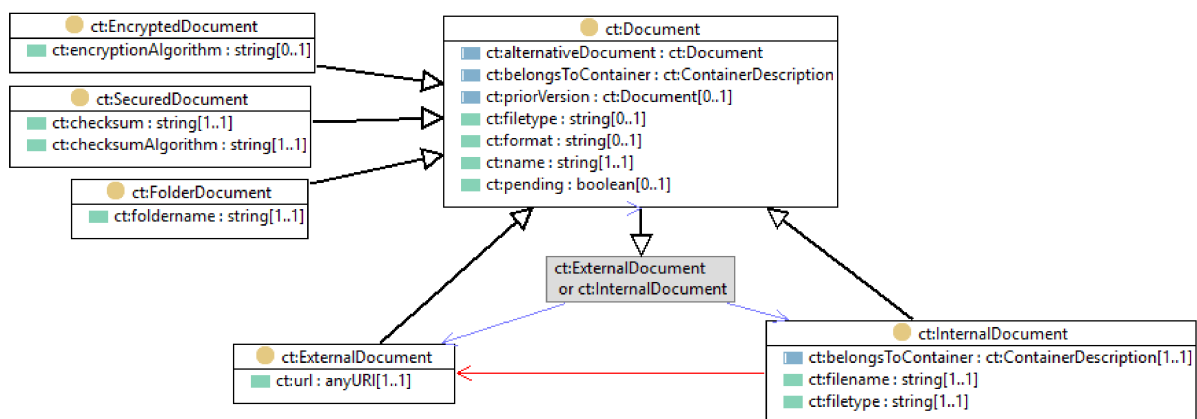


Bild 9 — Zusammenhang von *ct:Document*

In Abhängigkeit davon, welche Untertypen festgelegt werden, sind neue Eigenschaften verfügbar oder vorgeschrieben. Die mehrfache Eingabe von Dokumenten ist zulässig, es sei denn, zwischen diesen Klassen ist eine *owl:disjoint*-Aussage vorhanden. *ct:FolderDocument* stellt ein Dokument dar, das in einem Ordner vorhanden ist und aus mehreren Dateien bestehen darf. Eine Klasse *ct:EncryptedDocument* wird verwendet um anzugeben, dass ein Dokument verschlüsselt ist. *ct:SecuredDocument* kann verwendet werden, wenn ein Dokument durch einen Hash unter Anwendung eines Algorithmus geschützt ist, wie z. B. SHA256.

4.4.3 Linkset-Ontologie

Die Linkset-Ontologie ist eine RDF/OWL-Datei, welche die Objektklassen und Eigenschaften bereitstellt, die zur Erzeugung eines Link-Datensatzes verwendet werden müssen. Ein Link-Datensatz legt die Verknüpfungen zwischen Dokumenten fest und verwendet die Dokumentbeschreibungen aus dem Index-Datensatz erneut, weshalb er den Index-Datensatz importieren muss.

Ein Link kann Wechselbeziehungen zwischen zwei oder mehreren Dokumenten sowie zwischen Elementen, die in diesen Dokumenten enthalten sind, festlegen. Die meisten typischen Anwendungen von Links sind Verknüpfungen:

- zwischen einem einzelnen Element und einem zugehörigen Dokument (z. B. eine Zeile in einem Arbeitsblatt und eine GML-Datei);
- zwischen einem Element in einem Dokument und mehreren zugehörigen Elementen in anderen Dokumenten (z. B. ein Katalog-Element und die Häufigkeit des Vorkommens in einem Bauwerksmodell);
- zwischen einer Gruppe von Elementen in einem Dokument und zugehörigen Elementen in mehreren Dokumenten (z. B. eine Gruppe von Spalten und die entsprechenden Spezifizierungspunkte in einem Leistungsverzeichnis und die Prozesse in einem Terminplan-Dokument).

Zur Identifizierung bestimmter Elemente innerhalb eines Dokuments stellt diese Norm drei allgemeine Mechanismen bereit: ein stringbasierter Bezeichner, eine Abfrage oder ein URL-basierter Bezeichner. Die Wahl der verwendeten Elementattribute und der Syntax der Bezeichner und Abfragen ist den Implementierern überlassen.

In den Tabellen 7, 8 und 9 sind die Objekte, Datentyp-Eigenschaften bzw. Objekteigenschaften aufgeführt, die in der Linkset-Ontologie verwendet werden, wobei für alle jeweils Kurzbeschreibungen angegeben sind.

Tabelle 7 — In der Linkset-Ontologie festgelegte Klassen

Objektnamen	Beschreibung
ls:BinaryLink	ein <i>ls:Link</i> , der genau 2 Individuen der Klasse <i>ls:LinkElement</i> umfasst
ls:DirectedLink	ein <i>ls:Link</i> , der die Untereigenschaften <i>ls:hasFromLinkElement</i> und <i>ls:hasToLinkElement</i> nutzt, um eine Richtung dieses Links anzugeben
ls:DirectedBinaryLink	ein Untertyp eines binären Links (der genau 2 Instanzen von <i>ls:LinkElement</i> hat), der die Untereigenschaften <i>ls:hasFromLinkElement</i> und <i>ls:hasToLinkElement</i> verwendet, um eine Richtung dieses Links anzugeben
ls:Directed1toNLink	ein <i>ls:Directed1toNLink</i> ist ein Untertyp von <i>ls:DirectedLink</i> , der genau 1 <i>ls:hasFromLinkElement</i> anordnet
ls:Identifizier	eine abstrakte Klasse zur Identifizierung eines Elements innerhalb eines Dokuments; wenn ein Bezeichner berechnet werden darf, muss dies vom Implementierer vorgenommen werden, da in dieser Norm kein Verfahren dafür festgelegt ist
ls:Link	eine Gruppierung von 1 oder mehr <i>ls:Link-Elementen</i>

Objektname	Beschreibung
ls:LinkElement	eine Proxy-Klasse zur Bezugnahme auf ein Dokument oder auf ein Element in einem Dokument
ls:QueryBasedIdentifier	ein Bezeichner eines Elements in einem Dokument auf der Grundlage einer Abfrage
ls:StringBasedIdentifier	Identifizierung eines Elements innerhalb eines Dokuments über eine String-ID
ls:URIBasedIdentifier	URI-basierter Bezeichner für ein Dokument oder Element innerhalb eines Dokuments, das sich im Web befindet

