Core Information Model (CoreModel)

TR-512.14

Location

Version 1.6
January 2024

ONF Document Type: Technical Recommendation
ONF Document Name: Core Information Model version 1.6

Disclaimer

Open Networking Foundation
1000 El Camino Real, Suite 100, Menlo Park, CA 94025
www.opennetworking.org

Important note

This Technical Recommendations has been approved by the Project TST, but has not been approved by the ONF board. This Technical Recommendation is an update to a previously released TR specification, but it has been approved under the ONF publishing guidelines for 'Informational' publications that allow Project technical steering teams (TSTs) to authorize publication of Informational documents. The designation of '-info' at the end of the document ID also reflects that the project team (not the ONF board) approved this TR.

Table of Contents

List of Figures

Document History

Version	Date	Description of Change
1.0	September 2021	Initial Version
1.6	January 2024	Updated release and dates.

1 Introduction to the document suite

This document is an addendum to the TR-512 ONF Core Information Model and forms part of the description of the ONF-CIM. For general overview material and references to the other parts refer to TR-512.1.

1.1 References

For a full list of references see TR-512.1.

1.2 Definitions

For a full list of definition see TR-512.1.

1.3 Conventions

See TR-512.1 for an explanation of:

- UML conventions
- Lifecycle Stereotypes
- Diagram symbol set

1.4 Viewing UML diagrams

Some of the UML diagrams are very dense. To view them either zoom (sometimes to 400%) or open the associated image file (and zoom appropriately) or open the corresponding UML diagram via Papyrus (for each figure with a UML diagram the UML model diagram name is provided under the figure or within the figure).

1.5 Understanding the figures

Figures showing fragments of the model using standard UML symbols as well as figures illustrating application of the model are provided throughout this document. Many of the application-oriented figures also provide UML class diagrams for the corresponding model fragments (see TR-512.1 for diagram symbol sets). All UML diagrams depict a subset of the relationships between the classes, such as inheritance (i.e. specialization), association relationships (such as aggregation and composition), and conditional features or capabilities. Some UML diagrams also show further details of the individual classes, such as their attributes and the data types used by the attributes.

2 Introduction to Location

The focus of this document is on concepts that relate to 'where' something is.

This topic is a common one, but unlike party, there are no good open models that can be leveraged.

One approach that could be used in a model is just to have location (site) name and address attributes spread throughout the model as required. The problem is that name is often not a good identifier, address formats can be quite complex and different spellings can make it hard to compare them, and then updating the data will become very complex.

The best approach is to factor out all location related information into a separate set of classes and to reference them from the rest of the model as required, and this is the approach taken in this document.

In the past, most location information was textual based, but now a lot more information systems also support map-based locations, and our model needs to support this too.

Another complication is that locating things 'internally' to a building is often done differently from 'outside' locations, so this model will cover both these cases.

This model covers the most common alternatives for external location:

- (named) site
- (postal) Address
- (Geospatial) position (such as GPS coordinates)

For internal locations, the model will cover:

- Local address (such as a bin location in a warehouse)
- (local) position in a cartesian reference system (often relative to an internal reference point, and more accurate than GPS)

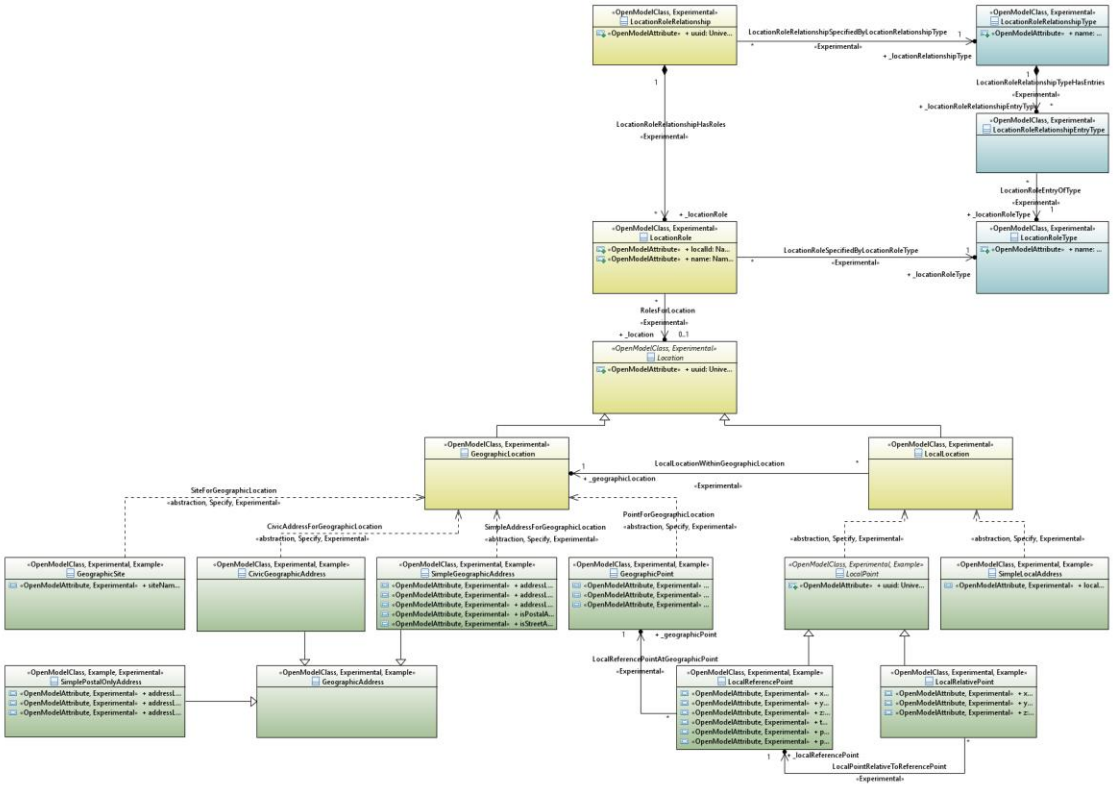
Note also that this document doesn't propose a full, robust enterprise grade location model as it is targeted at supporting a network management environment only.

