**Open Geospatial Consortium**

Submission Date: 2018-12-13

Approval Date:   <yyyy-mm-dd>

Publication Date:   <yyyy-mm-dd>

External identifier of this OGC® document: <[http://www.opengis.net/doc/[{doc-type/}]{standard}/{m.n}](http://www.opengis.net/def/%5b%7bdoc-type/%7d%5d%7bstandard%7d/%7bm.n%7d)>

Internal reference number of this OGC® document:    18-073r1

Version: 1.0

Category: OGC® Standard

Editor:   John Tisdale

OGC PipelineML Conceptual and Encoding Model Standard

**Candidate Standard**

**Copyright notice**

Copyright © 2018 Open Geospatial Consortium  
To obtain additional rights of use, visit <http://www.opengeospatial.org/legal/>.

**Warning**

This document is not an OGC Standard. This document is distributed for review and comment. This document is subject to change without notice and may not be referred to as an OGC Standard.

Document type:    OGC® Standard

Document subtype:    PipelineML Tech Spec

Document stage:    Draft

Document language:  English

Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

License Agreement

Permission is hereby granted by the Open Geospatial Consortium, ("Licensor"), free of charge and subject to the terms set forth below, to any person obtaining a copy of this Intellectual Property and any associated documentation, to deal in the Intellectual Property without restriction (except as set forth below), including without limitation the rights to implement, use, copy, modify, merge, publish, distribute, and/or sublicense copies of the Intellectual Property, and to permit persons to whom the Intellectual Property is furnished to do so, provided that all copyright notices on the intellectual property are retained intact and that each person to whom the Intellectual Property is furnished agrees to the terms of this Agreement.

If you modify the Intellectual Property, all copies of the modified Intellectual Property must include, in addition to the above copyright notice, a notice that the Intellectual Property includes modifications that have not been approved or adopted by LICENSOR.

THIS LICENSE IS A COPYRIGHT LICENSE ONLY, AND DOES NOT CONVEY ANY RIGHTS UNDER ANY PATENTS THAT MAY BE IN FORCE ANYWHERE IN THE WORLD.

THE INTELLECTUAL PROPERTY IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE DO NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE INTELLECTUAL PROPERTY WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE INTELLECTUAL PROPERTY WILL BE UNINTERRUPTED OR ERROR FREE. ANY USE OF THE INTELLECTUAL PROPERTY SHALL BE MADE ENTIRELY AT THE USER’S OWN RISK. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR ANY CONTRIBUTOR OF INTELLECTUAL PROPERTY RIGHTS TO THE INTELLECTUAL PROPERTY BE LIABLE FOR ANY CLAIM, OR ANY DIRECT, SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM ANY ALLEGED INFRINGEMENT OR ANY LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR UNDER ANY OTHER LEGAL THEORY, ARISING OUT OF OR IN CONNECTION WITH THE IMPLEMENTATION, USE, COMMERCIALIZATION OR PERFORMANCE OF THIS INTELLECTUAL PROPERTY.

This license is effective until terminated. You may terminate it at any time by destroying the Intellectual Property together with all copies in any form. The license will also terminate if you fail to comply with any term or condition of this Agreement. Except as provided in the following sentence, no such termination of this license shall require the termination of any third party end-user sublicense to the Intellectual Property which is in force as of the date of notice of such termination. In addition, should the Intellectual Property, or the operation of the Intellectual Property, infringe, or in LICENSOR’s sole opinion be likely to infringe, any patent, copyright, trademark or other right of a third party, you agree that LICENSOR, in its sole discretion, may terminate this license without any compensation or liability to you, your licensees or any other party. You agree upon termination of any kind to destroy or cause to be destroyed the Intellectual Property together with all copies in any form, whether held by you or by any third party.

Except as contained in this notice, the name of LICENSOR or of any other holder of a copyright in all or part of the Intellectual Property shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Intellectual Property without prior written authorization of LICENSOR or such copyright holder. LICENSOR is and shall at all times be the sole entity that may authorize you or any third party to use certification marks, trademarks or other special designations to indicate compliance with any LICENSOR standards or specifications. This Agreement is governed by the laws of the Commonwealth of Massachusetts. The application to this Agreement of the United Nations Convention on Contracts for the International Sale of Goods is hereby expressly excluded. In the event any provision of this Agreement shall be deemed unenforceable, void or invalid, such provision shall be modified so as to make it valid and enforceable, and as so modified the entire Agreement shall remain in full force and effect. No decision, action or inaction by LICENSOR shall be construed to be a waiver of any rights or remedies available to it.

Table of Contents

[1. Scope 10](#_Toc528593167)

[2. Conformance 10](#_Toc528593168)

[3. References 11](#_Toc528593169)

[4. Terms and Definitions 12](#_Toc528593170)

[5. Conventions 16](#_Toc528593195)

[6. Clauses not Containing Normative Material 18](#_Toc528593201)

[7. PipelineML Conceptual Model (Normative) 19](#_Toc528593202)

[7.1 UML Model Requirement Class 20](#_Toc528593203)

[7.2 PipelineML Abstract Requirements Class (Normative) 22](#_Toc528593204)

[7.3 Linked Open Data Requirements Class (Normative) 25](#_Toc528593205)

[7.4 PipelineML Conceptual Requirements Class (Normative) 27](#_Toc528593206)

[7.5 PMLComponent Class Requirements Class (Normative) 28](#_Toc528593207)

[8. XML Encoding Requirement Class (Normative) 199](#_Toc528593208)

[8.1 PipelineML Core XML Abstract Requirements Class (Normative) 200](#_Toc528593209)

[9. Media Types for any data encoding(s) 202](#_Toc528593210)

[9.1 Abbreviations 202](#_Toc528593211)

[Annex A: Conformance Classes (Normative) 203](#_Toc528593212)

[A.1 PipelineML Conceptual Conformance Class 203](#_Toc528593213)

[A.2 PipelineML Abstract Conformance Class 203](#_Toc528593214)

[A.3 PipelineML Linked Open Data Conformance Class 205](#_Toc528593215)

[A.4 PipelineML Conceptual Model Conformance Class 206](#_Toc528593216)

[A.5 PMLComponent Class Conformance Class 207](#_Toc528593217)

[A.6 Pipeline Class Conformance Class 209](#_Toc528593218)

[A.7 Assembly Class Conformance Class 212](#_Toc528593219)

[A.8 Collection Class Conformance Class 215](#_Toc528593220)

[A.9 Component Class Conformance Class 217](#_Toc528593221)

[A.10 Cap Class Conformance Class 220](#_Toc528593222)

[A.11 Compressor Class Conformance Class 226](#_Toc528593223)

[A.12 Elbow Class Requirements Class (Normative) 233](#_Toc528593224)

[A.13 Flange Class Requirements Class (Normative) 239](#_Toc528593225)

[A.14 Launcherreceiver Class Requirements Class (Normative) 243](#_Toc528593226)

[A.15 Linepipe Class Requirements Class (Normative) 247](#_Toc528593227)

[A.16 Meter Class Requirements Class (Normative) 254](#_Toc528593228)

[A.17 Pump Class Requirements Class (Normative) 258](#_Toc528593229)

[A.18 Reducer Class Requirements Class (Normative) 265](#_Toc528593230)

[A.19 Tap Class Requirements Class (Normative) 272](#_Toc528593231)

[A.20 Tee Class Requirements Class (Normative) 277](#_Toc528593232)

[A.21 Valve Class Requirements Class (Normative) 284](#_Toc528593233)

[A.22 Casing Class Requirements Class (Normative) 288](#_Toc528593234)

[A.23 Coating Class Requirements Class (Normative) 295](#_Toc528593235)

[A.24 Sleeve Class Requirements Class (Normative) 300](#_Toc528593236)

[A.25 PipelineML Core XML Abstract Requirements Class (Normative) 305](#_Toc528593237)

Table of Figures

[Figure 1. Use Case Grouping Levels 8](#_Toc515715056)

[Figure 2. Conceptual Model Requirements classes dependencies 21](#_Toc515715057)

[Figure 3. PMLComponent Core Requirements Class 31](#_Toc515715058)

[Figure 4. PMLDataset Requirements Class 37](#_Toc515715059)

[Figure 5. ISO 19103 Core Data Types 33](#_Toc515715060)

[Figure 6. ISO 19109 - Rules for application schema – Feature Type 34](#_Toc515715061)

[Figure 7. PMLComponent Requirements Class Classes 37](#_Toc515715062)

[Figure 8. Pipeline Requirements Class 40](#_Toc515715063)

[Figure 9. Assembly Requirements Class 45](#_Toc515715064)

[Figure 10. Collection Requirements Class 50](#_Toc515715065)

[Figure 11. Component Requirements Class 53](#_Toc515715066)

[Figure 12. ConnectorComponent Requirements Class 55](#_Toc515715067)

[Figure 13. Pipeconnector Requirements Class 57](#_Toc515715068)

[Figure 14. Virtualconnector Requirements Class 61](#_Toc515715069)

[Figure 15. CoterminousComponent Requirements Class 62](#_Toc515715070)

[Figure 16. Cap Requirements Class 65](#_Toc515715071)

[Figure 17. Compressor Requirements Class 73](#_Toc515715072)

[Figure 18. Elbow Requirements Class 84](#_Toc515715073)

[Figure 19. Flange Requirements Class 92](#_Toc515715074)

[Figure 20. Launcherreceiver Requirements Class 98](#_Toc515715075)

[Figure 21. Linepipe Requirements Class 106](#_Toc515715076)

[Figure 22. Meter Requirements Class 115](#_Toc515715077)

[Figure 23. Pump Requirements Class 123](#_Toc515715078)

[Figure 24. Reducer Requirements Class 134](#_Toc515715079)

[Figure 25. Tap Requirements Class 143](#_Toc515715080)

[Figure 26. Tee Requirements Class 150](#_Toc515715081)

[Figure 27. TeeOutsideDiameterRun 155](file:///C:\Users\John\Dropbox\EPROD\PipelineML\TechSpec%20Files\OGC%20PipelineML%20Conceptual%20Model%20Standard%200.1.22.docx#_Toc515715082)

[Figure 28. TeeOutsideDiameterOutlet 156](file:///C:\Users\John\Dropbox\EPROD\PipelineML\TechSpec%20Files\OGC%20PipelineML%20Conceptual%20Model%20Standard%200.1.22.docx#_Toc515715083)

[Figure 29. TeeCenterToEndRun 157](file:///C:\Users\John\Dropbox\EPROD\PipelineML\TechSpec%20Files\OGC%20PipelineML%20Conceptual%20Model%20Standard%200.1.22.docx#_Toc515715084)

[Figure 30. TeeCenterToEndOutlet 157](file:///C:\Users\John\Dropbox\EPROD\PipelineML\TechSpec%20Files\OGC%20PipelineML%20Conceptual%20Model%20Standard%200.1.22.docx#_Toc515715085)