Tabelle 8 — In der Linkset-Ontologie festgelegte Datentyp-Eigenschaften

Bezeichnung des Datentyps	Beschreibung
ls:identifizier	eine Eigenschaft des Datentyp-Strings, die den tatsächlichen ID-String enthält
ls:identifizierField	ein String-Datentyp zur Festlegung des Feldes/der Felder, in dem/in denen der Bezeichner gefunden werden kann; wenn der Bezeichner aus mehreren Feldern besteht, muss der Implementierer die Syntax-Regeln auswählen
ls:queryExpression	die Abfrage, die zu einem Bezeichner führt
ls:queryLanguage	eine Spezifikation der Abfragesprache
ls:uri	ein URI zur Bezugnahme auf ein Dokument

Tabelle 9 — In der Linkset-Ontologie festgelegte Objekt-Eigenschaften

Bezeichnung des Objekttyps	Beschreibung
ls:hasIdentifier	eine Beziehung von <i>ls:LinkElement</i> zu einem <i>ls:Identifier</i>
ls:hasLinkElement	eine Beziehung eines <i>ls:Link</i> zu einem <i>ls:LinkElement</i>
ls:hasFromLinkElement	eine Beziehung eines <i>ls:Link</i> zu einem <i>ls:LinkElement</i> . Es ist eine Untereigenschaft von <i>ls:hasLinkElement</i>
ls:hasToLinkElement	eine Beziehung eines <i>ls:Link</i> zu einem <i>ls:LinkElement</i> . Es ist eine Untereigenschaft von <i>ls:hasLinkElement</i>
ls:hasDocument	eine Beziehung von <i>ls:LinkElement</i> zu einem <i>ct:Document</i>

Bild 10 zeigt den Grundaufbau eines *Is:Link*-Datensatzes.

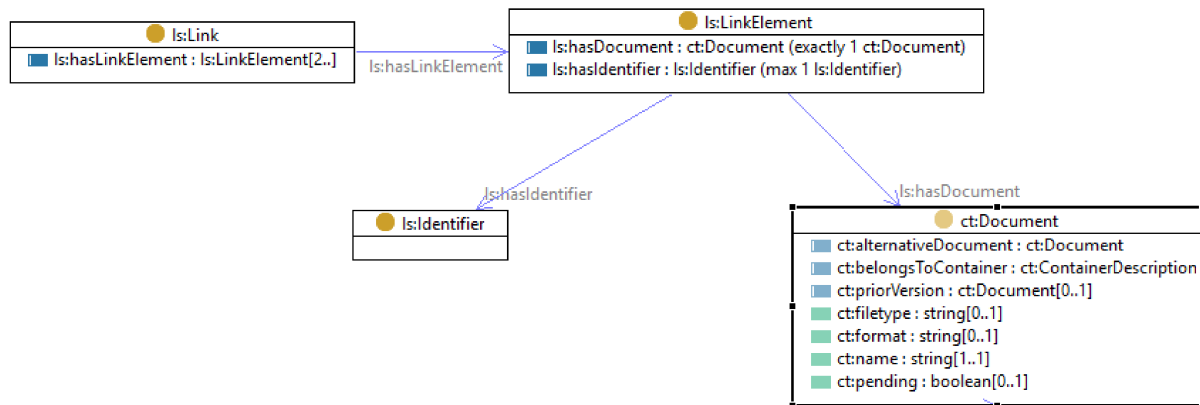


Bild 10 — Zusammenhang von *Is:Link*

Ein *Is:Link* hat mindestens 2 Instanzen von *Is:LinkElement* (*Is:hasLinkElement*). Ein *Is:LinkElement* verweist auf ein *ct:Document* (*Is:hasDocument*) und wahlweise auf einen Bezeichner (*Is:hasIdentifier*). Die Verweisung auf den Bezeichner ist für Deep-Linking notwendig, d. h. Bezugnahme auf einzelne Elemente innerhalb eines *ct:Document*, und wird nachstehend bei Bild 9 weiter erläutert.

In diesem Zusammenhang hat *Is:Link* die Unterklassen *Is:BinaryLink* und *Is:DirectedLink*.

Bild 11 zeigt die Struktur für *Is:BinaryLink*, mit genau zwei Instanzen von *Is:LinkElement* über die Eigenschaft *Is:hasLinkElement*. Des Weiteren wird ein *Is:DirectedBinaryLink* festgelegt als Unterklasse von *Is:BinaryLink* mit den vorgeschriebenen Eigenschaften *Is:hasFromLinkElement* und *Is:hasToLinkElement*. Diese Eigenschaften sind Untereigenschaften von *Is:hasLinkElement*. Ein *Is:DirectedBinaryLink* ist eine Unterklasse von sowohl *Is:BinaryLink* als auch *Is:DirectedLink* mit Übernahme aller Restriktionen.

