

# Composite structure diagrams

## On this page

- [Import elements](#)
- [Conversion details](#)
  - [Dummy class](#)
  - [Class](#)
  - [Class with attributes and operations](#)
  - [Interface](#)
  - [Part](#)
  - [Parts with Ports](#)
  - [Parts that set type to other elements](#)
  - [Port](#)
  - [Ports whose type is set to other elements](#)
  - [Port whose type is set to another element and containing Expose Interfaces](#)
  - [Collaboration](#)
  - [Expose Interface](#)
  - [Expose Interface with Ports](#)
  - [Expose Interface with Relationships](#)
  - [Package](#)
  - [Assembly](#)
  - [Dependency](#)
  - [Removed relationships](#)
- [Transformation report](#)

This section presents additional EA specific Composite Structure diagram information.

## Import elements

EA differs from MagicDraw in the Composite Structure diagram's content elements design. The table below shows the differences by focusing on the EA's elements and how they will be transformed or converted into a format MagicDraw can correctly load and display. The element names shown in the table are the same in both EA and MagicDraw's GUI. The contents in the brackets [...] are the XML element type references.

Enterprise Architect	MagicDraw
Interaction	
Class [uml:Class]	Class [uml:Class]
Interface [uml:Interface]	Interface [uml:Interface]
Part [uml:Class]	Part [uml:Property]
Part [uml:Property]	
Port [uml:Port]	Port [uml:Port]
Collaboration [uml:Collaboration]	Collaboration Use [uml:Collaboration]
Expose Interface [does not exist]	
Connector [element type does not exist]	Connector [uml:Connector]
Assembly [element type does not exist]	
Delegate [element type does not exist]	Connector [uml:Connector]
Role Binding [uml:Dependency]	Role Binding [uml:Dependency]
Represents [uml:Dependency]	Dependency [uml:Dependency]
Occurrence [uml:Dependency]	Dependency [uml:Dependency]

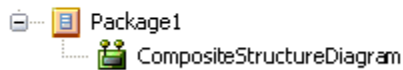
## Conversion details

### Dummy class

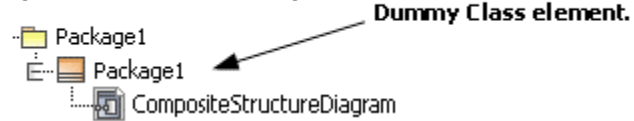
In MagicDraw, the Composite Structure diagram needs a Context element to contain itself. However, in EA there is no Context element. After conversion, a dummy Class element will be created to represent the Context element of the Composite Structure diagram.

A dummy Class will be named after the closest owner package of the Composite Structure diagram. The Composite Structure diagram and all of the Composite Structure elements will be placed inside the dummy Class element.

## EA (Before Conversion)



## MD (After Conversion)



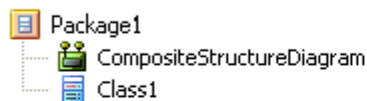
A Dummy Class.

## Class

The Class element data will be normally copied even if the Class element is drawn in a Composite Structure diagram. A Property element will be created to represent the Class element. The Property element will be named after the Class element.

## Containment tree

### EA (Before Conversion)

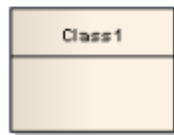


### MD (After Conversion)

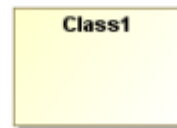


## Diagram View

### EA (Before Conversion)



### MD (After Conversion)



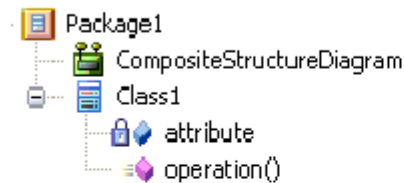
A Property element representing a Class element.

## Class with attributes and operations

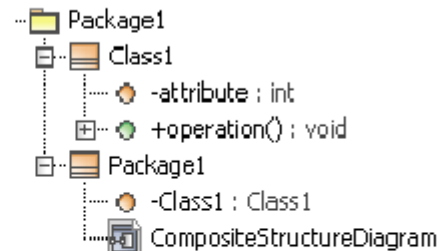
The Class view in EA will be converted to a Part in MagicDraw. Attributes and operations will not be shown in the Composite Structure diagram.

## Containment tree

### EA (Before Conversion)



### MD (After Conversion)

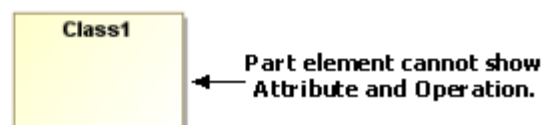


## Diagram View

### EA (Before Conversion)



### MD (After Conversion)



Part element cannot show Attribute and Operation.

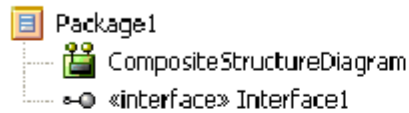
A Part element after conversion.