Also, the lack of a single standard for postal and internal address formats precludes a simple solution, so the model focusses on providing a framework that can be augmented as required.

The model only provides simple internal and geographic address classes and is designed to be easily extended using the augments / decorates pattern, for example by adding further geographic geometries or different geographic address formats.

A data dictionary that sets out the details of all classes, data types and attributes is also provided (TR-512.DD).

3 Location model detail



CoreModel diagram: Location

Figure 3-1 Location Model

3.1 Location Model

….

3.1.1 CivicGeographicAddress

Qualified Name: Location::CivicGeographicAddress

A complex address example conforming to :
- RFC4119: "A Presence-based GEOPRIV Location Object Format"
- RFC 4776: "Dynamic Host Configuration Protocol (DHCPv4 and DHCPv6) Option for Civic Addresses Configuration Information"
- RFC5139: "Revised Civic Location Format for Presence Information Data Format Location Object (PIDF-LO)"

Note that the civic attributes haven't been added to the example class.

Inherits properties from:

- GeographicAddress

This class is Experimental.

This class is Example.

3.1.2 GeographicAddress

Qualified Name: Location::GeographicAddress

The address(es) of the GeographicLocation.

This is an example structure that could be used directly where appropriate but could be replaced with other similar structures where necessary.
This class is Experimental.

This class is Example.

3.1.3 GeographicLocation

Qualified Name: Location::GeographicLocation

A class representing locations that are located using global definitions that takes the curvature of the earth into account (often these are 'outside' locations).
A GeographicLocation can have zero, one or more decorating classes.

Inherits properties from:

- Location

This class is Experimental.

3.1.4 GeographicPoint

Qualified Name: Location::GeographicPoint

A point location as would be returned from a GPS unit.
Note that the projection etc. details aren't shown – WGS84 can be assumed.
Note that a GeographicPoint can be related to LocalReferencePoints to allow conversion between global and local reference systems.

This is an example structure that could be used directly where appropriate but could be replaced with other similar structures where necessary.
This class is Experimental.

This class is Example.

Table 1: Attributes for GeographicPoint

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
latitude	Experimental	Latitude in degrees decimal.
longitude	Experimental	Longitude in degrees decimal.
elevation	Experimental	Elevation in meters.

3.1.5 GeographicSite

Qualified Name: Location::GeographicSite

Represents a well-known named location, such as a service provider's central office or GSM base station.
This is a convenience class for locations of interest.

This is an example structure that could be used directly where appropriate but could be replaced with other similar structures where necessary.
This class is Experimental.

This class is Example.

Table 2: Attributes for GeographicSite

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
siteName	Experimental	The name of the site.

3.1.6 LocalLocation

Qualified Name: Location::LocalLocation

Represents locations that use local definitions that ignore the curvature of the earth (often these are locations inside a building or a small, flat block of land).
A LocalLocation can have zero, one or more decorating classes.

Inherits properties from:

- Location

This class is Experimental.

Table 3: Attributes for LocalLocation

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
_geographicLocation	Experimental	The GeographicLocation that this LocalLocation is defined by.

3.1.7 LocalPoint

Qualified Name: Location::LocalPoint

An abstract class that covers both absolute and relative local point representations (other geometric options could also be considered to cover regions etc.).
LocalPoints defined in a cartesian projection don't take the curvature of the earth into account as it is not significant within a building or on a small block of land.

This is an example structure that could be used directly where appropriate but could be replaced with other similar structures where necessary.
This class is abstract.

Inherits properties from:

- GlobalClass

This class is Experimental.

This class is Example.

3.1.8 LocalReferencePoint

Qualified Name: Location::LocalReferencePoint

Defines the datum point and orientation (angles theta, pi, psi from north and straight up) of the local reference system in relationship to the WGS84 geographic reference system.
LocalRelativePoints are defined as an offset from a LocalReferencePoint.
Note also that relating the LocalReferencePoint back to a GeographicPoint allows the local and geographic points to be merged into one seamless representation.
The local reference point will often be chosen for convenience, for example one corner of a building or a corner of a block of land.