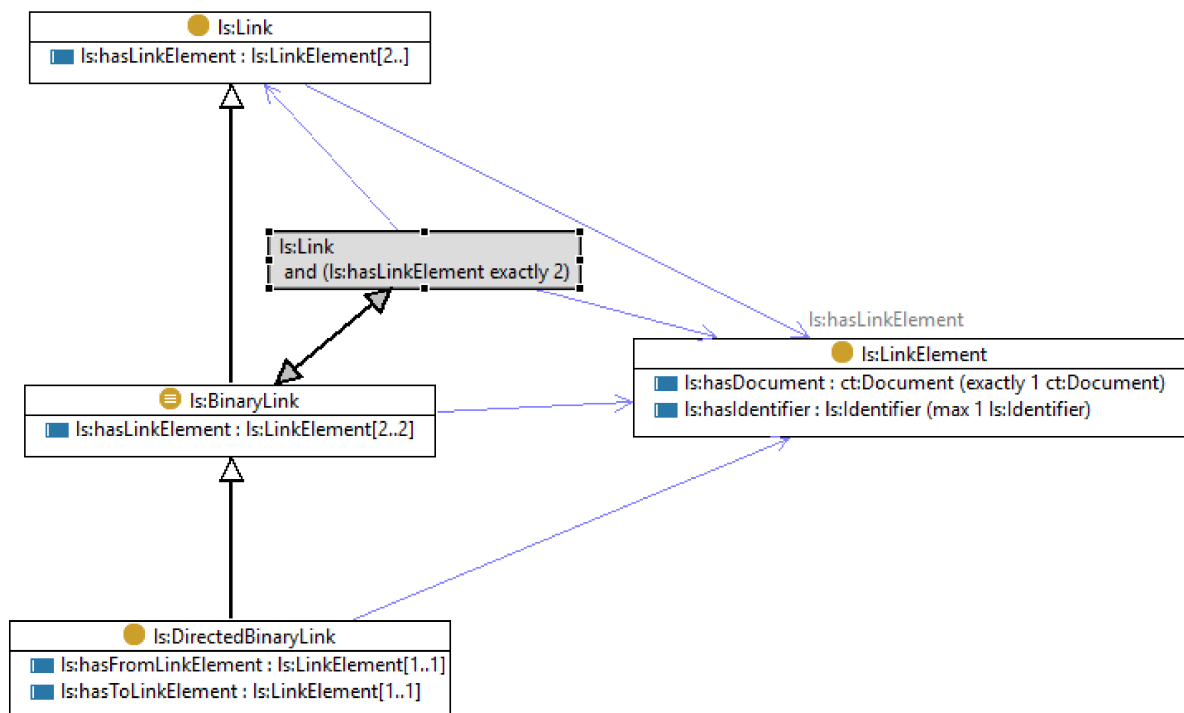


Bild 11 — Zusammenhang von *Is:BinaryLink*

Bild 12 zeigt die Struktur für *Is:DirectedLink*, d. h. eine Beziehung, bei der die Richtung der Beziehung von Bedeutung ist. Hier wird ein *Is:Directed1toNLink* als Unterklasse von *Is:DirectedLink* festgelegt, mit der zusätzlichen Restriktion „genau 1“ für die Eigenschaft *Is:hasFromLinkElement*.

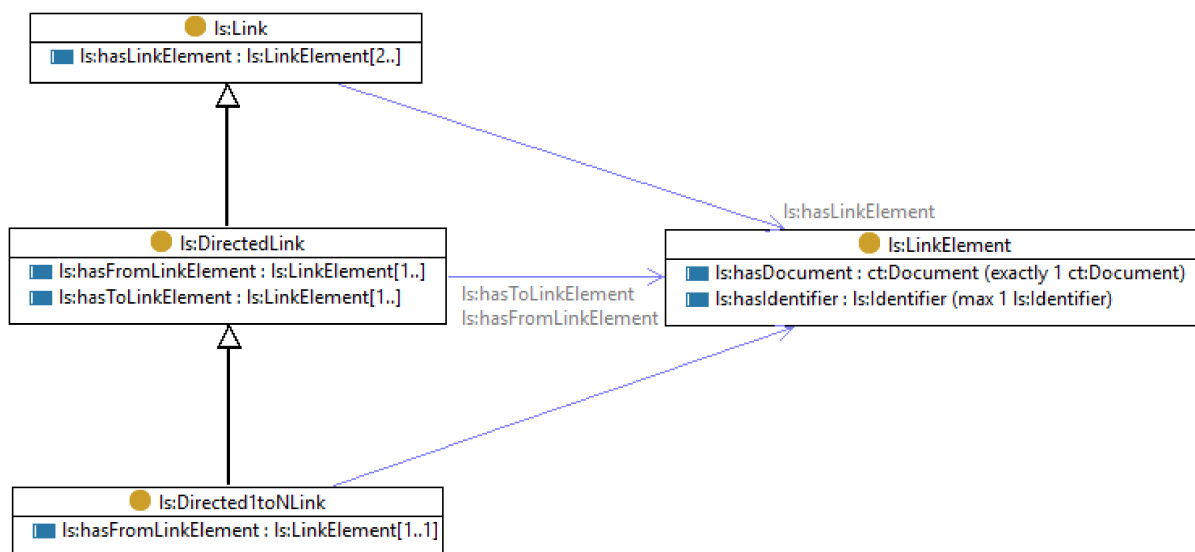


Bild 12 — Zusammenhang von *Is:DirectedLink*

Bild 13 zeigt die Objekte und Eigenschaften für *Is:Identifier*.

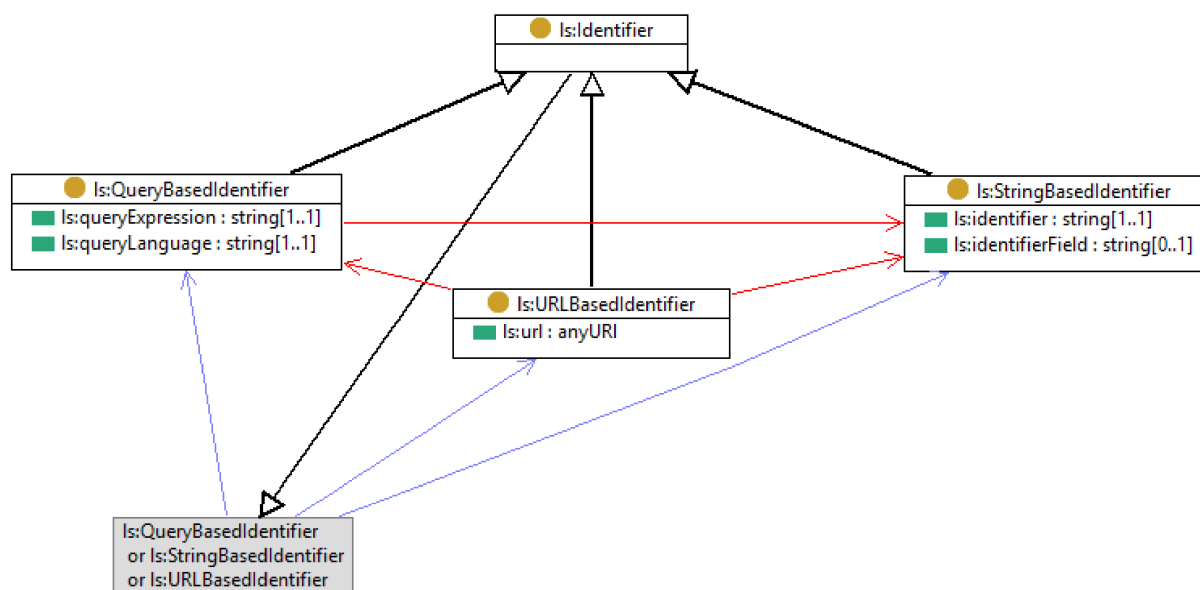


Bild 13 — Zusammenhang von *Is:Identifier*

Is:Identifier ist eine Unterklasse der owl:unionOf von *Is:StringBasedIdentifier*, *Is:URLBasedIdentifier* und *Is:QueryBasedIdentifier*. In dieser Norm wird dies als Bedingung interpretiert, was die Auswirkung hat, dass *Is:Identifier* zu einer abstrakten Oberklasse wird, die durch eine dieser drei Unterklassen realisiert werden muss.