## Interface

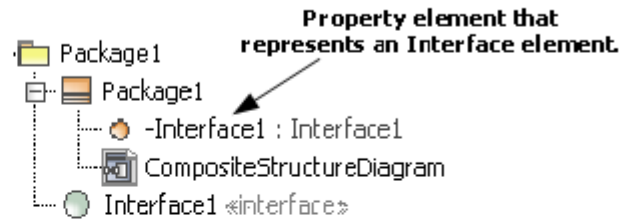
The Interface element data will be normally copied even if the Interface element is drawn in a Composite Structure diagram. A Property element will be created to represent the Interface element. The Property element will be named after the Interface element.

### Containment tree

#### EA (Before Conversion)

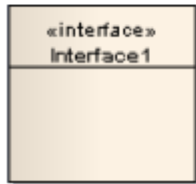


#### MD (After Conversion)

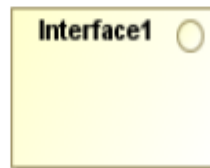


### Diagram View

#### EA (Before Conversion)



#### MD (After Conversion)



A Property element representing an Interface element.

## Part

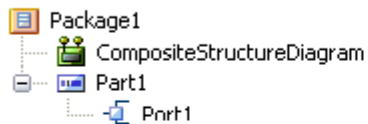
There are two types of Parts: (i) Parts with Ports; and (ii) Parts that set type to other elements.

### Parts with Ports

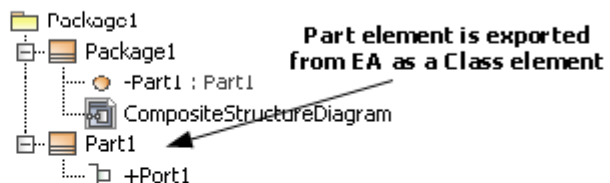
If you assign the type of a Part element to a Class, Component, or Node, the Part element can have a Port. If a Part is not nested to any element, the Part element will be exported from EA as a Class. Therefore, this Part element can have a Port.

### Containment tree

#### EA (Before Conversion)

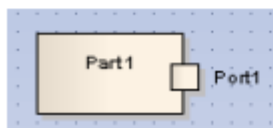


#### MD (After Conversion)

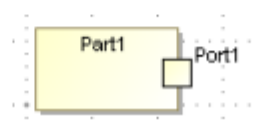


### Diagram View

#### EA (Before Conversion)



#### MD (After Conversion)



A Part that does not nest element(s) with Ports.

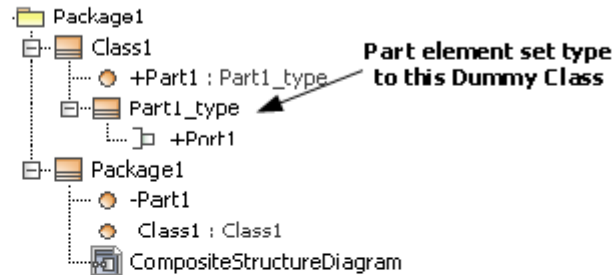
If a Part is nested to an element, and the type is not set, then the Part type will be used to set type to a dummy Class. This particular Part can have a Port. A dummy Class will be created at the same level of the Part element that sets type to it. The dummy Class will be named after the Part concatenated with '\_type'.

## Containment tree

### EA (Before Conversion)

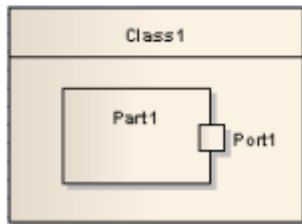


### MD (After Conversion)

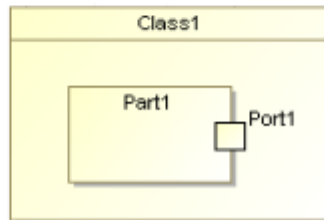


## Diagram View

### EA (Before Conversion)



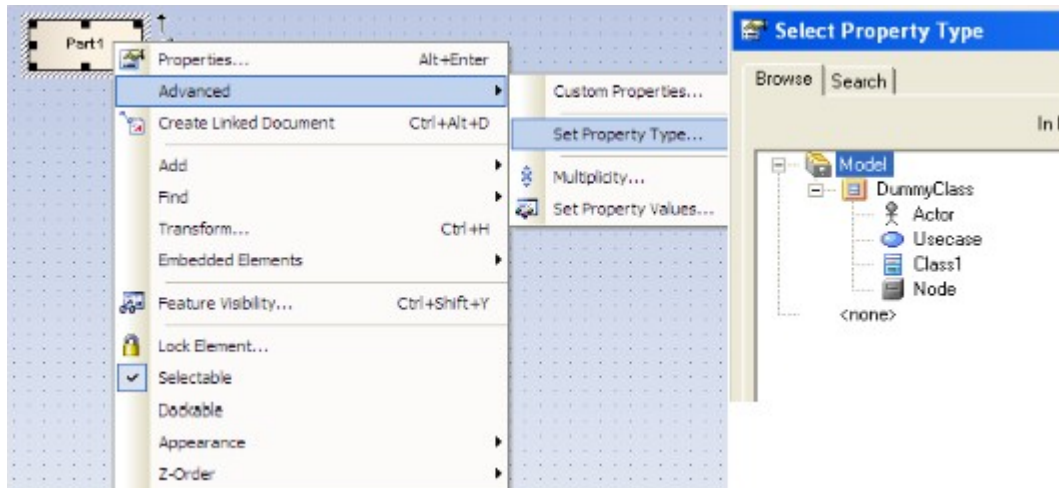
### MD (After Conversion)