Inherits properties from:

- LocalPoint

This class is Experimental.

This class is Example.

Table 4: Attributes for LocalReferencePoint

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
x	Experimental	The offset of the local datum from the geographic point location in the x direction.
y	Experimental	The offset of the local datum from the geographic point location in the y direction.
z	Experimental	The offset of the local datum from the geographic point location in the z direction.
theta	Experimental	The angle of rotation around the z-axis that the local reference system y-axis is from north, in degrees decimal.
pi	Experimental	The angle that the local reference system x-axis is from tangent, in degrees decimal. For a non-tilted floor, this is always 0.0.
psi	Experimental	The angle that the local reference system y-axis is from tangent, in degrees decimal. For a non-tilted floor, this is always 0.0.
_geographicPoint	Experimental	The GeographicPoint that this LocalReferencePoint is defined relative to.

3.1.9 LocalRelativePoint

Qualified Name: Location::LocalRelativePoint

A local point value as defined in the local cartesian reference system datum defined by the related LocalReferencePoint.

Inherits properties from:

- LocalPoint

This class is Experimental.

This class is Example.

Table 5: Attributes for LocalRelativePoint

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
x	Experimental	The distance from the local datum in the x direction in meters.
y	Experimental	The distance from the local datum in the y direction in meters.
z	Experimental	The distance from the local datum in the z direction in meters.
_localReferencePoint	Experimental	The datum that this point is relative to.

3.1.10 Location

Qualified Name: Location::Location

An abstract class that allows for decoupling of the different ways of locating something.
This class is abstract.

Inherits properties from:

- GlobalClass

This class is Experimental.

3.1.11 LocationRole

Qualified Name: Location::LocationRole

Represents location contextual behavior, in the context of a LocationRoleRelationship.
For example a central office site may be a 'trunk hub' for fiber transmission and a 'delivery

location' for given material orders.
A role can exist with no location fulfilling that role.

Inherits properties from:

- LocalClass

This class is Experimental.

Table 6: Attributes for LocationRole

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
_locationRoleType	Experimental	The specification of this LocationRole.
_location	Experimental	A location that takes this role.

3.1.12 LocationRoleRelationship

Qualified Name: Location::LocationRoleRelationship

Provides the context for related LocationRoles.
For example a central office site may be a 'trunk hub' for fiber transmission systems and a 'delivery location' for a given material order.
In the Party model, roles are expected to be a strong concept. In the location model, roles are a weaker concept and shouldn't be overused.

Inherits properties from:

- GlobalClass

This class is Experimental.

Table 7: Attributes for LocationRoleRelationship

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
_locationRelationshipType	Experimental	The specification for this relationship.

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
_locationRole	Experimental	A LocationRole in the LocationRoleRelationship. A LocationRoleRelationship is not meaningful with less than two role instances. A LocationRole cannot exist independent of a LocationRoleRelationship explaining the LocationRole with respect to one or more other LocationRoles.

3.1.13 LocationRoleRelationshipEntryType

Qualified Name: Location::LocationRoleRelationshipEntryType

Defines the LocationRole types used in a LocationRoleRelationshipType. Note that this is a use of the occurrence pattern.
This class is Experimental.

Table 8: Attributes for LocationRoleRelationshipEntryType

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
_locationRoleType	Experimental	The LocationRoleType for this relationship entry type.

3.1.14 LocationRoleRelationshipType

Qualified Name: Location::LocationRoleRelationshipType

The specification class for LocationRoleRelationship.
It allows us to define types of location relationships.

Inherits properties from:

- GlobalClass

This class is Experimental.

Table 9: Attributes for LocationRoleRelationshipType

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
_locationRoleRelationshipEntryType	Experimental	Entry type for the LocationRoleRelationshipType.

3.1.15 LocationRoleType

Qualified Name: Location::LocationRoleType

The specification class for LocationRole.

Inherits properties from:
- GlobalClass

This class is Experimental.

3.1.16 SimpleGeographicAddress

Qualified Name: Location::SimpleGeographicAddress

An address that doesn't break up the address string into all of its semantic parts.
It just opaquely represents the address as three lines of text.

This is an example structure that could be used directly where appropriate but could be replaced with other similar structures where necessary.
For example, a more complex decorator of GeographicLocation could also be added that splits up the structure of the address.

Inherits properties from:
- GeographicAddress

This class is Experimental.

This class is Example.

Table 10: Attributes for SimpleGeographicAddress

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
addressLine1	Experimental	The first line of text in the address.

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
addressLine2	Experimental	The second line of text in the address.
addressLine3	Experimental	The third line of text in the address.
isPostalAddress	Experimental	If this is a valid postal address (e.g., can be used to send a letter to).
isStreetAddress	Experimental	If this is a valid street address.

3.1.17 SimpleLocalAddress

Qualified Name: Location::SimpleLocalAddress

A local address that just opaquely represents a local address as a single line of text.
For example, a module location in a telephone exchange/central office could be a formatted string of name/value pairs like "suite:1/rack:7/chassis:2/slot:3".