Der *Is:StringBasedIdentifier* hat einen vorgeschriebenen *Is:Identifier*-Datentyp (xsd:string) für die tatsächliche ID. Der freigestellte *Is:identifierField* kann dazu verwendet werden, zusätzliche Informationen über das Feld zu liefern, wie z. B. wo die ID zu finden ist.

Der *ls:URLBasedIdentifier* kann verwendet werden, um auf ein Dokument oder ein Element innerhalb eines Dokuments zu verweisen, indem ein *xsd:anyUri* über die Datentyp-Eigenschaft *ls:url* angewendet wird.

Der *ls:URLBasedIdentifier* hat eine vorgeschriebene Datentyp-Eigenschaft *ls:queryExpression*, eine vorgeschriebene Datentyp-Eigenschaft *ls:queryLanguage* und eine freigestellte *ls:querySortExpression*. Eine SQL-Abfrage (z. B. "Select ID from table where Country='Mexico'") könnte die Eingabe für die *ls:queryExpression* sein, und „SQL“ könnte die Abfragesprache sein. Alternativ könnte eine XQuery (wie z. B. "for \$x in doc('costestimation.xml')/foundation/objects where \$x/price>30 order by \$x/id return \$x/id") die Eingabe von *ls:queryExpression* sein, und 'XQuery' könnte die Abfragesprache sein.

4.4.4 Index-Datensatz

Ein Container muss einen Index-Datensatz haben, der als *Index.rdf* bezeichnet wird. Der Index-Datensatz muss im Hauptverzeichnis des Containers abgelegt sein.

Der Index-Datensatz muss die Container-Ontologie über die Aussage *owl:import* in der *owl:Ontology*-Ressource importieren. Das Individuum *owl:Ontology* innerhalb des Graphen wird als eine *ct:ContainerDescription* eingegeben, und seine Eigenschaften können ausgefüllt werden. Individuen von *ct:Document* können dieser Beschreibung entsprechend der Container-Ontologie beigefügt werden. Jedes Individuum eines *ct:Document* beschreibt ein Dokument. Im Falle eines internen Dokuments ist eine Verweisung auf seine Lage im Ordner „Nutzdaten-Dokumente“ vorgeschrieben.

4.4.5 Link dataset

Der/die Link dataset(s) muss/müssen im Ordner „Nutzdaten-Tripeln“ des Containers abgelegt sein und muss/müssen mindestens die Linkset-Ontologie und den Index-Datensatz importieren. Er enthält alle Links zwischen Dokumenten als Individuen von *ls:Link* und *ls:LinkElement* entsprechend der Linkset-Ontologie.

4.5 Versionierung

Der Zweck einer Versionierung innerhalb dieser Norm besteht darin, die folgenden Funktionalitäten zu ermöglichen:

- Ermöglichung der Lieferung verschiedener alternativer Lösungen in einem Container (z. B. wenn verschiedene Konstruktionsszenarien vorgeschlagen werden);
- Ermöglichung einer Übertragung der Chronik von Ressourcen;
- Ermöglichung des Austauschs von Informationen über Versionen zwischen Partnern, um ein Mittel für die Verweisung auf bestimmte Stadien der Informationen bereitzustellen;
- Ermöglichung einer Rückverfolgung vorheriger Versionen.

Versionierung erfolgt durch Hinzufügen von Versionierungseigenschaften zu den Ontologien und deren Beschränkung auf bestimmte Domänen und Wertebereiche.

Die folgenden Subjekte dürfen versioniert werden:

- ein Individuum, das der Klasse *ct:ContainerDescription* angehört, wie in der Container-Ontologie festgelegt; die Versionsinformation, sofern vorgesehen, legt auf dieser Ebene eine Version für den Container als Ganzes fest;
- ein Individuum, das der Klasse *ct:Document* angehört, wie in der Container-Ontologie festgelegt; die Versionsinformation, sofern vorgesehen, legt auf dieser Ebene eine Version für das in Bezug genommene Dokument fest;

- ein Individuum, das *ct:Linkset* angehört, wie in der Container-Ontologie festgelegt; die Versionsinformation legt auf dieser Ebene eine Version für ein bestimmtes Linkset fest;

Wenn eine Ressource versioniert wird, muss die Eigenschaft *ct:versionID* verwendet werden, um die Version der Ressource anzugeben, und die entsprechende Eigenschaft *ct:versionDescription* darf verwendet werden, um eine erläuternde Beschreibung der Version einzuschließen. Diese Eigenschaft wird als funktional angegeben und es ist nur ein Wert zulässig. Die Eigenschaft *ct:versionID* muss den Wertebereich *xsd:string* haben. Diese Norm schreibt keine besondere Formatierung der Zeichenfolge für den Inhalt der Versionsidentifizierung vor und überlässt dies dem Anwender der Norm.

Um die Rückverfolgung der Versionschronik bei Ressourcen zu ermöglichen, darf eine versionierte Ressource neben *ct:versionID* mittels der Eigenschaft *ct:priorVersion* auch eine frühere Version anzeigen. Das Objekt eines *ct:priorVersion*-Prädikats muss vom selben Typ sein wie sein Subjekt.

Die Versionierung von Datenelementen innerhalb bestimmter Dokumente mit bestimmten Formaten (z. B. ENTITÄTS-Instanzen in .ifc-Dateien) ist nicht Gegenstand des Anwendungsbereichs dieser Norm.

Bild 14 stellt einen Beispiel-Datensatz mit Versionsinformationen bereit.

- ein Index-Datensatz, Version 1.3, der auf eine frühere Version desselben Containers verweist;
- der Container verweist auf eine Anzahl von Link-Datensätzen und Individuen der Klasse *ct:InternalDocument*;
- zwei Link-Datensätze, wovon ein Link-Datensatz die *ct:priorVersion* des anderen Link-Datensatzes ist;
- drei Dokumente:
 - zwei IFC-Dokumente, wovon eines die *ct:priorVersion* des anderen ist;
 - ein Arbeitsblatt-Dokument, das auf seine *ct:priorVersion* verweist (die nicht in diesem Container vorhanden ist).