A Part that nests element(s) with Ports.

## Parts that set type to other elements

The type of a Part can be set to another element, e.g., Class, Actor, Component, and Usecase, by right-clicking the Part element and selecting **Advanced > Set Property Type > Select Property Type**.



Setting Part Type.

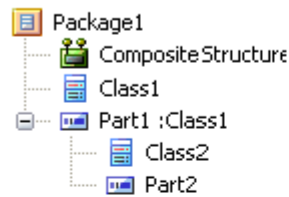
If the Part type is set to a type that cannot be the owner of a Port, the Port will be removed.

In MagicDraw, the Property element data cannot have any elements. However, in EA, it can have elements. Therefore, if you draw a Part element and it has an element inside it, that particular element will be relocated to an element, which is the Part type.

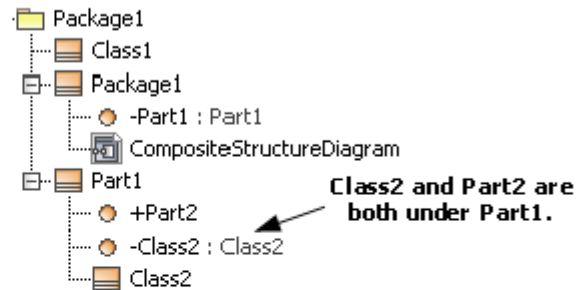
If a Part is not nested to any element, the Part element will be exported from EA as a Class. The element inside the Part will not be relocated. If that particular Part assigns a type to other elements, its type will always be set to Class.

## Containment tree

### EA (Before Conversion)

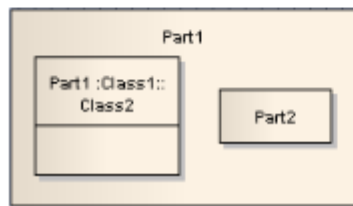


### MD (After Conversion)

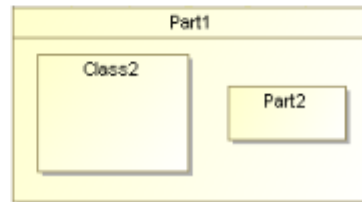


## Diagram View

### EA (Before Conversion)



### MD (After Conversion)

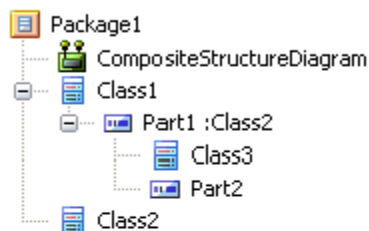


A Part that sets type to other elements.

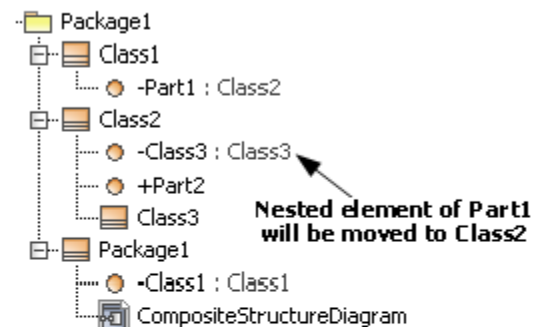
If a nested Part whose type is set to another element, e.g., Class, Actor, and Usecase, and this Part contains nested elements, all of the nested elements of the Part will be relocated to the element that is the type of this Part.