This is an example structure that could be used directly where appropriate but could be replaced with other similar structures where necessary.
For example, a more complex decorator of LocalLocation could also be added that splits up the structure of the address.
This class is Experimental.

This class is Example.

Table 11: Attributes for SimpleLocalAddress

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
localAddress	Experimental	The local address is a single string.

3.1.18 SimplePostalOnlyAddress

Qualified Name: Location::SimplePostalOnlyAddress

A simple address example where the address attributes aren't parsed, but just split into three strings as it would appear in three lines on an envelope.
A PostalOnlyAddress is one that doesn't directly relate to a geographic location and may be a:
- Post office box
- Mail Bag
- Locked Bag

Or other similar special postal address.

Inherits properties from:

- GeographicAddress

This class is Experimental.

This class is Example.

Table 12: Attributes for SimplePostalOnlyAddress

Attribute Name	Lifecycle Stereotype (empty = Mature)	Description
addressLine1	Experimental	The first line of text in the address.
addressLine2	Experimental	The second line of text in the address.
addressLine3	Experimental	The third line of text in the address.

3.2 Further detail

To link the Location model into the CIM core, it makes sense to decouple the network function classes from the Location model. The network functions deliberately don't have an abstract parent (for modularity reasons) which means that the best approach is to link the inventory model to the Location model via ConstraintDomain. This decouples the two modules (reducing the number of associations) and allows inventory items to be grouped into a ConstraintDomain before relating them to a Location, reducing the number of association *instances* to be managed and also giving more sensible semantics.

Linking local and geographic locations is a common need, and roles don't really play a factor, so a special association "LocalLocationWithinGeographicLocation" is provided for that case, simplifying the number of instances required.

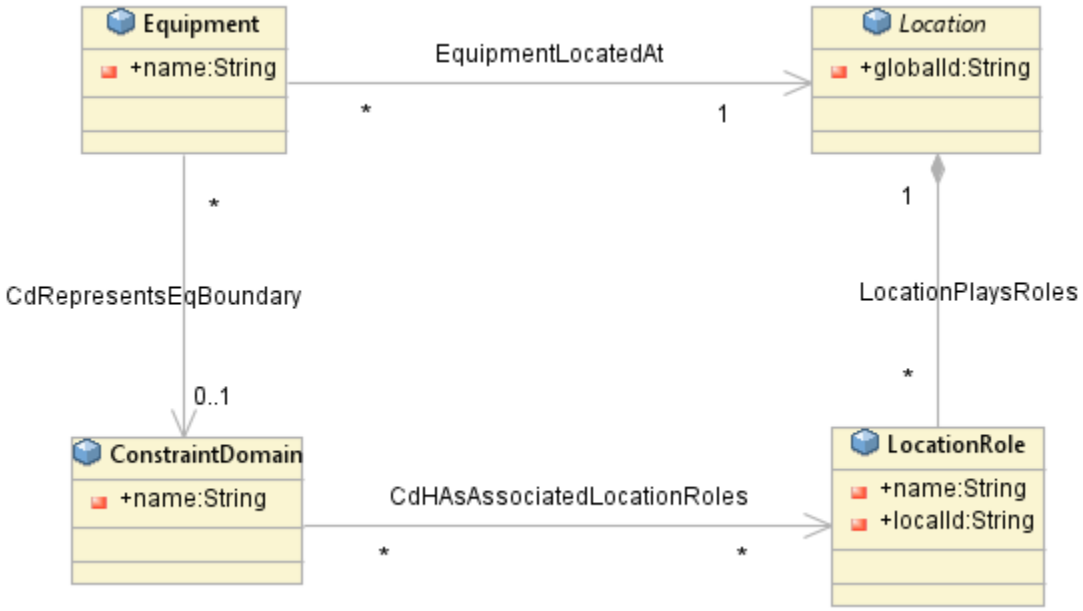


Figure 3-2 Equipment Location Association

Locating Equipment is a special case that deserves an optimized solution and for that case an association EquipmentLocatedAt is provided.

4 Location model examples

4.1 Site Contact

A site contact is the person who would be contacted to gain access to a Site.
We could create a special PartyRole of "site-contact" or we could just allow all employees to be site contacts.
The example below shows the latter option.