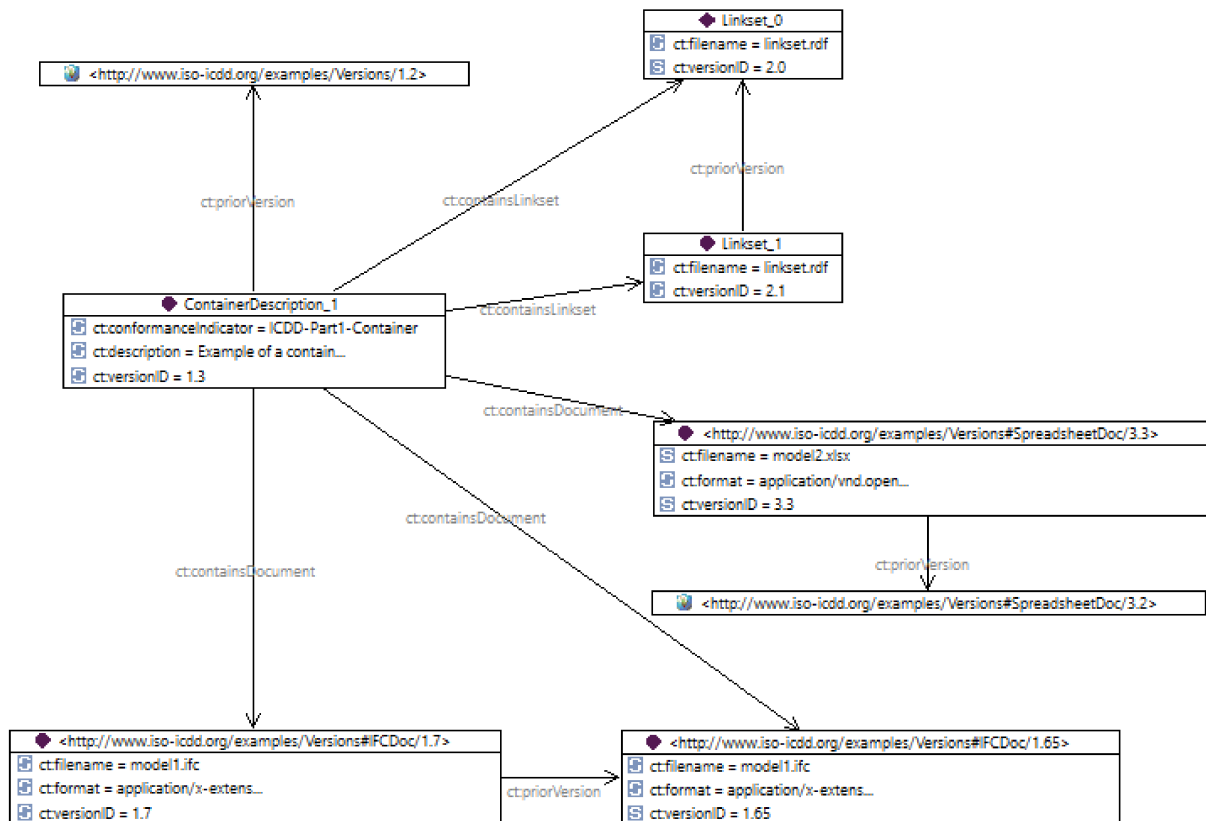


Bild 14 — Beispieldaten mit Versionsinformationen

4.6 Zusätzliche Eigenschaften in Datensätzen

Individuen in einem Datensatz dürfen nur den in dieser Norm festgelegten Klassen angehören. Jedoch ist es zulässig, weitere Prädikate, die außerhalb des Containers festgelegt sind, oder Linkset-Namensräume zu verwenden, um zusätzliche Eigenschaften festzulegen. Diese Prädikate und ihre Objekte können für den Austausch zusätzlicher Informationen für Individuen in den Datensätzen verwendet werden.

Das Objekt kann ein Literal oder ein URI sein. Die Norm schreibt keine weitere Interpretation dieser zusätzlichen Eigenschaften vor, weshalb jede zusätzliche Eigenschaft folglich als ein key-value-Paar behandelt werden muss. Anwendungsfall 1B in Anhang A enthält ein Beispiel für eine zusätzliche Eigenschaft.

5 Anforderungen an die Übereinstimmung

Jede Container-Datei, deren Übereinstimmung mit dieser Norm beansprucht wird, muss alle Anforderungen in diesem Abschnitt erfüllen.

Die folgenden Anforderungen gelten für den Container:

1. der Container muss eine gültige Zip64-Datei sein;
2. der Container muss eine Index.rdf -Datei im Hauptverzeichnis haben;
3. der Container muss die folgenden Ordner aufweisen:
 - a. Ontologie-Ressourcen;
 - b. Nutzdaten-Dokumente;
 - c. Nutzdaten-Tripeln;

4. wenn die Ontologie-Datei(en) Container.rdf und/oder Linkset.rdf im Container enthalten ist/sind, muss/müssen sie im Ordner „Ontologie-Ressourcen“ abgelegt sein;
5. der Container muss die Dateinamenerweiterung „icdd“ aufweisen.

Die folgenden Anforderungen gelten für die Index.rdf-Datei:

1. sie muss RDF/OWL entsprechen;
2. sie muss im RDF/XML-Serialisierungsformat vorliegen;
3. sie muss mit der Container.rdf -Datei übereinstimmen, wie in dieser Norm festgelegt;
4. sie muss die Ontologie Container.rdf durch Verwendung des Prädikats owl:import importieren;
5. sie muss alle internen und externen Dokumente auflisten;
6. sie muss alle Datensätze auflisten;
7. sie muss den Wert „ICDD-Part1-Container“ für die Eigenschaft *ct:conformanceIndicator* enthalten.

Die folgenden Anforderungen gelten für jede Link-Datensatz-Datei:

1. sie muss RDF/OWL entsprechen;
2. sie muss im RDF/XML-Serialisierungsformat vorliegen;
3. sie muss mit der Linkset.rdf-Ontologie übereinstimmen, wie in dieser Norm festgelegt;
4. sie muss im Ordner „Nutzdaten-Tripeln“ gespeichert sein.

Die folgenden Anforderungen gelten für jedes Dokument:

1. jedes im Container enthaltene Dokument muss im Ordner „Nutzdaten-Dokumente“ gespeichert werden.