## Containment tree

### EA (Before Conversion)

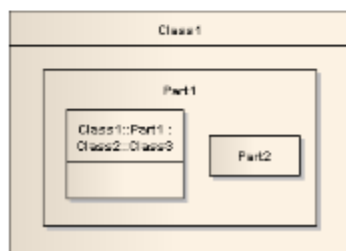


### MD (After Conversion)

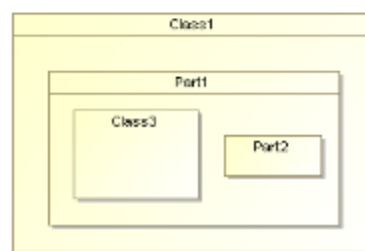


## Diagram View

### EA (Before Conversion)



### MD (After Conversion)



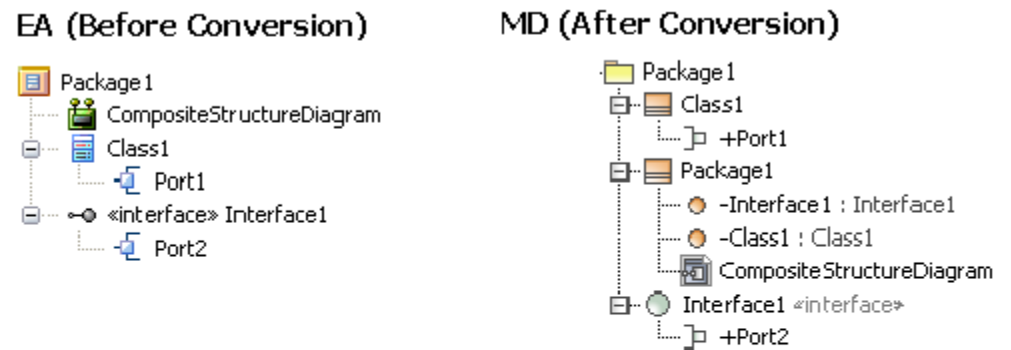
A Part that sets Type to other elements has nested elements inside itself.

## Port

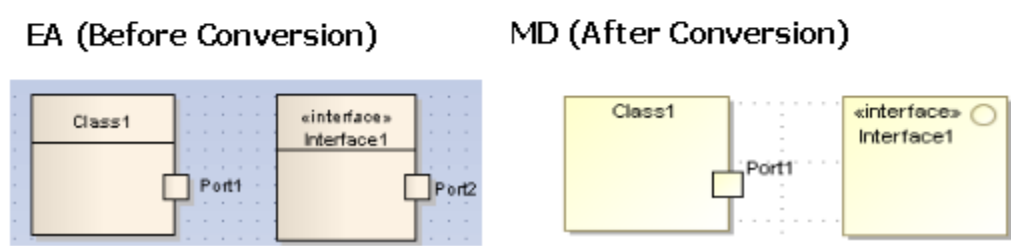
In MagicDraw, a Property element that can have a Port is the Property that sets type to a Class, Component, or Node, and other elements cannot have a Port. However, in EA, most of the elements can have a Port.

If a Port is created with an element that is not a Class, Component or Node, the Port will be removed from the diagram view. However, its data will be preserved.

### Containment tree



### Diagram View



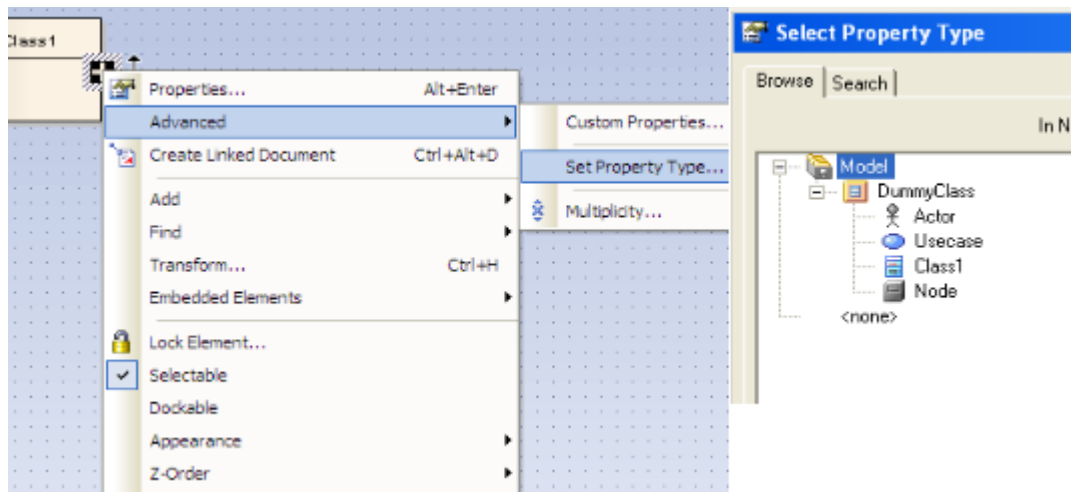
Only Class, Component, and Node can have a Port.

The following transformation message will open:

Removed view <xmi:id>: Invalid Port. Port can be added to Part that its type is set to Class, Component and Node only.

### Ports whose type is set to other elements

You can set the type of Port element to another element, e.g., Class, Actor, Component, and Usecase, by right-clicking the Port element and selecting **Advanced > Set Property Type > Select Property Type**.



Setting Port Type.

