

BAS TAG MODELLING STANDARDS

JUNE 2021 (rev 2)

Contents

Document Control.....	2
Source of truth.....	3
Spaces / Buildings / Levels / Rooms / Zones	3
Identification	3
Relationships.....	3
Tags.....	3
Asset Codes	3
Identification	3
Relationships.....	3
BAS Relationship Modelling.....	4
Definition of terms.....	4
Point.....	4
Geographical	4
Hierarchical.....	4
Asset.....	4
Cyclic Relationships.....	5
Containment Relationships.....	5
Serves Relationships.....	5
Space and Zone Modelling Concepts	7
Brick Schema Tagging Model.....	10
Minimum Required Relationships	10
Relationship Domains.....	11
Haystack	13
Minimum Required Relationships	13
Relationship Domains.....	14
BAS Tag Modelling	17
Geo Location	17
Haystack.....	17
Brick.....	17

Document Control

Date Modified	Revision	Description	Author
9/6/21	June 2021	Initial Document	JJ
17/6/21	June 2021, Rev 2	Update haystack site and campus modelling, Defined space and zone modelling rules	JJ

Source of truth

Tagging suppliers must all refer to the same source of truth database when tagging relationships between assets, across different systems. This ensures all entities maintain the same identity and cross system integration is seamless.

Spaces / Buildings / Levels / Rooms / Zones

For the purposes of this section, it is assumed that the BAS tags are running alongside an existing spatial database system.

Identification

All locations can be referenced directly by their space Management Database id.

E.g., urn:spaceManagement/000109601053 – campus 1 building 96, level 1, Room 53

E.g., urn:spaceManagement/000109601 – campus 1 building 96, level 1

E.g., urn:spaceManagement/0001096 – campus 1 building 96

E.g., urn:spaceManagement/00010 – campus

Please note the naming pattern pertaining to this example is not indicative of all location-based asset identification standards.

Relationships

Relationships between Space Management entities should not be redefined within controller tag databases.

Tags

Additional tags on Space Management entities are not permitted to be made within controller tag databases. as the Space Management database should be the source of truth for space assets.

Asset Codes

Identification

All equipment asset ids must come from an asset ID source of truth. This is to ensure that all systems referring to equipment all refer to consistent entities across multiple databases. E.g. work order systems, BAS, Space Management.

Relationships

Creating relationships between asset coded entities are allowed permitted.

BAS Relationship Modelling

Definition of terms

When creating BAS models, each entity (thing) can be placed into 1 one of the following fundamental categories:

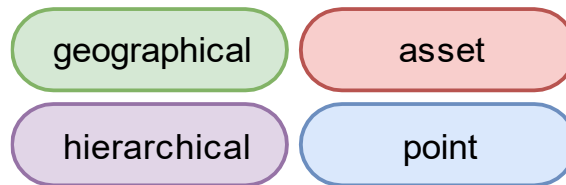


Figure 1 Fundamental BAS Tagging Entity Types

To ensure rich traversals and accurate hierarchical organization of assets the following relationship tagging models are proposed.

Point

Point entities represent a software or hardware values within a BAS driver network.

Geographical

Geographical entities represent either the abstract collection of other entities (including spaces) or physical volumes as contained by physical walls e.g.:

- Campuses are collections of buildings
- Building are collections of floors
- Floors are collections of rooms /zones

Hierarchical

Hierarchical entities represent the grouping of one or more related entities which form part of a larger system or network.

Asset

Assets represent physical equipment in the real world.

Cyclic Relationships

In most cases, Brick and haystack have 2 types of relationships, “containment” relationships and “serves” relationships.

Relationship cycle occurs when an entity can traverse 1 or more entity relationships back to itself.

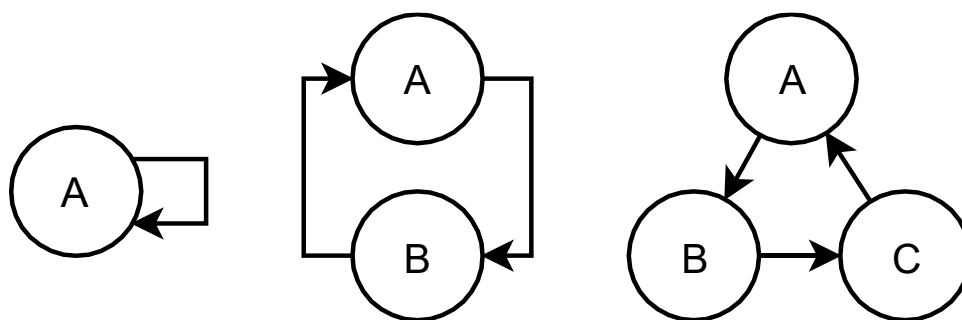


Figure 2 Examples of Cyclic relationship graphs

Containment Relationships

Containment relationships express that one entity hierarchy is part of another entity