Die folgenden Anforderungen gelten für die Erweiterbarkeit der Ontologien:

1. Es sind keine Erweiterungen zulässig, weshalb die Index.rdf-Datei und jede Link-Datensatz-Datei ausschließlich Individuen enthalten dürfen, die mit den Klassen übereinstimmen, die durch Container.rdf und Linkset.rdf festgelegt sind.

ANMERKUNG Für die Validierung von RDF-Graphen bietet W3C eine Shapes Constraint Language (SHACL) an, die seit dem 20. Juli 2017 eine offizielle Empfehlung von W3C ist. Es wird vorausgesehen, dass SHACL künftig von großer Bedeutung für die formale Validierung von RDF-Datei sein wird. Da es sich um eine neue Technologie handelt, beschränkt sich dieses Dokument diesbezüglich auf ein Beispiel in einem informativen Anhang (siehe Anhang F).

Anhang A (informativ)

Anwendungsfälle

A.1 Zusammenhang

Die Anwendungsfälle betreffen den Geschäftsablauf eines öffentlichen Auftraggebers im Infrastrukturbereich. Dieser Auftraggeber steuert die Infrastruktur in einer bestimmten Region. Als Teil des Managementprozesses werden regelmäßige Inspektionen durchgeführt. Die Durchführung der Inspektionen wird in einem Handbuch beschrieben. Inspektionen werden an Subunternehmer vergeben. Die Anwendungsfälle beziehen sich auf den Prozess an der Schnittstelle zwischen Auftraggeber und Auftragnehmer. Der Auftraggeber nutzt einen ICDD-Container, um die Austausch-Anforderungen für den Auftragnehmer verfügbar zu machen; der Auftragnehmer nutzt einen ICDD-Container, um die geforderten Informationen zu liefern.

A.2 Anwendungsfall 1A – Datenübergabe von Dokumenten

Der Auftraggeber fordert den Auftragnehmer auf, eine Instandhaltungsprüfung einer (in Bild A.1 dargestellten) Straßenüberführung vorzunehmen, die als Vermögensgegenstand mit dem Kennzeichnungscode 48D-100 bekannt ist. Nach Beendigung des Auftrags fordert der Auftraggeber eine Informationslieferung mit dem folgenden Inhalt an:

- Untersuchungsbericht (Excel, eine Zeile je Bauteil);
- eine oder mehrere Fotografien für jedes im Bericht angegebene Problem;
- ein 3D-Modell (zur Veranschaulichung des Bauwerks – im IFC-Format);
- ein Arbeitszeiterfassungsbogen (zur Angabe der für diesen Auftrag geleisteten Arbeitsstunden – im Excel-Format).

Für den Zweck des Auftrags liefert/endet der Auftraggeber einen Container, der über Slots für die angeforderten Dokumente verfügt. Die Spezifikation der Informationslieferung ist im Container enthalten, wie auch eine Dokumentvorlage für den Untersuchungsbericht und den Arbeitszeiterfassungsbogen.

Eine Straßenüberführung muss wie folgt zerlegt werden:

- Straßenüberführung;
 - Fundament;
 - Tragwerk;
 - Längsträger;
 - Brückenfahrbahn;
 - Auflager;
 - Stützpfeiler;
 - Säulen;
 - Dehnungsfuge.

Die Dokumentvorlage für einen Untersuchungsbericht legt die folgenden Datenelemente fest:

- Bezeichnung des Bauwerks;
- Vermögenswert-ID;
- Lage;
- Datum der Inspektion;
- Name des Prüfers 1;
- Name des Prüfers 2;
- Teil-ID;
- Teil-Typ;
- Zustand (gut, mäßig, schlecht, nicht maßgebend);
- festgestellte Mängel.

Die Dokumentvorlage für einen Arbeitszeiterfassungsbogen legt die folgenden Datenelemente fest:

- Vertrags-ID;
- Name des Mitarbeiters;
- ID des Mitarbeiters;
- Jahr;
- Woche;
- Tag;
- geleistete Arbeitsstunden;
- Beschreibung.

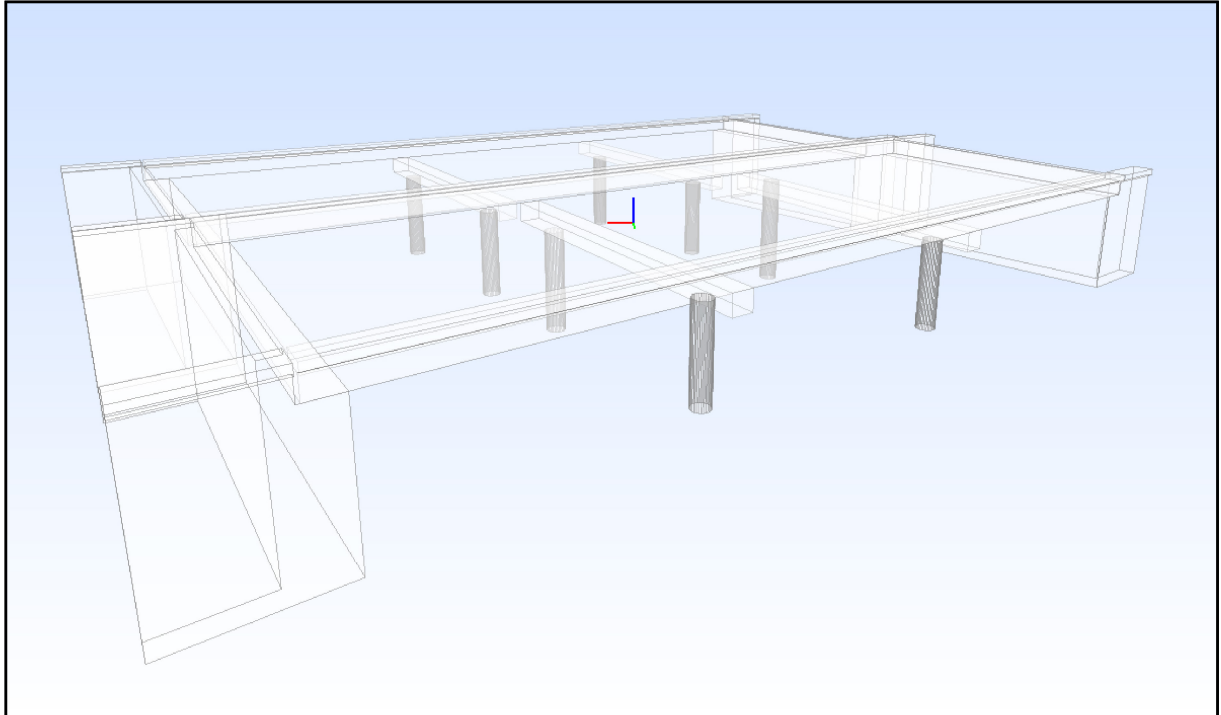


Bild A.1 — Beispiel für eine Straßenüberführung

Beispiel für einen Container — Anforderungen an die Informationslieferung

Dieses Beispiel erläutert den Container, der vom Auftraggeber bereitgestellt wird.

