

DIN EN ISO 19103



ICS 35.240.70

Einsprüche bis 2023-10-25

Entwurf**Geoinformation –
Konzeptuelle Beschreibungssprache (ISO/DIS 19103:2023);
Englische Fassung prEN ISO 19103:2023**

Geographic information –
Conceptual schema language (ISO/DIS 19103:2023);
English version prEN ISO 19103:2023

Information géographique –
Langage de schéma conceptuel (ISO/DIS 19103:2023);
Version anglaise prEN ISO 19103:2023

Anwendungswarnvermerk

Dieser Norm-Entwurf mit Erscheinungsdatum 2023-08-25 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 oder Am DIN-Platz, Burggrafenstr. 6, 10787 Berlin.

Es wird gebeten, mit den Kommentaren zu diesem Entwurf jegliche relevanten Patentrechte, die bekannt sind, mitzuteilen und unterstützende Dokumentationen zur Verfügung zu stellen.

Gesamtumfang 118 Seiten

DIN-Normenausschuss Bauwesen (NABau)

Nationales Vorwort

Dieses Dokument (prEN ISO 19103:2023) wurde vom Technischen Komitee ISO/TC 211 „Geographic information/Geomatics“ in Zusammenarbeit mit dem Technischen Komitee CEN/TC 287 „Geoinformation“ erarbeitet, dessen Sekretariat von BSI (Vereinigtes Königreich) gehalten wird.

Das zuständige deutsche Normungsgremium ist der Arbeitsausschuss NA 005-03-03 AA „Geoinformation (SpA zu CEN/TC 287+ISO/TC 211)“ im DIN-Normenausschuss Bauwesen (NABau).

Dieses Dokument enthält unter Berücksichtigung des DIN-Präsidialbeschlusses 1/2004 nur die Englische Originalfassung von prEN ISO 19103:2023.

Für die in diesem Dokument zitierten Dokumente wird im Folgenden auf die entsprechenden deutschen Dokumente hingewiesen:

ISO 2108	siehe	DIN ISO 2108
ISO 3166-1	siehe	DIN EN ISO 3166-1
ISO 8601-1	siehe	DIN ISO 8601-1
ISO 10241-1	siehe	DIN ISO 10241-1
ISO 19101-1:2014	siehe	DIN EN ISO 19101-1:2015-03
ISO 19107:2019	siehe	DIN EN ISO 19107:2020-09
ISO 19109:2015	siehe	DIN EN ISO 19109:2016-05
ISO 19108:2002	siehe	DIN EN ISO 19108:2005-05
ISO 19115-1:2014	siehe	DIN EN ISO 19115-1:2014-07
ISO 19118:2011	siehe	DIN EN ISO 19118:2012-01
ISO 19119:2005	siehe	DIN EN ISO 19119:2006-09
ISO 19123-1:2023	siehe	DIN EN ISO 19123-1:—*
ISO 19135-1:2015	siehe	DIN EN ISO 19135-1:2016-02
ISO 19136-1	siehe	DIN EN ISO 19136-1
ISO 19136-2	siehe	DIN EN ISO 19136-2
ISO 80000-3	siehe	DIN EN ISO 80000-3

Aktuelle Informationen zu diesem Dokument können über die Internetseiten von DIN (www.din.de) durch eine Suche nach der Dokumentennummer aufgerufen werden.

* Zur Zeit Entwurf.

Nationaler Anhang NA (informativ)

Begriffe

Reihenfolge und Inhalt der folgenden Begriffe sind identisch mit denen im Abschnitt Begriffe der englischen Fassung.

Für die Anwendung dieses Dokuments gelten die Begriffe nach ISO/IEC 11404:2007 [2] und die folgenden Begriffe.

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

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

ANMERKUNG UML 2.5.1 enthält weder Terminologieeinträge in Übereinstimmung mit ISO 10241-1 [7] noch ein Glossar — im Gegensatz zu früheren UML 1.x-Spezifikationen, siehe z. B. UML 1.5 [8] — deshalb besteht die Notwendigkeit, ausgewählte UML-Konzepte in Abschnitt 3 dieses Dokuments zu ergänzen. Die aus UML 2.5.1 übernommenen Konzepte sind in Terminologieeinträgen definiert, die

- „UML“ als Domain haben;
- Begriffe, Definitionen und Anmerkungen enthalten, die auf der Beschreibung der Konzepte im Hauptteil von UML 2.5.1 basieren und die redaktionell geändert worden sein können, um den Grundsätzen der Terminologie zu entsprechen;
- zusätzliche Beispiele und Anmerkungen zum Begriff beinhalten können, die für dieses Dokument relevant sind;
- mit einer Anmerkung zum Begriff mit Angabe des Abschnitts/der Abschnitte in UML 2.5.1 versehen sein können, in dem/denen das Konzept beschrieben wird/werden.

3.1

abstrakter Klassifikator

en **abstract classifier**

<UML> *Klassifikator*, der über keine direkten *Instanzen* verfügt

Anmerkung 1 zum Begriff: Jede Instanz eines abstrakten Klassifikators muss eine Instanz von einer seiner Spezialisierungen sein.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 9.2.3.2.

3.2

abstraktes Schema

en **abstract schema**

konzeptuelles Schema, das ohne weitere Spezifikation nicht implementierbar ist

BEISPIEL Das konzeptuelle Schema für die Beschreibung der räumlichen Merkmale der in ISO 19107:2019 [9] definierten Entitäten.

Anmerkung 1 zum Begriff: Ein abstraktes Schema kann auf viele Bereiche angewendet werden.

Anmerkung 2 zum Begriff: Ein abstraktes Schema kann mittels eines *Anwendungsschemas* realisiert werden.

3.3

Abstraktion

en **abstraction**

<Informationstechnologie> Ergebnis eines *Abstraktionsprozesses*

Anmerkung 1 zum Begriff: Im Bereich der Informationstechnologie repräsentiert der Begriff „Abstraktion“ auch das in 3.5 definierte Konzept.

3.4

Abstraktion

en **abstraction**

<UML> *Abhängigkeit*, die zwei *benannte Elemente* oder Gruppen von benannten Elementen zueinander in Beziehung setzt, die das gleiche *Konzept* auf unterschiedlichen *Abstraktionsebenen* oder aus verschiedenen Blickwinkeln repräsentieren

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 7.7.3.3.

3.5

Abstraktionsprozess

Abstraktion

en abstraction process

abstraction

<Informationstechnologie> Prozess des Herausfilterns von Details, die nicht innerhalb des Interessenbereichs liegen

Anmerkung 1 zum Begriff: Abstraktion ermöglicht das leichtere Verständnis der wesentlichen Punkte einer Sache und den Umgang mit Komplexität.

Anmerkung 2 zum Begriff: Im Bereich der Informationstechnologie repräsentiert der Begriff „Abstraktion“ auch das in 3.3 definierte Konzept.

3.6

Aggregation

geteilte Aggregation

en **shared aggregation**

<UML> *binäre Assoziation*, die eine *Teil-Ganzes-Beziehung* festlegt, bei der das Ganze keine Verantwortung für die Existenz seiner Teile besitzt

Anmerkung 1 zum Begriff: Ein Teil kann gleichzeitig in mehr als einem Ganzen enthalten sein.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 9.5.3.

3.7

Anwendung

Be- und Verarbeitung von Daten zur Unterstützung von Nutzeranforderungen

[QUELLE: ISO 19101-1:2014 [10], 4.1.1]

3.8

Anwendungsschema

en **application schema**

konzeptuelles Schema für Daten, das für eine oder mehrere *Anwendungen* erforderlich ist

[QUELLE: ISO 19101-1:2014 [10], 4.1.2]

3.9

Assoziation

en **association**

<UML> semantische *Beziehung*, die zwischen *Instanzen* eines bestimmten *Typs* bestehen kann

Anmerkung 1 zum Begriff: Eine Assoziation ist außerdem eine Art von *Klassifikator*.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 11.5.3.1.