The following relationship predicates are containment Relationships

- Brick
 - hasLocation (<https://brickschema.org/ontology/1.2/relationships/hasLocation>)
 - isPartOf (<https://brickschema.org/ontology/1.2/relationships/isPartOf>)
 - isPointOf (<https://brickschema.org/ontology/1.2/relationships/isPointOf>)
- Haystack
 - equipRef (<https://project-haystack.org/doc/lib-phIoT/equipRef>)
 - siteRef (<https://project-haystack.org/doc/lib-phIoT/siteRef>)
 - spaceRef (<https://project-haystack.org/doc/lib-phIoT/spaceRef>)

A containment relationship cycle occurs when any combination of the above containment relationships can traverse back to the same entity.

Cyclic containment relationships **are not** permitted.

Serves Relationships

Serves relationships indicate the dependence of one entity to another.

The following relationship predicates are Serves Relationships

- Brick
 - Feeds (<https://brickschema.org/ontology/1.2/relationships/feeds>)
 - Controls (<https://brickschema.org/ontology/1.2/relationships/controls>)
- Haystack

- airRef(<https://project-haystack.org/doc/lib-phloT/airRef>)
- blowdownWaterRef (<https://project-haystack.org/doc/lib-phloT/blowdownWaterRef>)
- chilledWaterRef (<https://project-haystack.org/doc/lib-phloT/chilledWaterRef>)
- condensateRef (<https://project-haystack.org/doc/lib-phloT/condensateRef>)
- condenserWaterRef (<https://project-haystack.org/doc/lib-phloT/condenserWaterRef>)
- deviceRef (<https://project-haystack.org/doc/lib-phlct/deviceRef>)
- domesticWaterRef (<https://project-haystack.org/doc/lib-phloT/domesticWaterRef>)
- elecRef (<https://project-haystack.org/doc/lib-phloT/elecRef>)
- fuelOilRef (<https://project-haystack.org/doc/lib-phloT/fuelOilRef>)
- gasolineRef (<https://project-haystack.org/doc/lib-phloT/gasolineRef>)
- hotWaterRef (<https://project-haystack.org/doc/lib-phloT/hotWaterRef>)
- makeupWaterRef (<https://project-haystack.org/doc/lib-phloT/makeupWaterRef>)
- naturalGasRef (<https://project-haystack.org/doc/lib-phloT/naturalGasRef>)
- networkRef (<https://project-haystack.org/doc/lib-phlct/networkRef>)
- refrigerantRef (<https://project-haystack.org/doc/lib-phloT/refrigRef>)
- steamRef (<https://project-haystack.org/doc/lib-phloT/steamRef>)
- submeterOf (<https://project-haystack.org/doc/lib-phloT/submeterOf>)
- weatherStationRef (<https://project-haystack.org/doc/lib-phloT/weatherStationRef>)

Cyclic Serves relationships **are** permitted.

Space and Zone Modelling Concepts

Space management systems often do not define the BAS zones contained within, often delegating the concept of zones, e.g., “HVAC zone”, “Lighting zone” to the BAS.

When defining zone and space containment relationships, often the model falls into one of 2 configurations:

- One space containing one or more zone(s)
- Two or more spaces sharing one or more zone(s)