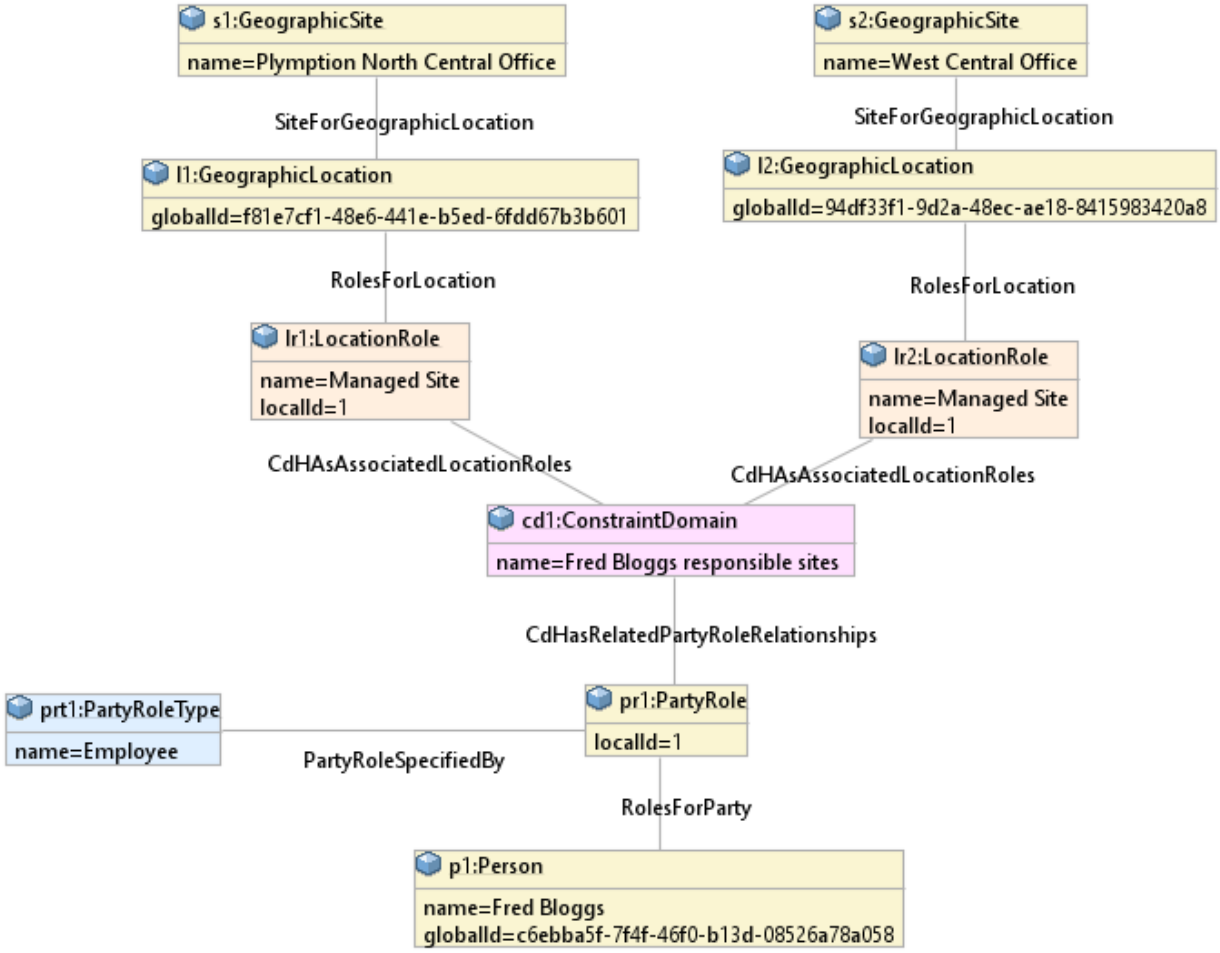


Figure 4-1 Simple Site Contact Example

If many people play roles in relation to the group of sites, then the ConstraintDomain should be made more generic, say to "Adelaide North Central Offices" and then specific site related roles can be used. This is better than creating and maintaining many equivalent groupings.

Below is shown another option.

Here, we have defined primary and secondary site contact Party roles.

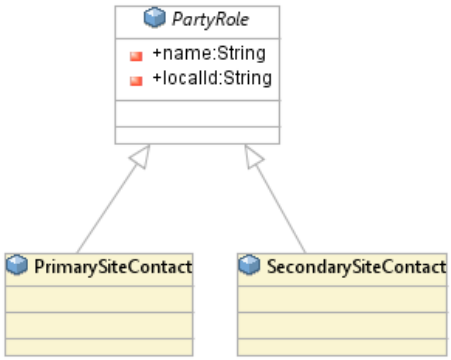


Figure 4-2 Site Contact Party Roles

So now we can show that Irma is the primary contact and Fred is the secondary contact for the "Adelaide North Central Offices" group of Locations.