[Figure 31. TeeWallThicknessRun 158](file:///C:\Users\John\Dropbox\EPROD\PipelineML\TechSpec%20Files\OGC%20PipelineML%20Conceptual%20Model%20Standard%200.1.22.docx#_Toc515715086)

[Figure 32. TeeWallThicknessOutlet 159](file:///C:\Users\John\Dropbox\EPROD\PipelineML\TechSpec%20Files\OGC%20PipelineML%20Conceptual%20Model%20Standard%200.1.22.docx#_Toc515715087)

[Figure 33. Valve Requirements Class 161](#_Toc515715088)

[Figure 34. AppurtenantComponent Requirements Class 167](#_Toc515715089)

[Figure 35. Method 1a - Absolute Geospatial, Point or Linestring 169](#_Toc515715090)

[Figure 36. Method 1b – Absolute Geospatial, Polygon 2D 169](#_Toc515715091)

[Figure 37. Method 2a – Relative, Girth Welds, Directionality 170](#_Toc515715092)

[Figure 38. Method 2b – Relative, Componentry, Directionality 172](#_Toc515715093)

[Figure 39. Method 3 – Relative Non-directionality Location 173](#_Toc515715094)

[Figure 40. Casing Requirements Class 176](#_Toc515715095)

[Figure 41. Coating Requirements Class 185](#_Toc515715096)

[Figure 42. Sleeve Requirements Class 193](#_Toc515715097)

Abstract

The OGC PipelineML Conceptual and Encoding Model Standard defines concepts supporting the interoperable interchange of data pertaining to oil and gas pipeline systems. PipelineML supports the common exchange of oil and gas pipeline information. This initial release of the PipelineML Core addresses two critical business use cases that are specific to the pipeline industry: new construction surveys and pipeline rehabilitation. This standard defines the individual pipeline components with support for lightweight aggregation. Additional aggregation requirements such as right-of-way and land management will utilize the OGC LandInfra standards with utility extensions in the future. Future extensions to PipelineML Core will include (non limitative): cathodic protection, facility and safety. PipelineML was advanced by an international team of contributors from the US, Canada, Belgium, Norway, Netherlands, UK, Germany, Australia, Brazil, and Korea.

This standard assumes the reader has a basic understanding of oil and gas pipeline industry concepts.

Keywords

The following are keywords to be used by search engines and document catalogues.

ogcdoc, OGC document, gml, PipelineML, pml, pipeline, oil and gas, energy, utility, underground, components

Preface

This document is a deliverable of the Open Geospatial Consortium (OGC) PipelineML Standards Working Group (SWG).

Version 1.0 of the PipelineML standard defined in this document solely focuses on addressing two critical business use cases within this grouping of component use cases: new construction as-built survey, and component rehabilitation surveys. This specification defines a minimal requirement for component groupings or aggregation.

The early work of the PipelineML SWG focused on defining pipeline industry use cases. We concluded these diverse set of use cases fall into three broad groups, as illustrated in the diagram below, starting with the highest level of aggregation and working toward increased specificity and granularity. The most common group of use cases are concerned with pipeline centerline or right of way. This includes such use cases as right of way management, foreign line crossings, regulatory reporting, land division, emergency response, high consequences areas (HCA) management, public awareness, and one call (call before you dig). We concluded these use cases should leverage the collaborative work of the LandInfra DWG and SWG. We expect this pending work will involve a utility-based update/extension to LandInfra that will service of the centerline/right of way needs of the pipeline industry.

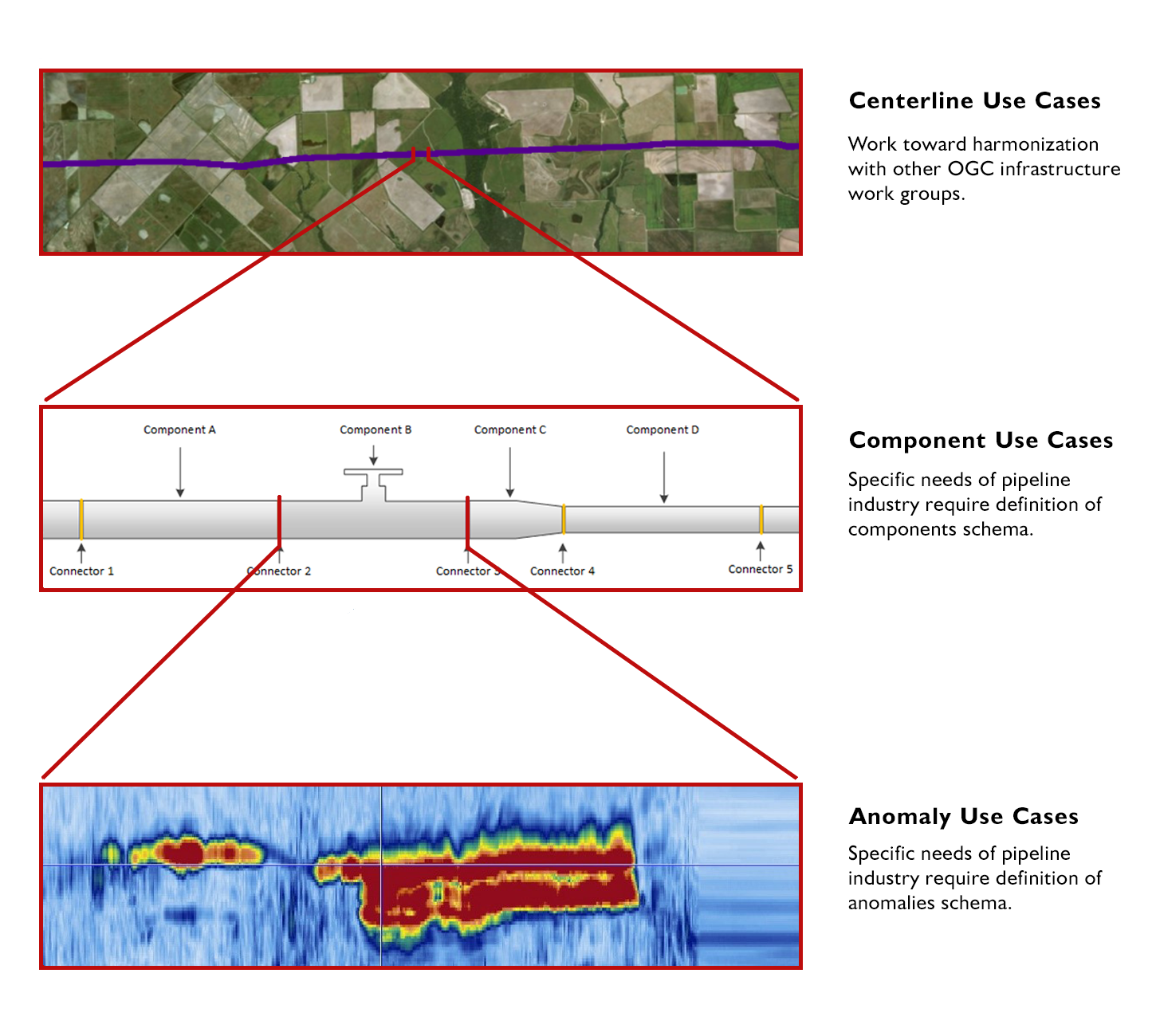


Figure 1. Use Case Grouping Levels

The second group of use cases is specific to the pipeline industry as they pertain to the management of individual pipeline component attributes required to assess and mitigate the risks of running highly volatile product through underground systems under high pressure. This includes such use cases as new construction as-built surveys, component rehabilitation surveys, component data management, and risk assessments.

The third group of use cases identify anomalies located on an individual pipeline component (subcomponent attributes). This includes such use cases as inline inspection (ILI) tool runs and remediation prioritization. These use cases shall be resolved at some point in the future as part of a modularized application schema of PipelineML that are specific to the pipeline industry.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights.

*Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.*

Security Considerations

No security considerations have been made for this standard.

Submitting organizations

The following organizations submitted this Document to the Open Geospatial Consortium (OGC):

* Enterprise Products LLP, Houston, Texas, United States of America
* Merkator NV/SA, Brussels, Belgium
* Enghouse Systems Ltd, Ontario, Canada
* BSD Consulting, Inc., Houston, Texas, United States of America

Submitters

All questions regarding this submission should be directed to the editor or the submitters:

|  |  |
| --- | --- |
| **Name** | **Affiliation** |
| John Tisdale | Enterprise Products LLP |
| Jan Stuckens | Merkator NV/SA |

# Scope

This OGC Standard defines a conceptual model and GML encoding rules pertaining to the data interchange needs of the oil and gas pipeline industry. Efforts have been made to minimize the footprint of this standard to avoid overlap with other standards pertaining to utility infrastructure. PipelineML specifies a single, modular, extensible application schema focused on defining pipeline components and their attributes with limited grouping or aggregation features.

# Conformance

This standard defines concepts for oil and gas pipeline systems. This includes a conceptual model and an XML encoding which conform to the OGC GML encoding rules. Requirements for three standardization target types are considered:

* Conceptual Model
* Encoding
* Data Instance

Such encoding standards might include, for example, GML, SQL, XML, JSON, etc. This version of the Standard supports an XML encoding that is GML-compliant.

Conformance with this standard shall be checked using all the relevant tests specified in Annex A (normative) of this document. The framework, concepts, and methodology for testing, and the criteria to be achieved to claim conformance are specified in the OGC Compliance Testing Policies and Procedures and the OGC Compliance Testing web site[[1]](#footnote-1).

In order to conform to this OGC™interface standard, a software implementation SHALL implement all requirements-classes and conformance-classes described in this document.

# References

The following normative documents contain provisions that, through reference in this text, constitute provisions of this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. For undated references, the latest edition of the normative document referred to applies.

ISO / TC 211: ISO 19103:2005 - Conceptual Schema Language (2005)

ISO / TC 211: ISO 19107:2003 - Spatial Schema (2003)

ISO 19109:2015, Geographic information – Rules for application schema

OGC 08-015r2, Abstract Specification Topic 2 – Spatial Referencing by Coordinates, (ISO 19111), April 27, 2010.

ISO / TC 211: ISO 19115:2003 – Geographic information - Metadata (also OGC Abstract Specification Topic 11) (2003)

OGC: OGC 07-036, Geography Markup Language (also ISO 19136:2007) (2007)

IETF: RFC 3986 - Uniform Resource Identifier (URI): Generic Syntax, (2005)

OMG: Unified Modeling Language (UML). Version 2.3. (2010)

W3C: Extensible Markup Language (XML) - Version 1.0 (Fourth Edition) (2006)

W3C: XML Schema - Version 1.0 (Second Edition) (2004)

# Terms and Definitions

This document uses the terms defined in Sub-clause 5.3 of [OGC 06-121r9], which is based on the ISO/IEC Directives, Part 2, Rules for the structure and drafting of International Standards. In particular, the word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this standard.