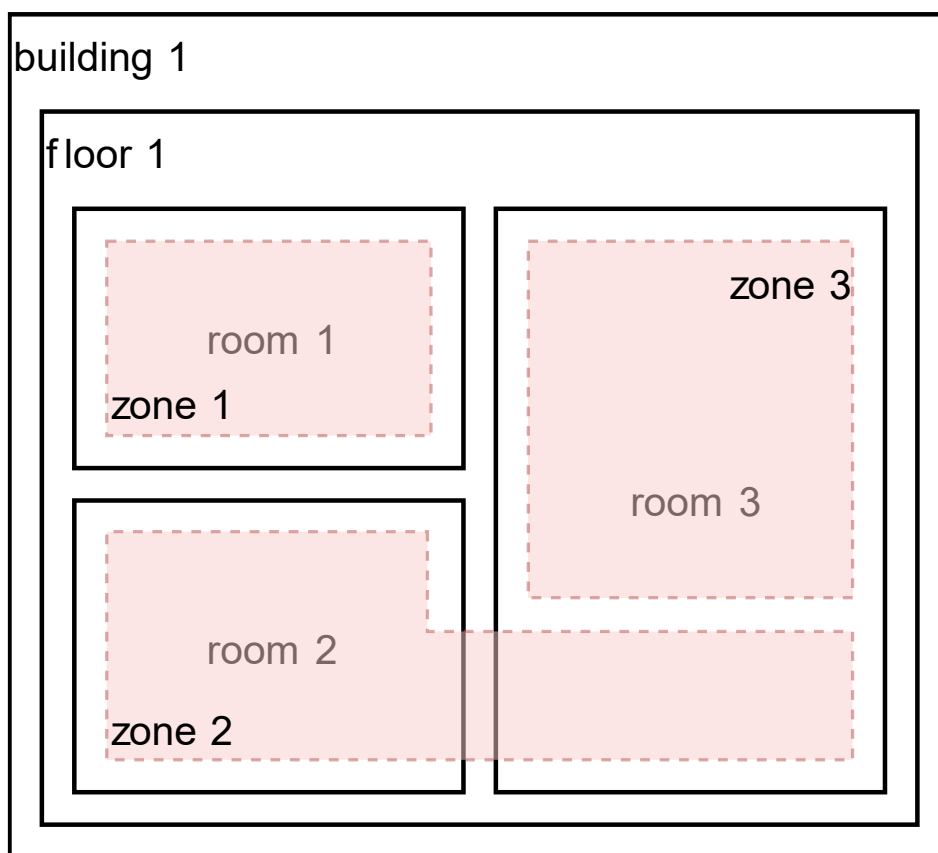
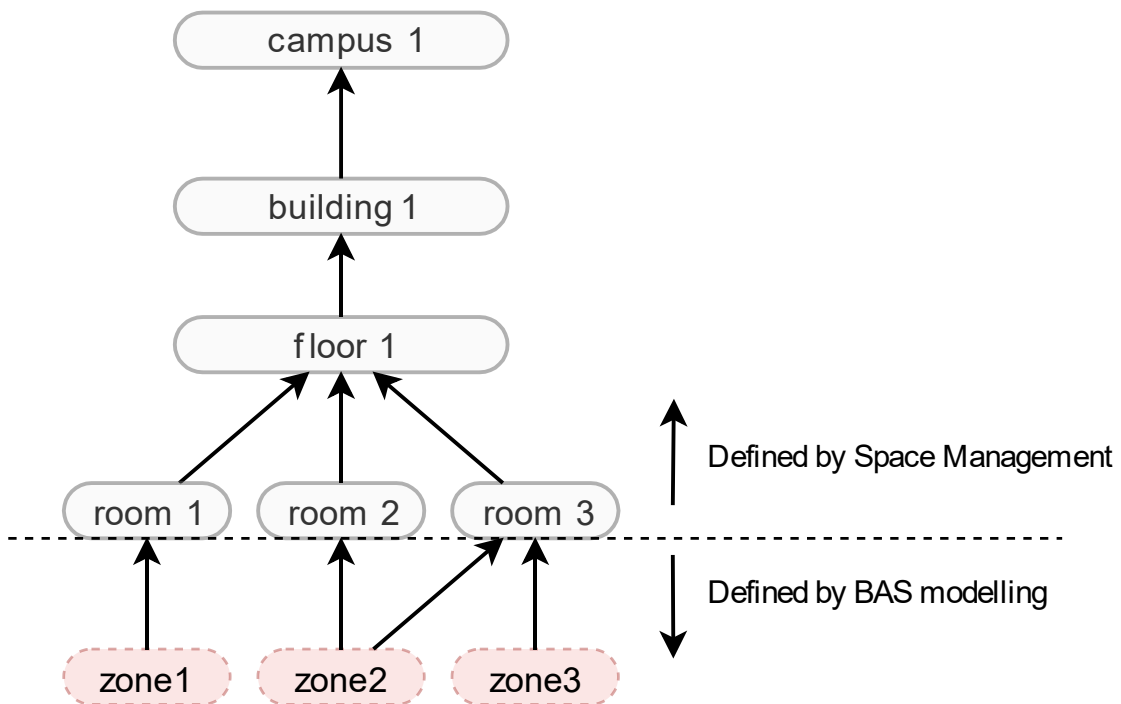


Figure 3 example Zone and space configuration

In the example shown in figure 3, zone 2 is shared by rooms 2 and 3. When modelling this topology, either the zone or the rooms could act as the “root node” when modelling the containment relationships.



