

# Innovator Data Model Graph View

V1 (April 2020)

# Overview

- **Purpose:**  
allows Administrators to graphically view the data model of an ItemType with all its Relationships, related Items Types and Item Property links.
- This solution is targeted for Aras Innovator Administrators only.
- 2 new Actions are introduced on Item Type „Item Type“.
- Each starts a different Graph View based on a different Query Definition
  - Graph View: „**Innovator Data Model Down**“ will resolve relationships to other Item Types recursively. And it will also resolve all item Property to their target Item Types.
  - Graph View: „**Innovator Data Model Surround**“ (360°) will resolve relationships to other Item Types one level down and one level up (where used). And it will also resolve all item Property to their target Item Types and item Properties pointing to this Item Type with their source Item Types.
- From the Graph View the displayed ItemTypes, Relationships, and Item Properties can be opened using right click action “Open Item”.
- If on an ItemType block, another right click action “Open Graph View” allows to start this GraphView again starting with the selected ItemType.

ItemTypes

Search

Clear

Simple

Name	Singular Label	Plural Label	Version
part			
Part	Part	Parts	

Open

Add to Favorites

Claim

Promote

Properties

Save As

Delete All Versions

Structure Browser

Where Used

Reset Item Access

Reset Life Cycle State

Set Default Life Cycle

Add to Package Definition

ItemType Definition Report

ItemType Permissions Report

ItemType History Configuration

Show Related Item Types Data Model

Show Surrounding Item Types Data Model

Graph View: „Innovator Data Model Down“ will resolve relationships to other Item Types recursively. And it will also resolve all item Property to their target Item Types.

Graph View: „Innovator Data Model Surround“ will resolve relationships to other Item Types 1 level down and 1 level up (where used). And it will also resolve all item Property to their target Item Types and item Properties pointing to this Item Type with their source Item Types.

# Graph View Cards

NOTE:  
Each RelationshipType will always show its corresponding ItemType of the same name when expanded (but with the background color of RelationshipTypes)

## Item Types

ItemType	<b>Part</b>
Labels:	Part (Parts)
Versions:	1 - 0
Is Relationship:	0
Table or Poly:	Table

Singular and (Plural) Labels

Configured icon on this ItemType (open\_icon)

Combination of  
„is\_versionable“ and „manual\_versioning“  
0- means versions are turned off  
1-0 means versions are turned on with automatic versioning  
1-1 means versions are turned on with manual versioning

If set to „polymorphic“, it means this is a Poly Item Type

If set to „1“, it means this is the item type that belongs to a relationshiptype of the same name

## Relationship Types

RelationshipType	<b>Part PSW</b>
Tab Label:	Part Submission Warrants
Behavior:	float
Hide in All:	0
Grid View:	0