For the purposes of this document, the following additional terms and definitions apply.

## *appurtenant component*

A pipeline component that is applied to a *coterminous component*, such as a coating, casing, or sleeve.

## *assembly*

A set of related components physically attached to each other that collectively support a single primary function as part of a pressurized pipeline system.

[ISO 6707-1:2014]

## *classifier*

A classifier is an abstract UML metaclass which describes (classifies) a set of instances having common features (not to be confused with the “Feature” stereotype from the OGC Feature Model). A feature declares a structural or behavioral characteristic of instances of classifiers. (http://www.uml-diagrams.org/classifier.html). Classes, Interfaces, Association, and Types are kinds of classifiers.

## *component*

A physical pipeline component that is part of a pressurized pipeline system.

## *conceptual model*

A model that defines concepts of a universe of discourse.

[ISO 19103, 4.11, 4.23]

## *conformance class*

A set of conformance test modules that must be applied to receive a single certificate of conformance.

[OGC 08-131r3]

## *connector*

Some means of connecting two components together such that they function cohesively together as members of a pressurized pipeline system (welded joint, bolted flange, etc.).

## *coterminous component*

A physical pipeline component that is welded, threaded, or bolted together into a series of connected components that collectively comprise a pressurized pipeline system.

## *dataset*

An identifiable collection of data.

[ISO 19115]

## *domain feature*

Feature of a type defined within a particular application domain.

NOTE: This may be contrasted with observations and sampling features, which are features of types defined for cross-domain purposes.

[ISO 19156, definition 4.4]

## *element <XML>*

Basic information item of an XML document containing child elements, attributes and character data.

NOTE: From the XML Information Set ― each XML document contains one or more elements, the boundaries of which are either delimited by start-tags and end-tags, or, for empty elements, by an empty-element tag. Each element has a type, identified by name, sometimes called its ‘generic identifier’ (GI), and may have a set of attribute specifications. Each attribute specification has a name and a value.

[ISO 19136:2007]

## *feature*

An abstraction of a real-world phenomenon.

[ISO 19101:2002, definition 4.11]

## *GML application schema*

Application schema written in XML Schema in accordance with the rules specified in OGC GML 3.3.

[ISO 19136:2007]

## *GML document*

XML document with a root element that is one of the elements AbstractFeature, Dictionary or TopoComplex, specified in the GML schema or any element of a substitution group of any of these elements.

[ISO 19136:2007]

## *GML schema*

Schema components in the XML namespace ― as specified in OGC GML 3.3.

[ISO 19136:2007]

## *measurement*

Set of operations having the objective of determining the value of a quantity.

[ISO/TS 19101-2:2008, definition 4.20]

## *observation*

Act of observing a property.

NOTE:            The goal of an observation may be to measure or otherwise determine the value of a property.

[ISO 19156:2011 definition 4.10]

## *observation procedure*

Method, algorithm or instrument, or system which may be used in making an observation.

[ISO19156, definition 4.11]

## *observation result*

Estimate of the value of a property determined through a known procedure.

[ISO 19156:2011]

## *pipeline*

A continuous connected set of components, including ancillary equipment, used for transporting liquids or gases as part of a pressurized system.

[ISO 6707-1:2014]

## *property <General Feature Model>*

Facet or attribute of an object referenced by a name.

EXAMPLE: Abby’s car has the color red, where “color red” is a property of the car instance.

## *sampled feature*

The real-world domain feature of interest, such as a geological unit or structure which is observed.

[ISO 19156:2011]

## *sampling feature*

Feature, such as a station, outcrop, borehole, section or specimen, which is involved in making observations of a domain feature.

NOTE: A sampling feature is purely an artefact of the observational strategy, and has no significance independent of the observational campaign.

[ISO 19156:2011, definition 4.16]

## *schema <XML Schema>*

XML document containing a collection of schema component definitions and declarations within the same target namespace.

Example Schema components of W3C XML Schema are types, elements, attributes, groups, etc.

NOTE: The W3C XML Schema provides an XML interchange format for schema information. A single schema document provides descriptions of components associated with a single XML namespace, but several documents may describe components in the same schema, i.e. the same target namespace.

 [ISO 19136:2007]

# Conventions

This section provides details and examples for any conventions used in the document. Examples of conventions are symbols, abbreviations, use of XML schema, or special notes regarding how to read the document.

All UML diagrams in this document follow the guidance as documented in OGC OWS Common 2.0 section 5.2.

## Requirement Class

Each normative statement (requirement or recommendation) in this specification is a member of a requirements class. Each requirements class is described in a discrete clause or sub-clause, and summarized using the following template:

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/{classM}** | |
| Target type | [artefact or technology type] |
| Name | [identifier] |
| Dependency | **[identifier for another requirements class]** |
| Requirement | **/req/{classM}/{reqN}** |
| Recommendation | **/req/{classM}/{recO}** |
| Requirement | **/req/{classM}/{reqP}** |
| Requirement/ Recommendation | [repeat as necessary] |

All requirements in a requirements class must be satisfied. Hence, the requirements class is the unit of re-use and dependency, and the value of a dependency requirement is another requirements class. All requirements in a dependency must also be satisfied by a conforming implementation. A requirements class may consist only of dependencies and introduce no new requirements.

## Requirement and Recommendation

All requirements and recommendations are normative, and each is presented using the following template:

|  |
| --- |
| **Req [N]** [Normative statement]  **/req/{classM}/[reqN]** |

The identifier /req/{classM}/[reqN] defines the requirement or recommendation. The use of this layout convention allows the normative provisions of this specification to be easily located by implementers.

## Conformance Class

Conformance to this specification is possible at a number of levels, specified by conformance classes (Annex A). Each conformance class is summarized using the following template:

|  |  |
| --- | --- |
| **Conformance Class** | |
| **/conf/{classM}** | |
| Dependency | **[identifier for another conformance class]** |
| Requirements | **/req/{classA}** |
| Tests | [reference to clauses(s) containing tests] |

All tests in a class must be passed. Each conformance class tests conformance to a set of requirements packaged in a requirements class.

## Identifiers

The normative provisions in this specification are denoted by the URI

http://www.opengis.net/spec/{standard}/{m.n}

All requirements and conformance tests that appear in this document are denoted by partial URIs which are relative to this base. The identifier supports cross-referencing of class membership, dependencies, and links from each conformance test to the requirements tested. In this specification identifiers are expressed as partial URIs or paths, which can be appended to a base URI that identifies the specification as a whole in order to construct a complete URI for identification in an external context.

The URI for each requirements class has the form:

http://www.opengis.net/spec/pipelineml/1.0/req/[classM]

The URI for each requirement or recommendation has the form:

http://www.opengis.net/spec/pipelineml/1.0/req/[classM]/[reqN].

The URI for each conformance class has the form:

http://www.opengis.net/spec/pipelineml/1.0/conf/[classM].

The URI for each conformance test has the form:

http://www.opengis.net/spec/pipelineml/1.0/conf/[classM]/[testN].

## Classifiers

This document contains a large number of references to classifiers that might sometimes be ambiguous. Classes and packages are simply referred by their name formed using “camelCase” name in mono space type. Duplicate names do exist and the scope (the package of a class or the class a property belongs to) must be made explicit.

OCL syntax will be used to identify a conceptual model classifier from the UML model.

Package::{…}Package::Classifier::Property:Type

Package names are not formal in UML and can change from one implementation to another. The reference model used by PipelineML, and several other domain models, is HollowWorld. For example, a complete path for a SF\_SamplingPoint in HollowWorld (from HollowWorld root) is

ISO TC211::ISO 19156 All::ISO 19156:2011 Observations and Measurements::Sampling Features::samplingPoint::SF\_SamplingPoint

For the sake of readability, and also because some HollowWorld package names do not have OCL friendly names (e.g. some package names contain ‘:’, as shown in the previous example), this document will use shortcuts to identify packages. For example, for OM::SF\_SamplingPoint, OM acts as a shortcut for (ISO TC211::ISO 19156 All::ISO 19156:2011 Observations and Measurements::\*) that includes all classifiers in all sub packages and avoids creating a shortcut for all sub packages. The list of shortcuts is provided in Section 8.1.2. PipelineML also uses the recently published ISO19115-3 model which has numerous classifier name overlaps with ISO19115 from HollowWorld.

W3C XPath will be used in XML instances. XML entities will be identified using their full qualified name (namespace, identified by its prefix, and entity name).

# Clauses not Containing Normative Material

ISO 19101 [9] defines universe of discourse to be a view of the real or hypothetical world that includes everything of interest. That standard then defines conceptual model to be a model that defines concepts of a universe of discourse.

The scope of this PipelineML Standard establishes the limits of the universe of discourse for this Standard. The next task is to discover and standardize the concepts within this scope. Because PipelineML attempts to avoid areas of discourse that contain clear overlap with other utility infrastructure concepts, it focuses on a subset of use cases specific to the pipeline industry to eliminate overlap and redundancy within the OGC community. Additionally, this limited scope of specialized use cases is designed to address a lack of application interoperability (in terms of data interchange) within the pipeline industry. These overarching design requirements directed the development of PipelineML toward structured rigidity over flexibility in coding patterns as well as detailed specificity.

The PipelineML conceptual model is defined in UML and follows the agreed OGC and TC211 conventions for using UML. Concepts were modeled in UML via Sparx Enterprise Architect and taken through a managed lifecycle development process. The UML was transformed into GML 3.3-compliant application schemas using ShapeChange. Sample PipelineML encoded files were then validated against the resulting schemas. Delta points (i.e., warnings and errors) between the sample encodings and the generated schemas were noted and the next round of refinements were made to the UML conceptual model and the sample files. The process repeated until all delta points were resolved and the UML conceptual model produced expected and agreed encoding results. These results provided a foundation for proposing a joint conceptual model and a GML-compliant encoding standard into a unified PipelineML Standard.

# PipelineML Conceptual Model (Normative)

The PipelineML Conceptual Model is a mostly rigid framework for modeling oil and gas pipeline components. Oil and gas pipelines carry highly volatile product under pressure and mostly underground. These characteristics require precise and detailed information to assess and mitigate risks using such methods as computational fluid dynamics, aerial dispersion (for gas product), and spill flow simulation based on available elevation data (for liquid product).

To maximize application interoperability, the current version of the PipelineML Standard does not provide any mechanism for extensibility. A future revision may consider adding an extensibility framework provided that it does not negatively impact interoperability.

Creating a conceptual model that mimics the connectivity and detailed attributes of physical components is crucial to the design because the model should support well-known use cases as well as provide a foundation for those yet to be defined. Providing a framework for future use case support is potentially achievable by closely matching the model to the physical components, connectivity, and properties.