### Port whose type is set to another element and containing Expose Interfaces

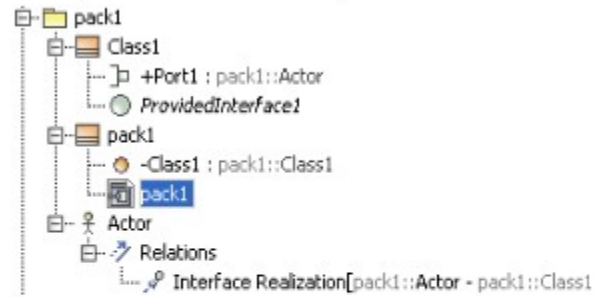
If an Expose Interface is created on a Port, and the Port type is set to another element, the Expose Interface data will be relocated to the Port's type element. For example, if the Port1 type is set to Actor, an Interface Realization (the Expose Interface data) will be placed inside the Actor after conversion.

## Containment tree

### EA (Before Conversion)

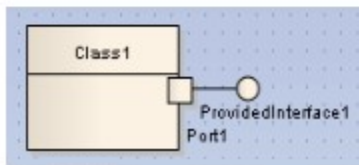


### MD (After Conversion)

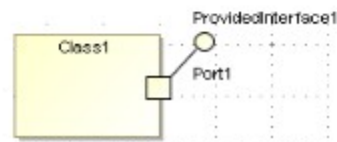


## Diagram View

### EA (Before Conversion)



### MD (After Conversion)



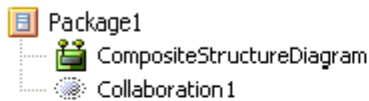
A Port that sets Type to other elements contains an Expose Interface.

## Collaboration

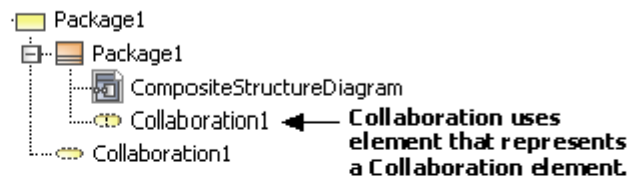
The Collaboration element data and its nested data will normally be copied. However, if the Collaboration element is drawn in a Composite Structure diagram, a Collaboration use element will be created to represent the Collaboration element. The Collaboration use element will be named after the Collaboration element.

## Containment tree

### EA (Before Conversion)



### MD (After Conversion)



## Diagram View

### EA (Before Conversion)



### MD (After Conversion)



A Collaboration use element.

## Collaboration containing other elements

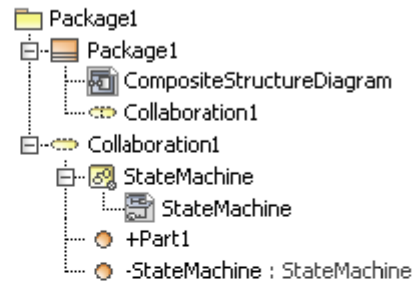
A Collaboration can contain only Property, Activity, State Machine, and Interaction. Elements other than these will be removed. If any element is drawn inside the Collaboration, it will be removed from the diagram view.

## Containment tree

### EA (Before Conversion)

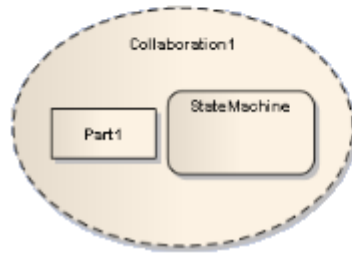


### MD (After Conversion)

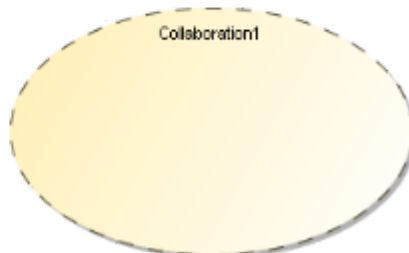


## Diagram View

### EA (Before Conversion)



### MD (After Conversion)



A Collaboration containing other elements.

The following transformation messages will open:

```
Removed element <xmi:id>: Invalid Element. Collaboration can contain Property, Activity, State Machine and Interaction only.  
Removed view <xmi:id>: Invalid Element view. Collaboration cannot be contained in any element view.
```

## **Expose Interface**

Expose Interfaces are the provided and required interfaces in MagicDraw.

In MagicDraw, you can draw a provided Interface or required Interface in a Port only and set the type of the Port to another element, e.g., Actor, Usecase, and Class. You can create that particular Port on a Part element whose type is set to a Class, Component, or Node only.

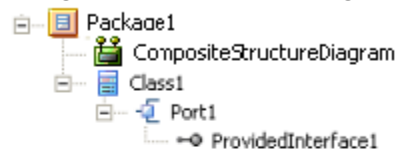
## **Expose Interface with Ports**

If you draw an Expose Interface in a Port, and the Port does not assign a Type to any elements, that particular Port will be used to set Type to a Dummy Class. The Dummy Class will be created and named after the Port concatenated with '\_type'. The Interface element will be moved to the same level of the Port.