If space nodes are always the parents for zones, the space management source of truth is respected and redefinitions of space relationships between the spaces are preserved.

Figure 4 Space nodes are always the roots

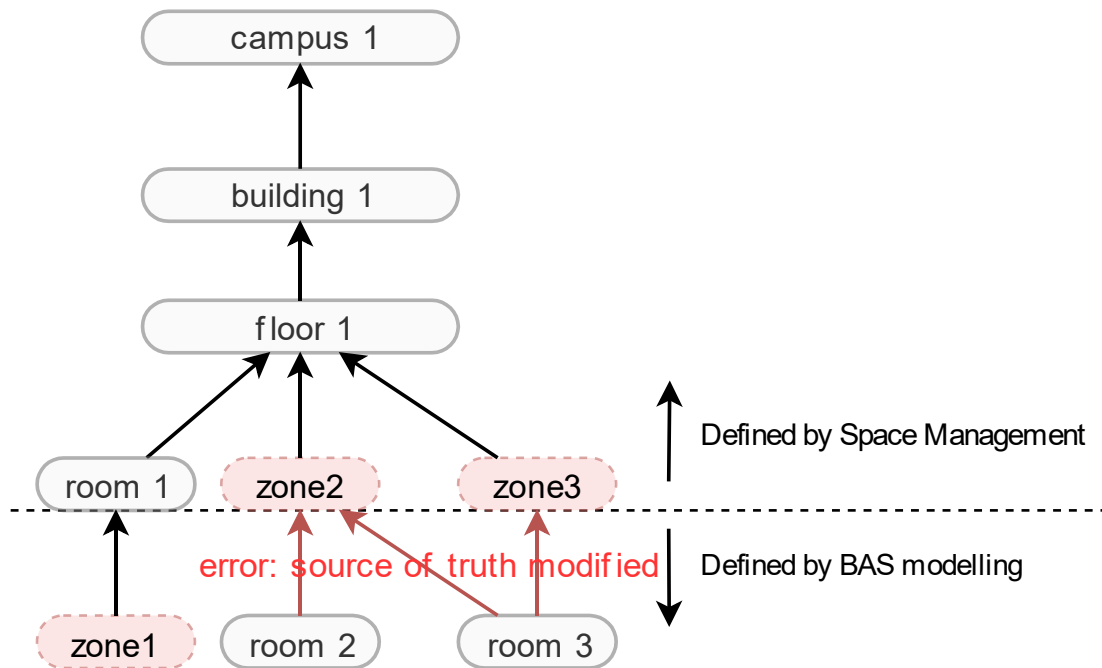


Figure 5 Nearest common ancestor is always the root

If topological sorting is priorities first, as shown in figure 5 e.g., zone 2 acting as the “parent” containing node for its two child nodes, it may cause invalid redefinitions of space relationships as originally defined in the source of truth.

Therefore, in order to respect the source of truth as managed by space management models, **zones should always be children under spaces**, if the zones are defined from within the BAS model.

