**ISO 191##-#:####(X)**

ISO TC 211 /SC /WG 7

Secretariat: SIS

**Addressing** — **Machine interchange syntax**

WD/CD/DIS/FDIS stage

**Warning for WDs and CDs**

This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

*To help you, this guide on writing standards was produced by the ISO/TMB and is available at* [*https://www.iso.org/iso/how-to-write-standards.pdf*](https://www.iso.org/iso/how-to-write-standards.pdf)

*A model manuscript of a draft International Standard (known as “The Rice Model”) is available at* [*https://www.iso.org/iso/model\_document-rice\_model.pdf*](https://www.iso.org/iso/model_document-rice_model.pdf)

© ISO 2017, Published in Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO’s member body in the country of the requester.

ISO copyright office

Ch. de Blandonnet 8 • CP 401

CH-1214 Vernier, Geneva, Switzerland

Tel.  + 41 22 749 01 11

Fax  + 41 22 749 09 47

copyright@iso.org

www.iso.org

Contents

[Foreword iv](#_Toc485815077)

[Introduction v](#_Toc485815078)

[1 Scope *(mandatory)* 1](#_Toc485815079)

[2 Normative references *(mandatory)* 1](#_Toc485815080)

[3 Terms and definitions *(mandatory)* 1](#_Toc485815081)

[4 Clause title autonumber 2](#_Toc485815082)

[5 Clause title 2](#_Toc485815083)

[5.1 Subclause autonumber 2](#_Toc485815084)

[5.1.1 Subclause autonumber 2](#_Toc485815085)

[6 Clause title 2](#_Toc485815086)

[Annex A (informative) Annex title e.g. Example of a figure and a table 3](#_Toc485815087)

[Bibliography 5](#_Toc485815088)

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 (see [www.iso.org/directives](https://www.iso.org/directives-and-policies.html)).

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 (see [www.iso.org/patents](https://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 voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: [www.iso.org/iso/foreword.html](https://www.iso.org/iso/foreword.html).

This document was prepared by Technical Committee ISO/TC 211, Geographic information / Geomantics, *Workgroup 7*.

Introduction

Addresses are among the most commonly exchanged information on the Internet, and the interchange of them is crucial to a number of Internet applications, such as electronic commerce, contact exchange, non-postal deliveries, as well as location scheduling.

Yet, addresses mean much more than just geolocation information:

* As an identity, such as an office address
* As reference points (waypoints) in routing information
* As a delivery point

The lifecycle of an address entry exchanged on the Internet typically starts with manual input of a human actor. This data, structured or unstructured, is then submitted to an Internet-connected application, and the application may in turn transmit this information to other applications or external parties on behalf of the user who provided the address. This transmission is usually performed to fulfil service delivery to the user. Within the process, there may be machine-human interactions that require display of the address in human-readable form, as well as machine-to-machine interactions on the address, such as for data validation.

Additional caution must be placed on the accuracy (or lack thereof) of human input addresses. While an address specified by a human actor may unambiguously distinguish a location, there may be intention or unintentional omissions or additions to an “official” address (if there was one).

This International Standard facilitates the interchange and interaction of addresses between humans and applications, and between applications, through the specification of a defined syntax for applications to interact and interchange international addresses, while supporting human input and human-machine interaction, without loss of fidelity.

Explain how this relates to the other parts of ISO 19160

Currently, profiles of ISO 19160-1 are published as documents (e.g. MS Word or PDF). This International Standard specifies a machine-readable encoding for such a human-readable document and for addresses conforming to such a profile. This International Standard can be used for maintaining a register of profiles.

**Addressing** — **Machine interchange syntax**

# Scope

This document specifies a machine-readable encoding for the digital storage and transmission of

* description of an address profiles conforming to ISO 19160-1, *Addressing – Part 1: Conceptual model*;
* address instances that conform to a specific profile of ISO 19160-1; and
* a template for entering and displaying addresses conforming to a profile of ISO 19160-1.

# Conformance

## General

This part of ISO 19160 defines four classes of requirements and conformance. Annex A specifies how conformance with these classes shall be tested.

## ProfileDescription

Any machine-readable description of a profile conforming to ISO 19160-1 which conformance is claimed shall pass all the requirements described in the abstract test suite in A.2.

## AddressInstance

Text…

## AddressEnteringTemplate

Text…

## AddressDisplayTemplate

Text…

# 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.

ISO 19160-1, *Addressing — Part 1: Conceptual Model*

ISO 19103:2015, *Geographic information – Conceptual schema language*

ISO 639–1, *Codes for the representation of names of languages – Part 1: Alpha-2 code*

ISO 3166–1, *Codes for the representation of names of countries and their subdivisions – Part 1: Country code*

ISO 15924, *Information and documentation -- Codes for the representation of names of scripts*

# Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 19160-1 and the following apply.

