ECLASS Technical Specification 15

URI Path

ECLASS e.V.

02.04.2024

Authors

Nikolaus Ondracek, Paradine GmbH

Contributors

ECLASS Center of Research and Development (CRD):

Dr. Oliver Drumm, Siemens

Petra Eder, SAP SE

Oliver Hadasch, Rösberg Engineering

Philippe Juhel, Schneider Electric SE

Thorsten Kroke, ECLASS Head Office

Stefan Mülhens, AmpereSoft

Matthias Redecker, Weidmüller Interface

Frank Scherenschlich, Class.Ing

Josef Schmelter, Phoenix Contact

and

Dr. Christian Block, ECLASS Head Office

Mihai Moldovan, Paradine

Fabian Schwanitz, ECLASS Head Office

Please send remarks to: crd@eclass.de

Revision History

Date	Version	Description	Who
06.12.2022	0.9	Final Draft	See list of contributors
30.03.2023	1.0	CRD decision about initial release	CRD
14.04.2023	1.0	Published	Schwanitz
20.03.2024	1.0.2	Text adjustments	Mülhens
02.04.2024	1.0.2	Published as 1.0.2	Schwanitz, Hassankhani

Table of contents

1 Introduction

1.1 Purpose

This document is ECLASS Technical Specification 15 for defining Property Path on basis of URI.

As a standardized proposal this specification is designed and agreed by different product and software manufacturers.

This Technical Specification is superseding Technical Specification 13.

1.2 Scope

In scope are:

- Definition of formal grammar for Property Path

Out of scope are:

- other Path syntax formats
- Detailed description of Cardinality and Polymorphism

1.3 Definitions

1.3.1 Context free grammar

A formal grammar is "context free" **if its production rules can be applied regardless of the context of a nonterminal**. No matter which symbols surround it, the single nonterminal on the left-hand side can always be replaced by the right-hand side. This is what distinguishes it from a context-sensitive grammar.

1.3.2 Support for LR Parser

LR Parser is **a bottom-up parser for context-free grammar** that is very generally used by computer programming language compiler and other associated tools. LR parser reads their input from left to right and produces a right-most derivation.

1.4 Acronyms, and Abbreviations

BNF	Backus–Naur Form (BNF)	metasyntax notation for context-free grammars

URI	Unified Resource Identifier	
IRDI	International Registration Data Identifier	[ISO 29002-5]

1.5 Other Definitions

Other definitions, acronyms, and abbreviations are described in the ECLASS Technical Support (https://eclass.eu/support).

1.6 References

ISO 29002-5

ISO 29002-10

ECLASS Technical Specification 11 – Conceptual Data Model

ECLASS Technical Specification 28 - How to transport ECLASS in the AAS

2 Introduction

2.1 Motivation

Different ECLASS application areas have the need to address Properties for encoding of product descriptive data, fully compliant to RFC 3986 which may be used in all internet representations for dictionaries/items/assets, for example:

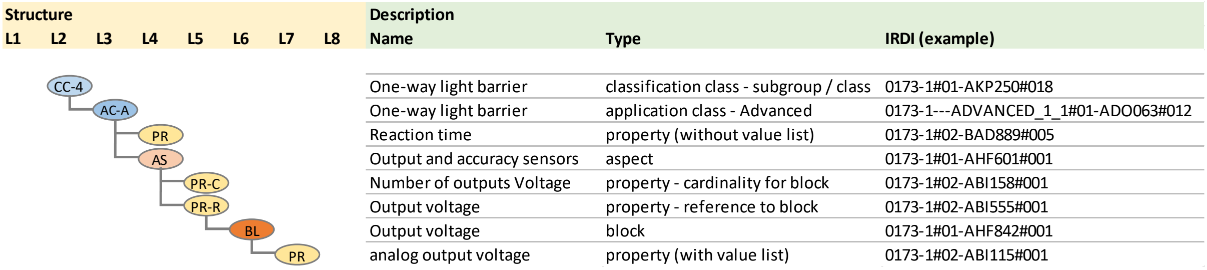
- Embedding ECLASS into AutomationML,
- In Accelerated ECLASS – Webservices to address a Property within an Aspect or Block (including unit), e.g., in mappings between different dictionaries,
- In Catalog Date Exchange (Item) Webservice to locate a value of a Property within an Aspect or Block.

2.2 Property Path

A Property Path is identifying the context of a property within a mereological structure.

Example

The following example describes how the Property at the leaf of the mereology is described based on ECLASS 13.0.