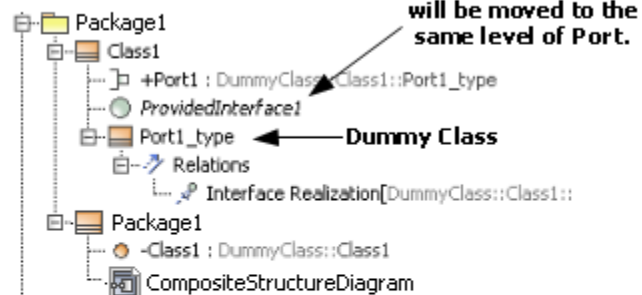


## Containment tree

### EA (Before Conversion)

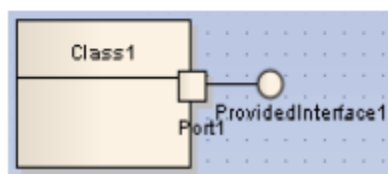


### MD (After Conversion)

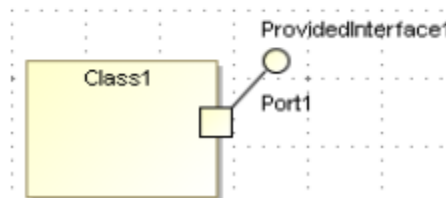


## Diagram View

### EA (Before Conversion)



### MD (After Conversion)



An Expose Interface with a Port.

If an Expose Interface is created on a Port, and the Type of the Port is set to another element, the Expose Interface data will be relocated to the Port's Type element (see (ii) Port whose type is set to another element and containing Expose Interfaces for more information).

In MagicDraw, some elements can be used as a Port Type. These elements can have either a provided Interface or required Interface, or both.

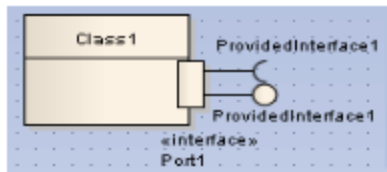
The following list shows the Port's Type elements with specific conditions:

- Interfaces can only have a provided Interface.
- Artifacts cannot have a provided Interface.
- Information Items cannot have a provided Interface.
- Signals cannot have a provided Interface.
- Components always have both provided and required Interfaces.
- The other elements can have both provided and required Interfaces.
- After conversion, if the type of a Port is set to an Interface and the Port has an Expose Interface, only the provided Interface will be shown and the Interface Realization data will be removed.
- Artifacts, Information Items, and Signals cannot hold Interface Realization. If the type of a Port is set to one of them, the provided Interface will be removed. However, this does not affect the required Interface.
- If the Type of a Port is set to a Component, and the Port has an Expose Interface, it will always show both the provided and required Interfaces.

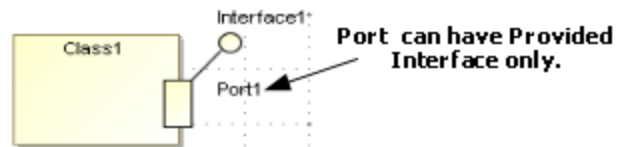
## Diagram View

### EA (Before Conversion)

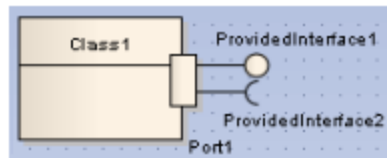
#### Set Port type to Interface



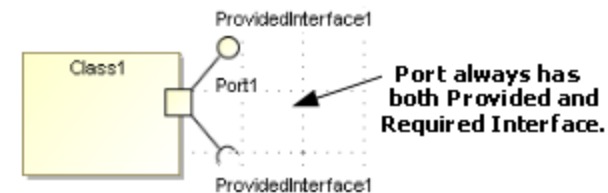
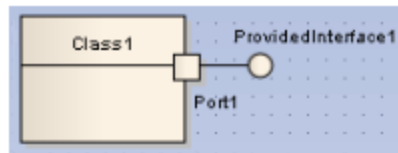
### MD (After Conversion)



#### Set Port type to Artifact, Information Item and Signal



#### Set Port type to Component



The Type element of a Port with specific conditions.

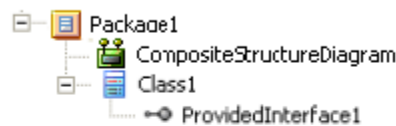
The following transformation message will open:

Updated element <xmi:id>: Provided Interface conflicts with Port type.  
The Port type is updated to an owner of Interface Realization.