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/pipelineml-conceptual** | |
| Target type | Encoding |
| Name | PipelineML Conceptual Model |
| Dependency | **ISO19101:2002 Reference Model Clause 7** |
| Dependency | **ISO19103 2015 Conceptual Model Language** |
| Dependency | **ISO19104:2008** |
| Dependency | **Unified Modeling Language (UML). Version 2.3. May 2010** |
| **Req 1** | **/req/pipelineml-conceptual/similarity** |

Target encodings that are compliant with the conceptual model shall implement components of the conceptual model respecting their semantics, i.e. their definition and intent. In other words, the encoding must be highly semantically similar to components of the conceptual model. Semantic similarity can be tested in multiple ways, including but not limited to: (i) direct comparison of UML components, (ii) comparison after mapping components to a common expressive knowledge representation language, such as first order logic or common logic, or (iii) comparison after mapping components to a reference ontology.

|  |
| --- |
| 1. An encoding, when claiming compliance with this conceptual model, SHALL implement its components (classes, attributes, relationships) respecting the conceptual model definitions and intent, such that high semantic similarity is obtained between the encoding and conceptual model components.   **/req/pipelineml-conceptual/similarity** |

The target implementation of the conceptual model is generally an encoding specification or a schema (which could use technologies like XSD, for example) and not a data instance. The conceptual model, expressed using UML, provides naming, structure and cardinality for any physical implementation. The UML model is a normative artefact as the official representation of PipelineML. Rules that can be unambiguously inferred from the UML model will not be documented as explicit requirements. Specific encoding idiosyncrasies shall be addressed in the requirement clauses pertaining to that encoding.

## UML Model Requirement Class

The UML model provides name, structure and cardinality for data elements composing various potential physical implementations of PipelineML. There are formal mappings between UML and GML (ISO-19136). Although it is assumed that UML is technologically neutral, in reality UML models actually address some of the encoding specification details. The current PipelineML UML model has been designed as a GML application according to ISO 19109 and borrows some of artefacts of GML. Several design decisions were guided by limitations of UML (e.g. single inheritance) and XSD (package dependencies artefacts). However, the UML model is detailed enough to constrain the main elements of any encoding; the names of entities and the cardinality of properties, the associations between entities and to some extent property types.

This section defines the minimum UML mapping requirements that shall be met by any target claiming compliance to this specification.

### Structural Overview of Requirement Classes

The Requirements Classes for this standard are structured as UML Packages in Figure 2. Below is a brief summary of the function of each of these Requirements Classes.

#### PMLComponent

PMLComponent is the core Requirements Class. This class contains information about the PipelineML dataset that can contain information about pipeline component classes, their attributes, and associations. PMLComponent also contains the definition of types such as data types and external reference code lists.

#### PMLAnomaly

PMLAnomaly is a potential future Requirements Class that may be implemented as part of a modular schema design in a future update to this Standard. This class contains information about anomalies observed and measured as spatial voids (corrosion), dents, scars, gouges, etc. in surfaces of pipeline components. This information is tracked in terms of geospatial location, percentage of wall depth loss, 360-degree angle of anomaly along component walls, proximity to welds, and temporal changes of anomaly area/volume to assess risk and determine remediation prioritization. No additional information is provided for this class as it is simply noted for future potential use.

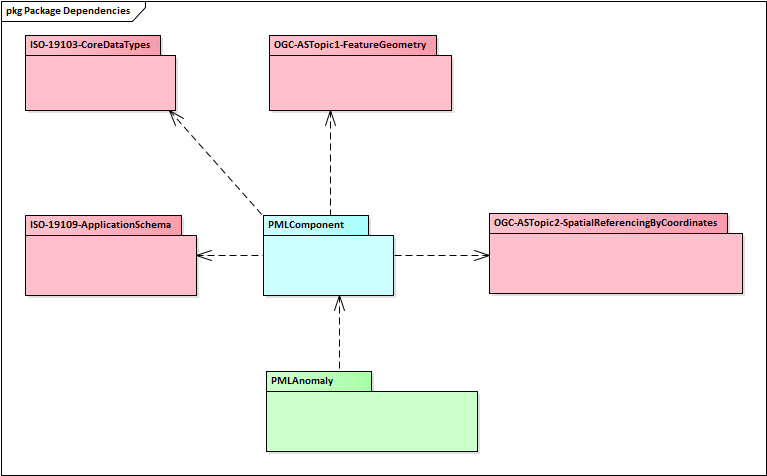


Figure 2. Encoding Requirements classes dependencies

Figure 2 also shows (external) OGC and ISO standards as Packages on which Requirements Classes in this Standard depend. Below is a brief summary of the function of each of these Standards.

#### ISO-19103-CoreDataTypes

Provides the core data types (e.g., CharacterString, Integer, Date) used in this Standard.

#### OGC-ASTopic1-FeatureGeometry

Provides most of the geometry types (e.g., Point, LineString, Polygon) used for spatial representations in this Standard.

#### OGC-ASTopic2 - SpatialReferencingByCoordinates

Defines Coordinate Reference Systems.

#### ISO-19109-ApplicationSchema

Defines the General Feature Model upon which this Standard is based.

#### Property Cardinality

All properties that could feasibly be made optional are optional in PipelineML. Structural rigidity is used to ensure consistency, application interoperability, and to streamline application development. Some degree of flexibility is provided through the omission of optional properties. This allow data packages to be as concise as possible.

### Package Shortcuts

The following shortcuts are used to refer to external (non-PipelineML) classifiers.

|  |  |
| --- | --- |
| **Shortcut** | **Full path** |
| **GEO** | ISO TC211/ISO 19107 All/ISO 19107:2003 Spatial Schema |
| **Primitive** | ISO TC211/ISO 19103 All/ISO 19103:2005 Conceptual schema language |
| **GML** | ISO TC211/ISO 19136:2007 GML |

## PipelineML Abstract Requirements Class (Normative)

|  |  |
| --- | --- |
| **Abstract Requirements Class** | |
| **/req/pipelineml-abstract** | |
| Target type | Encoding |
| Name | PipelineML Abstract Requirements Class |
| Dependency | **ISO19103 2015 Conceptual Model Language** |
| Dependency | **ISO19107:2003 Spatial Schema** |
| Dependency | **ISO19109:2015 Rules for application schemas** |
| Dependency | **RFC 3986 Uniform Resource Identifier (URI): Generic Syntax** |
| Dependency | **ISO19115-3 Metadata** |
| **Req 2** | **/req/pipelineml-abstract/uml-entity-name** |
| **Req 3** | **/req/pipelineml-abstract/uml-cardinality** |
| **Req 4** | **/req/pipelineml-abstract/uml-abstract** |
| **Req 5** | **/req/pipelineml-abstract/uml-polymorphism** |
| **Req 6** | **/req/pipelineml-abstract/quantities-uom** |
| **Req 7** | **/req/pipelineml-abstract/codelist** |

This section presents requirements to which all target encodings must conform in to order to claim compliance to PipelineML.

### Naming of entities

|  |
| --- |
| 1. When the target implementation allows it, the exact name of the classifier SHALL be used.   **/req/pipelineml-abstract/uml-entity-name** |

If a target implementation is capable of encoding all the artefacts (classes and properties) using the same names used in UML, it shall do so.

### Cardinality

|  |
| --- |
| 1. If the target implementation allows it, it SHALL implement the same cardinality of properties and associations as defined in the UML.   **/req/pipelineml-abstract/uml-cardinality** |

Cardinality shall be the same as defined in UML model.  Since essentially all properties are optional, this clause addresses the upper bounds of cardinality: “1” or “many” in almost all cases.  Therefore, if the UML model limits a property’s maximum cardinality to “1”, then the target implementation cardinality cannot be “many”.

### Abstract Classes

|  |
| --- |
| 1. Abstract classes SHALL NOT be materialised.   **/req/pipelineml-abstract/uml-asbstract** |

Not all physical implementations support the concept of an abstract class, or even inheritance and polymorphism. XSD does support that concept and all its implications, but JSON does not – although JavaScript can somewhat. This requirement specifies that the encoding specification shall not allow materialization of an instance of a class stereotyped as abstract. In the UML diagram shown in Figure 3, abstract classes are shown with yellow backgrounds while concrete classes are displayed with blue backgrounds.

### Polymorphism

|  |
| --- |
| 1. A target implementation SHALL implement type substitutions inferred from the UML model.   **/req/pipelineml-abstract/uml-polymorphism** |

The type hierarchy of the UML model implies type substitutions for property values. Many property types are abstract types and only a concrete subtype may be materialised (as per /req/pipelineml-abstrac/uml-abstract). A target implementation shall consider type substitutions using mechanisms available for this implementation.

### Quantities

|  |
| --- |
| 1. Quantities and measurements SHALL have explicit units of measure from a governed ontology.   **/req/pipelineml-abstract/quantities-uom** |

The quantities and measurements units of measure shall be taken from the PipelineML UOM class. This is an externally managed controlled vocabulary.

### Code List

|  |
| --- |
| 1. Classes with stereotype <<CodeList>> SHALL be implemented as externally governed vocabularies which terms are encoded as URIs (RFC 3986).   **/req/pipelineml-abstract/codelist** |

All properties that require formal vocabularies are modelled in UML as classes having the stereotype <<CodeList>>. All codelist classes in PipelineML define a restricted vocabulary with a prescribed URI (RFC 3986) containing authoritative codes and definitions defined as a GML 3.3 dictionary. Compliance to the PipelineML Standard includes compliance with the prescribed authoritative data. A mechanism is defined within the PipelineML Standard for identification of substandard data.

## Linked Open Data Requirements Class (Normative)

|  |  |
| --- | --- |
| **Abstract Requirements Class** | |
| **/req/pipelineml-lod** | |
| Target type | Encoding |
| Name | Linked Open Data Requirements Class |
| Dependency | **URI** |
| Dependency | **HTTP** |
| **Req 8** | **/req/pipelineml-lod/codelistURI** |
| **Recommendation 9** | **/req/pipelineml-lod/codelistURIexception** |
| **Req 10** | **/req/pipelineml-lod/identifier** |
| **Req 11** | **/req/pipelineml-lod/byref** |

Although OGC standards are not restricted to a web environment, they are strongly influenced by this environment. PipelineML was originally developed specifically for XML (due to its robust, mature capabilities as an ecosystem capable of supporting the weight of a data interchange standard), but many other encodings are suitable hypermedia formats (RDF/XML, JSON-LD, HTML). This requirements class describes extra rules that shall be implemented to turn PipelineML data instances into hypermedia compatible with Linked Open Data principles.

Linked Open Data is a method to publish structured data on the web. It leverages existing web technologies such as HTTP (transfer protocol) and URI (addressing over the web) to connect structured resources. The principle is similar to interconnected web pages through hyperlinks, except that pages are replace with structured information that can be processed by machines.