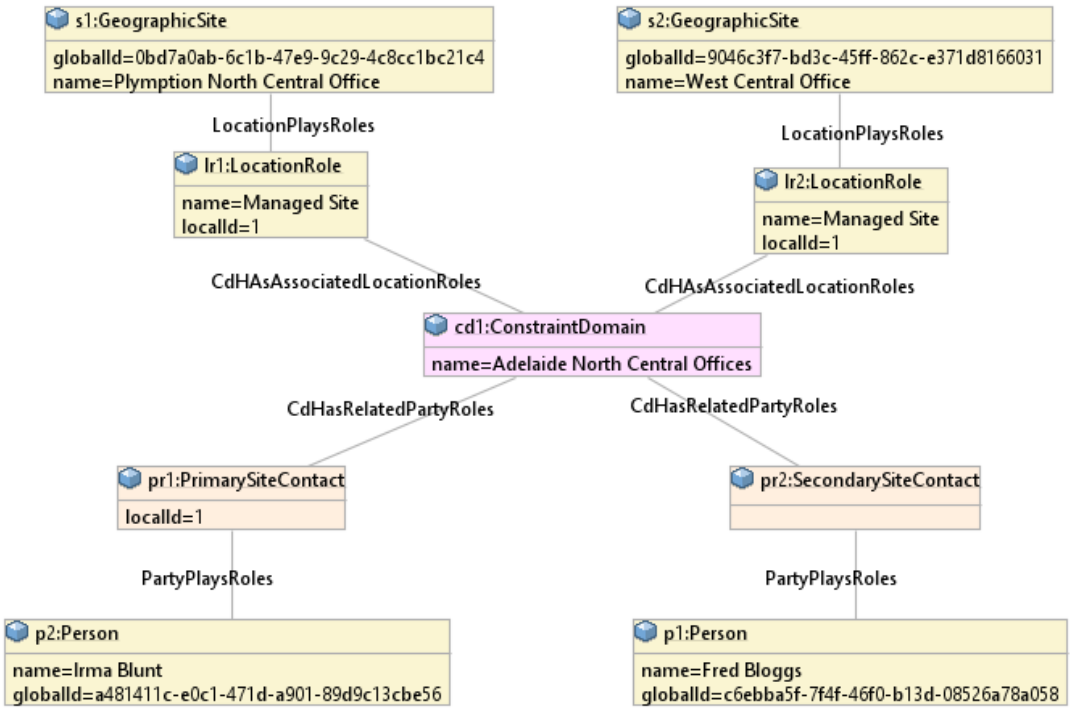


Figure 4-3 Primary and Secondary Site Contact Example

4.2 Global and Local Location Options

The global and local locations have been deliberately decoupled, so that they can be 'mixed and matched' as required.

This allows for various options, some of which are shown below.

Figure 4-4 Global and Local Address

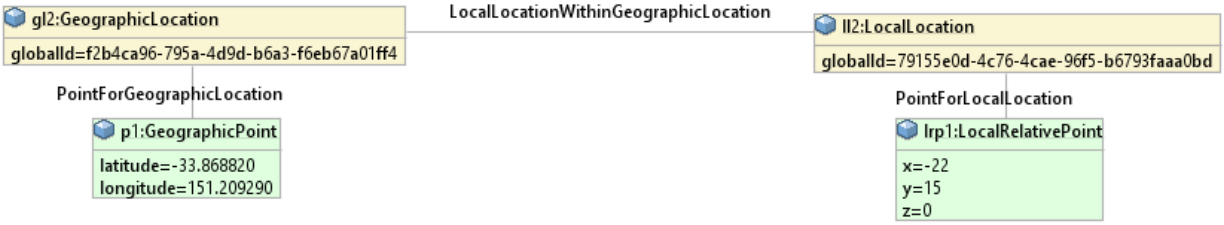


Figure 4-5 Global and Local points

In the example below, we have a GeographicSite that is related to an address and a point on a map.

Two LocalPositions are defined in the Site, one is a local reference point and the other is a local point measured relative to that local reference point.

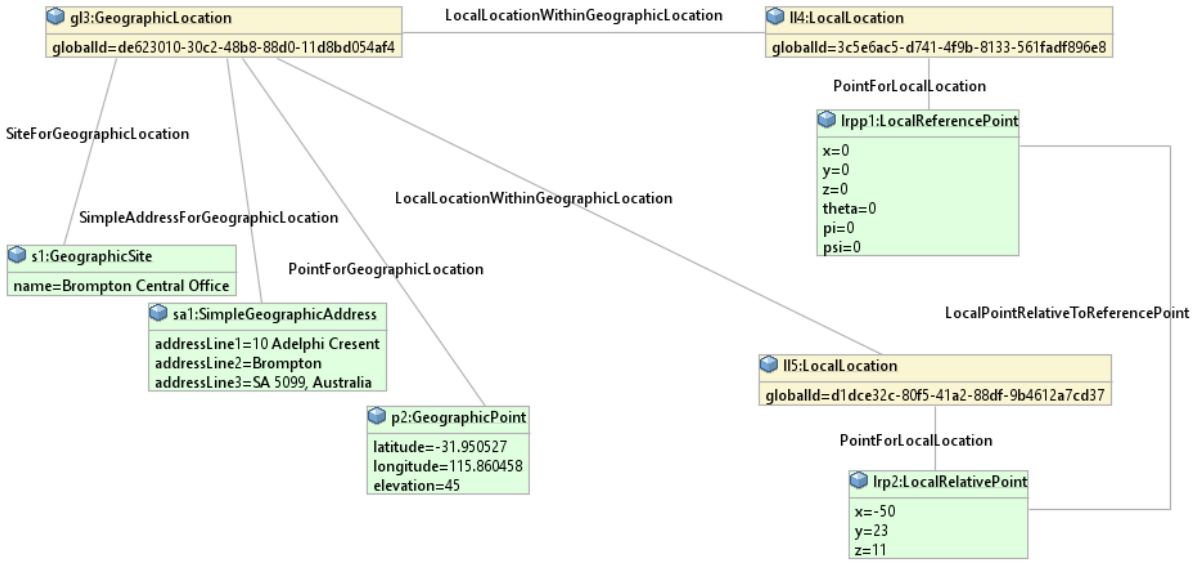


Figure 4-6 Site with local locations

4.3 Device Location

For Equipment, a special association is provided so that its representation can be optimized (without needing to use role instances).