3.10

Attribut

en **attribute**

<UML> *Eigenschaft*, die ein *Klassifikator* abgesehen von einer *Assoziation* besitzt

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 9.5.3.

3.11

verhaltensbezogenes Feature

en **behavioural feature**

<UML> *Feature*, das einen Aspekt des Verhaltens festlegt

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 9.9.2.1.

3.12

binäre Assoziation

en **binary association**

<UML> *Assoziation* zwischen genau zwei Elementen (Member Ends)

Anmerkung 1 zum Begriff: Die beiden Member Ends können denselben *Typ* haben.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 11.5.

3.13

Kardinalität (einer Sammlung)

en **cardinality (of a collection)**

<UML> Anzahl der Elemente in einer Sammlung

BEISPIEL Die Kardinalität einer Sammlung mit drei Elementen ist drei.

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 7.5.3.2.

3.14

Klasse

en **class**

<UML> *Klassifikator* einer Gruppe von *Objekten*

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 11.8.3.1.

3.15

Klassendiagramm

en **class diagram**

<UML> Strukturschema, bei dem die primären Symbole im Inhaltsbereich entweder *Klassen*-Symbole oder *Schnittstellen*-Symbole oder beides sind

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, Anhang A.

3.16

Klassifikator

en **classifier**

<UML> Klassifizierung von *Instanzen* anhand ihrer *Features*

Anmerkung 1 zum Begriff: Ein Klassifikator ist eine Art von *Typ*.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 9.2.1.

3.17

Kodierungsgruppe

Codeelement-Gruppe

Code

Code-Liste

en code set

code element set

code list

Ergebnis der Anwendung eines *Kodierungsschemas* auf alle Elemente einer *kodierten Gruppe*

BEISPIEL Alle aus drei Zeichen bestehenden Darstellungen von Flughafenbezeichnungen.

Anmerkung 1 zum Begriff: Der zulässige Begriff „Code“ repräsentiert außerdem die in 3.18, 3.20 und ISO 19118:2011 [11], 4.3 definierten Konzepte.

[QUELLE: ISO/IEC 2382:2015 [12], 2121556, modifiziert — zusätzlichen zulässigen Begriff hinzugefügt, Definition so angepasst, dass die bevorzugten Begriffe der in Bezug genommenen Begriffsdefinitionen verwendet werden, Anmerkung 1 zum Begriff in ein Beispiel umgewandelt, Anmerkungen 2 und 3 zum Begriff entfernt, Anmerkung zum Begriff bezüglich Homographen hinzugefügt]

3.18

Codewert

Codeelement

Code

en code value

code element

Ergebnis der Anwendung eines *Kodierungsschemas* auf ein Element einer *kodierten Gruppe*

BEISPIEL 1 „CDG“ als Darstellung von „Paris Charles-De-Gaulle“ im Code für Darstellungen von Flughafenbezeichnungen mit drei Buchstaben.

BEISPIEL 2 Die Hexadezimalzahl 0041 als Darstellung des „lateinischen Großbuchstabens A“ in ISO/IEC 10646 [13].

Anmerkung 1 zum Begriff: Der zulässige Begriff „Code“ repräsentiert außerdem die in 3.17, 3.20 und ISO 19118:2011 [11], 4.3 definierten Konzepte.

[QUELLE: ISO/IEC 2382:2015 [12], 2121555, modifiziert — Definition so angepasst, dass die bevorzugten Begriffe der in Bezug genommenen Begriffsdefinitionen verwendet werden, Anmerkung 1 zum Begriff in zwei Beispiele umgewandelt, Anmerkungen 2 und 3 zum Begriff entfernt, Anmerkung zum Begriff bezüglich Homographen hinzugefügt]

3.19

kodierte Gruppe

en **coded set**

Gruppe von Elementen, die anhand eines *Kodierungsschemas* auf eine andere Reihe abgebildet werden

BEISPIEL Eine Liste mit Bezeichnungen von Flughäfen, die auf eine Liste mit entsprechenden, aus drei Buchstaben bestehenden Abkürzungen abgebildet wird.

[QUELLE: ISO/IEC 2382:2015 [12], 2121553, modifiziert — Definition so angepasst, dass der bevorzugte Begriff der in Bezug genommenen Begriffsdefinition verwendet wird, Anmerkung 1 zum Begriff in ein Beispiel umgewandelt, Anmerkungen 2 und 3 zum Begriff entfernt]

3.20

Kodierungsschema

Code

en **coding scheme**

Zusammenstellung von Regeln, die die Elemente einer ersten Gruppe auf die Elemente einer zweiten Gruppe abbildet

Anmerkung 1 zum Begriff: Die Elemente in jeder der beiden Gruppen können Zeichen oder Zeichenketten sein.

Anmerkung 2 zum Begriff: Die erste Gruppe wird als *kodierte Gruppe* bezeichnet, die zweite Gruppe wird als *Kodierungsgruppe* bezeichnet.

Anmerkung 3 zum Begriff: Jedes Element der Kodierungsgruppe kann zu mehr als einem Element der kodierten Gruppe in Beziehung gesetzt werden, umgekehrt ist dies aber nicht der Fall.

Anmerkung 4 zum Begriff: Der zulässige Begriff „Code“ repräsentiert außerdem die in 3.17, 3.18 und ISO 19118:2011 [11], 4.3 definierten Konzepte.

[QUELLE: ISO/IEC 2382:2015 [12], 2121552, modifiziert — bevorzugten und zulässigen Begriff vertauscht, „darf“ in den Anmerkungen 1 und 2 zum Begriff durch „können“ ersetzt, Anmerkungen 4 und 5 zum Begriff entfernt, Anmerkung zum Begriff bezüglich Homographen hinzugefügt.]

3.21

Kodiersystem

en **coding system**

Kombination aus einer *kodierten Gruppe* und einer *Kodierungsgruppe* auf Grundlage eines *Kodierungsschemas*

[QUELLE: EN 1068:2005 [14], 3.8, modifiziert — Definition so angepasst, dass die bevorzugten Begriffe der in Bezug genommenen Begriffsdefinitionen verwendet werden.]

3.22

Komposition

en **composition**

<UML> *binäre Assoziation*, die eine *Teil-Ganzes-Beziehung* festlegt, bei der das Ganze Verantwortung für die Existenz seiner Teile besitzt

Anmerkung 1 zum Begriff: Ein Teil kann gleichzeitig immer nur in einem Ganzen enthalten sein.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 9.5.3.

3.23

Konzept

en **concept**

Wissenseinheit, die durch eine eindeutige Kombination von Eigenschaften beschrieben wird

Anmerkung 1 zum Begriff: Konzepte sind nicht unbedingt an bestimmte *natürliche Sprachen* gebunden. Sie werden jedoch durch den sozialen oder kulturellen Hintergrund beeinflusst, was oft zu unterschiedlichen Kategorisierungen führt.

Anmerkung 2 zum Begriff: Hierbei handelt es sich um das Konzept „Konzept“, wie es durch den Begriff „Konzept“ in der Terminologiarbeit verwendet und bezeichnet wird. Dieses Konzept unterscheidet sich wesentlich von jenem, das von anderen Bereichen wie industrielle Automatisierung oder Marketing bezeichnet wird.

[QUELLE: ISO 1087:2019 [15], 3.2.7]

3.24

konzeptueller Formalismus

en **conceptual formalism**

Zusammenstellung von Modellierungskonzepten zur Beschreibung eines *konzeptuellen Modells*

BEISPIEL objektorientierte Modellierung

Anmerkung 1 zum Begriff: Ein konzeptueller Formalismus kann in mehreren *Modellierungssprachen* ausgedrückt werden.

[QUELLE: ISO 19101-1:2014 [10], 4.1.4, modifiziert — Beispiele ersetzt.]

