INTERNATIONAL STANDARD

ISO 19155-2

First edition
2017-08

Geographic information — Place Identifier (PI) architecture —

Part 2:
Place Identifier (PI) linking

Information géographique — Architecture d'identifiants de lieu (IL) —

Partie 2: Liaison d'identifiants de lieu (IL)

Reference number
ISO 19155-2:2017(E)

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

Page

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).

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).

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.

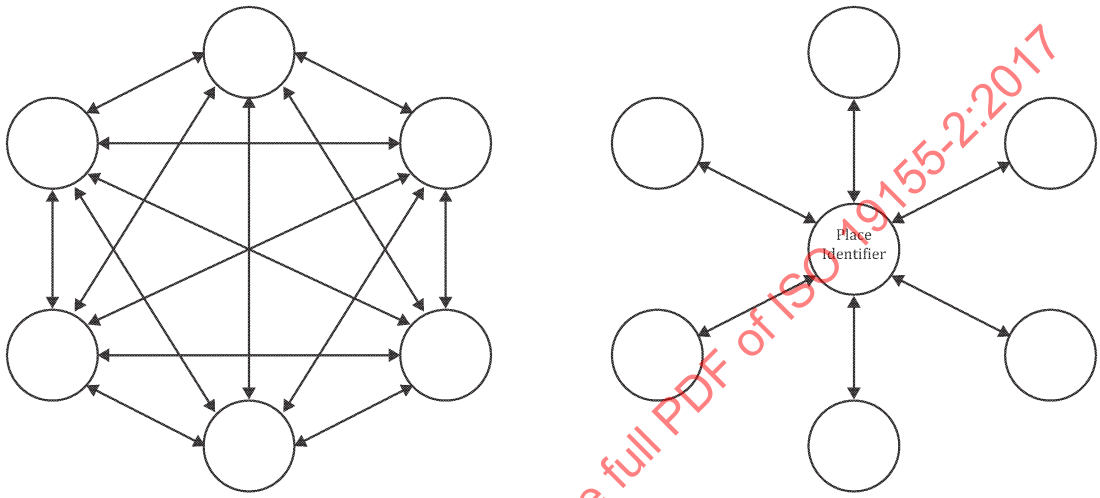
This document was prepared by Technical Committee ISO/TC 211, *Geographic information/Geomatics*.

A list of all parts in the ISO 19155 series can be found on the ISO website.

Introduction

The Place Identifier (PI) architecture (ISO 19155) defined the conceptual model of a place and specified normative encodings, for Place Identifiers, not specific to any type of geographic feature. In this document, three mechanisms are presented that define how Place Identifiers can be linked with features or objects in other encodings. Even though the identifiers of those features or objects may not specifically be a place, they may be referred to conceptually as “other identifiers.”

Figure 1 depicts the abstractions of linking mechanisms among feature/object encoding rules.



a) Without the use of a Place Identifier **b) With the use of a Place Identifier**

Key

encoding rules

linkages

Figure 1 — Linkages with other encoding rules

The linking mechanisms presented in this document are based on accepted information technology for object identification and reference using gml:id, UUID, or URL. By using these linking mechanisms with the rules defined here, and according to the type of encoding rule being linked to, Place Identifiers can more uniformly be related to features and objects — other identifiers — in other encodings. This extends the functionality of those other identifiers, in different encodings, by linking with Place Identifiers encoded in GML.

Existing PI data can complement a range of other encodings through the use of the linking mechanisms defined in this document.

For example, a group of Place Identifiers representing stores in a shopping mall can be associated with the specific locations inside the shopping mall represented by a GML data set.

Geographic information — Place Identifier (PI) architecture —

Part 2: Place Identifier (PI) linking

1 Scope

This document defines the following three mechanisms for linking Place Identifiers (PIs) (see ISO 19155) to features or objects existing in other encodings:

— Id attribute of a GML object (gml:id) as defined in ISO 19136;

— Universally Unique Identifier (UUID) as defined in IETF RFC 4122;

— Uniform Resource Locator (URL) as defined in IETF RFC 1738.

These PI linking mechanisms are enabled using xlink:href as defined in W3C XML Linking Language (XLink).

While the identifiers of these features or objects can sometimes identify a place, within the scope of this document, the identifiers of features or objects existing in other encoding domains are referred to conceptually as other identifiers.

This document further defines that when PIs are encoded, as specified in ISO 19155, using the Geography Markup Language (GML) (ISO 19136), they are linked using gml:id to other GML encoded features. The details of encoding GML instances using gml:id are specified in a normative annex.

Additional normative annexes define encodings for linking Place Identifiers to other identifiers using UUID and URL and present examples for their use.

This document is applicable to location-based services, linked open data, robotic assisted services and other application domains that require a relationship between PIs and objects in either the real or virtual world.

This document is not about creating a registry of Place Identifiers linked to specific features or objects, and support of linking mechanisms other than gml:id, UUID, and URL is out of the scope of this document.

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.

ISO 19103, *Geographic information — Conceptual schema language*

ISO 19136:2007, *Geographic information — Geography Markup Language*

ISO 19155:2012, *Geographic information — Place Identifier (PI) architecture*

IETF, *Universally Unique IDentifier (UUID) URN Namespace*, RFC 4122, July 2005

IETF, *Uniform Resource Locators (URL)*, RFC 1738, December 1994

W3C XML Linking Language (XLink) Version 1.1 — Recommendation, 06 May 2010

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 19155 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 http://www.iso.org/obp

3.1
Place Identifier (PI) link
relationship established between PIs and other identifiers in different encoding domains

Note 1 to entry: While the identifiers of these features or objects can sometimes identify a place, within the scope of this document, the identifiers of features or objects existing in other encoding domains are referred to conceptually as “other identifiers”.

Note 2 to entry: These “other identifiers” can exist outside of the PI architecture.

3.2
Place Identifier (PI) linking mechanism
means used to define a *place identifier (PI) link* (3.1)

4 Abbreviated terms

4.1 Abbreviated terms

BIM	Building Information Model
CSV	comma-separated values
IFC	Industry Foundation Class
GUID	Globally Unique IDentifier
OGC	Open Geospatial Consortium
PI	Place Identifier
RDF	Resource Description Framework
SVG	Scalable Vector Graphics
UML	Unified Modeling Language
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
URN	Uniform Resource Name
UUID	Universally Unique IDentifier
XML	eXtensible Markup Language

4.2 UML Notation

In this document, conceptual schemas are presented in the Unified Modeling Language (UML). The user shall refer to ISO 19103 for the specific profile of UML used in this document.

4.3 Backward compatibility

This document uses the concepts defined in the Place Identifier (PI) Architecture (ISO 19155) without modification. Therefore, no backward compatibility issues exist in this document.

5 Conformance

5.1 General

This document defines three conformance classes, specified in 5.2 to 5.4, matching the requirements classes of the three linking mechanisms defined in Clause 8. Any PI linking mechanism for which conformance with this document is claimed shall pass all of the requirements of the abstract test suite specified in Annex A.

5.2 Linking mechanism: gml:id

PI linking for which gml:id conformance is claimed shall pass all of the requirements specified in the abstract test suite in A.2.

Table 1 — Conformance class: Linking mechanism: gml:id

Conformance class	/conf/19155-2/5/5.2
Dependency	A.1
Requirements	/req/linking_mechanism/GML_ID
Tests	A.2

5.3 Linking mechanism: UUID

PI linking for which UUID conformance is claimed shall pass all of the requirements specified in the abstract test suite in A.3.