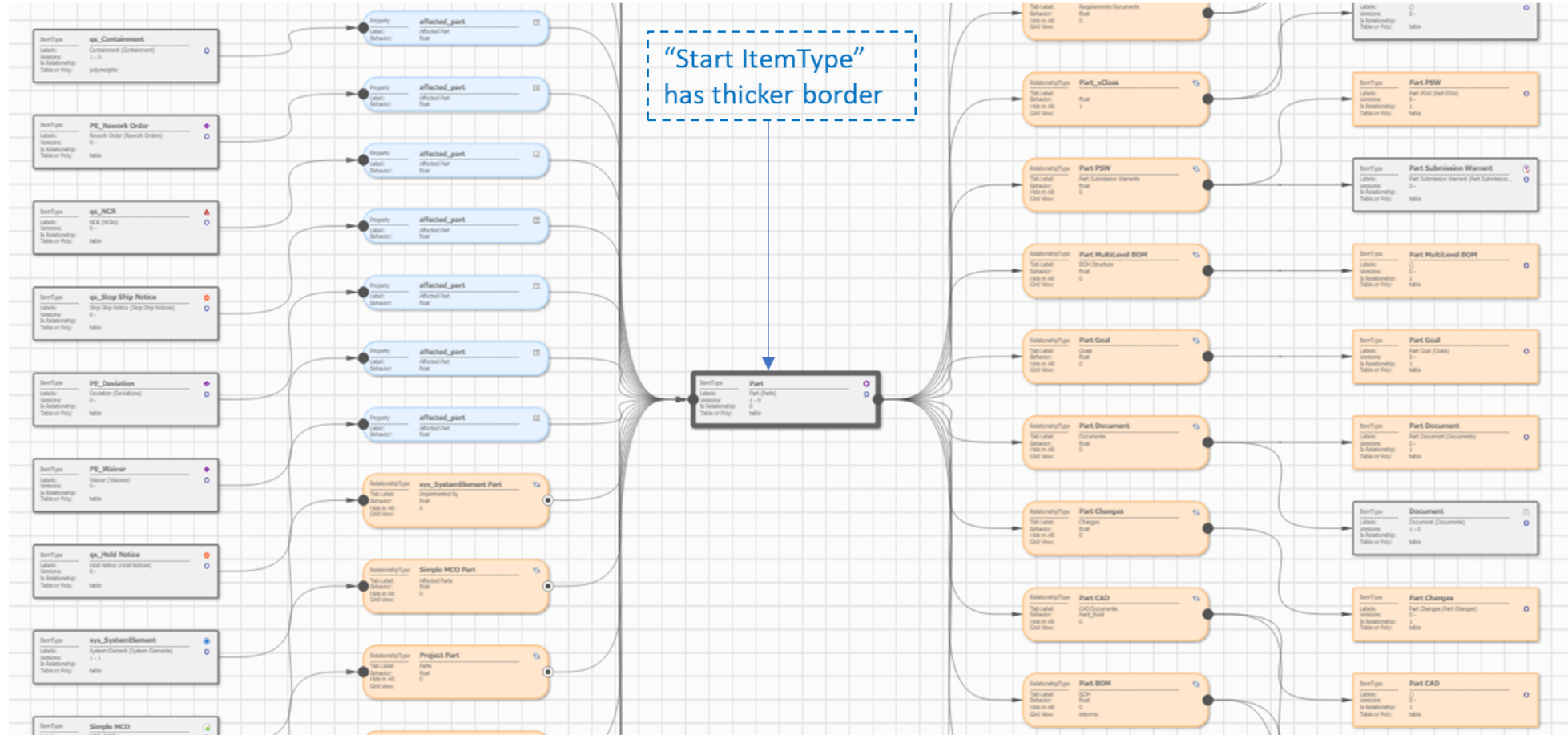
If set to „1“, it means this is a hidden Relationship

## Item Properties

Property	<b>affected_part</b>
Label:	Affected Part
Behavior:	float

“where used (up)” elements are displayed to the left of the “Start ItemType”