3.25

konzeptuelles Modell

en **conceptual model**

Modell, das einen abstrahierten *Ausschnitt der realen Welt* abbildet

[QUELLE: ISO 19101-1:2014 [10], 4.1.5]

3.26

konzeptuelles Schema

en **conceptual schema**

formale Beschreibung eines *konzeptuellen Modells*

[QUELLE: ISO 19101-1:2014 [10], 4.1.6]

3.27

Modellierungssprache

en **conceptual schema language**

formale Sprache auf Grundlage eines *konzeptuellen Formalismus* zum Zwecke der Darstellung *konzeptueller Schemata*

BEISPIEL UML, EXPRESS, IDEF1X.

Anmerkung 1 zum Begriff: Eine Modellierungssprache kann lexikalisch oder grafisch sein. Verschiedene Modellierungssprachen können auf dem gleichen konzeptuellen Formalismus beruhen.

[QUELLE: ISO 19101-1:2014 [10], 4.1.7, modifiziert — Anmerkung zum Begriff geändert: „können“ anstelle von „dürfen“.]

3.28

Einschränkung

en **constraint**

<UML> Bedingung oder Beschränkung, die in *natürlicher Sprache* als Text oder in einer maschinenlesbaren Sprache zwecks Deklaration der Semantik eines Elements oder einer Gruppe von Elementen ausgedrückt wird

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 7.8.3.1.

3.29

Datentyp

en **data type**

Menge von einzelnen Werten, charakterisiert durch Eigenschaften dieser Werte und durch Operationen in Bezug auf diese Werte

BEISPIEL Der Datentyp „Boolean“ mit den Eigenschaften „unordered“, „exact“ und „non-numeric“ und mit den Operationen „gleich“, „not“, „and“ und „or“.

Anmerkung 1 zum Begriff: Die Eigenschaften von Datentyp-Werten sind „ordered“ oder „unordered“, „exact“ oder „approximate“, „numeric“ oder „non-numeric“ und, wenn „ordered“, „bounded“ oder „unbounded“, wie in ISO/IEC 11404:2007 [2] beschrieben.

[QUELLE: ISO/IEC 11404:2007 [2], 3.12, modifiziert — Anmerkung zum Begriff und Beispiel hinzugefügt.]

3.30

Datentyp

en **data type**

<UML> *Klassifikator*, dessen *Instanzen* sich nur durch ihren Wert unterscheiden

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 10.2.1.

3.31

Datenwert

en **data value**

<UML> *Instanz* eines *Datentyps*

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 7.5.3.2.

3.32

Abhängigkeit

en **dependency**

<UML> *gerichtete Beziehung*, die besagt, dass ein einzelnes Modell oder eine Gruppe von Modellelementen andere Modellelemente für ihre Spezifikation oder Implementierung benötigt

Anmerkung 1 zum Begriff: Eine Abhängigkeit bezeichnet eine Supplier/Client-Beziehung zwischen Modellelementen, bei der die Modifizierung eines Suppliers möglicherweise Auswirkungen auf die Modellelemente des Clients hat. Die komplette Semantik des Client-Elements/der Client-Elemente ist entweder semantisch oder strukturell von der Definition des Supplier-Elements/der Supplier-Elemente abhängig.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 7.7.1 und 7.8.4.

3.33

gerichtete Beziehung

en **directed relationship**

<UML> *Beziehung* zwischen einer Sammlung von Quellmodellelementen und einer Sammlung von Zielmodellelementen

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 7.8.5.1.

3.34

Aufzählung

en **enumeration**

<UML> *Datentyp*, dessen Werte nacheinander im *Modell* benannt sind

Anmerkung 1 zum Begriff: Die Gruppe von *Aufzählungsliteralen* einer *Aufzählung* ist geordnet.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 10.5.3.

3.35

Aufzählungsliteral

en **enumeration literal**

<UML> benutzerdefinierter *Datenwert* für eine *Aufzählung*

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 10.5.4.1.

3.36

Erweiterung

en **extension**

<UML> *Assoziation*, die anzeigt, dass die *Eigenschaften* einer *Metaklasse* durch ein *Stereotyp* erweitert werden

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 12.4.1.1.

3.37

Feature

en **feature**

<UML> Merkmal

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 9.4.3.1.

3.38

formale Sprache

en **formal language**

Sprache, die maschinenlesbar ist und über eine genau festgelegte Semantik verfügt

Anmerkung 1 zum Begriff: Bei klar definierter Semantik handelt es sich üblicherweise um modelltheoretische Semantik.

[QUELLE: ISO/IEC 21838-1:2021 [16], 3.10]

3.39

Generalisierung

en **generalization**

<UML> taxonomische *gerichtete Beziehung* zwischen einem allgemeineren *Klassifikator* und einem spezifischen *Klassifikator*

Anmerkung 1 zum Begriff: Der allgemeinere Klassifikator wird als „Parent“ oder „Superclass“ (Oberklasse) bezeichnet, wenn der Klassifikator eine *Klasse* ist. Der spezifischere Klassifikator wird als „Child“ bezeichnet. Die Klassifikatoren, die erreicht werden können, indem den Generalisierungen gefolgt wird, die von einem gegebenen Klassifikator in Richtung auf die allgemeineren Klassifikatoren ausgehen, werden als Generalisierungen des Klassifikators bezeichnet. Die Klassifikatoren, die erreicht werden können, indem den Generalisierungen gefolgt wird, die von einem gegebenen Klassifikator in Richtung auf die spezifischeren Klassifikatoren ausgehen, werden als Spezialisierungen des Klassifikators bezeichnet.

Anmerkung 2 zum Begriff: Jede *Instanz* des spezifischen Klassifikators ist zugleich eine Instanz des allgemeinen Klassifikators. Der spezifische Klassifikator erbt die *Features* des allgemeineren Klassifikators.

Anmerkung 3 zum Begriff: Übernommen aus UML 2.5.1, 9.2.3.2 und 9.9.7.

3.40

Identifikator

en **identifier**

sprachlich unabhängige Zeichenkombination zur eindeutigen und permanenten Identifizierung

[QUELLE: ISO 19135-1:2015 [17], 4.1.5]

3.41

Identität

en **identity**

inhärentes Merkmal einer *Instanz*, das diese von allen anderen Instanzen unterscheidet

[QUELLE: ISO/IEC/IEEE 24765:2017 [18], 3.1865, modifiziert — Definition geändert („Merkmal“ statt „Eigenschaft“), Anmerkung zum Eintrag entfernt.]

3.42

Vererbung

en **inheritance**

<Informationstechnologie> Mechanismus, durch den spezifischere Entitäten Strukturen und Verhaltensweisen von allgemeineren Entitäten enthalten

3.43

Instanz

en **instance**

individuelle Entität

Anmerkung 1 zum Begriff: Der Begriff „Instanz“ repräsentiert das gleiche Konzept wie der in ISO/IEC 21838-1:2021 [16], 3.3 definierte Begriff „Partikulare“.

3.44

Schnittstelle

en **interface**

<UML> *Klassifikator*, der eine Deklaration einer Menge von kohärenten öffentlichen *Features* und Verpflichtungen darstellt, die zusammen einen kohärenten Dienst ergeben

Anmerkung 1 zum Begriff: Eine Schnittstelle spezifiziert einen Vertrag; jede *Instanz* eines Klassifikators, der die Schnittstelle umsetzt, muss diesen Vertrag erfüllen. Die Verpflichtungen, die mit einer Schnittstelle verknüpft sind, liegen in Form von *Einschränkungen* (wie Vor- und Nachbedingungen) oder Protokollspezifikationen vor, die den Interaktionen über die Schnittstelle Ordnungsbeschränkungen auferlegen können.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 10.4.3.1.

3.45

Schlüsselwort

en **keyword**

<UML> reserviertes Wort, das ein integraler Bestandteil der UML-Notation ist

Anmerkung 1 zum Begriff: Schlüsselwörter erscheinen üblicherweise als Text-Annotationen im Zusammenhang mit einem grafischen UML-Element oder als Teil einer Textzeile in einem UML-Diagramm.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, Anhang C.