Table 2 — Conformance class: Linking mechanism: UUID

Conformance class	/conf/19155-2/5/5.3
Dependency	A.1
Requirements	/req/linking_mechanism/UUID
Tests	A.3

5.4 Linking mechanism: URL

PI linking for which URL conformance is claimed shall pass all of the requirements specified in the abstract test suite in A.4.

Table 3 — Conformance class: Linking mechanism: URL

Conformance class	/conf/19155-2/5/5.4
Dependency	A.1
Requirements	/req/linking_mechanism/URL
Tests	A.4

6 Place Identifier (PI) concept

6.1 General

A "place" is defined as an identifiable part of any space (ISO 19155), either in the real world or virtual world.

ISO 19155:2012, 6.1 states that the same place can be identified with multiple Place Identifiers (PIs). If the place is identified with coordinates, it is called "position" and if the place is identified with geographic identifiers, it is called "location." Additionally, the place may be identified with online resource identifiers such as URI.

In ISO 19109:2015, 7.4, "position" is a spatial attribute of a feature, "location" is a location attribute of a feature, and a virtual identifier, such as a URI, is a thematic attribute of a feature. Therefore, a PI can be considered as an attribute of a feature.

6.2 PI structure

Figure 2 shows the structure of the PI, as defined in ISO 19155:2012, 7.2.2. The PI_PlaceIdentifier type is the basic data type for the PI.

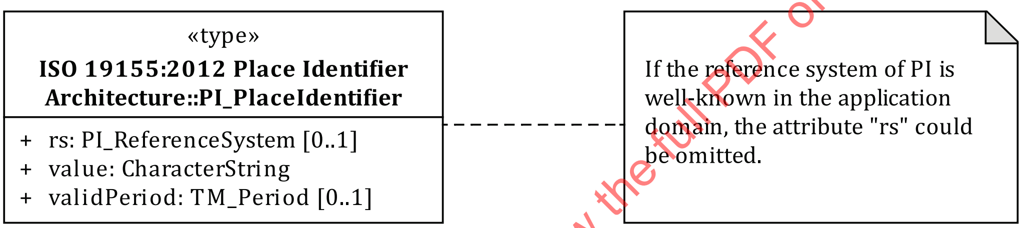


Figure 2 — PI_PlaceIdentifier

The PI_PlaceIdentifier class defines the following attributes.

a) The attribute *rs* specifies the reference system of the PI.

b) The attribute *value* is the identifier for a place and is unique within the context of the defined reference system.

c) The optional attribute *validPeriod* specifies the period in which the identifier is valid for the place.

The *validPeriod* can be used to set the expiration of the identifier which can be validated through implementation. The property of uniqueness of the identifier is determined by the reference system (*rs*).

7 PI linking

7.1 Overview

There are two main concepts that are standardized in this document.

The first concept is a mechanism for connecting PIs which identify the same place, where the connection is directly embedded into the PI.

PI matching (ISO 19155:2012, 4.11) is the matching of multiple PIs that represent the same place. This matching is accomplished through the use of the PI matching service (ISO 19155:2012, 7.3.2) and when implemented, is performed within the PI architecture. The logical structure of "PI matching" is shown in ISO 19155:2012, Figure 8, where a "PI_MatchedPISet" connects more than two "PI_PlaceIdentifier"s which identify the same place. The one way association from "PI_MatchedPISet" to "PI_PlaceIdentifier" enables users to make links independently with the instances of PI_PlaceIdentifier, and the data independence of the connection ensures the use of existing place identifiers because existing data need not change their schemas.

However, in many situations, relationships exist between PIs encoded using ISO 19155 and other identifiers in different encodings, existing independently or outside of the PI architecture. As these other identifiers are usually features or objects in different encodings, it is not possible for them to be included in or managed by the PI architecture. In these situations, users may want to simply embed the connection into their PI. Depending on the encodings of these other identifiers, a set of clearly defined mechanisms is required to expose and use those other identifiers for PI linking.

This is referred to as a Place Identifier (PI) link (3.1), similarly, a Place Identifier (PI) linking mechanism (3.2) is a means of defining a Place Identifier (PI) link.

While one specific linking mechanism is not suitable for all other identifiers in other encodings, similarities in linking mechanisms exist, such as the use of XLink for XML-based encodings.

The second concept standardized by this document is the structure of data instances for PI Linking. This document defines how to use existing constructs to enable linking to PIs without modification to those encoding rules for object identification and reference.

7.2 PI linking directionality

In this document, all links are directional. In cases where a bi-directional link is required, two distinct directional links shall be made, one in each direction.

/req/linking_directionality/directional_links	where a bi-directional link is required, two distinct directional links shall be made, one in each direction

A combination of directionality and encoding format limits the applicability of this document, as shown in the following cases:

a) if both identifiers are encoded in GML (ISO 19136:2007), the linking is normative;

b) if one identifier is in another encoding, the linking from a PI to that identifier is normative;

c) if one identifier is in another encoding, the linking from that identifier to a PI is informative.

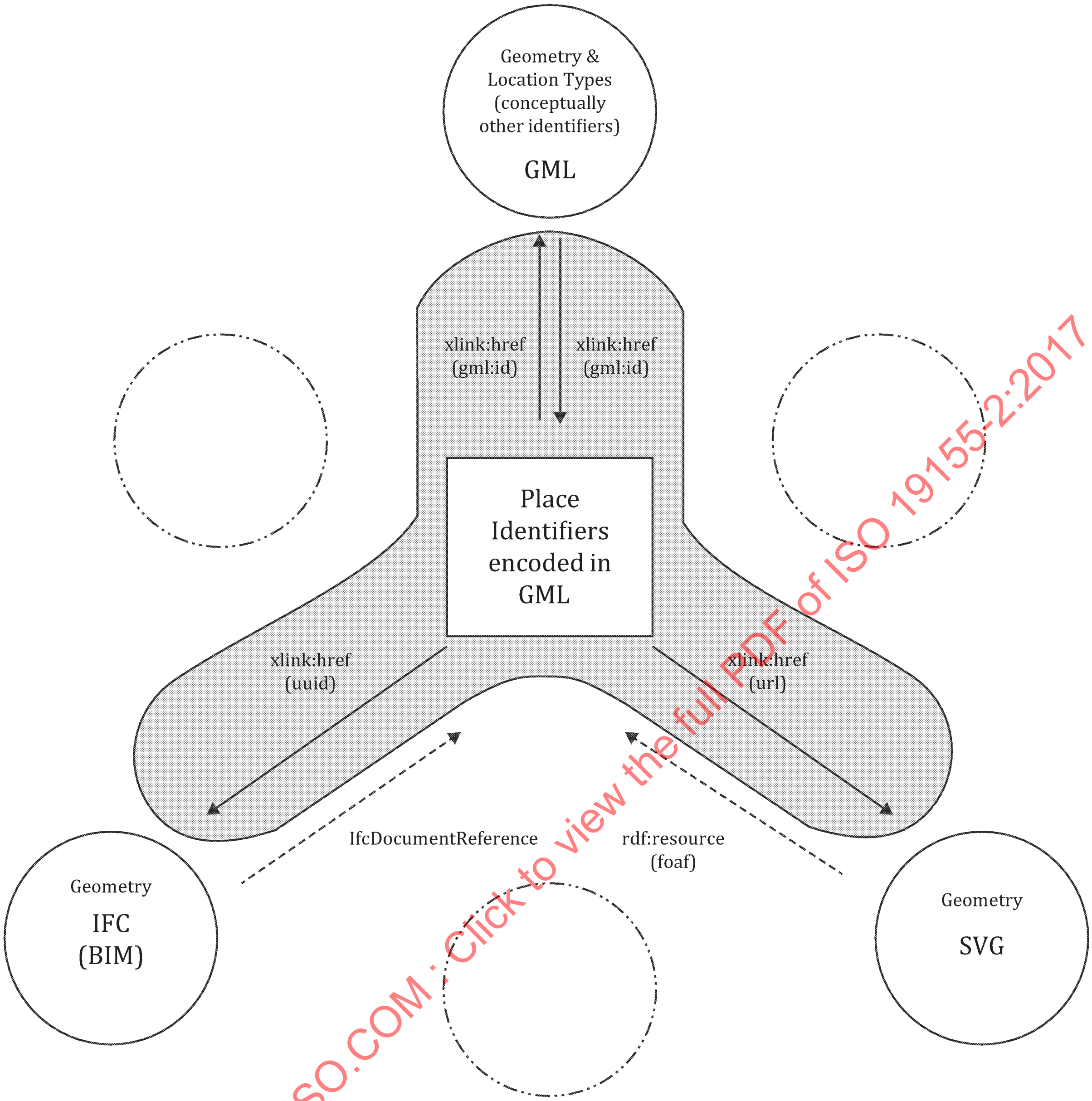
In Figure 3, the direction of the link between a PI encoded using ISO 19155 and an identifier in another encoding is shown using an arrow representing the directionality of the link. In addition, normative linking is shown using a solid line and informative linking is shown using a dashed line.