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

— IEC Electropedia: available at <http://www.electropedia.org/>

— ISO Online browsing platform: available at <https://www.iso.org/obp>

3.1

address

structured information that allows the unambiguous determination of an object for purposes of identification and location

[SOURCE: ISO 19160-1]

3.2

addressable object

object that may be assigned an *address* (3.1)

[SOURCE: ISO 19160-1]

3.3

address class

description of a set of addresses that share the same address components, operations, methods, relationships, and semantics

[SOURCE: ISO 19160-1]

3.4

address component

constituent part of an address (3.1)

[SOURCE: ISO 19160-1]

3.5

address reference system

defined set of address components and the rules for their combination into addresses (3.1)

[SOURCE: ISO 19160-1]

3.6

lineage

provenance (3.12), source(s) and production process(es) used in producing a resource

[SOURCE: ISO 19115-1:2014, 4.9]

3.7

locale

definition of the subset of a user’s environment that depends on language and cultural conventions

[SOURCE: ISO/IEC/IEEE 9945:2009, 3.211, modified - The notes given in ISO/IEC/IEEE 9945:2009 for this entry have been omitted. The note to entry has been added.]

3.8

parent address

*address* (3.1) of a *parent addressable object* (3.9)

[SOURCE: ISO 19160-1]

3.9

parent addressable object

*addressable object* (4.2) that fully encloses one or more other addressable objects

[SOURCE: ISO 19160-1]

3.10

profile

set of one or more base standards or subsets of base standards, and, where applicable, the identification of chosen clauses, classes, options and parameters of those base standards, that are necessary for accomplishing a particular function

[SOURCE: ISO 19106:2004, 4.5]

3.11

address instance

an *address* (3.1) that is set within the context of an *address class* (3.3)

**3.12  
provenance**organization or individual that created, accumulated, maintained and used records

3.13

value type

a data type definition for storing values of a certain type, used in the definition of an *address component* (3.4) within an *address class* (3.3)

3.14

primitive value type

a *value type* (3.13) that is provided by ISO 19103:2014

3.15

composed value type

a *value type* (3.13) that is custom defined in an address profile (3.1) through the composure of other *value types* (3.13) and provided constraints.

3.15

value type definition

the definition of a *composed* *value type* (3.15)

# Address Interchange Lifecycle

## Address Creation

This section describes how an address will be provided.

Typically, a user inputs an address manually.

Conforming applications should not expect general users to be able to input an address with a fully specified structure.

## Address Publication

After a user inputs an address into a structured address form, the user submits this address to the designated recipient, which could be an e-retailer, an electronic business card, or a calendar event.

## Address Usage

The service the receives the address submission will then process the address accordingly. It may opt to further clean up the address, such as by providing a better structure, or adding extra information such as delivery instructions or routes.

The service may wish to confirm with the user the resulting address, such as in the case where the address is used for shipping information.

If the user already has a verified, structured address, then the service could save the effort for verifying the address.

## Address Disposal

When a service no longer needs the address, it should be disposed and the user notified.

# Model

## General Structure

This document specifies the “Address Interchange Object Model” (“AXO”) with following components.

## ProfileDescription

The Profile is the top-level object and corresponds to an *address profile*.

A ProfileDescription contains:

— Attributes: ID, Type, Description

— One or more AddressClass

— Zero or more ValueType definitions

— One or more AddressComponent definitions

— Provenance Information: Issuer, Signature, Validity From/To

— Locale/Script Information: ISO 639-1, ISO 15924

A Profile can be specified in the following object structure (in JSON)

profile = {

id: "http://www.iso.org/tc211/tc211-sample.adp",

type: "iso-19160-address-profile",

publisher: "http://www.iso.org/tc211/",

signature: "...",

name: "TC 211 Minimal Address Profile",

locale: {

language: "en",

script: "en",

},

valueTypes: {

addressedObjectIdentifier: {

primitiveType: Integer,

}

}

addressComponents: { ... }

addressClasses: { ... }

}

## AddressClass

The AddressClass corresponds to the “addressClass” specified in ISO 19160-1.

An AddressClassDescription contains:

— One or more AddressComponent

— One ShowTemplate

— One FormTemplate

addressClasses: {

streetAddress: {

description: Street Address,

availableFields: [ (components) ... ]

showTemplates: [ ... ]

}

}

## Value Types

Primitive value types are defined in ISO 19103:2014, including:

— CharacterString

— DateTime, Date, Time

— Number, Integer, Decimal, Real

— Vector

— Boolean

Composed value types are defined within the profile that utilizes them. A composed value type can be made up of multiple primitive value types, and constraints can be set on them. For example, the maximum number of Post Office Box numbers.