3.46

Abstraktionsniveau

Abstraktionsebene

Angabe des Detailgrads, der außerdem des Interessenbereichs liegt

Anmerkung 1 zum Begriff: Ein Modell auf einer hohen Abstraktionsebene weist einen relativ geringen Detailgrad auf.

3.47

Metaklasse

en **metaclass**

<UML> *Klasse* in einem *Metamodell*

BEISPIEL Die Klasse „interface“ (Schnittstelle) ist eine Klasse im UML-Metamodell und daher eine Metaklasse.

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 6.2.

3.48

Metamodell

en **metamodel**

Modell, das eine Modellierungssprache definiert

BEISPIEL Das UML-Metamodell.

Anmerkung 1 zum Begriff: Ein Modell ist eine *Instanz* eines Metamodells, und ein Metamodell ist eine Instanz eines Meta-Metamodells.

Anmerkung 2 zum Begriff: Übernommen aus dem MDA Guide [19].

3.49

Modell

en **model**

Abstraktion von einigen Aspekten der Realität

[QUELLE: ISO 19109:2015 [20], 4.15]

3.50

Multiplizität (einer Sammlung)

en **multiplicity (of a collection)**

<UML> Spezifikation der gültigen *Kardinalitäten* einer Sammlung

BEISPIEL Eine *Instanz* einer Sammlung mit einer festgelegten Multiplizität von „1..3“ muss mindestens ein Element aufweisen und darf nicht mehr als drei Elemente aufweisen.

Anmerkung 1 zum Begriff: Eine Multiplizität ist eine Definition eines inklusiven Intervalls von nicht negativen Ganzzahlen, das mit einem unteren Grenzwert beginnt und mit einem oberen (möglicherweise unendlichen) Grenzwert endet.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 7.5.3.2 und 7.8.8.1.

3.51

benanntes Element

en **named element**

<UML> Element in einem *Modell*, das einen Namen haben kann

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 7.8.9.

3.52

Namensraum

en **namespace**

<UML> *benanntes Element*, das entweder Eigentümer einer Gruppe von *benannten Elementen* ist, die anhand eines Namens identifiziert werden können, oder diese importiert oder beides

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 7.8.10.1.

3.53

natürliche Sprache

en **natural language**

Sprache, die in einer Gemeinschaft von Menschen aktiv genutzt wird oder wurde und deren Regeln sich hauptsächlich aus ihrer Nutzung ableiten

[QUELLE: ISO 5127:2017 [21], 3.1.5.02, modifiziert — Anmerkung zum Eintrag entfernt.]

3.54

n-äre Assoziation

en **n-ary association**

<UML> *Assoziation* zwischen mehr als zwei Elementen (Member Ends)

Anmerkung 1 zum Begriff: Eine Assoziation mit drei Elementen (Member Ends) wird als „ternäre Assoziation“ bezeichnet.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 11.5.

3.55

Objekt

en **object**

<UML> einzelne Entität mit einem Zustand und *Beziehungen* zu anderen einzelnen Entitäten

Anmerkung 1 zum Begriff: Ein Objekt ist eine *Instanz* einer *Klasse*.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 6.3.1.

3.56

Operation

en **operation**

<UML> *verhaltensbezogenes Feature* einer *Schnittstelle*, eines *Datentyps* oder einer *Klasse*

Anmerkung 1 zum Begriff: Eine Operation kann direkt auf *Instanzen* seiner *Klassifikatoren* angewendet werden. Die Operation legt Name, *Typ*, Parameter und *Einschränkungen* dieser Anwendungen fest.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 9.6.3.1.

3.57

Paket

en **package**

<UML> Element, das verwendet wird, um Elemente zu Gruppen zusammenzufassen und das einen *Namensraum* für die gruppierten Elemente bereitstellt

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 12.4.5.1.

3.58

Paketdiagramm

en **package diagram**

<UML> Strukturschema, das bestimmte Aspekte eines oder mehrerer *Pakets/Pakete* darstellt

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, Anhang A.

3.59

Teil-Ganzes-Beziehung

partitive Relation

en part-whole relation

partitive relation

Beziehung zwischen zwei *Konzepten*, wobei eines der Konzepte das Ganze und das andere Konzept einen Teil dieses Ganzen darstellt

Anmerkung 1 zum Begriff: Eine Teil-Ganzes-Beziehung besteht zwischen „Woche“ und „Tag“ sowie zwischen „Molekül“ und „Atom“.

[QUELLE: ISO 5127:2017 [21], 3.1.7.06, modifiziert — bevorzugten und zulässigen Begriff vertauscht.]

3.60

primitiver Typ

en **primitive type**

<UML> vordefinierter *Datentyp* ohne jedwede Unterstruktur

Anmerkung 1 zum Begriff: Für einen primitiven Typen können algebraische und andere Operationen außerhalb der UML definiert sein, z. B. mathematisch.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 10.2.3.2.

3.61

Profil

en **profile**

<UML> *Paket*, das begrenzte Erweiterungen eines Referenz-*Metamodells* zwecks Anpassung des Metamodells an eine spezifische Plattform oder einen bestimmten Bereich definiert

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 12.4.7.

3.62

Eigenschaft

en **property**

<UML> *Strukturmerkmal*

Anmerkung 1 zum Begriff: Eigentümer einer Eigenschaft ist ein *Klassifikator*, eine *Assoziation* oder eine andere Eigenschaft.

Anmerkung 2 zum Begriff: Im UML-Metamodell ist die Eigenschaft das einzige Strukturmerkmal. UML 2.5.1 legt nicht genau fest, was der Unterschied ist zwischen einer Eigenschaft und einem Strukturmerkmal, das keine Eigenschaft ist.

Anmerkung 3 zum Begriff: Übernommen aus UML 2.5.1, 9.4.3.2, 9.5.3 und 9.9.17.1.

3.63

Realisierung

en **realization**

<UML> *Abstraktion* zwischen zwei *benannten Elementen* oder Gruppen von benannten Elementen, von denen eines eine Spezifikation und das andere dessen Implementierung darstellt

Anmerkung 1 zum Begriff: Der Supplier repräsentiert die Spezifikation, der Client repräsentiert eine Implementierung der Spezifikation.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 7.8.14.1.

3.64

Beziehung

en **relation**

<UML> Verbindung zwischen Elementen

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 7.8.15.1.

3.65

Schema

en **schema**

formale Beschreibung eines *Modells*

[QUELLE: ISO 19101-1:2014 [10], 4.1.34]

3.66

Stereotyp

en **stereotype**

<UML> Modellelement, das eine existierende *Metaklasse* erweitert und die Verwendung einer Plattform oder bereichsspezifischen Terminologie oder Notation anstelle jener oder zusätzlich zu jenen ermöglicht, die für die erweiterte Metaklasse verwendet werden

Anmerkung 1 zum Begriff: Ein Stereotyp ist in einem *Profil* enthalten.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 12.4.9.1.

3.67

Strukturmerkmal

en **structural feature**

<UML> *Feature*, das einen Aspekt der Struktur festlegt

Anmerkung 1 zum Begriff: Ein Strukturmerkmal repräsentiert Werte, die als Teil der Struktur der *Instanzen* des Elements (z. B. eines *Klassifikators*) betrachtet werden, das deren Eigentümer ist.

Anmerkung 2 zum Begriff: Ein Strukturmerkmal ist außerdem eine Art von *typisiertem Element*.

Anmerkung 3 zum Begriff: Übernommen aus UML 2.5.1, 7.5.3.1 und 9.9.21.

3.68

Eigenschaftsdefinition

en **tag definition**

<UML> *Eigenschaft* eines *Stereotyps*

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 12.3.3.4.

3.69

Eigenschaftswert

en **tagged value**

<UML> Wert einer *Eigenschaft* eines *Stereotyps*, angewendet auf ein Modellelement

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 12.3.3.4.

3.70

Schablone

en **template**

<UML> parametrisiertes Modellelement

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 7.3.