The normative linking mechanisms are:

a) gml:id: for linking between a PI and another identifier encoded in GML, see 8.2.2 and 8.2.3;

b) UUID or URL: for linking from a PI to an identifier in another encoding, see 8.3 and 8.4, respectively.

Figure 3 depicts linking to PIs from IFC(BIM) and SVG using IfcDocumentReference and rdf:resource as linking mechanisms. These mechanisms are shown as informative because those encodings are not being modified by this document.



Key

encoding rules (examples in this document)

other encoding rules

linking methods (normative)

linking methods (informative)

ISO 19155-2 scope (normative content)

ISO 19155 (normative PI encoding)

Figure 3 — Overview of content in ISO 19155-2

7.3 PI linking model

The conceptual model in Figure 4 defines the structure of PI linking.

The PI_PlaceIdentifier class defined in this document is an extended data type for the Place Identifier which enables users to connect a PI directly to other PIs.

The PI_PlaceIdentifier class also has a mandatory association “Linking” in addition to the derived three attributes. The association “Linking” is a mechanism for connecting PIs which identify the same place, where the connection is directly embedded into the PI.

NOTE While “PI_MatchedPISet” defined in ISO 19155 has an association to connect more than two “PI_PlaceIdentifier”s which identify the same place, “PI_PlaceIdentifier” defined in ISO 19155 itself does not have a property to connect other “PI_PlaceIdentifier”s.

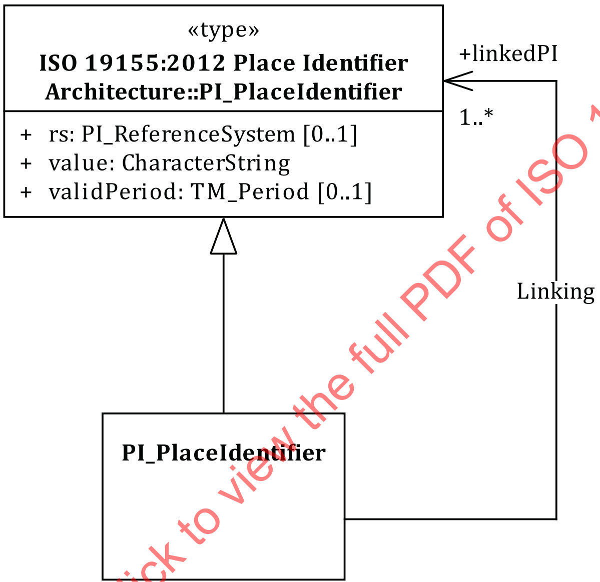


Figure 4 — PI linking model

8 PI linking mechanisms

8.1 Overview

This document defines three mechanisms for linking Place Identifiers to features or objects existing in other encodings. These PI linking mechanisms are enabled using the following information technologies for object identification and reference:

— using a gml:id linking both from and to a PI, see 8.2;

— using a Universally Unique Identifier (UUID) linking from a PI, see 8.3;

— using a Uniform Resource Locator (URL) linking from a PI, see 8.4.

When using the linking mechanisms defined in this document, the PIs shall be encoded using ISO 19136, which is the recommended normative encoding option, as specified in ISO 19155:2012, Annex B.

/req/linking_mechanisms/encoding	existing Place Identifiers shall be encoded using ISO 19136 as specified in ISO 19155:2012, Annex B

Implementation requirements are specified in 8.2, 8.3, and 8.4.

The encodings presented in this document are suitable for linking with a PI but do not represent a conclusive list and other encodings may also be suitable for linking with a PI using one of the defined mechanisms. Linkages may also exist directly between multiple encodings and can make use of a PI. For the purpose of this document, linkages not using a PI are out of scope.

Use case examples of PI linking are presented in Annex F.

The PI architecture and PI linking concepts have implications for use in the semantic web and linked data. To provide additional descriptions, Annex G includes RDF examples for PIs and PI linking.

8.2 Linking mechanism: gml:id

8.2.1 Overview

If gml:id is used, linking is defined as bi-directional, both from and to a PI, and linking Place Identifiers with other features encoded in GML and related application schema is specified based on the following content from ISO 19136:2007, 8.1:

> *Xlink components are used in GML to implement associations between objects by reference. GML property elements (ISO 19136:2007, 7.2.3) may carry XLink attributes, which support the encoding of an association relationship by reference, the name of the property element denoting the target role in the association.*

/req/linking_mechanisms/GML_ID	gml:id shall be used when linking to Place Identifiers encoded in GML

8.2.2 Linking from a PI

An xlink:href property in a PI instance defines a link that shall be used to point to a discrete GML instance and the value in that discrete GML instance is referenced by traversing the link.

/req/linking_mechanisms/GML_ID/from-a-PI	the xlink:href property in a PI instance shall use a gml:id to define a link that points to a discrete GML instance

8.2.3 Linking to a PI

The converse is also defined by this document, where an xlink:href property in a GML instance defines a link that shall be used to point to a discrete PI instance and the value in that discrete PI instance is referenced by traversing the link.

/req/linking_mechanisms/ GML_ID/to-a-PI	the xlink:href property in a GML instance shall use a gml:id to define a link that points to a discrete PI instance

8.2.4 Instance examples using gml:id

Examples showing the use of gml:id for linking between PIs and other GML encoded instances are presented in Annex B.

GML feature instances conforming to application schemas, such as OGC CityGML (OGC 12-019) and OGC IndoorGML 1.0 (OGC 14-005r3), may also be linked with a PI, as shown in Annex C.