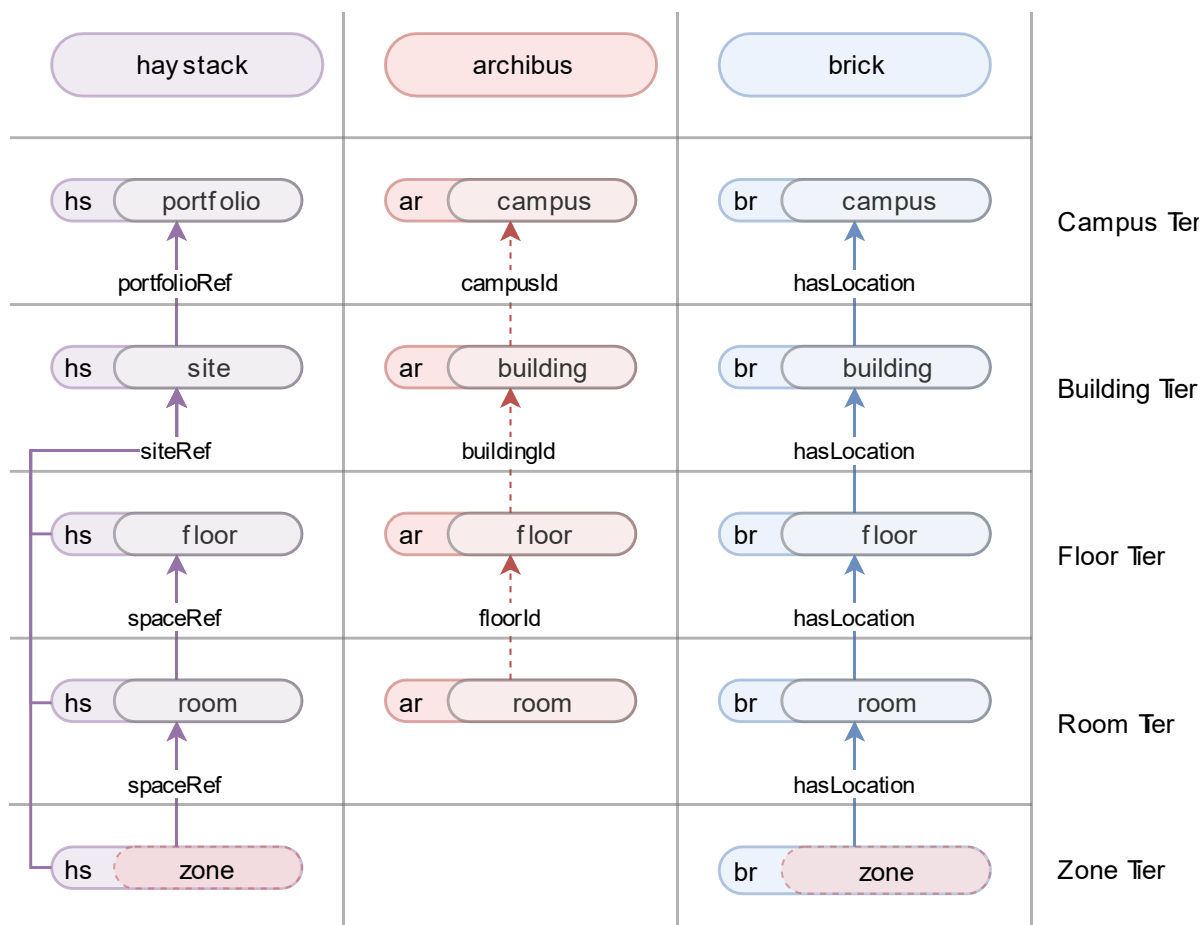


Figure 6 structure of zones and spaces within containment graphs

Brick Schema Tagging Model

Minimum Required Relationships

Brick relationship models must follow the following rules:

- All buildings must be located in a site instance.
- All floors must be located in a building instance.
- Each building must have at least 1 floor.
- Any other spaces associated with the building can only be associated to their appropriate floors.
- All systems and equipment must be associated with a floor or a space within a floor.
- All points must be associated with a system, equipment, and /or location.