Structure								Description		
L1	L2	L3	L4	L5	L6	L7	L8	Name	Type	IRDI (example)
	CC-4							One-way light barrier	classification class - subgroup / class	0173-1#01-AKP250#018
		AC-A						One-way light barrier	application class - Advanced	0173-1---ADVANCED_1_1#01-ADO063#012
			PR					Reaction time	property (without value list)	0173-1#02-BAD889#005
			AS					Output and accuracy sensors	aspect	0173-1#01-AHF601#001
				PR-C				Number of outputs Voltage	property - cardinality for block	0173-1#02-ABI158#001
				PR-R				Output voltage	property - reference to block	0173-1#02-ABI555#001
					BL			Output voltage	block	0173-1#01-AHF842#001
						PR		analog output voltage	property (with value list)	0173-1#02-ABI115#001

Figure 1: ECLASS example for Property path

2.3 Absolute Path

The Absolute Path is a Property Path which begins always with the outmost Application Class which is related to the ECLASS Classification Class hierarchy and traverses from this Application Class to the leaf Property.

```
0173-1---ADVANCED_1_1-01-ADO063-012/0173-1-01-AHF601-001/0173-1-02-ABI555-001~1/0173-1-01-AHF842-001/0173-1-02-ABI115-001
```

In the example above the Cardinality is given for the Block (and not for the reference Property to the Block) and is marked with the Cardinality number (here: “~1”).

2.4 Relative Path

In contrast to an Absolute Path, a Relative Path is a Property Path which is starting at any Characterization Class (Block, Aspect, Asset) of the mereological structure. Except for the starting Characterization Class, the Relative Path follows all the same rules as the Absolute Path.

Note: Relative Paths are introduced for the use in the area of mappings to AAS submodels (see Technical Specification 28) and are currently not used in other applications today.

2.5 URI Path

A URI Path is a Property Path which follows the W3C specification for URI.

2.6 URI Path in Dictionary Layer

The URI Path may be used for addressing the context of Properties. The specification includes traversing of

- Blocks
- Aspects
- Level
- Axis

Dictionary Layer does not contain interpretation of

- Cardinality
- Polymorphism

Note: The Accelerated ECLASS REST Services will be extended to use the URI Path for accessing dictionary elements.

2.7 URI Path in Item Layer

The URI Path may be used within Items, which contains Values encoded in context of Property within a mereology, in this case the following additional pattern must be considered in the syntax.

The URI Path will contain additionally:

- Values
- Units
- Cardinality
- Polymorphism

Note: The fact if Advanced or Basic representation is used is indicated by different Application Class IRDI’s.

3 URI Path Syntax

3.1 General Assumptions

3.1.1 Context free grammar

The URI Path grammar shall be defined as a context free grammar.

3.1.2 Support for LR Parser

The URI Path grammar shall be defined to allow implementation by LR Parsers.

3.1.3 Compliant to RFC 3986

URI Path syntax shall be fully compliant to RFC 3986 which will be called URI Path, which should not use % encoding.

3.1.4 One Syntax for Dictionary and Items

There shall be one Syntax for URI Path in dictionary and items exchange.

3.1.5 Slash based URI Path

We use an approach where a trimmed down slash-based path aims at eliminating the need for encoding characters and presenting a valid structure path.

3.1.6 Use Hyphens to avoid Ambiguity

The URI Path grammar shall use hyphens to eliminate ambiguity.

Note: Hyphens are used for avoiding overuse of slash separated groups of characters of varying lengths and purposes.

3.1.7 Use Org Identifier on each IRDI

The URI Path grammar shall use the Organization Identifier mandatorily in each path element.

Example

```
https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0112-3-01-AAA789-001/0173-1-02-AAA357-001
```

3.1.8 Use Code Space Identifier

The URI Path grammar shall use Code Space Identifiers according to ISO 29002-5.

Example

```
https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001
```

Note: This is needed, since in ECLASS overlapping sets of codes per dictionary element type (e.g. Property, Class, Value List, etc.) exist.

3.1.9 Use Version in Identifier

The URI Path grammar shall use the version identifier for the traversed Structure Elements within the path for allowing receiving systems to detect if the dictionary is up to date.

Example

```
https://cdp.com/0173-1-01-AAA123-002/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001
```

Note: The version is contained for each node of the URI Path, in order that an interpreting system can detect, if the path is traversing through a dictionary which contains at least same or newer versions of the traversed Structure Elements.

4 Dictionary Assumptions

4.1 Use Reference Property AND Referenced Block Identifier

The URI Path shall contain Reference Property AND Referenced Block Identifier if a Reference Property is traversed. The Reference Property itself as well as the referenced Characterization Class are contained in the path.

Example

```
https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001
```

4.2 Level-Type

The URI Path grammar shall take care about Level-Type interpretation.

The terminal symbols MIN / MAX / NOM / TYP are used in the grammar for denoting the Levels of a Level-Type property.

```
https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.MAX
```

4.3 AXIS1 Type

The URI Path grammar shall allow about interpretation of Properties with Property-Data-Type AXIS1.

Note: According to ISO 10303 axis1_placement is the direction and location in three-dimensional space of a single axis and is defined by a point (first three numbers) and an axis direction (last three numbers).

The URI Path contains Axis-Suffixes for the leaf Property.

Axis Suffixes (X, Y, Z) for the point and (A1X, A1Y, A1Z) for the direction.

Example

```
https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.X

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.Y

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.Z
```

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.A1X

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.A1Y

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.A1Z

4.4 AXIS2 Type

The URI Path grammar shall allow about interpretation of Properties with Property-Data-Type AXIS2.