8.3 Linking mechanism: UUID

If UUID is used, linking is defined in a single direction, from a PI, where a Namespace URN shall conform to IETF RFC 4122, and a Universally Unique Identifier (UUID) is stored in the attribute of the linkedPI, as shown in the following example fragment.

```
…
<PI_PlaceIdentifier gml:id="pi006">
       <rs>TokyoInternationalAirport_POI_list.net</rs>
       <value>Terminal1_Gate15</value>
       <linkedPI xlink:href="urn:uuid:dddbf24a-e0df-1f48-fd9e-7646bc510cf7"/>
</PI_PlaceIdentifier>
…
```

/req/linking_mechanisms/UUID	when a UUID is used, the xlink:href property in a PI instance shall use a UUID to define a link from a PI that points to a discrete instance of another identifier

Examples showing the use of UUID for linking PIs are presented in Annex D.

8.4 Linking mechanism: URL

If a URL is used, linking is defined in a single direction, from a PI, where a URL conforming to IETF RFC 1738, is stored in the attribute of the linkedPI, as shown in the following example fragment.

```
…
<PI_PlaceIdentifier gml:id="pi007">
       <rs>gazetteer_tokyo.net</rs>
       <value>Roppongi</value>
       <linkedPI xlink:href="http://some-svg-server.org/tokyo/minatoku.svg#Roppongi"/>
</PI_PlaceIdentifier>
…
```

/req/linking_mechanisms/URL	when a URL is used, the xlink:href property in a PI instance shall use a URL to define a link from a PI that points to a discrete instance of another identifier

Examples showing the use of a URL for linking PIs are presented in Annex E.

Annex A
(normative)

Abstract test suite

A.1 Conformance test for PI linking mechanisms

a) Test purpose: Verify a defined PI linking mechanism.

b) Test method: Check that the PI linking mechanism follows requirements specified in this document.

c) Reference: Clause 7.

d) Test type: Basic.

A.2 Conformance test for a PI linking mechanism using gml:id

a) Test purpose: Verify the defined PI linking mechanism using gml:id. Each PI encoded as GML shall be implemented based on the requirements of this document when it is necessary to have a linkage between other identifiers in different encoding domains.

b) Test method: Check the document to confirm that the xlink:href attribute is used for the linking between the other identifiers when the document is encoded as GML and that the xlink:href points to a gml:id.

c) Reference: 8.2; ISO 19155:2012; ISO 19136:2007, 8.1.

d) Test type: Capability.

A.3 Conformance test for a PI linking mechanism using UUID

a) Test purpose: Verify the defined PI linking mechanism using UUID. Each PI encoded as GML shall be implemented based on the requirements of this document when it has a linkage between the other identifiers using a UUID.

b) Test method: Check the document to confirm that the xlink:href attribute is used for linking to another identifier pointed to by a UUID.

c) Reference: 8.3; ISO 19155:2012; ISO 19136:2007, 8.1.

d) Test type: Capability

A.4 Conformance test for a PI linking mechanism using URL

a) Test purpose: Verify the defined PI linking mechanism using URL. Each PI encoding as GML shall be implemented based on the requirements of this document when it has a linkage between the other identifiers using a URL.

b) Test method: Check the document to confirm that the xlink:href attribute is used for linking to another identifier pointed to by a URL.

c) Reference: 8.4; ISO 19155:2012; ISO 19136:2007, 8.1

d) Test type: Capability

Annex B
(normative)

Encoding using gml:id to link with GML

B.1 Encoding introduction

This annex defines the encoding procedures for using gml:id for linking, as specified in 8.2.

In the following encoded instance fragments, the PIs are encoded using ISO 19136 (GML), the normative option, specified in ISO 19155:2012, Annex B.

B.3 contains XSD and XML documents of the encoding samples shown in B.2.

All components of the GML schema for PI are defined in the name space with the identifier "http://standards.iso.org/iso/19155/gpi/1.0", for which the prefix **gpi** or the default namespace is used within this document.

B.2 Instance encodings

B.2.1 Link from a PI

As specified in 8.2.2, a PI can be linked to a discrete GML instance using gml:id with the following constructs.

A PI encoded in GML shall link to a GML feature instance using xlink:href as specified in ISO 19136:2007, 7.2.3.

```
<!-- place identifier -->
<PI_PlaceIdentifier gml:id="PI01">
    <rs>park-registry.net</rs>
    <value>Hibiya Park</value>
    <linkedPI xlink:href="http://sample.net/sample_app_data01.xml#park001">
</linkedPI>
</PI_PlaceIdentifier>
```

In the instance example above, a link is established to the gml:id, as shown in the following fragment, which is part of a larger GML file

```
<!-- example: http://sample.net/sample_app_data_01.xml-->
<parkName gml:id="park001">
    <name>Hibiya Park</name>
</parkName>
```

If there is another GML point instance which identifies Hibiya Park, the PI_PlaceIdentifier can have a link to the GML point instance.

```
<PI_PlaceIdentifier gml:id="PI01">
    <rs>park-registry.net</rs>
    <value>Hibiya Park</value>
    <linkedPI xlink:href="http://sample.net/sample_app_data01.xml#park001"/>
    <linkedPI xlink:href="http://sample.net/jp13_park_centroids.xml#pt327"/>
</PI_PlaceIdentifier>
```

In the previous example, the second linked GML instance is also part of a larger GML file, which identifies the position of Hibiya Park.

```
<!-- point instance from a larger GML file
example: http://sample.net/jp13_park_centroids.xml-->
<gml:Point gml:id="pt327"
<!-- centroid for Hibiya park -->
    srsName="http://www.opengis.net/def/crs/EPSG/0/4326">
    <gml:coordinates>35.674393, 139.756522</gml:coordinates>
</gml:Point>
```

B.2.2 Link to a PI

As specified in 8.2.3, a GML instance can be linked to a PI instance using gml:id with the following constructs. The default encoding of a link from a GML instance to a PI uses a property element defined in a GML application schema. The property element to link to a PI or PIs should be declared in the GML application schema.

In the examples below, `landmarkName` and `landmarkPosition` both have a `linkedPI` property that is used to link to a PI.

```
<landmarkName gml:id="lm001">
        <name>Hibiya Park</name>
        <linkedPI xlink:href="http://park-registry.net/pidata.xml#PI01"/>
</landmarkName>

<landmarkPosition gml:id="lm02">
   <pos>
     <gml:Point gml:id="pt328" srsName="http://www.opengis.net/def/crs/EPSG/0/4326">
        <gml:coordinates>35.674393, 139.756522</gml:coordinates>
     </gml:Point>
   </pos>
   <linkedPI xlink:href="http://park-registry.net/pidata.xml#PI01"/>
</landmarkPosition>
```

The referred PI instance should be declared in the specified location:

```
<!-- place identifier: http://park-registry.net/pidata.xml-->
<PI_PlaceIdentifier gml:id="PI01">
     <rs>park-registry.net</rs>
     <value>Hibiya Park</value>
</PI_PlaceIdentifier>
```

In this case, an attribute for PI linking should be declared in a GML application schema:

```
<element name="linkedPI" type="gml:ReferenceType">
     <annotation>
        <appinfo>
            <gml:targetElement>gpi:PI_PlaceIdentifier</gml:targetElement>
        </appinfo>
     </annotation>
</element>
```

An entire GML application schema that includes the property for PI linking is shown in B.3.1.

NOTE It can be difficult to add a new property for PI linking to GML application schemas that already exist. In these cases, the "PI_MatchedPISet" class (ISO 19155:2012, 7.2.5) can be used to connect more than one PI without needing to modify the already existing GML application schema.

B.3 Sample GML application schema and data

B.3.1 Sample GML application schema

The element which refers to a PI should be declared in a GML application schema to realize PI linking.