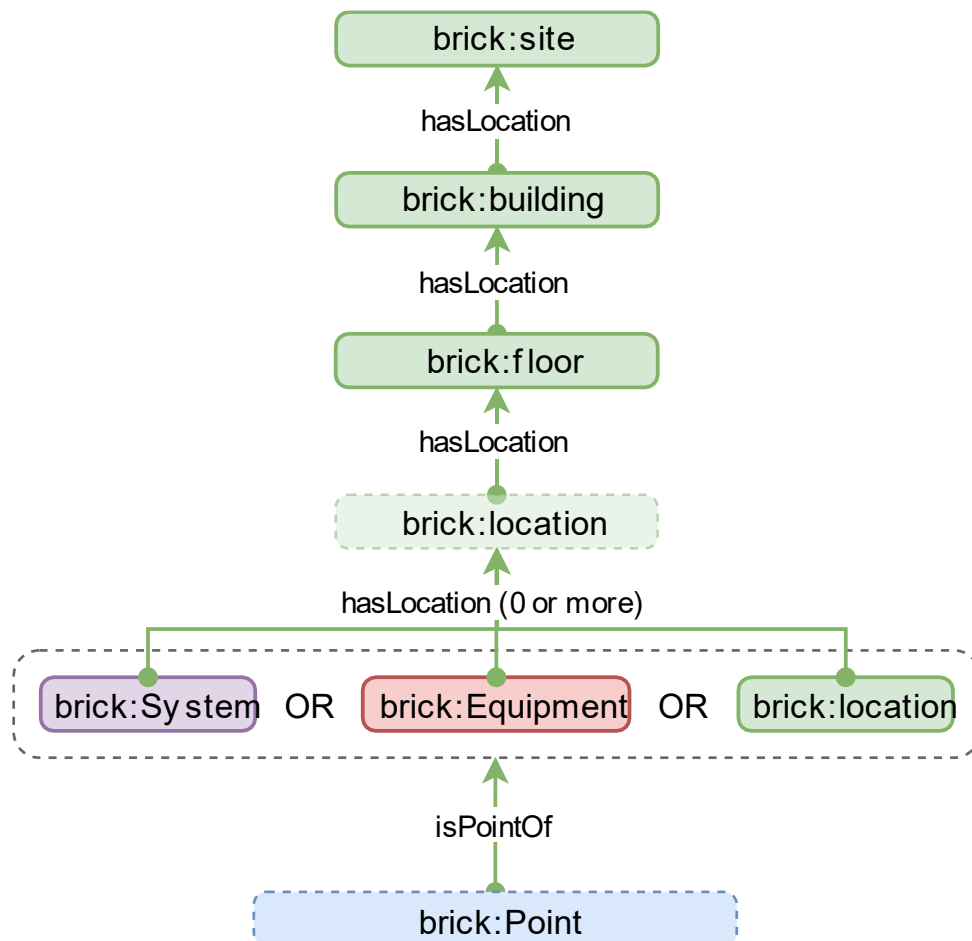


Figure 7 Brick Minimum Relationship tagging

Relationship Domains

To ensure consistency of relationships between brick models, the following constraints on relationship domains are proposed to allow for “build once, run anywhere” applications for BAS tags.

Feeds (<https://brickschema.org/ontology/1.2/relationships/feeds>)

		target			
		location	system	Equipment	point
source	Location				
	System	TRUE	TRUE	TRUE	
	equipment	TRUE	TRUE	TRUE	
	point				

hasLocation (<https://brickschema.org/ontology/1.2/relationships/hasLocation>)

		target			
		location	system	Equipment	point
source	Location	TRUE			
	System	TRUE			
	equipment	TRUE			
	point				

isPartOf (<https://brickschema.org/ontology/1.2/relationships/isPartOf>)

		target			
		location	system	Equipment	point
source	Location				
	System		TRUE	TRUE	
	equipment		TRUE	TRUE	
	point				

Controls (<https://brickschema.org/ontology/1.2/relationships/controls>)

		target			
		location	system	Equipment	point
source	Location				
	System				
	equipment				
	point				TRUE

isPointOf (<https://brickschema.org/ontology/1.2/relationships/isPointOf>)

		target			
		location	system	Equipment	point
source	Location				
	System				
	equipment				
	point	TRUE	TRUE	TRUE	