3.71

Typ

en **type**

<UML> Modellelement, das eine Gruppe zulässiger Werte festlegt

Anmerkung 1 zum Begriff: Die Gruppe zulässiger Werte ist als die *Instanzen* des Typs bekannt.

Anmerkung 2 zum Begriff: Der Begriff „Typ“ repräsentiert außerdem das in 3.72 definierte Konzept.

Anmerkung 3 zum Begriff: Übernommen aus UML 2.5.1, 7.5.3.1.

3.72

Typ

en **type**

<UML> *Stereotyp* einer *Klasse*, das einen Bereich von *Objekten* zusammen mit den auf die Objekte anwendbaren *Operationen* festlegt, ohne die physische Implementierung dieser Objekte zu definieren

Anmerkung 1 zum Begriff: Ein Typ kann *Attribute* und *Assoziationen* besitzen.

Anmerkung 2 zum Begriff: Hierbei handelt es sich um einen vordefinierten Standard-Stereotyp, der im Standardprofil der UML definiert ist.

Anmerkung 3 zum Begriff: Der Begriff „Typ“ repräsentiert außerdem das in 3.71 definierte Konzept.

Anmerkung 4 zum Begriff: Übernommen aus UML 2.5.1, 22.3.

3.73

typisiertes Element

en **typed element**

benanntes Element, für das ein *Typ* festgelegt werden kann

Anmerkung 1 zum Begriff: Wenn ein typisiertes Element einen zugehörigen Typ besitzt, muss jeder Wert, der durch das typisierte Element (zu jedwedem Zeitpunkt) repräsentiert wird, eine *Instanz* des angegebenen Typs sein. Ein typisiertes Element ohne zugehörigen Typ kann einen beliebigen Wert repräsentieren.

Anmerkung 2 zum Begriff: Übernommen aus UML 2.5.1, 7.8.22.1.

3.74

Gegenstandsbereich

Ausschnitt der realen Welt

en **universe of discourse**

Sicht der realen oder hypothetischen Welt, die die Gesamtheit dessen umfasst, was für eine Anwendung von Interesse ist

[QUELLE: ISO 19101-1:2014 [10], 4.1.38]

3.75

Verwendungsbeziehung

en **usage**

<UML> *Abhängigkeit*, bei der das Client-Element für seine vollständige Implementierung oder Operation das Supplier-Element (oder eine Gruppe von Elementen) benötigt

Anmerkung 1 zum Begriff: Übernommen aus UML 2.5.1, 7.8.23.1.

3.76

Wertebereich

en **value domain**

Gruppe von angenommenen Werten

BEISPIEL Der Bereich 3–28, alle Ganzzahlen, beliebige ASCII-Zeichen, Aufzählung von Werten (grün, blau, weiß).

Anmerkung 1 zum Begriff: Es handelt sich um das gleiche Konzept wie „value space“ in ISO/IEC 11404:2007 [2], 3.61.

Nationaler Anhang NB (informativ)

Symbole und Abkürzungen

CSL	Conceptual Schema Language (Modellierungssprache)
EBNF	Erweiterte Backus-Naur-Form
GML	Geography Markup Language (Auszeichnungssprache für raumbezogene Objekte)
HMMG	Harmonized Model Maintenance Group (Gruppe zur Pflege des harmonisierten Modells)
IANA	Internet Assigned Numbers Authority (Institution zur Zuordnung von Nummern und Namen im Internet, insbesondere IP-Adressen)
IETF	Internet Engineering Task Force (Internettechnik-Arbeitsgruppe)
ITU	International Telecommunication Union (Internationale Fernmeldeunion)
JCGM	Joint Committee for Guides in Metrology (gemeinsames Komitee für die Herausgabe metrologischer Handbücher)
MOF	Meta Object Facility (spezielle Metadaten-Architektur)
OCL	Object Constraint Language (Sprache zur Festlegung notwendiger Randbedingungen)
OMG	Object Management Group (Konsortium zur Entwicklung von Standards für die objektorientierte Programmierung)
OWL	Web Ontology Language (Spezifikation zur Erstellung von Ontologien)
RDF	Resource Description Framework (System zur Beschreibung von Ressourcen)
RFC	Request for Comments
SBVR	Semantics of Business Vocabulary and Business Rules (Semantik von Geschäftsvokabular und Geschäftsregeln)
SFA	Simple Feature Access (Spezifikation für eine allgemein gültige Architektur für geografische Daten)
SKOS	Simple Knowledge Organisation System (einfaches System zur Organisation von Wissen)
SQL	Structured Query Language (strukturierte Datenbanksprache)
UML	Unified Modelling Language (vereinheitlichte Modellierungssprache)
uom	unit of measure (Maßeinheit)
URI	Uniform Resource Identifier (Identifikator für Ressourcen im Internet)
URL	Uniform Resource Locator (einheitlicher Anzeiger für Ressourcen im Internet)
UUID	Universally Unique Identifier (universeller Identifikator für Informationen in Computersystemen)

XML eXtensible Markup Language (erweiterbare Auszeichnungssprache)

Nationaler Anhang NC (informativ)

Literaturhinweise

DIN EN ISO 3166-1, *Codes für die Namen von Ländern und deren Untereinheiten — Teil 1: Codes für Ländernamen*

DIN EN ISO 19101-1:2015-03, *Geoinformation — Referenzmodell — Teil 1: Grundsätze (ISO 19101-1:2014); Englische Fassung EN ISO 19101-1:2014*

DIN EN ISO 19107:2020-09, *Geoinformation — Raumbezugsschema (ISO 19107:2019); Englische Fassung EN ISO 19107:2019*

DIN EN ISO 19108:2005-05, *Geoinformation — Zeitliches Schema (ISO 19108:2002); Englische Fassung EN ISO 19108:2005*

DIN EN ISO 19109:2016-05, *Geoinformation — Regeln zur Erstellung von Anwendungsschemata (ISO 19109:2015); Englische Fassung EN ISO 19109:2015*

DIN EN ISO 19115-1:2014-07, *Geoinformation — Metadaten — Teil 1: Grundsätze (ISO 19115-1:2014); Englische Fassung EN ISO 19115-1:2014*

DIN EN ISO 19118:2012-01, *Geoinformation — Kodierung (ISO 19118:2011); Englische Fassung EN ISO 19118:2011*

DIN EN ISO 19119:2006-09, *Geoinformation — Dienste (ISO 19119:2005); Englische Fassung EN ISO 19119:2006*

DIN EN ISO 19123-1:—^{*}, *Geoinformation — Coverage Geometrie- und Funktionsschema — Teil 1: Grundlagen (ISO 19123-1:2023); Englische Fassung EN ISO 19123-1:2023*

DIN EN ISO 19135-1:2016-02, *Geoinformation — Registrierungsverfahren für geographische Informationseinheiten — Teil 1: Grundsätze (ISO 19135-1:2015); Englische Fassung EN ISO 19135-1:2015*

DIN EN ISO 19136-1, *Geoinformation — Geography Markup Language (GML) — Teil 1: Grundsätze*

DIN EN ISO 19136-2, *Geoinformation — Geography Markup Language (GML) — Teil 2: Erweitertes Schema und Kodierregeln*

DIN EN ISO 80000-3, *Größen und Einheiten — Teil 3: Raum und Zeit*

DIN ISO 2108, *Information und Dokumentation — Internationale Standard-Buchnummer (ISBN)*

DIN ISO 8601-1, *Datum und Uhrzeit — Darstellung für den Informationsaustausch — Teil 1: Grundlegende Regeln*

DIN ISO 10241-1, *Terminologische Einträge in Normen — Teil 1: Allgemeine Anforderungen und Beispiele für ihre Darstellung*

* Zur Zeit Entwurf.