### Code List URI

|  |
| --- |
| 1. All URIs used for reference code lists SHALL conform to the official URI governed by the OGC PipelineML Standards Working Group (SWG) or an officially designated governing body granted responsibility for managing PipelineML reference code lists, codes, terms, definitions, and associated attributes (PipelineML Authoritative Data Governance Body).   **/req/pipelineml-lod/codelistURI** |

All reference code lists that conform to the codelistURI construct SHALL conform to the official URI governed by the OGC PipelineML Standards Working Group (SWG) or an officially designated governing body granted responsibility for managing PipelineML reference code lists, codes, terms, definitions, and associated attributes (PipelineML Authoritative Data Governance Body). A URI used for vocabulary terms that utilizes PipelineML authoritative code values SHALL be dereferenceable to one or more representations of the vocabulary term.

|  |
| --- |
| 1. **[Recommendation]** Certain PipelineML codelists are considered to be immature as the codes have not been fully developed and vetted across pipeline industry experts. As such, these codelists are designated to permit the use of non-authoritative codes. In these designated code lists, if a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following 5 guidelines.   **/req/pipelineml-lod/codelistURIexception** |

Certain PipelineML codelists are considered to be immature as the codes have not been fully developed and vetted across pipeline industry experts. As such, these codelists are designated to permit the use of non-authoritative codes. In these designated code lists, if a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:

1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone wanted to encode the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used.
2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier.
3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset.
4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain.
5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.

### Identifier

|  |
| --- |
| 1. HTTP URI used as entity identifiers SHALL be to one or more representations of that entity.   **/req/pipelineml-lod/identifier** |

This requirement demands that the target ensures that a data instance exposes a URI as a unique identifier for this feature and this identifier SHALL be dereferenceable to one or more representations of that feature.

### ByReference Associations

|  |
| --- |
| 1. External references to an entity conforming to [/req/gsml4-lod/identifier] shall be expressed using this entity identifier.   **/req/pipelineml-lod/byref** |

Serialization of a dataset will often omit the full description of a feature and replace the property value with an external reference. A reference to this feature is formed by the dereferenceable identifier described in clause 8.3.3. A client ingesting the dataset can use this reference to extract a feature representation if need be. Over the web, this reference shall be a HTTP URI that can be dereferenced to one or more representations of that feature.

## PipelineML Encoding Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/pipelineml-encoding** | |
| Target type | Encoding |
| Name | PipelineML Encoding Model |
| Dependency | **/req/pipelineml-conceptual** |
| Dependency | **ISO19101:2002 Reference Model Clause 7** |
| Dependency | **ISO19103 2015 Conceptual Model Language** |
| Dependency | **ISO19104:2008** |
| Dependency | **Unified Modeling Language (UML). Version 2.3. May 2010** |
| **Req 12** | **/req/pipelineml-encoding/similarity** |

Target encodings that are compliant with the conceptual model shall implement components of the conceptual model respecting their semantics, i.e. their definition and intent. In other words, encodings must be highly semantically similar to components of the conceptual model. Semantic similarity can be tested in multiple ways, including but not limited to: (i) direct comparison of UML components, (ii) comparison after mapping components to a common expressive knowledge representation language, such as first order logic or common logic, or (iii) comparison after mapping components to a reference ontology.

|  |
| --- |
| 1. A target encoding, when claiming compliance with this conceptual model, SHALL implement its components (classes, attributes, relationships) respecting the conceptual model definitions and intent, such that high semantic similarity is obtained between the encoding and conceptual model components.   **/req/pipelineml-encoding/similarity** |

## PMLComponent Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/pmlcomponent** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent |
| Dependency | **/req/pipelineml-conceptual** |
| Dependency | **urn:iso:is:iso:19103:clause:7** |
| Dependency | **http://www.opengis.net/doc/AS/Topic1** |
| Dependency | **urn:iso:is:iso:19109:req:uml:feature** |
| Dependency | **http://www.opengis.net/doc/AS/Topic2** |
| **Req 13** | **/req/pmlcomponent/dataset** |
| **Req 14** | **/req/pmlcomponent/classes** |
| **Req 15** | **/req/pmlcomponent/crs** |
| **Req 16** | **/req/pmlcomponent/19103** |
| **Req 17** | **/req/pmlcomponent/topic-1** |
| **Req 18** | **/req/pmlcomponent/19109** |
| **Req 19** | **/req/pmlcomponent/boundtypes** |

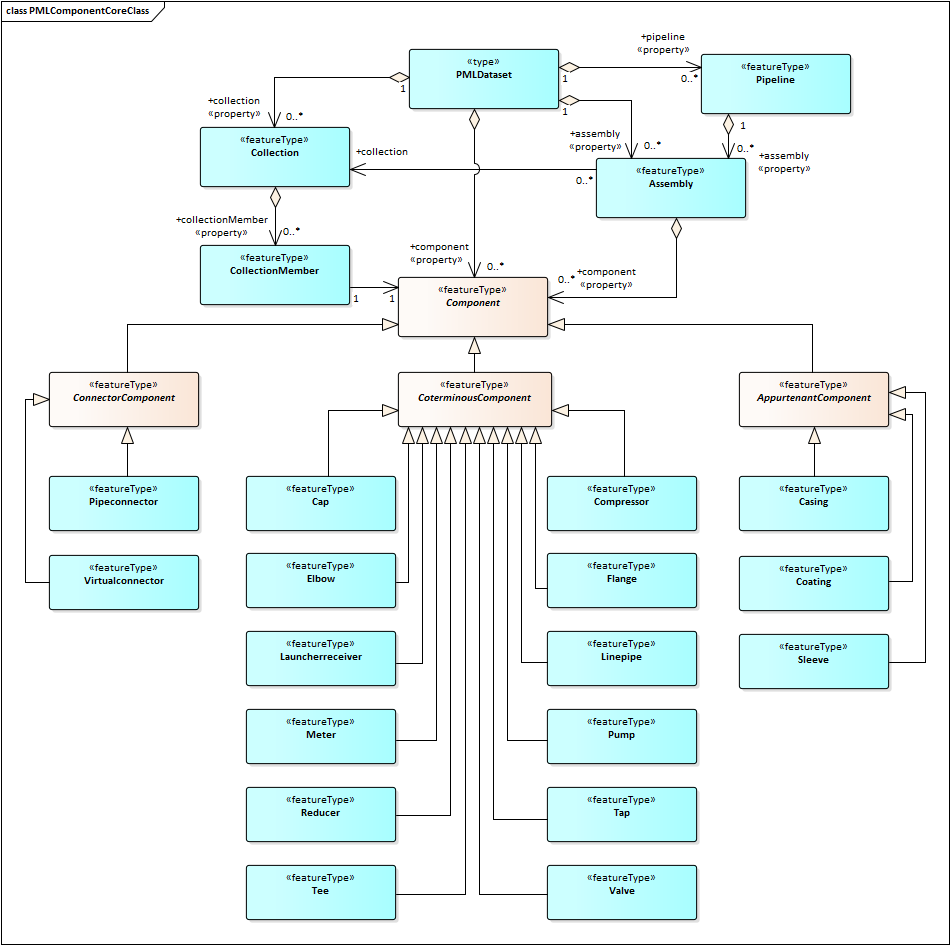


Figure 3. PMLComponent Core Requirements Class

Although PipelineML is designed to support a modular framework, the initial release of the standard contains only a single GML application schema by the name of PMLComponent(.xsd). This small footprint schema is designed to address some specific use cases within the pipeline industry. PMLComponent is the core requirements class, as shown in Figure 3. The root class, PMLComponent, specifies the PMLDataset and all classes contained within this requirement class.

|  |
| --- |
| 1. A PipelineML encoding shall specify a PMLDataset class in whatever format that is appropriate to that encoding (e.g., an XML document for GML encoding).   **/req/pmlcomponent/dataset** |

The dataset may contain any number of occurrences of each of the following classes: pipeline, assembly, collection, component, pipeconnector, virtualconnector, cap, compressor, elbow, flange, launcherreceiver, linepipe, meter, pump, reducer, tap, tee, valve, casing, coating, and sleeve.

|  |
| --- |
| 1. The PMLComponent Requirements Class Classes (shown in blue in Figure 3) SHALL be provided by the encoding in a manner consistent with the encoding.   **/req/pmlcomponent/classes** |

|  |
| --- |
| 1. A PipelineML encoding SHALL support coordinate reference systems in accordance with OGC Abstract Specification Topic 2, Spatial Referencing by Coordinates.   **/req/pmlcomponent/crs** |



Figure 4. ISO 19103 Core Data Types

|  |
| --- |
| 1. A PipelineML encoding SHALL support the core data types specified in ISO 19103 that are appropriate to the supported subject area(s).   **/req/pmlcomponent/19103** |

|  |
| --- |
| 1. A PipelineML encoding SHALL support the geometry types specified in the OGC Abstract Specification Topic 1, Feature Geometry that are appropriate to the supported subject area(s). Additional geometry types, not found in Topic 1, but required by a specific requirements class, are specified in that requirements class.   **/req/pmlcomponent/topic-1** |



Figure 5. ISO 19109 - Rules for application schema – Feature Type

|  |
| --- |
| 1. A PipelineML encoding SHALL support the uml:feature requirement(s) in ISO 19109:2015 that are appropriate to the supported subject area(s). Each instance of FeatureType shall be implemented by the encoding’s equivalent of a UML Class having a generalization association with AnyFeature and with a stereotype of <<FeatureType>>.   **/req/pmlcomponent/19109** |

|  |
| --- |
| 1. A PipelineML encoding SHALL support the use of bound data constructs. These constructs support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type is dereferencable to a concatenated string of attributes separated by semicolons. The utilization of bound data types is considered a best practice over the use of unbound data types, when available, as it represents the highest level of data pedigree.   **/req/pmlcomponent/boundtypes** |