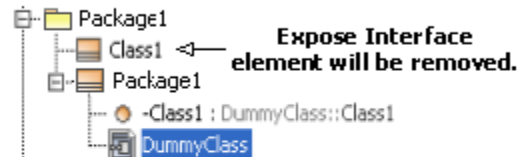
If an Expose Interface is created with an element that is not a Port, the element will be removed both from the Containment tree and diagram view.

## Containment tree

### EA (Before Conversion)

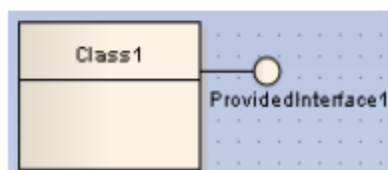


### MD (After Conversion)

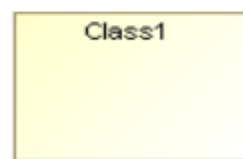


## Diagram View

### EA (Before Conversion)



### MD (After Conversion)



An Expose Interface with the element that is not a Port.

The following transformation message will open:

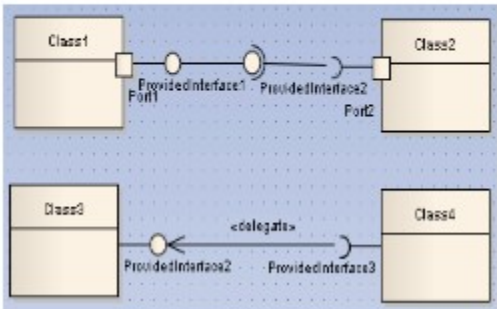
Removed element <xmi:id>: Invalid Element. Expose Interface can be added to Port only.

## Expose Interface with Relationships

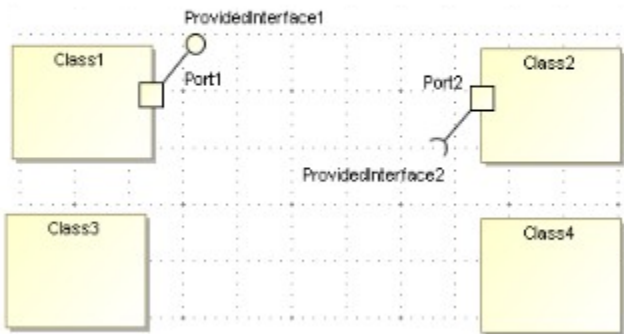
In MagicDraw, the provided and required Interfaces cannot be connected with any Relationship, as shown in the figure below. If there is any Relationship connecting to the Expose Interface, it will be removed from both the Containment tree and diagram view.

**Diagram View**

EA (Before Conversion)



MD (After Conversion)



An Expose Interface with Lines.

The following transformation message will open:

Removed element <xmi:id>: Invalid Element. Expose Interface cannot be connected with the element.

**Package**

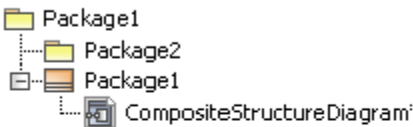
In MagicDraw, a Package cannot be drawn in a Composite Structure diagram. If it is drawn in EA, it will be removed from the diagram once it has been converted to MagicDraw.

**Containment tree**

EA (Before Conversion)

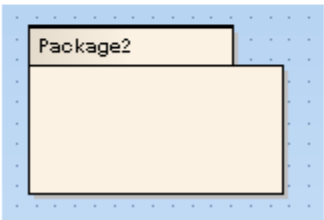


MD (After Conversion)

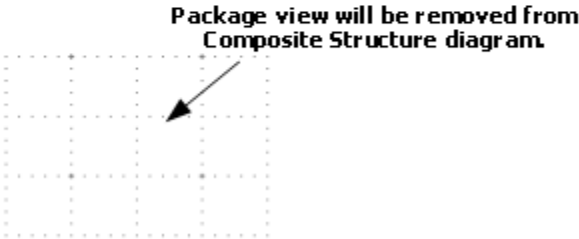


**Diagram View**

EA (Before Conversion)



MD (After Conversion)



A Package element.

**Assembly**

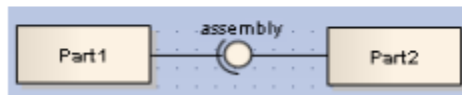
In MagicDraw, there is no Assembly Line. It will be updated to a Connector.

The following transformation message will open:

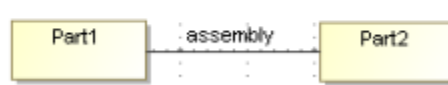
Updated element <xmi:id>: Assembly updated to Connector.

## Diagram View

### EA (Before Conversion)



### MD (After Conversion)



Updating the Assembly Line to the Connector Line.