– *Entwurf* –

Contents

Foreword	vi
Introduction.....	vii
1 Scope	1
2 Normative references.....	1
3 Terms and definitions	1
4 Symbols and abbreviated terms	13
5 Conformance	14
5.1 Conformance overview.....	14
5.2 Conceptual schemas modelled in UML	14
5.3 Supplementary naming conventions	14
6 Overview.....	14
7 Use of UML.....	15
7.1 General use of UML	15
7.2 Classifiers	16
7.2.1 General.....	16
7.2.2 Classes.....	18
7.2.3 Data types.....	18
7.2.4 Enumerations	19
7.2.5 Interfaces	20
7.3 Features	21
7.3.1 General.....	21
7.3.2 Properties.....	21
7.3.3 Operations.....	24
7.4 Relationships	24
7.4.1 General.....	24
7.4.2 Associations.....	25
7.4.3 Generalizations	27
7.4.4 Realizations	27
7.4.5 Template bindings	28
7.5 Packages	28
7.6 Notes.....	29
7.7 Constraints.....	29
7.8 UML profile	30
7.9 Basic naming conventions.....	36
7.10 Diagrams.....	39
7.10.1 General.....	39
7.10.2 Package diagrams	40
7.10.3 Class diagrams.....	40

7.11	Reusable types	41
7.11.1	General.....	41
7.11.2	Core data types.....	41
7.11.3	Common types.....	41
8	Supplementary naming conventions	42
9	Core data types	42
9.1	General	42
9.1.1	Relation with ISO/IEC 11404:2007	42
9.1.2	Modelling choice for the core data types.....	44
9.1.3	Categories of core data types	45
9.2	Primitive data types.....	46
9.2.1	Truth data types	46
9.2.2	Numeric data types.....	46
9.2.3	Text data types.....	47
9.2.4	Date and time data types.....	48
9.3	Defined data types	50
9.3.1	CharacterString.....	50
9.3.2	IRI	51
9.3.3	URI.....	51
9.3.4	UUID.....	51
9.3.5	Bit.....	51
9.3.6	Binary.....	52
9.3.7	Digit.....	52
9.3.8	Sign.....	52
9.3.9	Vector	52
9.3.10	Measure.....	53
9.4	Aggregate data types	53
9.4.1	Record.....	53
9.4.2	HomogeneousRecord	54
9.4.3	Array.....	55
9.4.4	OneDimensionalArray.....	55
9.4.5	TwoDimensionalArray.....	56
9.4.6	ThreeDimensionalArray.....	57
Annex A (normative)	Abstract test suite	58
A.1	Conceptual schemas modelled in UML	58
A.2	Supplementary naming conventions	59
Annex B (informative)	Backward compatibility.....	61
Annex C (informative)	On conceptual schema languages	65
Annex D (informative)	UML notation reference.....	67
D.1	Notation for UML elements used for modelling conceptual schemas.....	67

D.1.1	Notation for classifiers	67
D.1.2	Notation for properties.....	68
D.1.3	Notation for relationships	69
D.2	Notation for UML elements used for modelling profiles	71
D.3	Difference between UML keywords and stereotypes	72
Annex E (informative)	Differences between UML 2.5.1 and UML 2.4.1	74
Annex F (informative)	Mapping between ISO 19103 and ISO/IEC 11404 data types.....	76
Annex G (informative)	Conceptual schema representations	79
Annex H (informative)	Code sets	80
H.1	General	80
H.2	Principles.....	80
H.2.1	Coding objectives	80
H.2.2	Coding schemes	82
H.2.3	Governance and maintenance of coding systems	84
H.2.4	Special code values and exhaustiveness of coded sets	84
H.3	Coding system interchange.....	86
H.4	Coding systems and conceptual schemas	87
H.4.1	Conceptual schemas for modelling coding systems	87
H.4.2	Conceptual schemas using coding systems.....	87
Bibliography		89

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2. www.iso.org/directives

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received. www.iso.org/patents

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword - Supplementary information

The committee responsible for this document is ISO/TC 211, *Geographic information/Geomatics*.

This second edition of ISO 19103 cancels and replaces ISO 19103:2015 [1].

This edition includes the following significant changes compared with the previous edition:

- increased conformance to UML 2.5.1:
 - improvement of the UML profile and deprecation of stereotypes Leaf, CodeList and Union;
 - removal of the collection data types;
- increased alignment with the data types described in ISO/IEC 11404:2007 [2];
- removal of the conformance classes for conceptual schemas modelled in UML 1.x and for conceptual schemas modelled in another conceptual schema language;
- update of the normative references, in particular
 - addition of UML 2.5.1 and removal of ISO/IEC 19505-2:2012 [3] (equivalent to UML 2.4.1, Superstructure [4]);
 - removal of the Object Constraint Language (OCL) specification.

Introduction

This document of the ISO geographic information suite of standards is concerned with the adoption and use of a conceptual schema language (CSL) for developing computer interpretable models, or schemas, of geographic information. Standardization of geographic information requires the use of a formal CSL to specify unambiguous schemas that can serve as a basis for data interchange. An important goal of the ISO geographic information suite of standards is to create a framework in which data interchange and service interoperability can be realized across multiple implementation environments. The adoption and consistent use of a CSL to specify geographic information is of fundamental importance in achieving this goal.

There are two aspects to this document. First, a CSL is selected that meets the requirements for rigorous representation of geographic information. This document identifies a subset of the Unified Modeling Language (UML) as the conceptual schema language for the specification of conceptual schemas, and it provides guidelines on how UML should be used to create conceptual schemas that are a basis for achieving the goal of interoperability. In addition, this document defines a set of core data type definitions for use in conceptual schemas.

One goal of the ISO geographic information suite of standards using conceptual schemas specified in UML is that they will provide a basis for model based mapping to encoding schemas like the ones defined in ISO 19118 [5], as well as a basis for creating implementation specifications for implementation profiles for various other environments.

This document describes the general metamodel for use of UML in the context of the ISO geographic information series of standards. Aspects specifically dealing with the modelling of application schemas are described in ISO 19109 [6].

– *Entwurf* –

Geographic information — Conceptual schema language

1 Scope

This document provides rules and guidelines for the use of a conceptual schema language within the context of modelling geographic information. The chosen conceptual schema language is the Unified Modeling Language (UML).

This document defines a UML profile for modelling geographic information.

This document defines a set of core data types for use in conceptual schemas.

The standardization target type of this document is conceptual schemas describing geographic information.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

UML 2.5.1: OBJECT MANAGEMENT GROUP (OMG). Unified Modeling Language (UML) [online]. Version 2.5.1. December 2017. Available from: <https://www.omg.org/spec/UML/2.5.1>

xmlschema-2:; WORLD WIDE WEB CONSORTIUM. XML Schema Part 2: Datatypes [online]. Available from: <http://www.w3.org/TR/xmlschema-2/>

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 11404:2007 [2] and the following apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <https://www.electropedia.org>

NOTE UML 2.5.1 contains neither terminological entries in accordance with ISO 10241-1 [7] nor a glossary — as opposed to the earlier UML 1.x specifications, see e.g. UML 1.5 [8] — hence the need to add selected UML concepts to Clause 3 in this document. The concepts adopted from UML 2.5.1 are defined in terminological entries that

- have “UML” as domain;
- have terms, definitions and notes that are based on the description of the concepts in the main body of UML 2.5.1, and that can have been changed editorially to conform to the principles of terminology;
- can have additional examples and notes to entry relevant for this document;
- have a note to entry indicating the clause or clauses in UML 2.5.1 that describe the concept.

3.1

abstract classifier

⟨UML⟩ *classifier* that has no direct *instances*

Note 1 to entry: Every instance of an abstract classifier shall be an instance of one of its specializations.

Note 2 to entry: Adopted from UML 2.5.1, 9.2.3.2.

3.2

abstract schema

conceptual schema that is not implementable without further specification

EXAMPLE The conceptual schemas for describing the spatial characteristics of geographic entities defined in ISO 19107:2019 [9].

Note 1 to entry: An abstract schema can be applied to many domains.

Note 2 to entry: An abstract schema can be realized by an *application schema*.