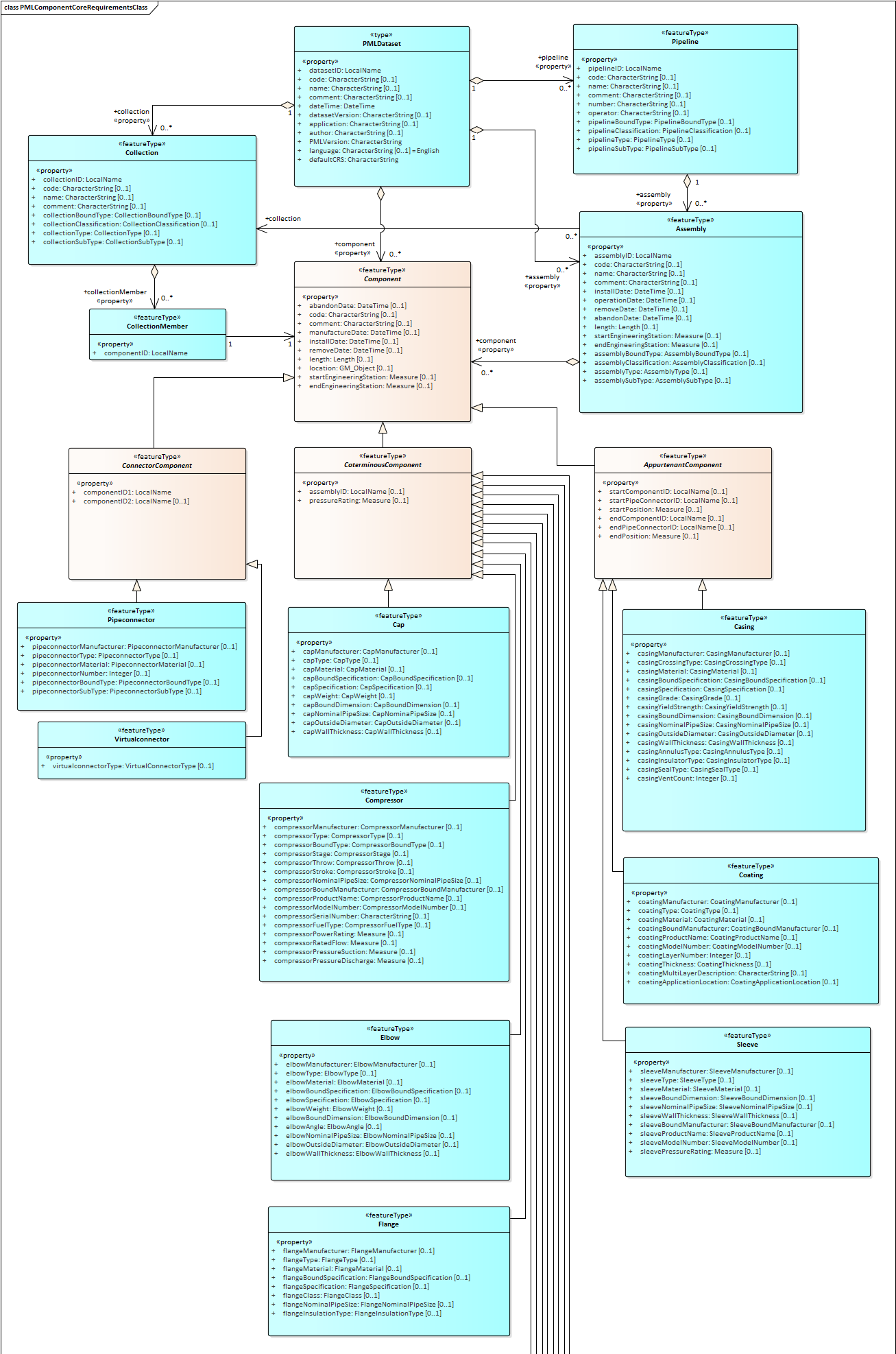




Figure 6. PMLComponent Requirements Class Classes

### PMLDataset Class Requirements Class (Normative)

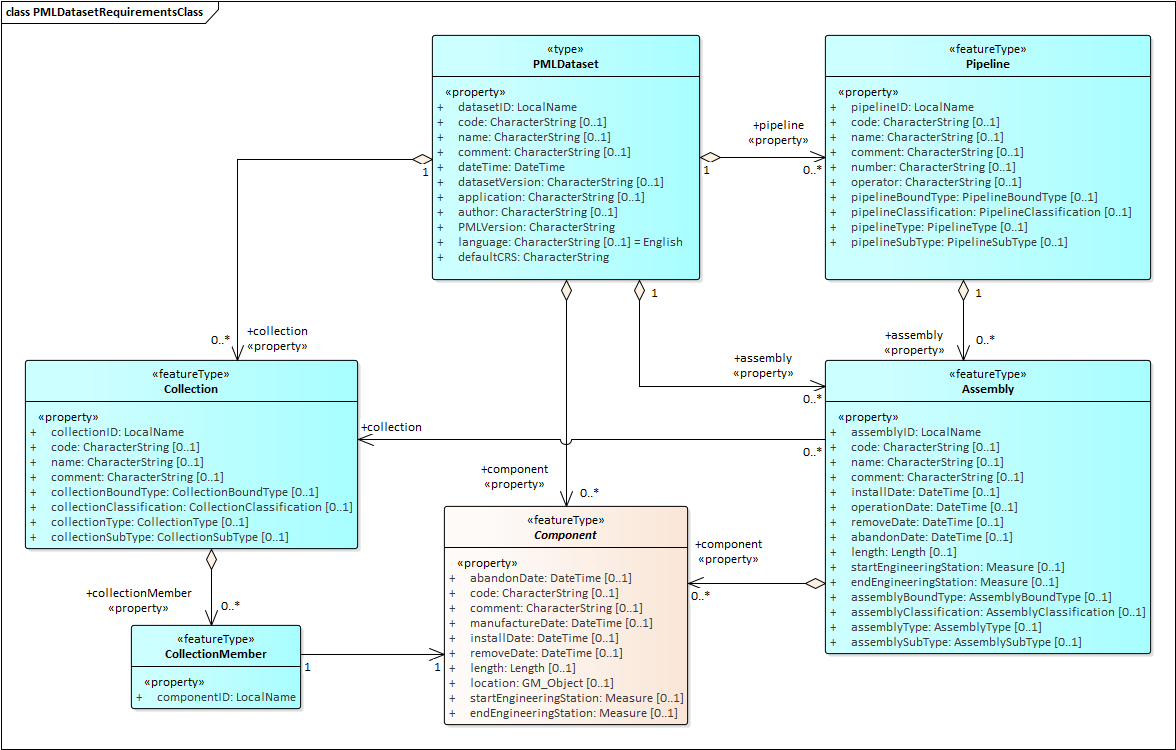


Figure 7. PMLDataset Requirements Class

The PMLDataset Class is a concrete class and therefore may be encoded into a PipelineML dataset. Figure 4 shows the key class relationships related to the PMLDataset class (note that the abstract and concrete classes that inherit from the Component class are not shown in this diagram for readability). The PMLDataset class contains header information (metadata) about the PipelineML dataset and relationships to other classes:

**datasetID**: A required machine-readable globally unique instance identifier for this feature. This identifier is unique within a PipelineML data package and is retained for the same piece of componentry throughout all subsequent PipelineML versions being distributed.

**code**: An optional human-readable string that is unique within the context of this entity (component) type, but not globally unique across all entity types

**name**: An optional call name by which the classes grouped in this class can commonly be references

**comment**: An optional comment pertaining to the dataset

**dateTime**: The required date and time that the dataset was created and therefore the point-in-time for which the data is valid

**datasetVersion**: The optional version number of the dataset

**application**: The optional software application (including version) used to generate the dataset

**author**: The optional person or organization which created this dataset

**pmlVersion**: The required version number of the PipelineML encoding standard used to generate this dataset

**language**: The optional language used in the dataset for CharacterString data types, default = English

**defaultCRS**: The required default coordinate reference system which is used for all spatial representations within the dataset except where overridden by the coordinate reference system defined for an individual pipeline component

**pipeline**: any number of Pipelines

**assembly**: any number of Assemblies

**collection**: any number of Collections

**collectionMember**: any number of CollectionMembers

**component**: The Component Class is an abstract class and SHALL NOT be encoded.

**connectorComponent**: The ConnectorComponent Class is an abstract class and SHALL NOT be encoded.

**coterminousComponent**: The CoterminousComponent Class is an abstract class and SHALL NOT be encoded.

**appurtenantComponent**: The AppurtenantComponent Class is an abstract class and SHALL NOT be encoded.

**pipeconnector**: any number of PipeConnectors

**virtualconnector**: any number of VirtualConnectors

**cap**: any number of Caps

**compressor**: any number of Compressors

**elbow**: any number of Elbows

**flange**: any number of Flanges

**launcherreceiver**: any number of Launcherreceivers

**linepipe**: any number of Linepipes

**meter**: any number of Meters

**pump**: any number of Pumps

**reducer**: any number of Reducers

**tap**: any number of Taps

**tee**: any number of Tees

**valve**: any number of Valves

**casing**: any number of Casing

**coating**: any number of Coatings

**sleeve**: any number of Sleeves

#### Pipeline Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/pipeline** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Pipeline |
| Dependency | **/req/pmlcomponent** |
| **Req 20** | **/req/pipeline/pipelineboundtype-boundtype** |
| **Req 21** | **/req/pipeline/pipelineboundtype-codelistURI** |
| **Req 22** | **/req/pipeline/pipelineboundtype-exclusivity** |
| **Req 23** | **/req/pipeline/pipelineclassification-codelistURI** |
| **Req 24** | **/req/pipeline/pipelineclassification-exclusivity** |
| **Req 25** | **/req/pipeline/pipelinetype-codelistURI** |
| **Req 26** | **/req/pipeline/pipelinetype-exclusivity** |
| **Req 26** | **/req/pipeline/pipelinesubtype-codelistURI** |
| **Req 27** | **/req/pipeline/pipelinesubtype-exclusivity** |

The Pipeline Class is a concrete class and therefore MAY be encoded into a PipelineML dataset. This class is an aggregation of features designed to represent a comprehensive set of pipeline components that collectively define a pressurized pipeline system capable of carry product. This class contains classes and attributes as shown in Figure 8.

The Pipeline Class is rigidly modeled after the physical connectivity of pipeline components. If a Pipeline feature is defined, it SHALL contain one or more Assembly Classes. An Assembly Class defines a segment of pipeline components that perform a single primary function (e.g. mainline assembly, launcherreceiver assembly, storage rack assembly, etc.).

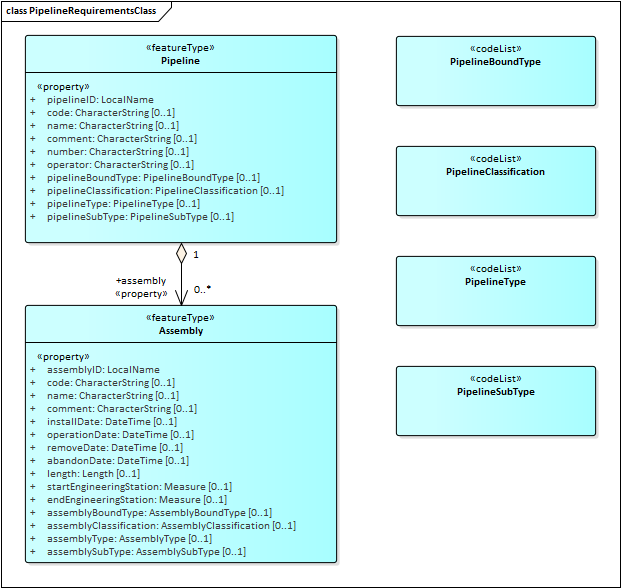


Figure 8. Pipeline Requirements Class

The Pipeline Requirements Class contains attributes and relationships to other classes:

**pipelineID**: A required machine-readable globally unique instance identifier for this feature. This identifier is unique within a PipelineML data package and is retained for the same piece of componentry throughout all subsequent PipelineML versions being distributed.

