

# Ecore Documentation

# Table of Contents

Contents of EPackage1.....	1
Data Types .....	1
SpecialDataType.....	1
Enumerations .....	1
Enum1 .....	1
Classes.....	1
Abstract Class EClass1.....	1
Class MyEClass .....	2
Class Class3 .....	3
Interface Iface.....	4
Contents of EPackage2.....	4
Class SomeClass .....	4

# Contents of EPackage1

## Data Types

TODO: Create template for EDataType

### SpecialDataType

*Used at*

- [EPackage1.MyEClass.attr](#)

## Enumerations

### Enum1

Enum 1 Description

*Literals*

Symbol	Value	Description
<a href="#">Literal1</a>	0	Literal1 Description  Second sentence of Literal1 Description.
<a href="#">Literal2</a>	1	Literal2 Description

*Used at*

- [EPackage1.MyEClass.eeeenum](#)
- [EPackage1.Class3.something](#)

## Classes

### Abstract Class EClass1

TODO: Find a good way to show class attributes like abstract.

Description of EClass1

*Sub-concepts*

- [EPackage1.MyEClass](#)
- [EPackage2.SomeClass](#)

*Attributes*

Name	Type	Multiplicity / Order	Default Value	Description
id	EInt	1		Description of id.
name	EString	3..5		Name desc.

#### References

Name	Type	Multiplicity / Order	Opposite	Description
myRelation	<a href="#">EPackage1.MyEClass</a>	0..* / unordered	<a href="#">backwards</a>	Description.

#### Used at

- [EPackage1.MyEClass.ref](#)
- [EPackage1.MyEClass.backwards](#)

## Class MyEClass

Description

#### Supertypes

- [EPackage1.EClass1](#)

#### Attributes

Name	Type	Multiplicity / Order	Default Value	Description
attr	<a href="#">EPackage1.SpecialDataType</a>	0..1		Description. Second sentence.
eeenum	<a href="#">EPackage1.Enum1</a>	0..6	<a href="#">Literal1</a>	Deschkriptschion.
id ( <a href="#">△ EPackage1.EClass1</a> )	EInt	1		Description of id.
name ( <a href="#">△ EPackage1.EClass1</a> )	EString	3..5		Name desc.

#### Containments

Name	Type	Multiplicity / Order	Opposite	Description
clazzes	<a href="#">EPackage1.Class3</a>	1..* / unordered	<a href="#">specialParent</a>	Desc.

Name	Type	Multiplicity / Order	Opposite	Description
otherClasses	EPackage1.Class3	0..* / ordered		Desc.  Containments could also be inherited.

#### References

Name	Type	Multiplicity / Order	Opposite	Description
ref	EPackage1.EClass1	0..1		Whatever.
backwards	EPackage1.EClass1	1	myRelation	
myRelation ( $\triangle$ EPackage1.EClass1)	EPackage1.MyEClass	0..* / unordered	backwards	Description.

#### Used at

- EPackage1.MyEClass.myRelation
- EPackage1.Class3.ref
- EPackage1.Class3.myRelation
- EPackage1.EClass1.myRelation

## Class Class3

#### Description

#### Supertypes

- EPackage1.EClass1

#### Attributes

Name	Type	Multiplicity / Order	Default Value	Description
something	EPackage1.Enum1	0..6		Something.
id ( $\triangle$ EPackage1.EClass1)	EInt	1		Description of id.
name ( $\triangle$ EPackage1.EClass1)	EString	3..5		Name desc.

#### References

Name	Type	Multiplicity / Order	Opposite	Description
<a href="#">specialParent</a>	<a href="#">EPackage1.MyEClass</a>	5	<a href="#">clazzes</a>	Bla
<a href="#">myRelation</a> ( <a href="#">△ EPackage1.EClass1</a> )	<a href="#">EPackage1.MyEClass</a>	0..* / <a href="#">unordered</a>	<a href="#">backwards</a>	Description.

## Interface Iface

Description

*Known Implementations*

- [EPackage2.SomeClass](#)

Attributes, Containments, References

# Contents of EPackage2

## Class SomeClass

*Supertypes*

- [EPackage1.EClass1](#)
- [EPackage1.Iface](#)

*Attributes*

Name	Type	Multiplicity / Order	Default Value	Description
<a href="#">id</a> ( <a href="#">△ EPackage1.EClass1</a> )	EInt	1		Description of id.
<a href="#">name</a> ( <a href="#">△ EPackage1.EClass1</a> )	EString	3..5		Name desc.

*References*

Name	Type	Multiplicity / Order	Opposite	Description
<a href="#">myRelation</a> ( <a href="#">△ EPackage1.EClass1</a> )	<a href="#">EPackage1.MyEClass</a>	0..* / <a href="#">unordered</a>	<a href="#">backwards</a>	Description.