3.3

abstraction

⟨information technology⟩ result of an *abstraction process*

Note 1 to entry: In the information technology domain, the term “abstraction” also represents the concept defined in 3.5.

3.4

abstraction

⟨UML⟩ *dependency* that relates two *named elements* or sets of named elements that represent the same *concept* at different *levels of abstraction* or from different viewpoints

Note 1 to entry: Adopted from UML 2.5.1, 7.7.3.3.

3.5

abstraction process

abstraction

⟨information technology⟩ process of filtering out detail that is not within the scope of interest

Note 1 to entry: Abstraction allows understanding something's essence more easily and allows handling complexity.

Note 2 to entry: In the information technology domain, the term “abstraction” also represents the concept defined in 3.3.

3.6

aggregation

shared aggregation

⟨UML⟩ *binary association* that specifies a *part-whole relation* where the whole does not have responsibility for the existence of its parts

Note 1 to entry: A part can be included in more than one whole simultaneously.

Note 2 to entry: Adopted from UML 2.5.1, 9.5.3.

3.7

application

manipulation and processing of data in support of user requirements

[SOURCE: ISO 19101-1:2014 [10], 4.1.1]

3.8

application schema

conceptual schema for data required by one or more *applications*

[SOURCE: ISO 19101-1:2014 [10], 4.1.2]

3.9

association

⟨UML⟩ semantic *relationship* that can occur between *instances* that have a *type*

Note 1 to entry: An association is also a kind of *classifier*.

Note 2 to entry: Adopted from UML 2.5.1, 11.5.3.1.

3.10

attribute

⟨UML⟩ *property* owned by a *classifier* other than an *association*

Note 1 to entry: Adopted from UML 2.5.1, 9.5.3.

3.11

behavioural feature

⟨UML⟩ *feature* that specifies an aspect of behaviour

Note 1 to entry: Adopted from UML 2.5.1, 9.9.2.1.

3.12

binary association

⟨UML⟩ *association* having two member ends

Note 1 to entry: The two member ends can have the same *type*.

Note 2 to entry: Adopted from UML 2.5.1, 11.5.

3.13

cardinality (of a collection)

⟨UML⟩ number of elements in a collection

EXAMPLE The cardinality of a collection having three members is three.

Note 1 to entry: Adopted from UML 2.5.1, 7.5.3.2.

3.14

class

⟨UML⟩ *classifier* of a set of *objects*

Note 1 to entry: Adopted from UML 2.5.1, 11.8.3.1.

3.15

class diagram

⟨UML⟩ structure diagram where the primary symbols in the contents area are either *class* symbols or *interface* symbols, or both

Note 1 to entry: Adopted from UML 2.5.1, Annex A.

3.16

classifier

⟨UML⟩ classification of *instances* according to their *features*

Note 1 to entry: A classifier is a kind of *type*.

Note 2 to entry: Adopted from UML 2.5.1, 9.2.1.

3.17

code set

code element set

code

code list

result of applying a *coding scheme* to all elements of a *coded set*

EXAMPLE All the three-letter representations of airport names.

Note 1 to entry: The admitted term “code” also represents the concepts defined in 3.18, 3.20 and ISO 19118:2011 [11], 4.3.

[SOURCE: ISO/IEC 2382:2015 [12], 2121556, modified — Additional admitted term added, definition adjusted to use the preferred terms of the cross-referenced entries, note 1 to entry converted to an example, notes to entry 2 and 3 removed, note to entry regarding homographs added.]

3.18

code value

code element

code

result of applying a *coding scheme* to an element of a *coded set*

EXAMPLE 1 “CDG” representing Paris Charles-de-Gaulle in the code for three-letter representation of airport names.

EXAMPLE 2 The hexadecimal number 0041 representing “Latin capital letter A” in ISO/IEC 10646 [13].

Note 1 to entry: The admitted term “code” also represents the concepts defined in 3.17, 3.20 and ISO 19118:2011 [11], 4.3.

[SOURCE: ISO/IEC 2382:2015 [12], 2121555, modified — Definition adjusted to use the preferred terms of the cross-referenced entries, note 1 to entry converted to two examples, notes to entry 2 and 3 removed, note to entry regarding homographs added.]

3.19

coded set

set of elements that is mapped onto another set according to a *coding scheme*

EXAMPLE A list of the names of airports that is mapped onto a corresponding set of three-letter abbreviations.

[SOURCE: ISO/IEC 2382:2015 [12], 2121553, modified — Definition adjusted to use the preferred term of the cross-referenced entry, note 1 to entry converted to an example, notes to entry 2 and 3 removed.]

3.20

coding scheme

code

collection of rules that maps the elements of a first set onto the elements of a second set

Note 1 to entry: The elements of either set can be characters or character strings.

Note 2 to entry: The first set is called *coded set* and the second set is called *code set*.

Note 3 to entry: Each element of the code set can be related to more than one element of the coded set but the reverse is not true.

Note 4 to entry: The admitted term “code” also represents the concepts defined in 3.17, 3.18 and ISO 19118:2011 [11], 4.3.

[SOURCE: ISO/IEC 2382:2015 [12], 2121552, modified — Preferred and admitted term swapped, replaced “may” with “can” in notes to entry 1 and 2, notes to entry 4 and 5 removed, note to entry regarding homographs added.]

3.21

coding system

combination of a *coded set* and a *code set*, based on a *coding scheme*

[SOURCE: EN 1068:2005 [14], 3.8, modified — Definition adjusted to use the preferred terms of the cross-referenced entries.]

3.22

composition

⟨UML⟩ *binary association* that specifies a *part-whole relation* where the whole has responsibility for the existence of its parts

Note 1 to entry: A part can only be included in at most one whole at a time.

Note 2 to entry: Adopted from UML 2.5.1, 9.5.3.

3.23

concept

unit of knowledge created by a unique combination of characteristics

Note 1 to entry: Concepts are not necessarily bound to particular *natural languages*. They are, however, influenced by the social or cultural background which often leads to different categorizations.

Note 2 to entry: This is the concept ‘concept’ as used and designated by the term “concept” in terminology work. It is a very different concept from that designated by other domains such as industrial automation or marketing.

[SOURCE: ISO 1087:2019 [15], 3.2.7]

3.24

conceptual formalism

set of modelling *concepts* used to describe a *conceptual model*

EXAMPLE object-oriented modelling

Note 1 to entry: One conceptual formalism can be expressed in several *conceptual schema languages*.

[SOURCE: ISO 19101-1:2014 [10], 4.1.4, modified — Examples replaced.]

3.25

conceptual model

model that defines *concepts* of a *universe of discourse*

[SOURCE: ISO 19101-1:2014 [10], 4.1.5]

3.26

conceptual schema

formal description of a *conceptual model*

[SOURCE: ISO 19101-1:2014 [10], 4.1.6]

3.27

conceptual schema language

formal language based on a *conceptual formalism* for the purpose of representing *conceptual schemas*

EXAMPLE UML, EXPRESS, IDEF1X.

Note 1 to entry: A conceptual schema language can be lexical or graphical. Several conceptual schema languages can be based on the same conceptual formalism.

[SOURCE: ISO 19101-1:2014 [10], 4.1.7, modified — Note to entry changed: “can” instead of “may”.]

3.28

constraint

⟨UML⟩ condition or restriction expressed in *natural language* text or in a machine readable language for the purpose of declaring some of the semantics of an element or set of elements

Note 1 to entry: Adopted from UML 2.5.1, 7.8.3.1.

3.29

data type

set of distinct values, characterized by properties of those values, and by operations on those values

EXAMPLE The data type “Boolean” with properties “unordered”, “exact”, and “non-numeric” and with operations “equal”, “not”, “and” and “or”.

Note 1 to entry: Properties of data type values are ordered or unordered, exact or approximate, numeric or non-numeric and, if ordered, bounded or unbounded, as described in ISO/IEC 11404:2007 [2].

[SOURCE: ISO/IEC 11404:2007 [2], 3.12, modified — Note to entry and example added.]