Bild A.2 stellt die Struktur des Containers dar und zeigt die Dokumente, die zur Verfügung gestellt werden.

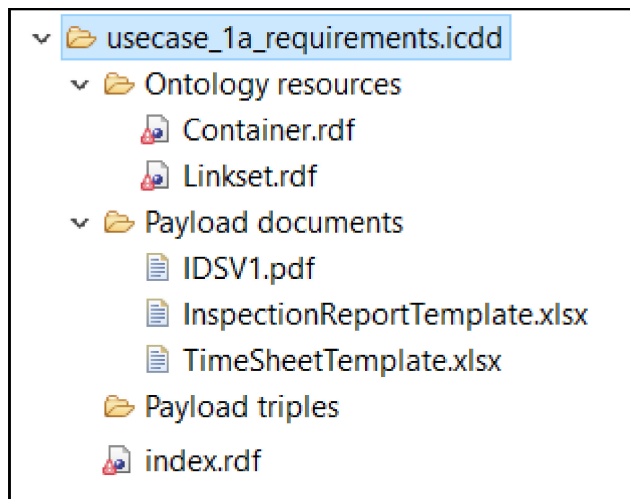


Bild A.2 — Struktur des Containers

Index.rdf im Hauptverzeichnis enthält eine Auflistung aller geforderten Dokumente und Dokumente, die bereits im Container zur Verfügung stehen. Bild A.3 zeigt eine Instanz einer *ct:ContainerDescription*.

Die Instanz hat einen URI, der im Ressourcen-Formular auf dem Bildschirmfoto oben unterlegt ist, und als eindeutiger Bezeichner für diesen Container behandelt wird. Dieses Individuum hat ein *rdf:type*-Prädikat zu *ct:ContainerDescription* und *owl:Ontology*, wodurch diese Ressource zu einem Angehörigen dieser beiden Klassen gemacht wird. Wie gezeigt wird dieses Individuum durch mehrere weitere Prädikate beschrieben, von denen jedes entweder einen vorgeschriebenen Wert hat (z. B. *ct:description*) oder auf URIs verweist, die andere Objekte identifizieren (z. B. verweist *ct:creator* über seinen URI auf eine Instanz von *ct:Party*).

In diesem Beispiel bezieht sich das Individuum über das Prädikat *ct:containsDocument* auf sieben weitere URIs. Einer dieser URIs ist geöffnet, wodurch gezeigt wird, dass er vom Typ (*rdf:type*) *ct:InternalDocument* ist, mit den Eigenschaften *ct:description*, *ct:filename*, *ct:filetype*, *ct:name* und dem Boole'schen Wert *ct:pending*, der in diesem Fall auf „wahr“ gesetzt ist, weil das Arbeitsblatt vom Auftragnehmer geliefert werden muss. Die anderen URIs können ebenfalls geöffnet werden.

Es wird darauf hingewiesen, dass einige Dokumente noch hängig sind, da der Boole'sche Wert *ct:pending* auf „wahr“ gesetzt ist. Weitere Dokumente sind im Container verfügbar und sind folglich nicht hängig (*ct:pending* ist auf falsch gesetzt).

Resource Form

URL: http://www.iso-icdd.org/examples/part1/usecase1A/requirements/indexexample1a_requirements.rdf

Annotations

owl:backwardCompatibleWith

owl:incompatibleWith

owl:priorVersion

Other Properties

icdd:containsDocument

id004f7641-fdf6-4a13-8a9e-e2ec97b9f20f

id00ebf1a5-a992-4ee6-a2ed-1ed25b198e19

id5cf50ea1-5354-4287-bddd-5b2ea8915de3

rdf:type

icdd:InternalDocument

icdd:description

Excel inspection report. One row in Excel must be for each construction part

icdd:filename

inspection_48D-100.xls

icdd:filetype

xls

icdd:name

inspection_48D-100.xls

icdd:pending

true

id64e79dfe-d68a-42d4-8a72-1b003a8cedd8

id9fd2b12f-f848-49e4-9c51-5c05e24cdd4f

idac195517-9126-4f1f-afca-c985e7a3357a

iddfab63d7-d6e3-4f18-9a79-32d7327c456a

icdd:containsLinkset

icdd:creator

id22638c4c-9b12-4ce3-b0be-e152f661e707

icdd:description

icdd showcase 1a: Requirements container

icdd:priorVersion

icdd:publisher

id24b576a7-6e29-4f9b-9a89-630183bc4e51

rdf:type

icdd:ContainerDescription

owl:Ontology

owl:imports

<http://www.iso-icdd.org/draft/Container.rdf>

Bild A.3 — *ct:ContainerDescription* in der Index.rdf-Datei eines Containers

Dieser Container kann unter http://www.iso-icdd.org/examples/part1/usecase_1a_requirements.icdd heruntergeladen werden.

(URIs in dem Datensatz dürfen von den URIs auf den Bildschirmfotos abweichen).

Beispiel für einen Container — Informationslieferung

Dieses Beispiel erläutert den Container, der vom Auftragnehmer zurückgesendet wird.

Der Liefer-Container enthält den Untersuchungsbericht in Form eines Arbeitsblatts und den Arbeitszeiterfassungsbogen im Nutzdaten-Ordner des Containers. Der Nutzdatenordner enthält außerdem auch ein 3D-IFC-Modell und einige Fotos, die den Untersuchungsbericht unterstützen. Siehe Bild A.4 bezüglich der Struktur des Containers.