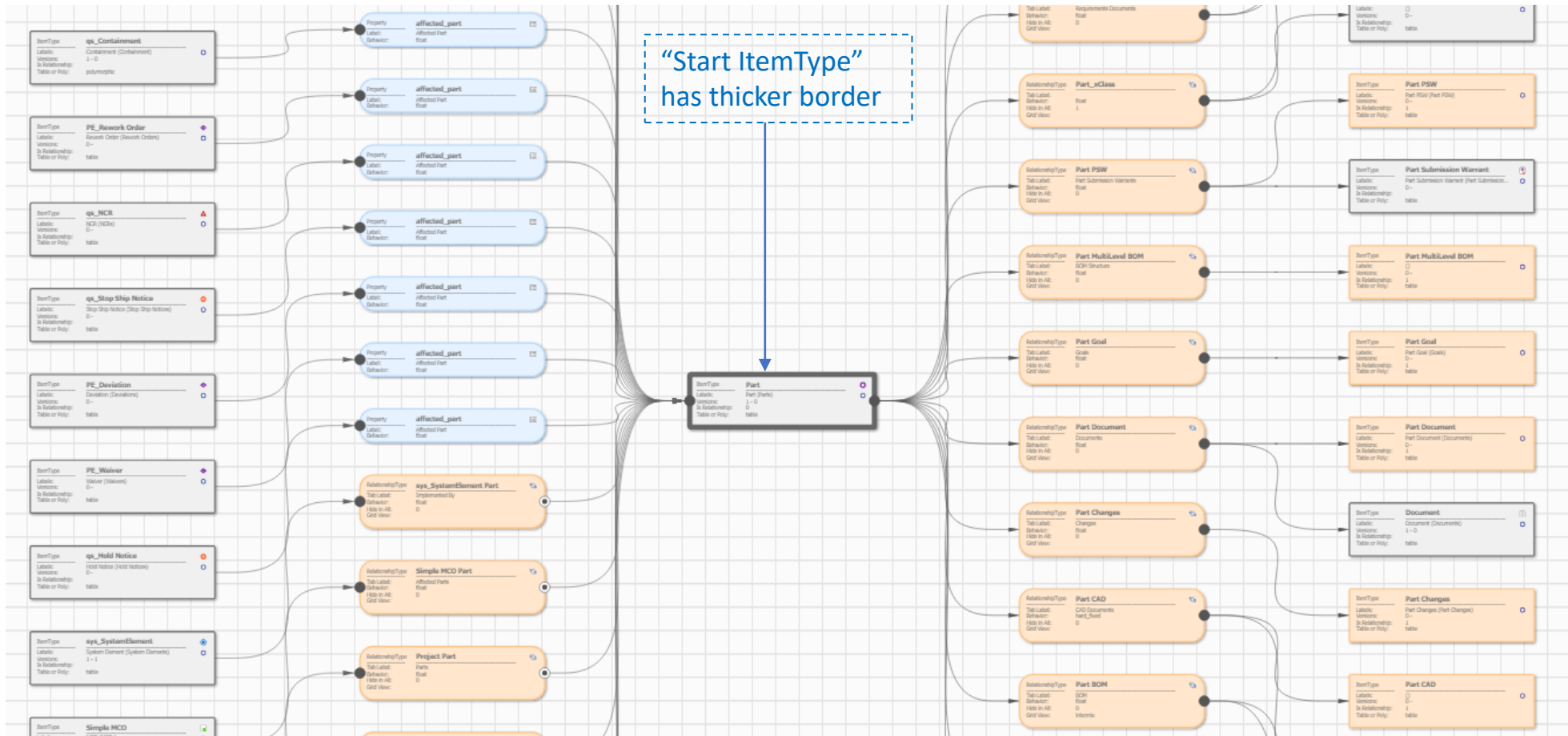
“down” elements are displayed to the right of the “Start ItemType”