A sample GML application schema which contains two features that refer to a PI as one of their properties is shown in the listing below.

In this example, the feature `Park` has a geographic name property `landmarkName`, which has a link to a PI, and the feature `Shop` has a geometric property `landmarkPosition`, which also has a link to a PI.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:ex="http://sample.net/sample" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" targetNamespace="http://sample.net/sample"
elementFormDefault="qualified" version="1.0">
   <xsd:import namespace="http://www.opengis.net/gml/3.2" schemaLocation="http://standards
.iso.org/ittf/PubliclyAvailableStandards/ISO_19136_Schemas/gml.xsd"/>
   <!-- Feature "Park" and "Shop" for example -->
   <xsd:element name="Dataset">
     <xsd:complexType>
        <xsd:choice maxOccurs="unbounded">
            <xsd:element ref="ex:Park"/>
            <xsd:element ref="ex:Shop"/>
            <xsd:any namespace="http://www.opengis.net/gml/3.2"/>
        </xsd:choice>
     </xsd:complexType>
   </xsd:element>
   <xsd:element name="Park" type="ex:ParkType" substitutionGroup="gml:AbstractFeature"/>
   <xsd:complexType name="ParkType">
     <xsd:complexContent>
        <xsd:extension base="gml:AbstractFeatureType">
            <xsd:sequence>
                <xsd:element ref="ex:landmarkName" minOccurs="0"/>
                <xsd:element name="area" type="gml:SurfacePropertyType"/>
            </xsd:sequence>
        </xsd:extension>
     </xsd:complexContent>
   </xsd:complexType>
   <xsd:element name="Shop" type="ex:ShopType" substitutionGroup="gml:AbstractFeature"/>
   <xsd:complexType name="ShopType">
     <xsd:complexContent>
        <xsd:extension base="gml:AbstractFeatureType">
            <xsd:sequence>
                <xsd:element name="name" type="xsd:string"/>
                <xsd:element name="telNumber" type="xsd:string"/>
                <xsd:element ref="ex:landmarkPosition" minOccurs="0"/>
            </xsd:sequence>
        </xsd:extension>
     </xsd:complexContent>
   </xsd:complexType>
   <xsd:element name="landmarkName" type="ex:landmarkNameType" substitutionGroup="gml:
AbstractGML"/>
   <xsd:complexType name="landmarkNameType">
     <xsd:complexContent>
        <xsd:extension base="gml:AbstractGMLType">
            <xsd:sequence>
                <xsd:element name="name" type="xsd:string"/>
                <xsd:element name="linkedPI" type="gml:ReferenceType" minOccurs="0"
maxOccurs="unbounded">
                    <xsd:annotation>
                        <xsd:appinfo>
                            <gml:targetElement>gpi:PI_PlaceIdentifier</gml:targetElement>
                        </xsd:appinfo>
                    </xsd:annotation>
                </xsd:element>
            </xsd:sequence>
        </xsd:extension>
     </xsd:complexContent>
```

```
   </xsd:complexType>
   <xsd:element name="landmarkPosition" type="ex:landmarkPositionType"
substitutionGroup="gml:AbstractGML"/>
   <xsd:complexType name="landmarkPositionType">
     <xsd:complexContent>
        <xsd:extension base="gml:AbstractGMLType">
            <xsd:sequence>
                <xsd:element name="pos" type="gml:PointPropertyType"/>
                <xsd:element name="linkedPI" type="gml:ReferenceType" minOccurs="0"
maxOccurs="unbounded">
                    <xsd:annotation>
                        <xsd:appinfo>
                            <gml:targetElement>gpi:PI_PlaceIdentifier</gml:targetElement>
                        </xsd:appinfo>
                    </xsd:annotation>
                </xsd:element>
            </xsd:sequence>
        </xsd:extension>
     </xsd:complexContent>
   </xsd:complexType>
</xsd:schema>
```

B.3.2 Sample GML application data

Sample application data based on the GML application schema in B.3.1.

```
<?xml version="1.0" encoding="UTF-8"?>
<Dataset xmlns="http://sample.net/sample" xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xlink="http://www.w3.org/1999/
xlink" xsi:schemaLocation="http://sample.net/sample sample_appschema">
   <Park gml:id="pk">
     <landmarkName gml:id="lm001">
        <name>Hibiya Park</name>
        <linkedPI xlink:href="http://park-registry.net/pidata.xml1#PI01"/>
     </landmarkName>
     <area xlink:href="sf1"/>
   </Park>
   <Shop gml:id="sp">
     <name>Hibiya Restaurant</name>
     <telNumber>+81-0-1234-5678</telNumber>
<landmarkPosition gml:id="lm02">
        <pos>
            <gml:Point gml:id="pt328" srsName="http://www.opengis.net/def/crs/EPSG/
0/4326">
                <gml:coordinates>35.674393, 139.756522</gml:coordinates>
            </gml:Point>
        </pos>
        <linkedPI xlink:href="http://park-registry.net/pidata.xml1#PI02"/>
     </landmarkPosition>
   </Shop>
   <gml:Surface gml:id="sf1">
     <gml:patches>
        <gml:PolygonPatch>
            <gml:exterior>
                <gml:Ring>
                    <gml:curveMember xlink:href="cv1"/>
                </gml:Ring>
            </gml:exterior>
        </gml:PolygonPatch>
     </gml:patches>
   </gml:Surface>
   <gml:Point gml:id="pt1">
     <gml:pos>35.673824 139.756163</gml:pos>
   </gml:Point>
   <gml:Curve gml:id="cv1">
     <gml:segments>
        <gml:LineStringSegment>
            <gml:pos>35.6733163 139.7577562</gml:pos>
            <gml:pos>35.6746498 139.7587111</gml:pos>
            <gml:pos>35.6766972 139.7551579</gml:pos>
            <gml:pos>35.6723144 139.7530255</gml:pos>
```



```
            <gml:pos>35.6710583 139.7559275</gml:pos>
            <gml:pos>35.6733163 139.7577562</gml:pos>
        </gml:LineStringSegment>
     </gml:segments>
   </gml:Curve>
</Dataset>
```

Annex C
(normative)

Encoding using gml:id to link with GML application schemas

C.1 Linking with CityGML

C.1.1 General

CityGML is a model for the representation of 3D objects commonly found in an urban environment. CityGML is implemented as an application schema of GML 3. The current release of CityGML is revision 2.0 (12-019, April 2012). Work on CityGML is managed through the Open Geospatial Consortium.

CityGML supports different Levels of Detail (LOD). As specified in CityGML (12-019:6.2 Multi-scale modelling):

LODs are required to reflect independent data collection processes with differing application requirements. Further, LODs facilitate efficient visualisation and data analysis. In a CityGML dataset, the same object may be represented in different LOD simultaneously, enabling the analysis and visualisation of the same object with regard to different degrees of resolution.

Furthermore, two CityGML data sets containing the same object in different LOD may be combined and integrated. However, it will be within the responsibility of the user or application to make sure objects in different LODs refer to the same real-world object.

The coarsest level LOD0 is essentially a two and a half dimensional Digital Terrain Model over which an aerial image or a map may be draped. Buildings may be represented in LOD0 by footprint or roof edge polygons.

LOD1 is the well-known blocks model comprising prismatic buildings with flat roof structures.

In contrast, a building in LOD2 has differentiated roof structures and thematically differentiated boundary surfaces.

LOD3 denotes architectural models with detailed wall and roof structures potentially including doors and windows.

LOD4 completes a LOD3 model by adding interior structures for buildings. For example, buildings in LOD4 are composed of rooms, interior doors, stairs, and furniture.

C.1.2 CityGML instance example

As specified in 8.2.4, a PI shall be linked to a discrete CityGML instance using gml:id.

In this example, a PI which is registered in the building fire registry and is occasionally checked by the fire brigade, contains a link to another id which references the Place Identifier in the CityGML file. The linked CityGML id `GML-8f988da9-22d7-41e5-ae94-880afd46a3c9` identifies the geometry of the fire exit door, therefore, the fire brigade can specify the location of the fire exit door through PI linking.

The CityGML id `GML-8f988da9-22d7-41e5-ae94-880afd46a3c9` is used to link to the larger CityGML instance file. The value element of the PI contains another CityGML id which references the actual object in the CityGML file, in this example, a specific fire exit door, which may be registered in the building fire registry and is occasionally checked by the fire brigade. Given the following PI instance fragment:


```
<!-- place identifier -->
<PI_PlaceIdentifier gml:id="Fire-Exit-pi001"
    <rs>bldg-fire-registry.net</rs>
    <value>Bristol Hotel FireExit East-003</value>
    <linkedPI xlink:href="http://some-citygml2-server.net/LOD3-datafile.gml#GML-8f988da9
-22d7-41e5-ae94- 880afd46a3c9">
</PI_PlaceIdentifier>
```

A link is established to the gml:id, as shown in this CityGML fragment, which is part of a larger CityGML file.

```
    <!-- instance from larger CityGML file (building … 3eb9ee0a8e68) -->
    <bldg:opening>
        <bldg:Door gml:id="GML-93096bbb-5155-47fb-ae2c-e2f9327f3007">
            <gml:name>FireExit East003</gml:name>
            <bldg:lod3MultiSurface>
                <gml:MultiSurface>
                    <gml:surfaceMember>
                        <gml:Polygon gml:id="GML-8f988da9-22d7-41e5-ae94-
                         880afd46a3c9">
                            <gml:exterior>
                                <gml:LinearRing>
                                    <gml:posList>
                                        458884.9 5438352 112.2 458884.9 5438353 112.2
                                        458884.9 5438353 114.2 458884.9 5438352 114.2
                                        458884.9 5438352 112.2
                                    </gml:posList>
                                </gml:LinearRing>
                            </gml:exterior>
                        </gml:Polygon>
                    </gml:surfaceMember>
                </gml:MultiSurface>
            </bldg:lod3MultiSurface>
        </bldg:Door>
    </bldg:opening>
```


C.2 Linking with IndoorGML

C.2.1 Overview

IndoorGML (Revision 1.0: 14-005r3) is a model for the representation of indoor navigation applications implemented as an application schema of GML 3. Work on IndoorGML is managed through the Open Geospatial Consortium.

C.2.2 IndoorGML instance example

As specified in 8.2.4, a PI shall be linked to a discrete IndoorGML instance using gml:id with the following constructs. The fragment below is an example of a linking from a GML encoded PI to an IndoorGML node instance object.

```
<PI_PlaceIdentifier gml:id="PI02">
    <rs>SampleBldg-Exit</rs>
    <value>Emergency Exit of the 2nd Floor</value>
    <linkedPI xlink:href="http://sample.net/sample_indoorGML01.xml#P1"/>
</PI_PlaceIdentifier>
```

Where the P1 is a Place Identifier instance from a larger IndoorGML file (sample_indoor01.xml)

```
<!-- node instance from IndoorGML file (sample_indoorGML01.xml) -->
...
<MultiLayeredGraph gml:id="MG1">
    <spaceLayers gml:id="SL1">
        <spaceLayerMember>
            SpaceLayer gml:id="IS1">
```



```
<nodes gml:id="N1">
    <stateMember>
       <State gml:id="R1">
           <gml:name>002</gml:name>
               <duality xlink:href="#C1"/>
               connects xlink:href="#T0"/>
               <connects xlink:href="#T1"/>
               geometry>
                  <gml:Point gml:id="P1">
                     <gml:pos>445536.48 5444906.24 -2.02</gml:pos>
                  </gml:Point>
               </geometry>
       </State>
    </stateMember>
...
```

Annex D
(normative)

Encoding using UUID for linking

D.1 Encoding introduction

This annex defines the encoding procedures for using UUID for linking, as specified in 8.3, and shows an example with IFCxml.

In the following examples, the PIs are encoded using ISO 19136 (GML), the normative option specified in ISO 19155:2012, Annex B.

In the IFC model, every object instance is assigned a unique identifier based on Universal Unique Identifier (UUID) as specified by IETF RFC 4122. Over time, organizations have also developed standards using the UUID definitions, such as

— ISO/IEC 11578,

— ITU-T/Rec. X.667 [currently being revised 201210 draft], and

— ISO/IEC 9834-8.

NOTE ISO/IEC 9834-8 and ITU-T/Rec. X.667 are essentially the same standard.

Within the IFC specification, the implementation of UUID is referred to as a "globally unique identifier" or GUID. In addition, the IFC specification requires the GUID be compressed for exchange purposes following a published compression function. The final compressed GUID is referred to as an "IFC-GUID." The IFC-GUID is created using the 128bit UUID applying a 64-character encoding, resulting in a fixed 22-character string.

D.2 Instance encodings

As specified in 8.3, when a UUID is used, a PI shall be linked using the following constructs.

Given the following example PI instance fragment:

```
<!-- place identifier -->
<PI_PlaceIdentifier gml:id="pi006">
    <rs>TokyoInternationalAirport_POI_list.net</rs>
    <value>Terminal1_Gate15</value>
    <linkedPI xlink:href="urn:uuid:dddbf24a-e0df-1f48-fd9e-7646bc510cf7”/>
</PI_PlaceIdentifier>
```

A link is established from the value attribute of the PI to the GUID of the ifcXML fragment, which is part of a larger feature instance.

In IFC, the IFC-GUID is specified as a base64 string. In the example above, the base64 conversion of `uuid:dddbf24a-e0df-1f48-fd9e-7646bc510cf7` is `3dvySuDfH0j9nnZGvFEM99`.

```
…
<IfcSite CompositionType="element" GlobalId="3dvySuDfH0j9nnZGvFEM99" Name="Site"
RefElevation="0" id="i3">
        OwnerHistory ref="i2" xsi:nil="true" xsi:type="IfcOwnerHistory"/>
        <ObjectPlacement id="i10" xsi:type="IfcLocalPlacement">
            <RelativePlacement>
                <IfcAxis2Placement3D id="i41">
```

```
                    <Location Coordinates="0. 0. 0." id="i42" xsi:type=
"IfcCartesianPoint"/>
                    <Axis DirectionRatios="0. 0. 1." id="i43" xsi:type="IfcDirection"/>
                    <RefDirection DirectionRatios="1. 0. 0." id="i44" xsi:type=
"IfcDirection"/>
                </IfcAxis2Placement3D>
            </RelativePlacement>
        </ObjectPlacement>
        …
```

Annex E
(normative)

Encoding using URL for linking

E.1 Encoding introduction

This annex defines the encoding procedures for using URL for linking, as specified in 8.4, and shows an example with SVG.

In the following examples, the PIs are encoded using ISO 19136 (GML), the normative option specified in ISO 19155:2012, Annex B.

E.2 Instance encodings

As specified in 8.4, when a URL is used, a PI shall be linked using the following constructs.

Given the following example PI instance fragment:

```
<PI_PlaceIdentifier gml:id="pi007">
    <rs>gazetteer_tokyo.net</rs>
    <value>Roppongi</value>
    <linkedPI xlink:href="http://some-svg-server.org/tokyo/minatoku.svg#Roppongi"/>
</PI_PlaceIdentifier>
```

A link is established from the value attribute of the PI to the SVG fragment, which is part of a larger feature instance.