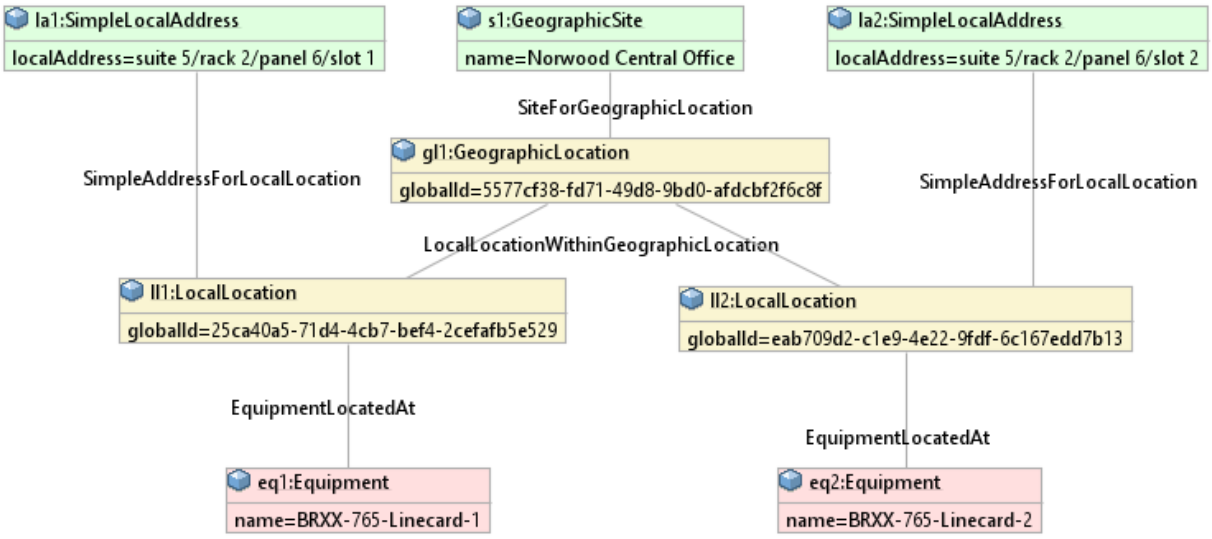


Figure 4-7 Device Location Example

Note that for SimpleLocalAddress, we are just using a single string with some sort of structure and delimiters.

If required, a more complex local address, such as one that had fields for each level in the local address could be added as another decorator of LocalLocation. The issue is that the naming is likely to be company specific and may not follow simple rules, so it can't really be added to this document.

4.4 Wifi Heat Map

A common task is to be able to show a WiFi 'heat map".

In the diagram below, assume that the grid is a 1 meter spacing and that the local reference point is at the bottom left of the diagram and that the Y axis is aligned with north and the Z axis point straight up.