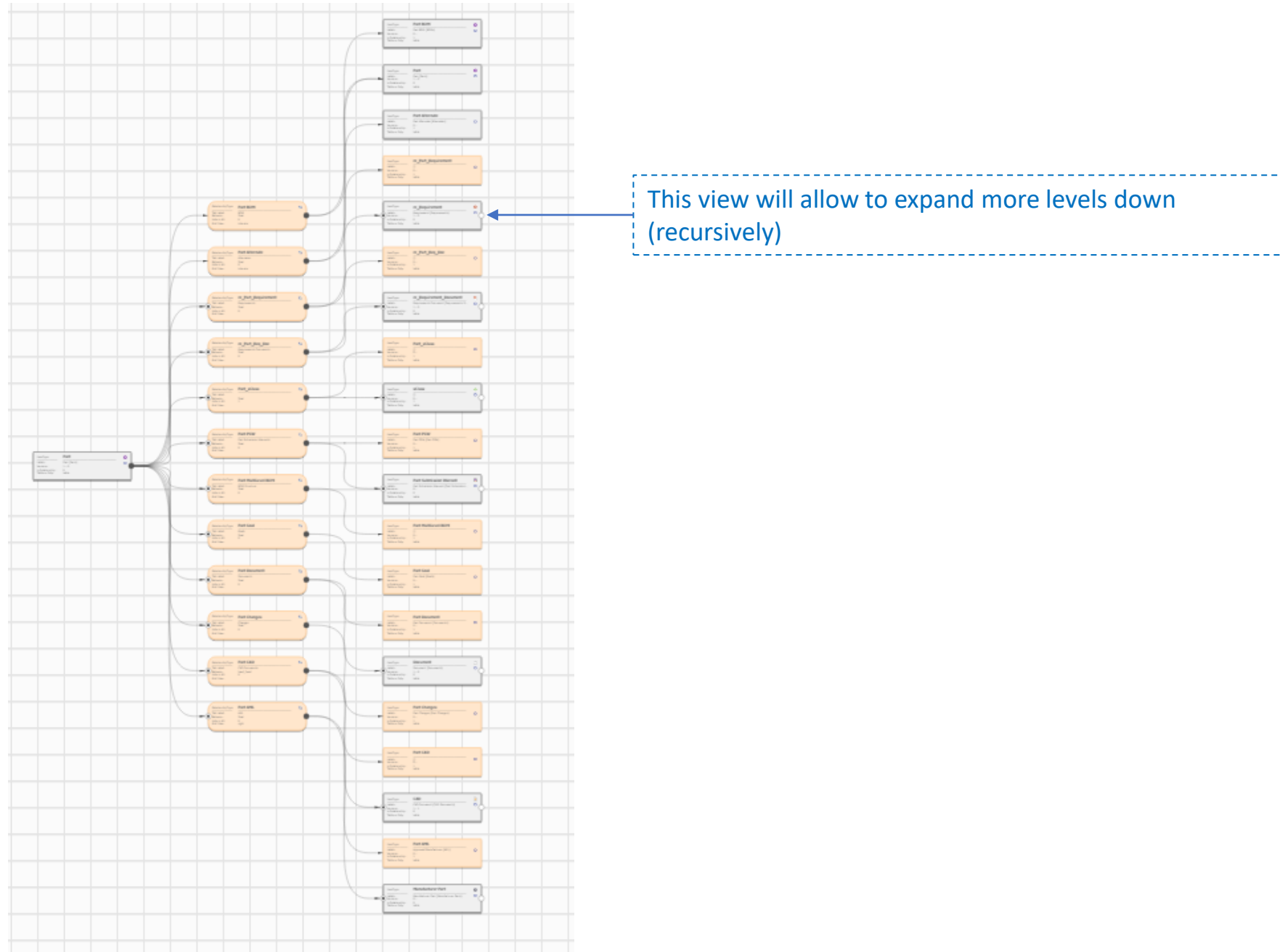
# Where Used – Connectors

“where used (up)” elements are displayed to the left of the “Start ItemType”