**code**: An optional string that is unique within the context of this entity (component) type, but not globally unique across all entity types

**name**: An optional name of the Pipeline

**comment**: An optional string that allows for text comment to be provided pertaining to a specific instance of an assembly

**number**: An optional human-readable number used to define this class

**pipelineBoundType**: An optional value that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: pipelineClassification; pipelineType; pipelineSubType.

|  |
| --- |
| 1. This optional bound attribute pipelineBoundType SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/pipeline/pipelineboundtype-boundtype** |

|  |
| --- |
| 1. This optional bound attribute pipelineBoundType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pipelineboundtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/pipeline/pipelineboundtype-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute pipelineBoundType MAY be populated, or its corresponding set of unbound attributes (pipelineClassification, pipelineType, and pipelineSubType) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pipeline/pipelineboundtype-exclusivity** |

**pipelineClassification**: An optional value that, if populated, SHALL be dereferenceable to a string value is used to define the classification of the pipeline. This attribute SHALL NOT be provided if the attribute pipelineBoundType is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute pipelineClassification SHALL utilize the PipelineML codelistURIconstruct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pipelineclassification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/pipeline/pipelineclassification-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute pipelineClassification MAY be populated, or its corresponding bound attribute pipelineBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pipeline/pipelineclassification-exclusivity** |

**pipelineType**: An optional value is used to define the type of the pipeline. This attribute SHALL NOT be provided if the attribute pipelineBoundType is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute pipelineType SHALL utilize the PipelineML codelistURIconstruct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pipelinetype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/pipeline/pipelinetype-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute pipelineType MAY be populated, or its corresponding bound attribute pipelineBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pipeline/pipelinetype-exclusivity** |

**pipelineSubType**: An optional value is used to define the subtype of the pipeline. This attribute SHALL NOT be provided if the attribute pipelineBoundType is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute pipelineSubType SHALL utilize the PipelineML codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pipelinesubtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/pipeline/pipelinesubtype-codelistURI** |

|  |
| --- |
| 1. The optional unbound attribute pipelineSubType MAY be populated, or its corresponding bound attribute pipelineBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pipeline/pipelinesubtype-exclusivity** |

**assembly**: any number of Assemblies

#### Assembly Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/assembly** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Assembly |
| Dependency | **/req/pmlcomponent** |
| **Req 29** | **/req/assembly/componentexclusivity** |
| **Req 30** | **/req/assembly/assemblyboundtype-boundtype** |
| **Req 31** | **/req/assembly/assemblyboundtype-codelistURI** |
| **Req 32** | **/req/assembly/assemblyboundtype-exclusivity** |
| **Req 33** | **/req/assembly/assemblyclassification-codelistURI** |
| **Req 34** | **/req/assembly/assemblyclassification-exclusivity** |
| **Req 35** | **/req/assembly/assemblytype-codelistURI** |
| **Req 36** | **/req/assembly/assemblytype-exclusivity** |
| **Req 37** | **/req/assembly/assemblysubtype-codelistURI** |
| **Req 38** | **/req/assembly/assemblysubtype-exclusivity** |

The Assembly Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class is an aggregation of features designed to provide a single primary function (e.g. mainline assembly, launcherreceiver assembly, storage rack assembly, etc.). This class contains classes and attributes as shown in Figure 9. This diagram does not include concrete classes that inherit from the ConnectorComponent, Component, and AppurtenantComponent class as these classes are shown on previous diagrams.

|  |
| --- |
| 1. Components SHALL be exclusive to a single assembly; that is, no component may exist in more than one assembly (as defined by the component identity attribute).   **/req/assembly/componentexclusivity** |

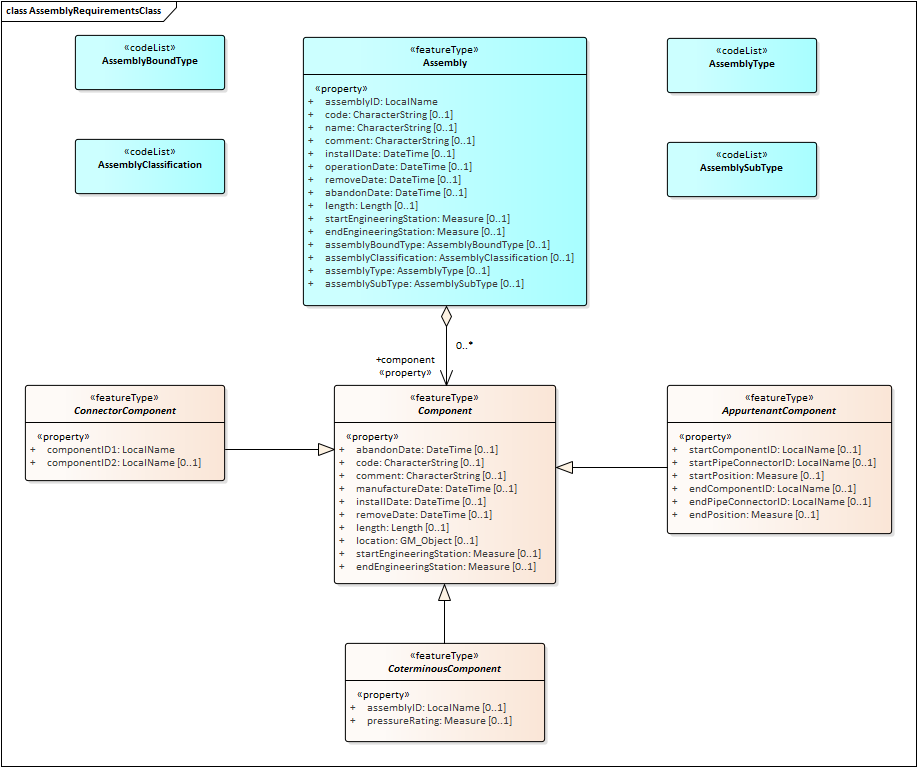


Figure 9. Assembly Requirements Class

The Assembly Class contains attributes and relationships to other classes:

**assemblyID**: A required machine-readable globally unique instance identifier for this feature. This identifier is unique within a PipelineML data package and is retained for the same piece of componentry throughout all subsequent PipelineML versions being distributed.

**code**: An optional human-readable string that is unique within the context of this entity (component) type, but not globally unique across all entity types

**name**: An optional call name by which the classes grouped in this class can commonly be references

**comment**: An optional human-readable comment pertaining to this class

**installDate**: An optional date when the assembly was installed on the pipeline system

**operationDate**: An optional date when the assembly was put into operation in the pipeline system

**removeDate**: An optional date when the assembly was removed from the pipeline system

**abandonDate**: An optional date when the assembly was abandoned from service

**length**: An optional total length of the assembly

**startEngineeringStation**: An optional surveyed engineering station value at the start of the object. This static attribute is designed to support legacy business use cases but is not part of a dynamic linear reference system.

**endEngineeringStation**: An optional surveyed engineering station value at the end of the object. This static attribute is designed to support legacy business use cases but is not part of a dynamic linear reference system.

**assemblyBoundType**: An optional value that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: assemblyClassification; assemblyType; assemblySubType.

|  |
| --- |
| 1. This optional bound attribute assemblyBoundType SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/assembly/assemblyboundtype-boundtype** |

|  |
| --- |
| 1. This optional bound attribute assemblyBoundType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/assemblyboundtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/assembly/assemblyboundtype-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute assemblyBoundType MAY be populated, or its corresponding set of unbound attributes (assemblyClassification, assemblyType, and assemblySubType) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/assembly/assemblyboundtype-exclusivity** |

**assemblyClassification**: A value that, if populated, SHALL be dereferenceable to a string value is used to define the classification of the pipeline. This attribute SHALL NOT be provided if the attribute assemblyBoundType is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute assemblyClassification SHALL utilize the PipelineML codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/assemblyclassification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/assembly/assemblyclassification-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute assemblyClassification MAY be populated, or its corresponding bound attribute assemblyBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/assembly/assemblyclassification-exclusivity** |

**assemblyType**: An optional value is used to define the type of the pipeline. This attribute SHALL NOT be provided if the attribute assemblyBoundType is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute assemblyType SHALL utilize the PipelineML codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/assemblytype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/assembly/assemblytype-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute assemblyType MAY be populated, or its corresponding bound attribute assemblyBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/assembly/assemblytype-exclusivity** |

**assemblySubType**: An optional value is used to define the subtype of the pipeline. This attribute SHALL NOT be provided if the attribute assemblyBoundType is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute assemblySubType SHALL utilize the PipelineML codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/assemblysubtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/assembly/assemblysubtype-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute assemblySubType MAY be populated, or its corresponding bound attribute assemblyBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/assembly/assemblysubtype-exclusivity** |

**component**: The Component Class is an abstract class and SHALL NOT be encoded.

**connectorComponent**: The ConnectorComponent Class is an abstract class and SHALL NOT be encoded.

**coterminousComponent**: The CoterminousComponent Class is an abstract class and SHALL NOT be encoded.

**appurtenantComponent**: The AppurtenantComponent Class is an abstract class and SHALL NOT be encoded.

**pipeconnector**: any number of PipeConnectors

**virtualconnector**: any number of VirtualConnectors

**cap**: any number of Caps

**compressor**: any number of Compressors

**elbow**: any number of Elbows

**flange**: any number of Flanges

**launcherreceiver**: any number of Launcherreceivers

**linepipe**: any number of Linepipes

**meter**: any number of Meters

**pump**: any number of Pumps

**reducer**: any number of Reducers

**tap**: any number of Taps

**tee**: any number of Tees

**valve**: any number of Valves

**casing**: any number of Casing

**coating**: any number of Coatings

**sleeve**: any number of Sleeves

#### Collection Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/collection** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Collection |
| Dependency | **/req/pmlcomponent** |
| **Req 39** | **/req/collection/collectionboundtype-boundtype** |
| **Req 40** | **/req/collection/collectionboundtype-codelistURI** |
| **Req 41** | **/req/collection/collectionboundtype-exclusivity** |
| **Req 42** | **/req/collection/collectionclassification-codelistURI** |
| **Req 43** | **/req/collection/collectionclassification-exclusivity** |
| **Req 44** | **/req/collection/collectiontype-codelistURI** |
| **Req 45** | **/req/collection/collectiontype-exclusivity** |
| **Req 46** | **/req/collection/collectionsubtype-codelistURI** |
| **Req 47** | **/req/collection/collectionsubtype-exclusivity** |

The Collection Class is a concrete class and therefore may be encoded into a PipelineML dataset. Whereas the Pipeline and Assembly Classes are rigidly designed to mimic the connectivity of physical pipeline components, the Collection Class provides flexible aggregation of components to support a variety of business needs. This class contains classes and attributes as shown in Figure 10.