```
<?xml version="1.0" encoding="UTF-8"?>
<svg xmlns="http://www.w3.org/2000/svg" version="1.1">
    <!-- This SVG encodes subdivisions in Minato-ku, Tokyo. -->
    <title>Minato-ku, Tokyo</title>
    <metadata>
        <rdf:RDF xmlns:svg="http://www.w3.org/2000/svg"
            xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:crs="http://
opengis.org/xmldtds/transformations.dtd">
            <rdf:Description rdf:about="">
                <crs:CoordinateReferenceSystem
                    rdf:resource="http://purl.org/crs/84" svg:transform="matrix(1,0,0,1,0
,0)" />
            </rdf:Description>
        </rdf:RDF>
    </metadata>
    <g id="akasaka">
        <polygon points="139.73149600 35.68149500 ..." />
    </g>
    <g id="roppongi">
        <polygon points="139.73107600 35.68124400 ..." />
    </g>
    ...
</svg>
```

Annex F
(informative)

Use case examples

F.1 Personal navigation example

F.1.1 "3D Slice Model Format"

The "3D Slice model format" was developed for a domestic Location Based Service (LBS) testbed in Japan during 2010. The format is compound, comprising of data files in SVG, CSV and HTML. The focus of the testbed was indoor navigation. Floor plan diagrams are encoded using SVG. Basic content information regarding the buildings is encoded in HTML. A concise network structure is encoded in CSV.

The SVG data represented the layout of a floor 2D and was referred to as a slice model. Each floor was encoded in SVG along with a value representing the approximate elevation of the floor. All of the objects on the same floor had the same elevation. The network structure also has elevation values.

The HTML files contained information about the building that could not be easily encoded into the SVG. This content included the name of the building, address, post-code, latitude and longitude, location of the entrances. The elevation information was described as the number of floors both above and below ground. The number of SVG files (that represented the floors) was also encoded. This content was placed in the header of the HTML files using meta tags as shown in the following example:

```
<head>
    <title>Marunouchi Building</title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta name="geo.position" content="35.68118264026908;139.76376950740814"/>
    <meta name="BLDGNAME" content="Marunouchi Building"/>
    <meta name="BLDGKANA" content="japanese characters"/>
    <meta name="BLDGNIC" content="Maru Bil"/>
    <meta name="ADDRESS" content="2-4-1 Marunouchi, Chiyoda-ku, Tokyo"/>
    <meta name="BASEFLR" content="1"/>
    <meta name="TOTALFLR" content="Above Ground 37:Below Ground 4"/>
    <meta name="AVAILFLR" content="41"/>
</head>
```

Links to both the SVG files and network CSV files are encoded in the body of the HTML file using hyper link tags. Finally, the entire group of SVG, HTML and CSV files are then assigned a license designation, based on Creative Commons, and shown in the following translated example:

```
<h2>Floor Information Links</h2>
<ul>
    <li><a href="b2.svg" rel="Floor Plan">B2 Level</a></li>
    <li><a href="b1.svg" rel="Floor Plan">B1 Level</a></li>
    <li><a href="f1.svg" rel="Floor Plan Base Floor" >1F Base Floor</a></li>
    <li><a href="f2.svg" rel="Floor Plan">2F Level</a></li>
    <li><a href="network.csv" rel="network">network</a></li>
</ul>
<h1>Information about this content</h1>
<dl>
    <dt>License</dt>
    <dd>
        <a rel="license" href=" http://creativecommons.org/licenses/by/2.1/jp/deed.en">
        <img alt="Creative Commons License" style="border-width:0"
        src="http://creativecommons.org/images/public/somerights20.png"/>
        </a>
    </dd>
</dl>
```


The network structure describes one or more nodes connected by a series of arcs. The nodes were intersections, entrances to shops with links between them for navigation. The network data was intended to be used for helping people navigate in buildings.

F.1.2 “Registered Place Code”

A “Registered Place Code” is an authorized reference point for use in both indoor and outdoor environments. The “Registered Place Code” was developed for a domestic research project by the Geospatial Information Authority of Japan (GSI). The “Registered Place Code” uses the ucode (ITU-T H.642) encoding format. The ucode format comprises of a 128 bit character string. The first 64 bits contain the official custodial code number. The second 64 bit space is available for independent encoding use by the custodian. In the “Registered Place Code” project, the category, latitude, longitude, special encoded floor and a sequential number are encoded into the 64 bit independent space.

F.1.3 Instance example

The following instance example of the “3D Slice Model Format” makes use of the “Registered Place Code” presenting three different encoding options. The “stand alone format” is the simplest encoding. In fragment “example 1” here, a lat/long pair is tied to a “Registered Place Code.” In fragment “example 2,” a lat/long pair (W3C Basic Geo) and plus other content contained in the `foaf:name` is tied to a “Registered Place Code.” Fragment “example 3” shows how an external file can be referenced using `rdf:resource` and a URI to another XML (SVG Tiny 2.1) file.

```
<svg>
    <metadata>
            <rdf:RDF>
            <!-- standalone form: example 1-->
            <geo:Point geo:lat="35.679699" geo:long="139.764466">
                <foaf:depiction rdf:resource="#registered_place_code_id1"/>
            </geo:Point>
            <!--  standalone form and other information: example 2  -->
            <geo:Point geo:lat="35.680086" geo:long="139.765096"
                foaf:name="East Entrance Main Point">
                    foaf:depiction rdf:resource="#registered_place_code_id2"/>
                    <dcterms:conformsTo rdf:resource="{uri-to-specify-standard}"/>
                </geo:Point>
                <!-- external reference: example 3-->
                <geo:Point rdf:resource="{uri-to-specify-anchorpoint}">
                    <foaf:depiction rdf:resource="#registered_place_code_id3"/>
            </geo:Point>
            </rdf:RDF>
        </metadata>