Note: According to ISO 10303 axis2_placement_2d is the location and orientation in two-dimensional space of two mutually perpendicular axes defined by a point (first two numbers) and an axis (last two numbers).

The URI Path grammar take care about AXIS2 Type interpretation, by adding Axis-Suffixes to the leaf Property.

Axis Suffixes (X, Y) for the point and (A1X, A1Y) for the direction.

Example

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.X

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.Y

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.A1X

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.A1Y

4.5 AXIS3 Type

The URI Path grammar shall allow about interpretation of Properties with Property-Datatype AXIS3.

Note: According to ISO 10303 axis2_placement_3d is the location and orientation in three-dimensional space of two mutually perpendicular axes defined by a point (first three numbers) and two axes (middle and last three numbers).

The URI Path grammar take care about AXIS3 interpretation, by adding Axis-Suffixes to the leaf Property.

Axis-Suffixes (X, Y, Z) for the point and (A1X, A1Y, A1Z) for the direction of AXIS1 and (A2X, A2Y, A2Z) for the direction of AXIS2.

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.X

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.Y

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.Z

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.A1X

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.A1Y

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.A1Z

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.A2X

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.A2Y

https://cdp.com/0173-1-01-AAA123-001/0173-1-02-AAA345-001~1/0173-1-01-AAA789-001.A2Z

4.6 No ECLASS Release in path

The URI Path grammar **shall not** contain any information about the ECLASS Release.

5 Item Assumptions

5.1 Special Character for Pattern of Cardinality

Since item exchange is needing additional information for Cardinality, this Cardinality Values are separated by a special character (“~”) which may be omitted in case of dictionary exchange, but which denotes the Cardinality Value on item exchange.

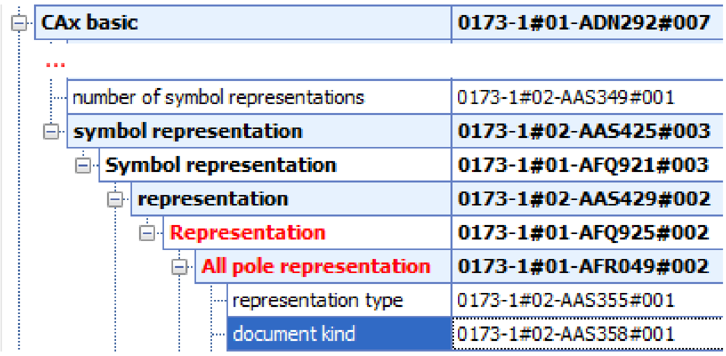
5.2 Reference Properties pointing to specialized Characterization Class

If in the pattern of Polymorphism, a Reference Property references a specialized Characterization Class (e.g. Block), the Property Path does not contain the IRDI of the generic class, but directly the reference to the specialized Characterization Class.

Note: During interpretation of the dictionary, the specialized Characterization Class is obtained by selecting a Value which is used as Class-Value-Assignment acting as the discriminator Property in the specialized class. For details see ECLASS Technical Specification 11.

6 Examples

Example



CAx basic	0173-1#01-ADN292#007
...	
number of symbol representations	0173-1#02-AAS349#001
symbol representation	0173-1#02-AAS425#003
Symbol representation	0173-1#01-AFQ921#003
representation	0173-1#02-AAS429#002
Representation	0173-1#01-AFQ925#002
All pole representation	0173-1#01-AFR049#002
representation type	0173-1#02-AAS355#001
document kind	0173-1#02-AAS358#001

The example shows an Absolute Path, which

- starts at Application Class (0173-1---ADVANCED_1_1-01-ADN533-008), which relates to
- Aspect “CAx basic” (0173-1-01-ADN292-007), which is described by
- Reference Property “symbol representation” (0173-1-02-AAS425-003), which
- has Cardinality Value 2 (0173-1-02-AAS425-003~2), and is referencing
- Block “symbol representation” (0173-1-01-AFQ921-003), which is described by
- Reference Property “representation” (0173-1-02-AAS429-002), referencing to
- Block Representation (not part of path), since its specialization is selected in Polymorphism
- Block “All pole representation” (0173-1-02-AAS429-002), which is described by
- Property “document kind” (0173-1-02-AAS358-001) has

```
https://cdp.com/0173-1---ADVANCED_1_1-01-ADN533-008/0173-1-01-ADN292-007/0173-1-02-AAS425-003~2/0173-1-01-
AFQ921-003/0173-1-02-AAS429-002/0173-1-01-AFR049-002/0173-1-02-AAS358-001
```

If we would need to also include one of the Values for this Property, we can expand the path with

- a reference to the Coded-Value (0173-1-07-AAV599-001) assigned

```
https://cdp.com/0173-1---ADVANCED_1_1#01-ADN533#008/0173-1-01-ADN292-007/0173-1-02-AAS425-003~2/0173-1-01-
AFQ921-003/0173-1-02-AAS429-002/0173-1-01-AFR049-002/0173-1-02-AAS358-001/0173-1-07-AAV599-001
```