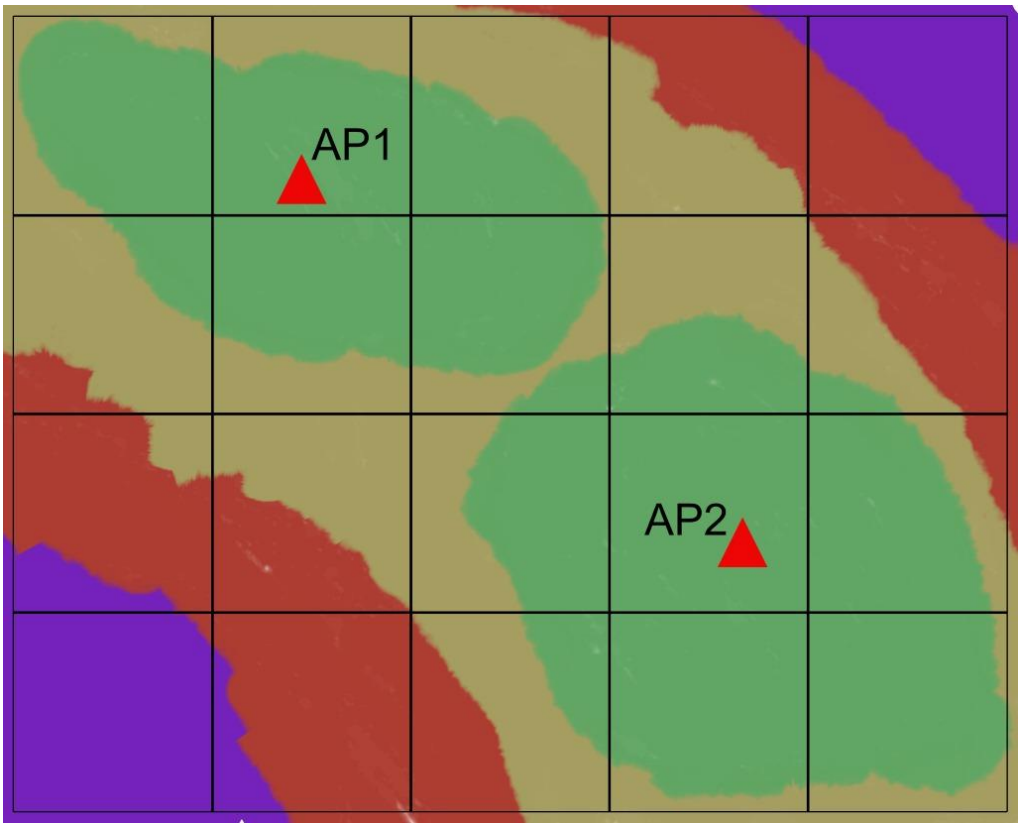


Figure 4-8 Heat Map Example

For this example, assume that the reference point is at floor level on the fifth floor of the building, and the access points are mounted on the ceiling (which is 4.0m from the floor).
Note that in this example, a structured local address of the form suite/rack/panel/slot is not appropriate, a relative position is the best way of representing where the APs are.

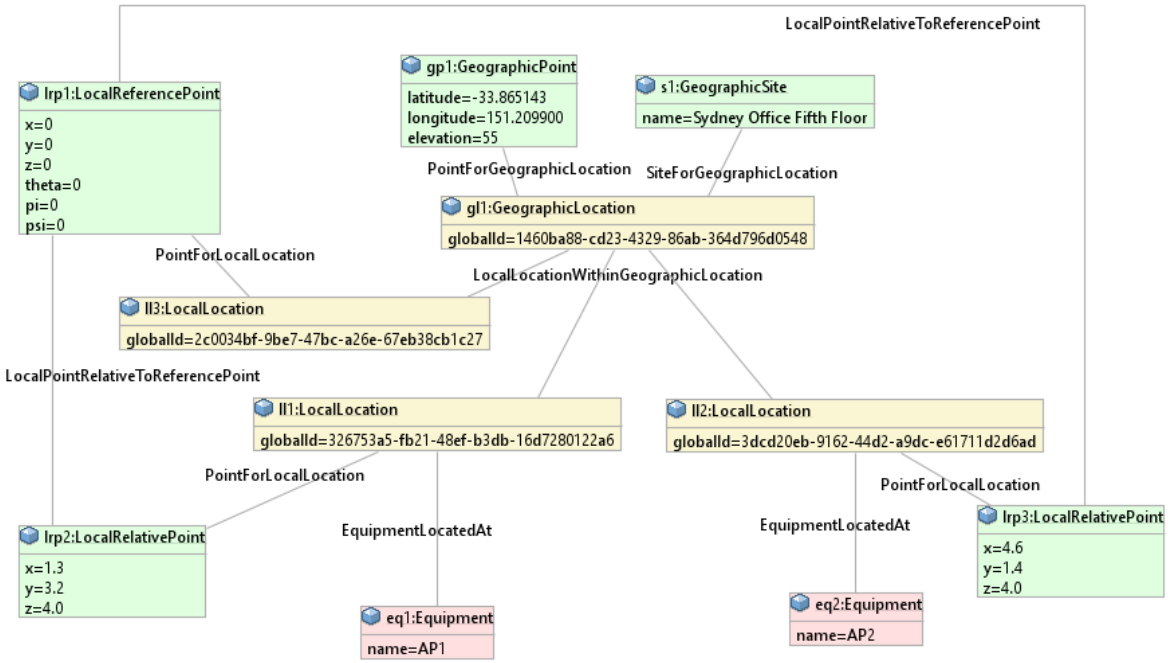


Figure 4-9 Heat Map Instance Diagram

4.5 Complex Address

As well as the current simple address, the model could be extended by adding a more detailed address class as a subclass of GeographicAddress.

For example, in MEF standard 57.1, they define a Fielded address with attributes

- Fielded Address Identifier
- Street Number
- Street Number Suffix
- Street Number Last
- Street Number Suffix Last
- Street Name
- Street Type
- Street Suffix
- Locality
- City
- Postal Code
- Postal Code Extension
- State Or Province
- Country
- Sub Unit List
- Level Type
- Level Number
- Building Name
- Private Street Number
- Private Street Name

As well as a FormattedAddress (which is similar to our SimpleGeographicAddress) with attributes

- Formatted Address Identifier
- Locality
- City
- Postal Code
- Postal Code Extension
- State Or Province
- Country
- Address Line 1
- Address Line 2

Another definition of a complex address can be found in IETF RFCs

- RFC 5774
- RFC 5139
- RFC 5491

- RFC 7459
- RFC4776

End of Document