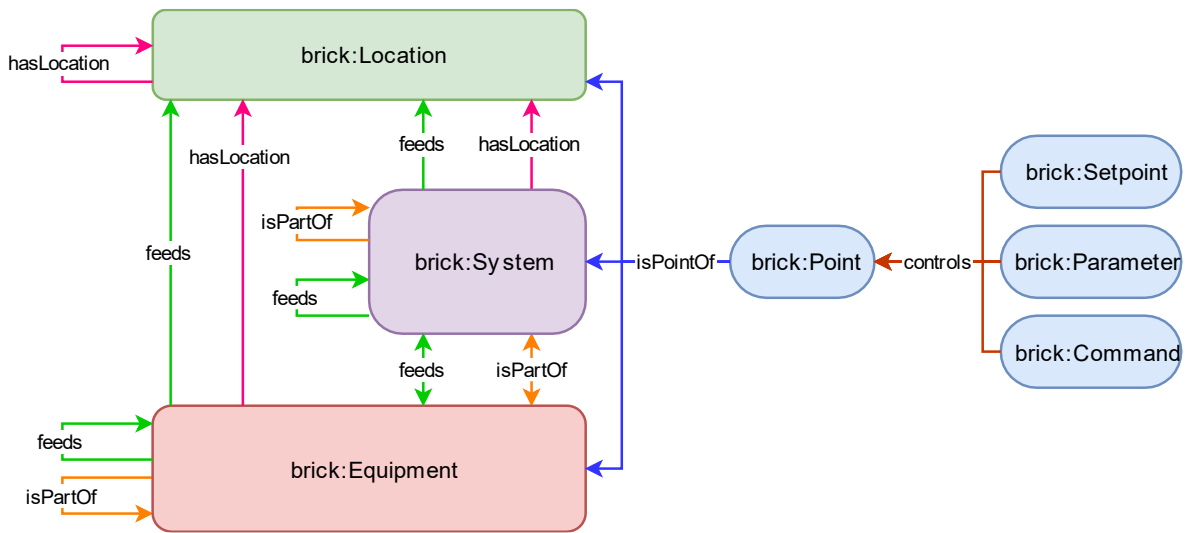


Figure 8 Brick Relationship Domains

Haystack

Minimum Required Relationships

- All buildings must be associated with a portfolio subclass instance with the portfolioRef tag.
- All floors must be associated with a building instance with a siteRef tag.
- All buildings must have at least one floor.
- All other spaces within a building must be associated with its nearest floor or space via the spaceRef tag.
- Devices and equipment must have either a (deviceRef or Equipref) OR to a spaceRef.
- Points must have a deviceRef, equipRef or SpaceRef.
- All entities sharing a common ancestor with a site must have a siteRef tag relating it to the site.

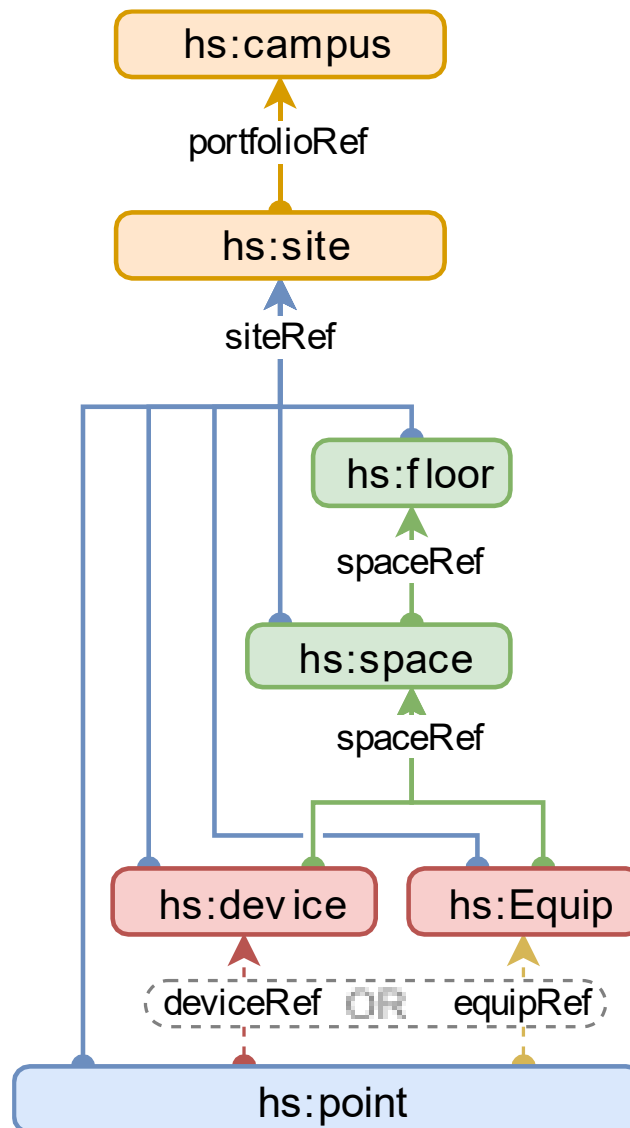


Figure 9 Haystack Minimum Relationship Requirements

Relationship Domains

portfolioRef

		target	
		site	portfolio
source	Site		True
	portfolio		True

siteRef (<https://project-haystack.org/doc/lib-phIoT/siteRef>)

		target						
		site	space	equip	device	system	network	point
source	Site	True						
	Space	True						
	Equip	True						
	Device	True						
	System	True						
	Network	True						
	Point	True						

spaceRef (<https://project-haystack.org/doc/lib-phIoT/spaceRef>)

		target						
		site	space	equip	device	system	network	point
source	Site							
	Space		True					
	Equip		True					
	Device		True					
	System		True					
	Network							
	Point		True					

deviceRef (<https://project-haystack.org/doc/lib-phIoT/deviceRef>)

		target						
		site	space	equip	device	system	network	point
source	Site							
	Space							
	Equip				True			
	Device				True			
	System							
	Network							
	Point				True			

equipRef (<https://project-haystack.org/doc/lib-phIoT/equipRef>)

		target

		site	space	equip	device	system	network	point
source	Site							
	Space							
	Equip			True				
	Device			True				
	System							
	Network							
	Point			True				