Constraints that can be set depend on the underlying primitive type. Number types support “maxValue” and “minValue”. CharacterString’s support “maxLength” and “minLength”. As a constraint, it is also possible to provide a fixed list of accepted values to this type, such as the number of states in a country or districts in a city.

valueTypes: {

addressNumberValue: {

primitiveType: Integer,

maxValue: 10000,

minValue: 1,

},

boxNumberValue: {

primitiveType: Integer,

maxValue: 100000,

minValue: 1,

}

}

## AddressComponents

AddressComponents correspond to the addressComponent defined in ISO 19160-1.

They can be specified as follows inside a profile.

addressComponents: {

addressNumber: {

valueType: addressNumberValue,

},

boxNumber: {

valueType: boxNumberValue,

},

thoroughfareName: {

valueType: thoroughfareNameValue,

},

localityName: {

valueType: CharacterString,

},

postOfficeName: {

valueType: CharacterString

},

postCode: {

valueType: CharacterString

},

countryName: {

valueType: thoroughfareName,

}

addressNumber: {

valueType: addressedObjectIdentifier,

},

}

## Show Templates

This section should be filled in.

A show template indicates how an address instance belonging to an addressClass should be shown to a user. This will be based on a subset of PADTL and may contain further extensions.

This may be performed using SVG for its layout capabilities.

The resulting language should fulfil these criteria:

— Simple to use and easy to understand

— Not verbose like XML

— Allows flexible formatting, such as hiding certain lines when there is not enough space

— Can be specified by the profile authority, as well as by the address owner (“my address should be shown this way”).

Each ShowTemplate should also provide an example for display purposes (e.g., Help section).

ShowTemplates should support RTL languages as well as horizontal and vertical layouts.

## Form Templates

This section shows how an Input Form can be rendered according to the addressClass requirements. It should be assumed that users will not be often able enter an address thoroughly in the correct, detailed structure. Therefore, the form should be easy to understand and take into account that easy addressClass switching is important.

An application that adheres to this document may find that it is easiest for the user to first select the appropriate addressClass, then allowing users to (or automatically) add necessary components as they type.

A sample input should be provided for illustration purposes.

FormTemplates should support RTL languages as well as horizontal and vertical layouts.

This section has to be completed.

# Address Instances

## General

An address instance is the having an address put into the form that fits an address class.

Here are two general examples of them.

addressInstance1 = {

profile: "http://www.iso.org/tc211/tc211-sample.adp#streetAddress",

components: [

{

type: addressNumber,

value: 99

},

{

type: thoroughfareName,

value: {

name: Lombardy,

type: Street

}

},

{

type: placeName,

value: The Hills,

},

{

type: postCode,

value: 0039,

},

{

type: countryName,

value: South Africa

}

]

}

boxInstance1 = {

profile: "http://www.iso.org/tc211/tc211-sample.adp#boxAddress",

components: [

{

type: boxNumber,

value: 345

},

{

type: postOfficeName,

value: Orlando,

},

{

type: postCode,

value: 2020

},

{

type: countryName,

value: South Africa

}

]

}

## Structural Status

When an address is first manually entered by a common person, it is unrealistic to expect this person to fully adhere to the defined structure.

There are three states of a structured address after being input:

— Unstructured. The user has entered free-form text with no regard of structuring them.

— Partially structured. Country, city and postcodes may be listed correctly, but the other components may be listed in the same row due to previous habits.

— Fully structured. All components are accurately separated.

Therefore, an organization who accepts these addresses may wish to re-structure them or fill in any missing address components.

The status of the address can be specified within an address instance, which can be signed by the structurer as provenance.

## Accuracy and Verification

An authority, such as the local post office, could “verify” a structured address that it is confirmed that this address instance is a “deliverable address”.

The owner of the address, such as the tenant of an office, could provide its signed, structured address on an electronic business card. This allows the recipient of the business card to know whether the senders address is authentic. If this address is verified to be a “deliverable address”, the recipient will know that items sent to this address will very likely be deliverable.

Extra steps need to be taken here to allow this.

## Address As Identity

Addresses do not only specify a location, in some cases they are part of the identity. For example, in business cards, an address can mean more than just an address, such as with vanity value.

This document must support this functionality for it to be useful in contact exchange.

*// TODO: This is a comment  
// As an example, the following address while complete, is more difficult to read / remember compared with the intentionally incomplete one below.*

Suites 1107-1111,

Floor 11,

Central Building,

1-3 Pedder Street,

Central,

Central & Western District,

Hong Kong Island,

Hong Kong

*Preferred address, as equally understandable (and deliverable):*

Suite 1111,

1 Pedder Street,

Central,

Hong Kong

*// TODO: This ends the comment*

## Address As Destination

In certain cases, an address is expected to be reachable either by person and/or post.

An address instance should support being used in conjunction with routing information, acting as a waypoint, and/or supporting a source-defined route.