3.30

data type

⟨UML⟩ *classifier* whose *instances* are distinguished only by their value

Note 1 to entry: Adopted from UML 2.5.1, 10.2.1.

3.31

data value

⟨UML⟩ *instance* of a *data type*

Note 1 to entry: Adopted from UML 2.5.1, 7.5.3.2.

3.32

dependency

⟨UML⟩ *directed relationship* which signifies that a single model element or a set of model elements requires other model elements for their specification or implementation

Note 1 to entry: A dependency signifies a supplier/client relationship between model elements where the modification of a supplier may impact the client model elements. The complete semantics of the client element(s) are either semantically or structurally dependent on the definition of the supplier element(s).

Note 2 to entry: Adopted from UML 2.5.1, 7.7.1 and 7.8.4.

3.33

directed relationship

⟨UML⟩ *relationship* between a collection of source model elements and a collection of target model elements

Note 1 to entry: Adopted from UML 2.5.1, 7.8.5.1.

3.34

enumeration

⟨UML⟩ *data type* whose values are named one by one in the *model*

Note 1 to entry: The set of *enumeration literals* of an *enumeration* is ordered.

Note 2 to entry: Adopted from UML 2.5.1, 10.5.3

3.35

enumeration literal

⟨UML⟩ user-defined *data value* for an *enumeration*

Note 1 to entry: Adopted from UML 2.5.1, 10.5.4.1.

3.36

extension

⟨UML⟩ *association* which indicates that the *properties* of a *metaclass* are extended through a *stereotype*

Note 1 to entry: Adopted from UML 2.5.1, 12.4.1.1.

3.37

feature

⟨UML⟩ characteristic

Note 1 to entry: Adopted from UML 2.5.1, 9.4.3.1.

3.38

formal language

language that is machine readable and has well-defined semantics

Note 1 to entry: Well-defined semantics will typically be model-theoretic semantics.

[SOURCE: ISO/IEC 21838-1:2021 [16], 3.10]

3.39

generalization

⟨UML⟩ taxonomic *directed relationship* between a more general *classifier* and a more specific classifier

Note 1 to entry: The more general classifier is called the parent, or the superclass if the classifier is a *class*. The more specific classifier is called the child. The classifiers that can be reached by following the generalizations from a given classifier in the direction towards the more general classifiers are called the classifier's generalizations. The classifiers that can be reached by following the generalizations from a given classifier in the direction towards the more specific classifiers are called the classifier's specializations.

Note 2 to entry: Each *instance* of the specific classifier is also an instance of the general classifier. The specific classifier inherits the *features* of the more general classifier.

Note 3 to entry: Adopted from UML 2.5.1, 9.2.3.2 and 9.9.7.

3.40

identifier

linguistically independent sequence of characters capable of uniquely and permanently identifying that with which it is associated

[SOURCE: ISO 19135-1:2015 [17], 4.1.5]

3.41

identity

inherent characteristic of an *instance* that distinguishes it from all other instances

[SOURCE: ISO/IEC/IEEE 24765:2017 [18], 3.1865, modified — Definition changed (“characteristic” instead of “property”), note to entry removed.]

3.42

inheritance

⟨information technology⟩ mechanism by which more specific entities incorporate structure and behaviour defined by more general entities

3.43

instance

individual entity

Note 1 to entry: The term “instance” represents the same concept as the term “particular” defined in ISO/IEC 21838-1:2021 [16], 3.3.

3.44

interface

⟨UML⟩ *classifier* that represents a declaration of a set of coherent public *features* and obligations that together constitute a coherent service

Note 1 to entry: An interface specifies a contract; any *instance* of a classifier that realizes the interface must fulfil that contract. The obligations associated with an interface are in the form of *constraints* (such as pre- and postconditions) or protocol specifications, which can impose ordering restrictions on interactions through the interface.

Note 2 to entry: Adopted from UML 2.5.1, 10.4.3.1.

3.45

keyword

⟨UML⟩ reserved word that is an integral part of the UML notation

Note 1 to entry: Keywords normally appear as text annotations attached to a UML graphic element or as part of a text line in a UML diagram.

Note 2 to entry: Adopted from UML 2.5.1, Annex C.

3.46

level of abstraction

abstraction level

indication of the amount of detail that is outside the scope of interest

Note 1 to entry: A model at a high level of abstraction has a relatively low amount of details.

3.47

metaclass

⟨UML⟩ *class* in a *metamodel*

EXAMPLE The class “Interface” is a class in the UML metamodel, and is therefore a metaclass.

Note 1 to entry: Adopted from UML 2.5.1, 6.2.

3.48

metamodel

model that defines a modelling language

EXAMPLE The UML metamodel.

Note 1 to entry: A model is an *instance* of a metamodel, and a metamodel is an instance of a meta-metamodel.

Note 2 to entry: Adopted from the MDA Guide [19].

3.49

model

abstraction of some aspects of reality

[SOURCE: ISO 19109:2015 [20], 4.15]

3.50

multiplicity (of a collection)

⟨UML⟩ specification of the valid *cardinalities* of a collection

EXAMPLE An *instance* of a collection specified as having a multiplicity of “1..3” must have at least one member and must not have more than three members.

Note 1 to entry: A multiplicity is a definition of an inclusive interval of non-negative integers beginning with a lower bound and ending with a (possibly infinite) upper bound.

Note 2 to entry: Adopted from UML 2.5.1, 7.5.3.2 and 7.8.8.1.

3.51

named element

⟨UML⟩ element in a *model* that can have a name

Note 1 to entry: Adopted from UML 2.5.1, 7.8.9.

3.52

namespace

⟨UML⟩ *named element* that either owns or imports, or both, a set of *named elements* that can be identified by a name

Note 1 to entry: Adopted from UML 2.5.1, 7.8.10.1.

3.53

natural language

language which is or was in active use in a community of people, and the rules of which are mainly deduced from usage

[SOURCE: ISO 5127:2017 [21], 3.1.5.02, modified — Note to entry removed.]

3.54

n-ary association

⟨UML⟩ *association* having more than two member ends

Note 1 to entry: An association with three members ends is called a ternary association.

Note 2 to entry: Adopted from UML 2.5.1., 11.5.

3.55

object

⟨UML⟩ individual with a state and *relationships* to other individuals

Note 1 to entry: An object is an *instance* of a *class*.

Note 2 to entry: Adopted from UML 2.5.1, 6.3.1.

3.56

operation

⟨UML⟩ *behavioural feature* of an *interface*, *data type* or *class*

Note 1 to entry: An operation can be directly invoked on *instances* of its featuring *classifiers*. The operation specifies the name, *type*, parameters, and *constraints* for such invocations.

Note 2 to entry: Adopted UML 2.5.1, 9.6.3.1.

3.57

package

⟨UML⟩ element that is used to group elements, and provides a *namespace* for the grouped elements

Note 1 to entry: Adopted from UML 2.5.1, 12.4.5.1.

3.58

package diagram

⟨UML⟩ structure diagram that shows certain aspects of one or more *packages*

Note 1 to entry: Adopted from UML 2.5.1, Annex A

3.59

part-whole relation

partitive relation

relation between two *concepts* where one of the concepts constitutes the whole and the other concept a part of that whole

Note 1 to entry: A part-whole relation exists between “week” and “day”, “molecule” and “atom”.

[SOURCE: ISO 5127:2017 [21], 3.1.7.06, modified — Preferred and admitted term swapped.]

3.60

primitive type

⟨UML⟩ predefined *data type* without any substructure

Note 1 to entry: A primitive type can have algebra and operations defined outside of UML, for example, mathematically.

Note 2 to entry: Adopted from UML 2.5.1, 10.2.3.2.

3.61

profile

⟨UML⟩ *package* that defines limited extensions to a reference *metamodel* with the purpose of adapting the metamodel to a specific platform or domain

Note 1 to entry: Adopted from UML 2.5.1, 12.4.7.