    <!-- section where the registered place code is anchored to the reference -->
        <use xml:id="registered_place_code_id1" x="120" y="200" class="calc
anchorpointID"/>
</svg>
```

F.2 Colloquial place name relationship example

Different communities (people by age group) often use different identifiers to refer to the same place. To children, a park may be referred to by the type of play equipment present — “the park with the big slide” but adults who do not have any connection to the children, such as emergency service personnel, would most likely only know the park by its official name “Newton Street Park.” Individual PIs could be registered for every colloquial and official reference for a specific place with linkages to the same geographic coordinates, address and map data. The colloquial references could be collected using a collaborative or “crowd-sourced” technique, such as involving different age groups, such as students, families and neighbours, in addition to city officials, emergency service personnel and public safety officers.

F.3 Building Information Model (BIM) examples

F.3.1 Background

An example of a vendor neutral definition of Building Information Modeling (BIM) is the following, provided from the National BIM Standard of the United States:

> *BIM is a digital representation of physical and functional characteristics of a facility. A BIM is a shared knowledge resource for information about a facility forming a reliable basis for decisions during its life-cycle; defined as existing from earliest conception to demolition.*

Also, within the BIM community is an initiative to form a more uniform and open workflow built on BIM called "OpenBIM." As the level of interoperability increases between models, there will be increased opportunities to link Place Identifiers with various elements inside of those models.

F.3.2 Linking PIs with BIM

The following non-exhaustive list is a collection describing the varied and many uses where PIs may be linked with spatial object/zone object attribute information in IFC (BIM) elements and what merits may result from such relationships:

a) urban planning and architectural planning and visualizations; observer locations, walk-throughs, etc.;

b) site/environment simulation; site/transportation network simulation/interaction with the building; traffic simulation for human or cars, using integrated models, such as buildings, underground structures, transportation networks, etc.; energy simulations;

c) linkages from a legal database, such as government laws and regulations, to the model to access specific legal attributes such as sunlight/shade regulations, parcel and land use zoning, road abutments, etc.;

d) validation of disaster evacuation protocols; feasible region can be estimated by using the spatial object linked with feature information including hazardous material, firefighting equipment, refuge accommodation, etc.;

e) emergency response; operation and training of first-responders, emergency and security personnel;

f) facilities maintenance and management; attribute information of facility object; manufacturer information, operating history, repair history; asset management where BIM objects are linked with asset management ID that is utilized for depreciation, preventive maintenance, etc.

F.3.3 PI linking example using BIMServer

The following instance example is built around the premise of the open source Building Information Modelserver (BIMServer) available at <http://bimserver.org>.

The BIMServer enables IFC data to be interpreted and stored using an underlying database, which allows querying and other operations to be performed on the database. It is then possible to request the data as IFC or in other encodings.

The BIMServer API provides the following method to specify an object by using a request URL

```
http://example.com/bimserver/rest/getDataObjectByGuid?roid=65&guid=3dvySuDfH0j9nnZGvFEM99
```

NOTE The IFC-GUID is at the end of this request.

The resulting example output from the BIMServer request is shown below.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<sDataObject>
    <guid>3dvySuDfH0j9nnZGvFEM99</guid>
```

```
    <name>Building</name>
    <oid>143</oid>
    <type>IfcBuilding</type>
    …
</sDataObject>
```

The following linking example shows one method where the URL containing the request with IFC-GUID can be encoded in the value attribute of the linkedPI.

```
<PI_PlaceIdentifier gml:id="pi001">
<rs>gazetteer_tokyo.net</rs>
    <value>Roppongi Building</value>
    <linkedPI xlink:href="http://example.com/bimserver/rest/
getDataObjectByGuid?roid=65&guid=3dvySuDfH0j9nnZGvFEM99"/>
</PI_PlaceIdentifier>
```

F.4 Linking PIs to XML content using UUID

In many web-based resource hierarchies, XML is commonly used. It may be practical to make linkages between PIs in a spatial database encoded in GML with content from a web-based resource hierarchy encoded in XML. One method, uses a UUID in the xlink:href call in the PI instance to link to the XML content. In this example, a system for content retrieval called geogratis uses UUID as a form of a unique content identifier referred to as a "geogratis id."

Given the following PI example encoded as GML with the "geogratis id" used by the xlink:href call:

```
<!-- place identifier -->
<PI_PlaceIdentifier gml:id="pi001"
            <rs>http://geogratis.gc.ca/api/en/nrcan-rncan/ess-sst/</rs>
            <value>Geology, Porcupine River (southeast), Yukon</value>
                <linkedPI xlink:href=http://geogratis.gc.ca/api/en/nrcan-rncan/
            ess-sst/14b3cf37-cff2-5398-8b05-fa3df1d0fd33?alt=xml/>
</PI_PlaceIdentifier>
```

A link is established using the *"geogratis id"* as shown in the following XML fragment, which is part of the output from a larger resource hierarchy.

```
<?xml version="1.0" encoding="UTF-8"?>
<entry xmlns="http://www.w3.org/2005/Atom"
        xmlns:georss="http://www.georss.org/georss"
        xmlns:gml="http://www.opengis.net/gml" xml:base="http://geogratis.gc.ca/api/
en/nrcan-rncan/ess-sst/14b3cf37-cff2-5398-8b05-fa3df1d0fd33" xml:lang="en">
        <id>urn:uuid:14b3cf37-cff2-5398-8b05-fa3df1d0fd33</id>
        <link href="./14b3cf37-cff2-5398-8b05-fa3df1d0fd33.xml" rel="self"/>
        <link href="http://geogratis.gc.ca/api/" rel="profile"/>
        <link href="./14b3cf37-cff2-5398-8b05-fa3df1d0fd33.xml/17" rel="edit"/>
        <link href="./14b3cf37-cff2-5398-8b05-fa3df1d0fd33.xml" rel="alternate"
type="application/xml"/>
        <!-- more XML content follows -->
        …
        <link href="./14b3cf37-cff2-5398-8b05-fa3df1d0fd33.html" rel="alternate"
type="text/html"/>
        <link href="../../../fr/nrcan-rncan/ess-sst/14b3cf37-cff2-5398-8b05-fa3df1d0fd33.
xml" rel="alternate" hreflang="fr" title="Français"/>
        <updated>2014-08-07T09:31:07.000Z</updated>
        <app:edited>2014-08-08T04:26:12.608Z</app:edited>
        <published>2013-03-07T00:00:00.000Z</published>
        <title type="text">Geology, Porcupine River (southeast), Yukon</title>
        <author>
            <name>Lane, L S</name>
        </author>
        <summary type="text">
            The southeast quadrant of Porcupine River (NTS 116-J) has been mapped only
once previously, during Operation Porcupine (1962-1976).
            <!-- more XML content follows -->
            …
```

```
        </summary>
        <!-- more XML content follows -->
        …
</entry>
```

Annex G
(informative)

RDF examples of linking PIs

G.1 Overview

This annex presents a series of RDF/Turtle (Terse RDF Triple Language) encoding examples for linking PIs. To accomplish this encoding, an ontology of the Place Identifier (PI) Architecture (ISO 19155) model was developed. Based on this ontology, the RDF/Turtle encodings were derived.

To ensure an unambiguous visualization for Figures G.2, G.4 and G.5 which represents the structure and data types of the ontology, a diagram notation as shown in Figure G.1 is used.

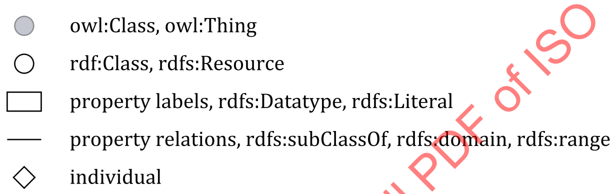


Figure G.1 — Notations

G.2 Requirements of the ontology development

The following requirements were considered in the development of the Place Identifier ontology.

a) All linked PIs should be queryable using SPARQL/GeoSPARQL.

EXAMPLE 1 For a well-known PI composed of a WGS84 coordinate tuple, all linked PIs should be queryable.

b) It should be possible to filter PIs using SPARQL based on the valid date and reference system (RS). Based on this requirement, SPARQL queries are possible returning only valid PIs in a specific reference system.

c) For the encoding of a PI representing a geometry, either GML or WKT should be used based on the encoding rules defined in GeoSPARQL. This requirement enables a request of PIs based on geometries and their topological relationships.

EXAMPLE 2 Return all PIs within a defined area (BBOX) or return the PIs which are closest to a specific point.

d) The OWL Ontologies shall conform to ISO 19150-2.

G.3 Ontology structure

Figure G.2 shows an example Place Identifier Ontology (PIO) for modelling PIs and PI linking.