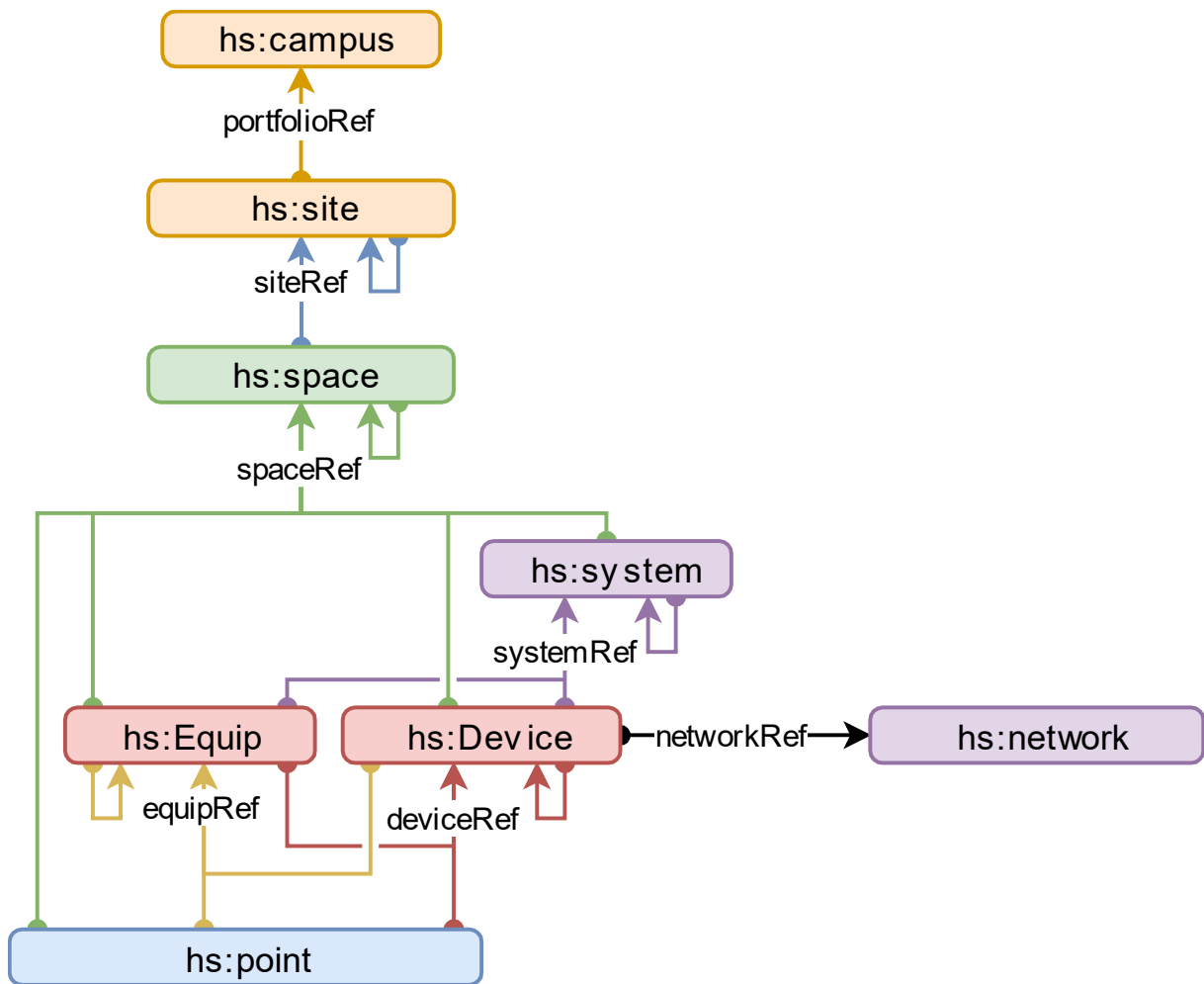


Figure 10 Haystack Relationship Domains

BAS Tag Modelling

Geo Location

Haystack

All haystack equipment and devices with a spaceRef to a space must have the geoCoord tag associated with it, indicating the entity's location on the floorplan

<https://project-haystack.org/doc/lib-ph/geoCoord>

Brick

All Brick equipment instances tagged with a hasLocation predicate must have a GPS geolocation tag associated with it.

Brick does not have an ontological definition of geolocation, however, the w3c wgs84 geo schema has the following predicate to denote geolocation.

http://www.w3.org/2003/01/geo/wgs84_pos#lat_long