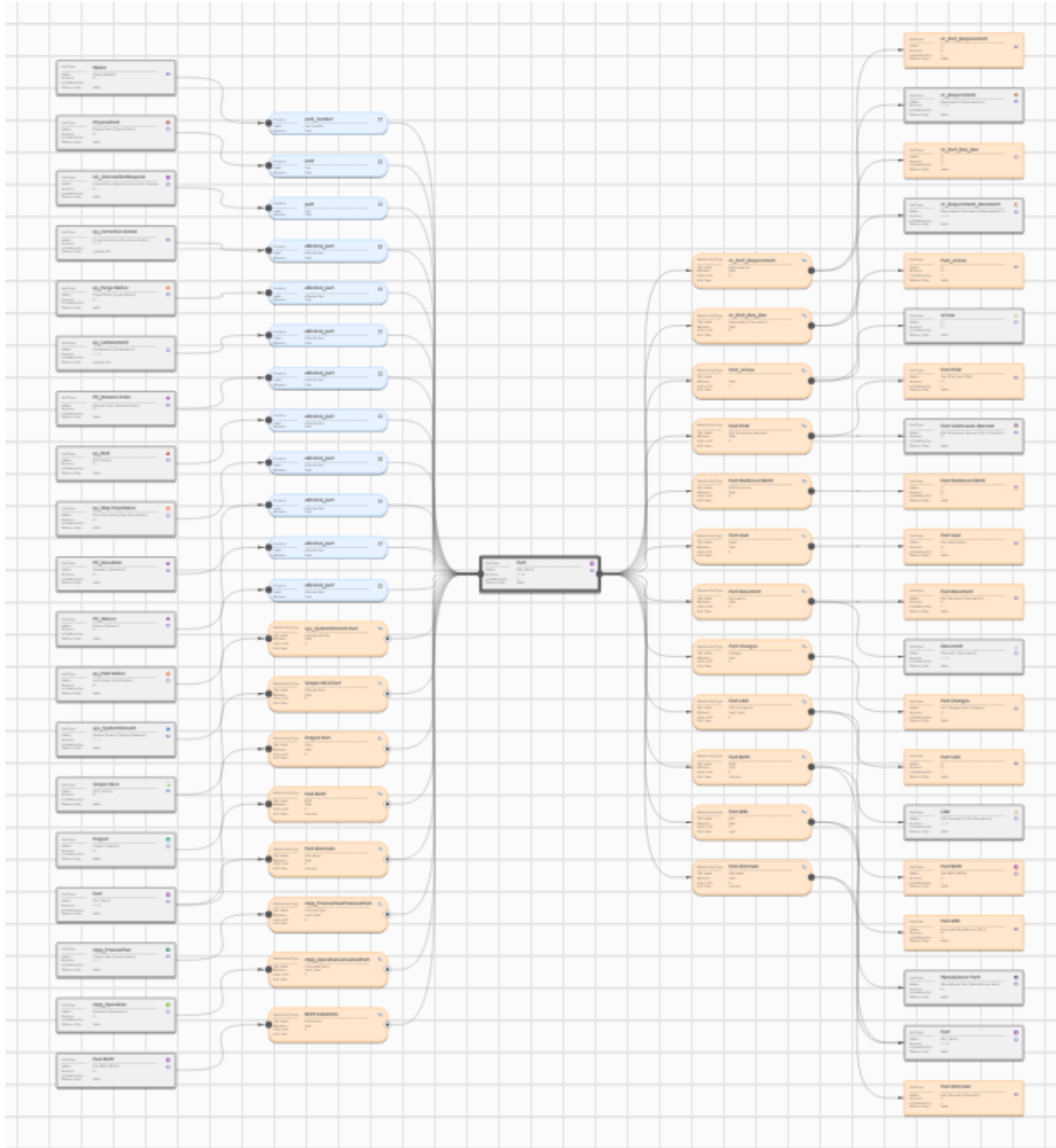
“down” elements are displayed to the right of the “Start ItemType”