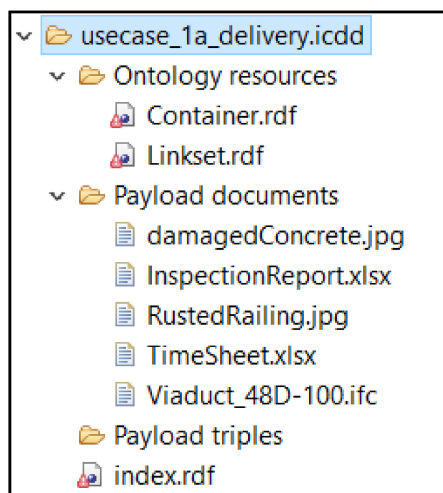


Bild A.4 — Struktur des Containers

Die *ct:ContainerDescription* ist über Individuen von *ct:InternalDocuments* mit den Dokumenten verknüpft. Bild A.5 zeigt die Container-Beschreibung, die mit einem internen Dokument verknüpft ist.

Resource Form

URI:

▼ Annotations

owl:backwardCompatibleWith ▼

owl:incompatibleWith ▼

owl:priorVersion ▼

▼ Other Properties

ct:checksum ▼

ct:checksumAlgorithm ▼

ct:conformanceIndicator ▼

ct:containsDocument ▼

- ☒ id23927804-835e-4a27-848a-ef0f0370cefe
- ☒ id2f6ad082-aff4-48e6-b55f-0d45e600e42b
- ☒ id74b4d6a7-abc3-4a46-9cc2-5c4a2d1118f1
- ☒ id9a230d37-79df-4f88-90cd-05aacfaa3c09

rdf:type ▼

☒ ct:InternalDocument

ct:creator ▼

- ☒ id5cd9d643-b474-4b6f-a65a-5c1f441fb073

ct:description ▼

ct:filename ▼

ct:filetype ▼

ct:format ▼

ct:name ▼

ct:pending ▼

☐ false

- ☒ ide83dce39-0bd5-44e8-bc10-a218cb908cbc

ct:containsLinkset ▼

ct:creator ▼

- ☒ id69f6c5d1-df3e-413e-8a8c-c24acf0ab4d8

Bild A.5 — *ct:ContainerDescription* in der Index.rdf-Datei eines Containers

Da alle Daten im RDF-Format vorliegen, ist es möglich, die Daten mit SPARQL abzufragen. Beispielsweise ist es möglich, Links über eine einzige SPARQL-Abfrage wiederzufinden. Das folgende Bild zeigt eine SPARQL-Abfrage, die die Daten nach Links absucht und die Bezeichner und Dateinamen eines Dokuments wiederfindet.

Query Editor	Query Library	[doc1]	identifier1	doc2	identifier2
<pre> prefix owl:<http://www.w3.org/2002/07/owl#> prefix ls:<http://www.iso-icdd.org/draft/Linkset.rdf#> prefix ct:<http://www.iso-icdd.org/draft/Container.rdf#> select ?doc1 ?identifier1 ?doc2 ?identifier2 where { ?link ls:hasLinkElement ?l1. ?link ls:hasLinkElement ?l2. filter (?l1=?l2) ?l1 ls:hasIdentifier [ls:identifier ?identifier1]. ?l2 ls:hasIdentifier [ls:identifier ?identifier2]. ?l1 ls:hasDocument [ct:filename ?doc1]. ?l2 ls:hasDocument [ct:filename ?doc2]. } </pre>		<ul style="list-style-type: none"> InspectionReport.xlsx 4 InspectionReport.xlsx 2 InspectionReport.xlsx 3 InspectionReport.xlsx 3 Viaduct_48D-100.ifc 20dnvrgv16XurcDISZ1ql3 Viaduct_48D-100.ifc 1D_dzaxZf4_QcddQY3uvjg Viaduct_48D-100.ifc 2ptPWWWHJCCfWE1B3yaT2A Viaduct_48D-100.ifc 2ptPWWWHJCCfWE1B3yaT2A damagedConcrete.jpg 3 damagedConcrete.jpg 3 		<ul style="list-style-type: none"> Viaduct_48D-100.ifc 20dnvrgv16XurcDISZ1ql3 Viaduct_48D-100.ifc 1D_dzaxZf4_QcddQY3uvjg Viaduct_48D-100.ifc 2ptPWWWHJCCfWE1B3yaT2A Viaduct_48D-100.ifc 2ptPWWWHJCCfWE1B3yaT2A damagedConcrete.jpg 3 damagedConcrete.jpg 3 Viaduct_48D-100.ifc 2ptPWWWHJCCfWE1B3yaT2A InspectionReport.xlsx 3 	

Bild A.6 — Anwendung von SPARQL zum Wiederfinden der Links

Dieser Container kann unter http://www.iso-icdd.org/examples/part1/usecase_1a_delivery.icdd heruntergeladen werden.

(URIs in dem Datensatz dürfen von den URIs auf den Bildschirmfotos abweichen).

A.3 Anwendungsfall 1B – Datenübergabe von Dokumenten mit Links

Dieser Anwendungsfall ist eine Erweiterung von Anwendungsfall 1A. In diesem Anwendungsfall fordert der Auftraggeber zusätzlich auch die Lieferung von Folgendem an:

- die Links zwischen den im Untersuchungsbericht angegebenen Bauteilen und den IFC-Elementen;
- die Links zwischen den im Untersuchungsbericht angegebenen Bauteilen und den Fotos.

Für den Zweck des Auftrags liefert der Auftraggeber einen Container, der über Slots für die angeforderten Dokumente verfügt. Die vorgenannten Anforderungen werden zur Spezifikation der Informationslieferung hinzugefügt und als Nachfolger der im vorhergehenden Anwendungsfall verwendeten Spezifikation der Informationslieferung gespeichert.

Der Auftraggeber verwendet Container, die von Auftragnehmern für den Zweck der Überprüfung von Untersuchungsberichten geliefert wurden. Dies bedeutet, dass der Auftraggeber über eine Anwendung verfügt, die den Untersuchungsbericht (Excel-Tabelle) zeigt, die entsprechenden Bauteile in einem 3D-Modell darstellt und verfügbare Fotos für jedes Bauteil zeigt.

Beispiel für einen Container — Anforderungen an die Informationslieferung

Dieser Container ist dem vorherigen Beispiel sehr ähnlich und enthält zwei Spezifikationen der Informationslieferung, die über die Eigenschaft *ct:priorVersion* miteinander in Beziehung stehen. Das folgende Bildschirmfoto (Bild A.7) zeigt den Link zu dem neuen Spezifikationsdokument der Informationslieferung.