For example, written instructions on how to deliver to a place that is unambiguous but terribly difficult to locate.

*E.g., some buildings have split floors -– rooms may have the same floor identifier, but is actually inaccessible from the same floor.*

1. (informative)  
     
   Examples
   1. Example of address profiles defined in ISO 19160-1
      1. ISO 19160-1 C2

profile = {

id: "http://www.iso.org/tc211/tc211-minimal.adp",

type: "iso-19160-address-profile",

publisher: "http://www.iso.org/tc211/",

signature: "...",

name: "TC 211 Minimal Address Profile",

locale: {

language: "en",

script: "en",

},

addressComponents: {

addressLine: {

valueType: CharacterString,

minCardinality: 1,

maxCardinality: n

},

},

addressClasses: {

minimalAddress: {

availableFields: [

{

componentType: addressLine,

min: 1,

max: n,

description: "One line of this address",

require: true,

}

],

showTemplates: [

{

/\* TODO \*/

orientation: horizontal,

text: "({{ addressLine }}\n)\*"

}

}

]

}

}

}

----

----

addressInstance1 = {

profile: "http://www.iso.org/tc211/tc211-minimal.adp",

components: [

{

type: addressLine,

value: 14 Church Street,

},

{

type: addressLine,

value: Hatfield

},

{

type: addressLine,

value: South Africa

}

]

}

addressInstance2 = {

profile: "http://www.iso.org/tc211/tc211-minimal.adp",

components: [

{

type: addressLine,

value: Statue of Liberty

},

{

type: addressLine,

value: Liberty Island

},

{

type: addressLine,

value: New York

},

{

type: addressLine,

value: NY

}

]

}

* + 1. ISO 19160-1 C3

----

profile = {

id: "http://www.iso.org/tc211/tc211-sample.adp",

type: "iso-19160-address-profile",

publisher: "http://www.iso.org/tc211/",

signature: "...",

name: "TC 211 Minimal Address Profile",

locale: {

language: "en",

script: "en",

},

valueTypes: {

addressNumberValue: {

primitiveType: Integer,

maxValue: 10000,

minValue: 1,

},

boxNumberValue: {

primitiveType: Integer,

maxValue: 100000,

minValue: 1,

}

}

addressComponents: {

addressNumber: {

valueType: addressNumberValue,

},

boxNumber: {

valueType: boxNumberValue,

},

/\* Table C.3. Address component type \*/

thoroughfareName: {

valueType: thoroughfareNameValue,

},

localityName: {

valueType: CharacterString,

},

postOfficeName: {

valueType: CharacterString

},

postCode: {

valueType: CharacterString

},

countryName: {

valueType: thoroughfareName,

}

addressNumber: {

valueType: addressedObjectIdentifier,

},

},

addressClasses: {

streetAddress: {

description: Street Address,

availableFields: [

{

componentType: addressNumber,

minCardinality: 1,

maxCardinality: 1,

required: true,

},

{

componentType: thoroughfareName,

minCardinality: 1,

maxCardinality: 1,

required: true,

},

{

componentType: placeName

valueType: CharacterString,

minCardinality: 1,

maxCardinality: 1,

required: true,

},

{

componentType: postCode

minCardinality: 1,

maxCardinality: 1,

required: true,

},

{

componentType: countryName

minCardinality: 1,

maxCardinality: 1,

required: false,

},

],

showTemplates: [

{

/\* TODO \*/

}

]

},

boxAddress: {

availableFields: [

{

componentType: boxNumber,

minCardinality: 1,

maxCardinality: 1,

required: true,

},

{

componentType: postOfficeName,

valueType: CharacterString,

minCardinality: 1,

maxCardinality: 1,

required: true,

},

{

componentType: postCode

minCardinality: 1,

maxCardinality: 1,

required: true,

},

{

componentType: countryName

minCardinality: 1,

maxCardinality: 1,

required: false,

},

],

showTemplates: [

{

/\* TODO \*/

}

]

}

}

}

----

----

addressInstance1 = {

profile: "http://www.iso.org/tc211/tc211-sample.adp#streetAddress",

components: [

{

type: addressNumber,

value: 99

},

{

type: thoroughfareName,

value: {

name: Lombardy,

type: Street

}

},

{

type: placeName,

value: The Hills,

},

{

type: postCode,

value: 0039,

},

{

type: countryName,

value: South Africa

}

]

}

boxInstance1 = {

profile: "http://www.iso.org/tc211/tc211-sample.adp#boxAddress",

components: [

{

type: boxNumber,

value: 345

},

{

type: postOfficeName,

value: Orlando,

},

{

type: postCode,

value: 2020

},

{

type: countryName,

value: South Africa

}

]

}

Bibliography

[1] ISO #####‑#, *General title — Part #: Title of part*

[2] ISO #####‑##:20##, *General title — Part ##: Title of part*