# Sample of „Show Related Item Types Data Model”



# Sample of „Show Surrounding Item Types Data Model”



|| This view will only resolve to 1 level up and 1 level down



# Innovator Data Model Graph View

Implementation Details

# Query1 Defintion

Where Condition Join Condition

```
[[Related ItemType] != [Source ItemType]] OR ((Count([[Related]] ItemType) = 0)
```

## IMPORTANT:

If source and related item type are of same type (like "Part BOM" has it), then recursive resolution will not work !!

Prevent this case with this condition to allow recursion of the data model Or condition to include if related count = 0 ensures NULL relationships are in.

Where Condition Join Condition

```
[Related ItemType] = [Source ItemType]
```

Same "Related" definition for the case when source and related item type are the same. Whith this configuration these are included, but will not allow recursion of the model from here on.

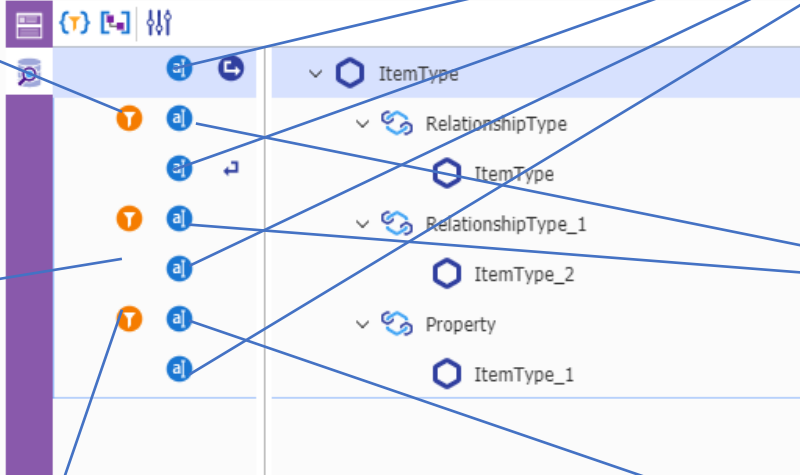
Where Condition Join Condition

```
[[Data Type] = 'item'] AND (Name != 'id') AND (Name != 'config_id') AND (Name != 'locked_by_id') AND (Name != 'created_by_id') AND (Name != 'owned_by_id') AND (Name != 'managed_by_id') AND (Name != 'current_state') AND (Name != 'team_id') AND (Name != 'modified_by_id') AND (Name != 'permission_id') AND (Name != 'source_id') AND (Name != 'related_id')
```

Filter system item properties every item type has but are not relevant to show

Innovator\_Data\_Model\_Down

Edit



Properties

Name  
Plural Label  
Implementation Type  
Manual Versioning  
Dependent  
Small Icon  
Versionable  
Singular Label