## Dependency

There are many relationships that EA exporter exports to Dependency as the following elements:

- Delegate
- Role Binding
- Represents
- Occurrence
- Nest
- Derive
- Import
- Instantiate
- Usage
- Realize
- Trace

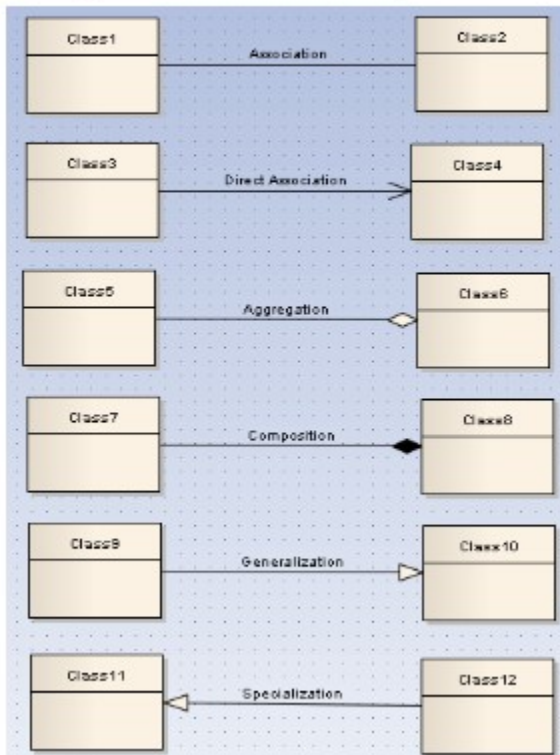
The Relationships above will be shown in the diagram view as Dependencies with stereotype.

## Removed relationships

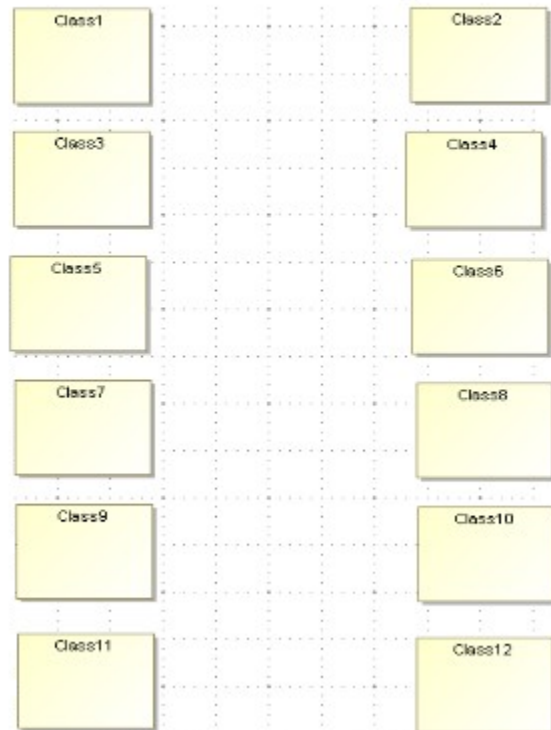
If an Association, Direct Association, Aggregation, Composition, Generalization, or Specialization is created in a Composite Structure diagram, it will be removed from the diagram view, but its data will be preserved.

## Diagram View

### EA (Before Conversion)



### MD (After Conversion)



Removing Lines.

The following transformation messages will open:

```
Removed view <xmi:id>: Association cannot be shown in Composite Structure diagram.  
Removed view <xmi:id>: Direct Association cannot be shown in Composite Structure diagram.  
Removed view <xmi:id>: Aggregation cannot be shown in Composite Structure diagram.  
Removed view <xmi:id>: Composition cannot be shown in Composite Structure diagram.  
Removed view <xmi:id>: Generalization cannot be shown in Composite Structure diagram.
```

## Transformation report

A report containing the conflicts solved during transformation, along with other transformation information, e.g., special mapping and removal of some irrelevant data, is required to be provided for the users.

The following list shows the transformation messages:

```
Removed view <xmi:id>: Invalid Port. Port can be added to Part that set type to Class, Component, and Node only.  
Removed element <xmi:id>: Invalid Element. Collaboration can contain Property, Activity, State Machine, and  
Interaction only.  
Removed view <xmi:id>: Collaboration cannot be contained in any element view.  
Removed view <xmi:id>: UseCase cannot be contained in any element view.  
Removed element <xmi:id>: Invalid Element. Expose Interface can be added to Port only.  
Removed element <xmi:id>: Invalid Element. Expose Interface cannot be connected with the element.  
Updated element <xmi:id>: Assembly updated to Connector.  
Updated element <xmi:id>: Provided Interface conflicts with Port type. The Port type is updated to an owner of  
Interface Realization.  
Removed view <xmi:id>: Association cannot be shown in CompositeStructure diagram.  
Removed view <xmi:id>: Direct Association cannot be shown in Composite Structure diagram.  
Removed view <xmi:id>: Aggregation cannot be shown in Composite Structure diagram.  
Removed view <xmi:id>: Composition cannot be shown in Composite Structure diagram.  
Removed view <xmi:id>: Generalization cannot be shown in Composite Structure diagram.
```