Properties

Relationship Name  
Behavior  
Tab Label  
Hide In All  
Create Related  
Grid View

Properties

Name  
Label  
Behavior

# Query2 Defintion

Use the same Property Definitions as used in Query 1 !

```
Where Condition  Join Condition  &  II  !  X
[[Data Type] = 'item' AND (Name != 'id') AND (Name != 'config_id') AND (Name !=
'locked_by_id') AND (Name != 'created_by_id') AND (Name != 'owned_by_id') AND
(Name != 'managed_by_id') AND (Name != 'current_state') AND (Name != 'team_id'
) AND (Name != 'modified_by_id') AND (Name != 'permission_id') AND (Name !=
'source_id') AND (Name != 'related_id')]
```

Filter system item properties every item type has but are not relevant to show

Innovator\_Data\_Model\_Surround

Edit

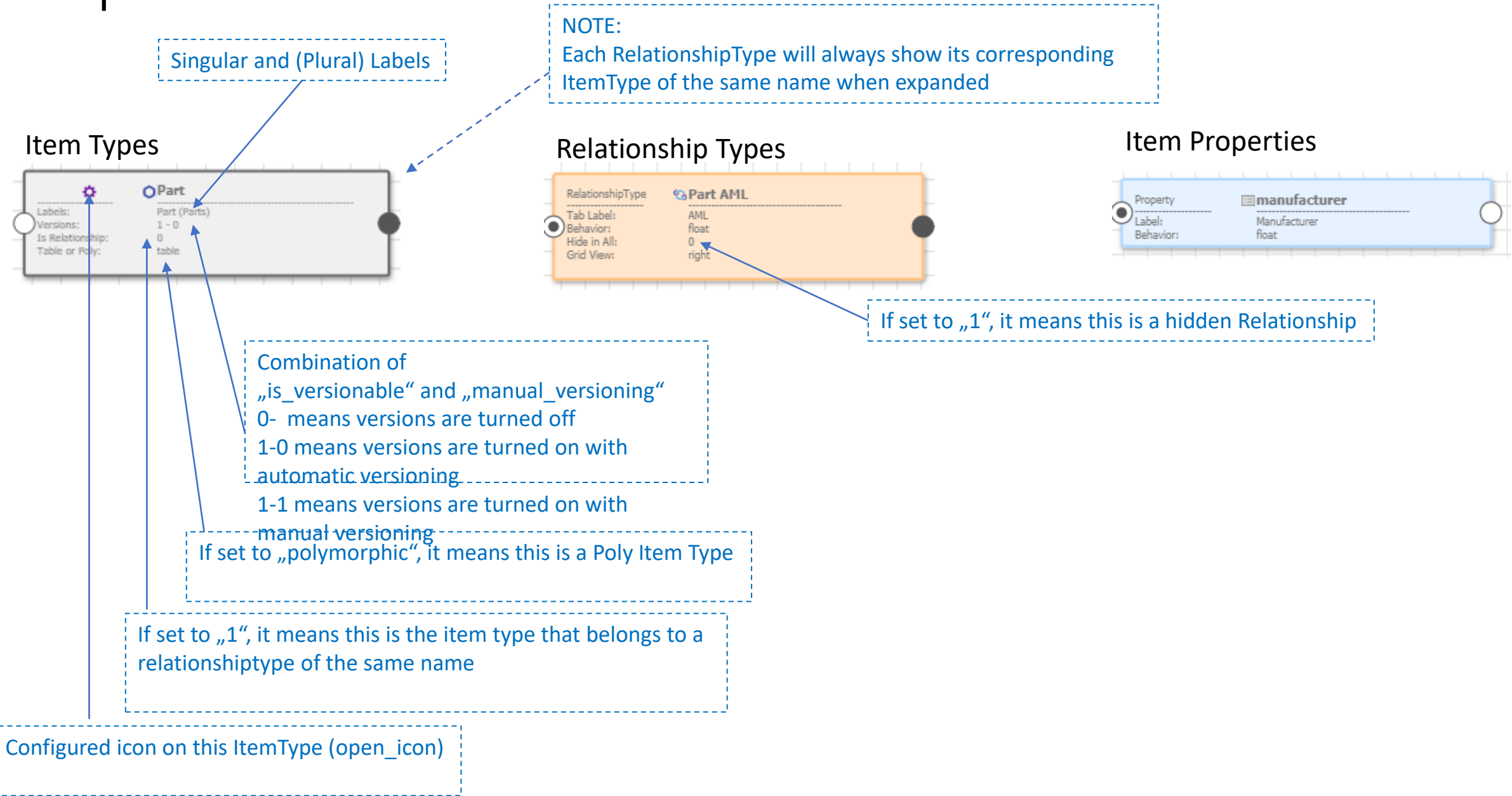
ItemType

- RelationshipType
  - ItemType\_1
- Property
  - ItemType\_3
- RelationshipType related\_id
  - ItemType\_2
- Property data\_source
  - ItemType\_4

# Backup

- Older Version Concepts

# Graph View Cards

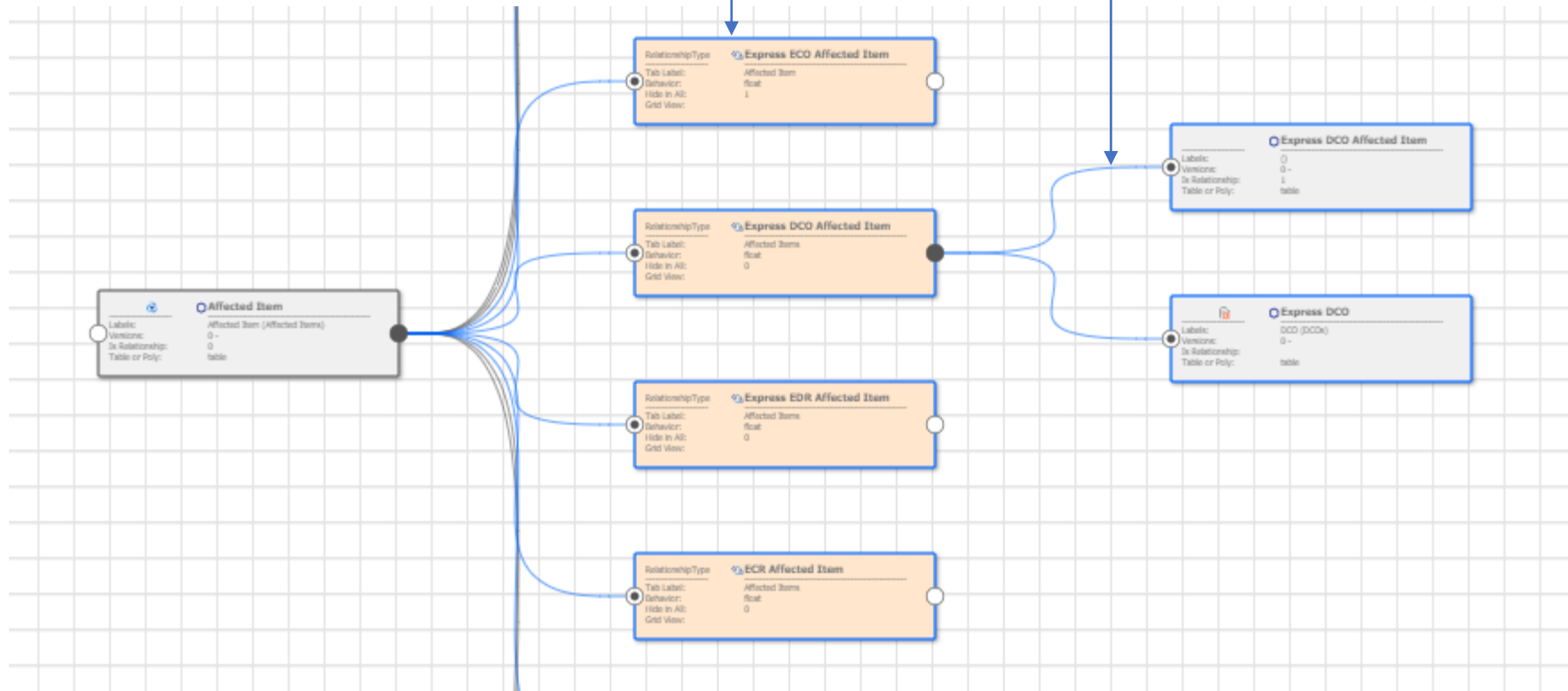


# Where Used – Connectors and Colors

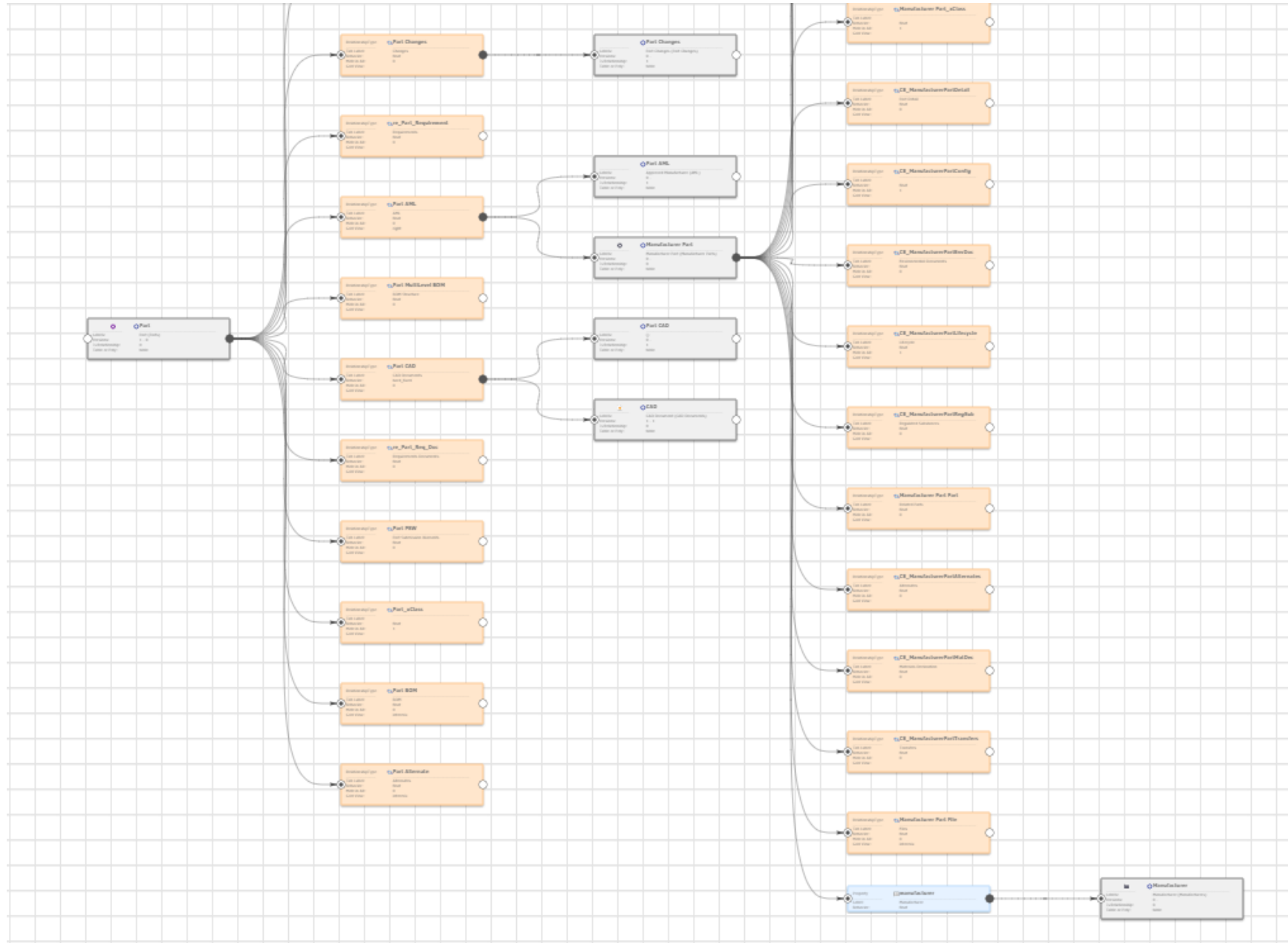
Graph View: „Innovator Data Model Surround“ applies color coding to mark the elements of “where used” (1 level up)

Blue borders indicate the “where used” elements

Blue connectors indicate the “where used” path



# Sample of „Show Related Item Types Data Model”



# Sample of „Show Surrounding Item Types Data Model”