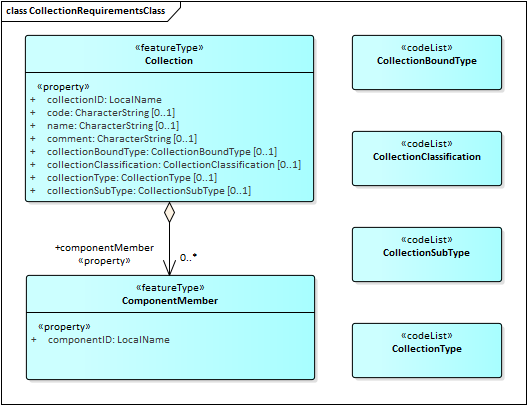


Figure 10. Collection Requirements Class

The Collection Class contains attributes and relationships to other classes:

**collectionID**: A required machine-readable globally unique instance identifier for this feature. This identifier is unique within a PipelineML data package and is retained for the same piece of componentry throughout all subsequent PipelineML versions being distributed.

**code**: An optional human-readable string that is unique within the context of this entity (component) type, but not globally unique across all entity types

**name**: An optional call name by which the classes grouped in this class can commonly be references

**comment**: An optional human-readable comment pertaining to this class

**collectionBoundType**: An optional value that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: collectionClassification; collectionType; collectionSubType.

|  |
| --- |
| 1. This optional bound attribute collectionBoundType SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/collection/collectionboundtype-boundtype** |

|  |
| --- |
| 1. This optional bound attribute collectionBoundType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/collectionboundtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/collection/collectionboundtype-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute collectionBoundType MAY be populated, or its corresponding bound set of attributes (collectionClassification, collectionType, and collectionSubType) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/collection/collectionboundtype-exclusivity** |

**collectionClassification**: An optional value that, if populated, SHALL be dereferenceable to a string value is used to define the classification of the pipeline. This attribute SHALL NOT be provided if the attribute collectionBoundType is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute collectionClassification SHALL utilize the PipelineML codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/collectionclassification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/collection/collectionclassification-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute collectionClassification MAY be populated, or its corresponding bound attribute collectionBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/collection/collectionclassification-exclusivity** |

**collectionType**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of collection. This attribute SHALL NOT be provided if the attribute collectionBoundType is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute collectionType SHALL utilize the PipelineML codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/collectiontype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/collection/collectiontype-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute collectionType MAY be populated, or its corresponding bound attribute collectionBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/collection/collectiontype-exclusivity** |

**collectionSubType**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the subtype of collection. This attribute SHALL NOT be provided if the attribute collectionBoundType is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute collectionSubType SHALL utilize the PipelineML codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/collectionsubtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/collection/collectionsubtype-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute collectionSubType MAY be populated, or its corresponding bound attribute collectionBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/collection/collectionsubtype-exclusivity** |

**collectionMember**: Any number of CollectionMembers. A CollectionMember Class is a concrete class that has a one-to-one relationship with the abstract Component Class.

#### Component Class Requirements Class (Normative)

The Component Class is an abstract class and SHALL NOT be encoded into a PipelineML dataset. This abstract class provides an inheritance mechanism for passing attributes on to other classes that inherit from it. This class also establishes relationships with other classes in the model. This class contains classes and attributes as shown in Figure 11 (the concrete classes generalized from the ConnectorComponent, CoterminousComponent, and AppurtenantComponent classes are not shown as they are shown in other diagrams in this document).

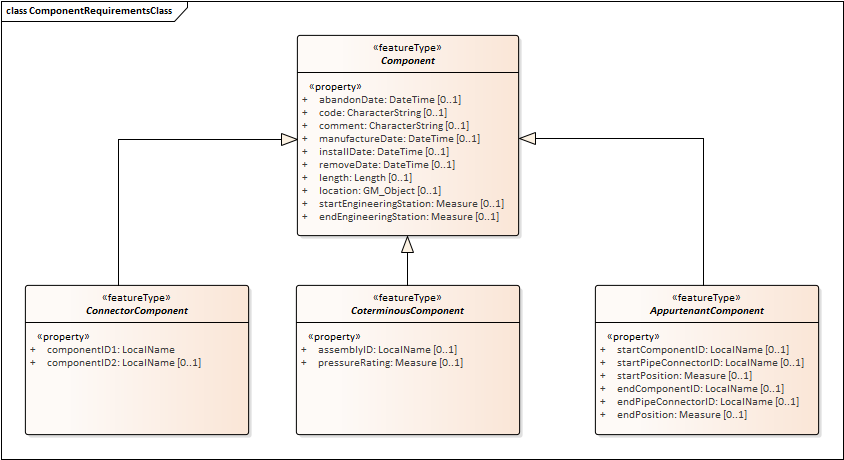


Figure 11. Component Requirements Class

The Component Class contains attributes and relationships to other classes:

**abandonDate**: An optional date when the assembly was abandoned from service

**code**: An optional human-readable string that is unique within the context of this entity (component) type, but not globally unique across all entity types

**comment**: An optional human-readable comment pertaining to this class

**manufacturerDate**: An optional date when the component was manufactured

**installDate**: An optional date when the component was installed on the pipeline system

**removeDate**: An optional date when the assembly was removed from the pipeline system

**length**: An optional total length of the assembly

**location**: One or more optional geospatial coordinates (x, y; x, y, z,; x, y, z, m) based on a specified Coordinate Reference System (CRS) in 0, 1, or 2 spatial dimensions, derived from the GML GM\_Object class.

**startEngineeringStation**: An optional surveyed engineering station value at the start of the object. This static attribute is designed to support legacy business use cases but is not part of a dynamic linear reference system.

**endEngineeringStation**: An optional surveyed engineering station value at the end of the object. This static attribute is designed to support legacy business use cases but is not part of a dynamic linear reference system.

**connectorComponent**: The ConnectorComponent Class is an abstract class and SHALL NOT be encoded.

**coterminousComponent**: The CoterminousComponent Class is an abstract class and SHALL NOT be encoded.

**appurtenantComponent**: The AppurtenantComponent Class is an abstract class and SHALL NOT be encoded.

**pipeconnector**: any number of PipeConnectors

**virtualconnector**: any number of VirtualConnectors

**cap**: any number of Caps

**compressor**: any number of Compressors

**elbow**: any number of Elbows

**flange**: any number of Flanges

**launcherreceiver**: any number of Launcherreceivers

**linepipe**: any number of Linepipes

**meter**: any number of Meters

**pump**: any number of Pumps

**reducer**: any number of Reducers

**tap**: any number of Taps

**tee**: any number of Tees

**valve**: any number of Valves

**casing**: any number of Casing

**coating**: any number of Coatings

**sleeve**: any number of Sleeves

#### ConnectorComponent Class Requirements Class (Normative)

The ConnectorComponents Class is abstract and SHALL NOT be encoded into a PipelineML dataset. This abstract class provides an inheritance mechanism for passing attributes on to other classes that inherit from it. This class also establishes relationships with other classes in the model. This class contains classes and attributes as shown in Figure 12.

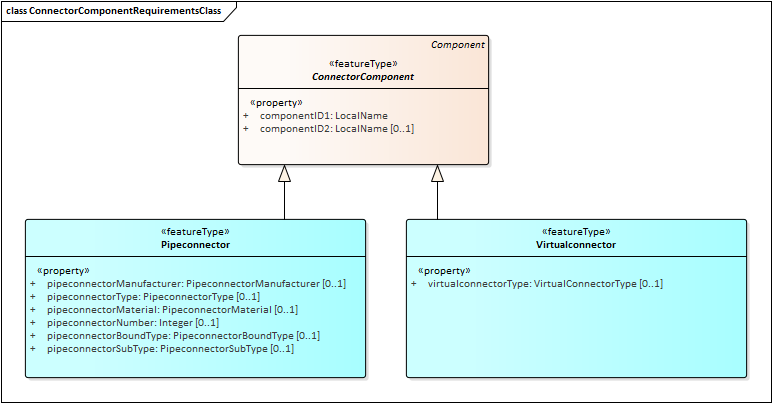


Figure 12. ConnectorComponent Requirements Class

The ConnectorComponent Class contains attributes and relationships to other classes:

**componentID1**: A required unique identifier for the first component that this connector connects

**componentID2**: An optional unique identifier for the second component that this connector connects. This attribute is optional to support applications that only identify a single component with a connection.

**pipeconnector**: The ConnectorComponent abstract class can be instantiated as a concrete Pipeconnector class/encoding

**virtualconnector**: The ConnectorComponent abstract class can be instantiated as a concrete Virtualconnector class/encoding

#### Pipeconnector Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/pipeconnector** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Pipeconnector |
| Dependency | **/req/pmlcomponent** |
| **Req 48** | **/req/pipeconnector/pipeconnectormanufacturer-codelistURIexception** |
| **Req 49** | **/req/pipeconnector/pipeconnectortype-codelistURI** |
| **Req 50** | **/req/pipeconnector/pipeconnectortype-exclusivity** |
| **Req 51** | **/req/pipeconnector/pipeconnectormaterial-codelistURI** |
| **Req 52** | **/req/pipeconnector/pipeconnectorboundtype-boundtype** |
| **Req 53** | **/req/pipeconnector/pipeconnectorboundtype-codelistURI** |
| **Req 54** | **/req/pipeconnector/pipeconnectorboundtype-exclusivity** |
| **Req 55** | **/req/pipeconnector/pipeconnectorsubtype-codelistURI** |
| **Req 56** | **/req/pipeconnector/pipeconnectorsubtype-exclusivity** |

The Pipeconnector Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 13.

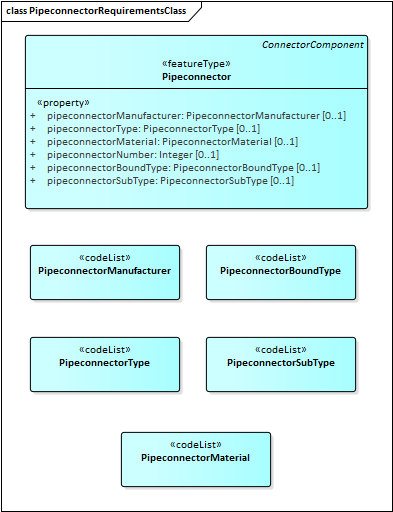


Figure 13. Pipeconnector Requirements Class

The Pipeconnector Class contains attributes and relationships to other classes:

**pipeconnectorManufacturer**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the name of the company that manufactured the component. In the case of welding rods, this would be the company who manufactured the rods. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute pipeconnectorManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pipeconnectormanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/pipeconnector/pipeconnectormanufacturer-codelistURIexception** |

**pipeconnectorType**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of connector

|  |
| --- |
| 1. This optional unbound attribute pipeconnectorType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pipeconnectortype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/pipeconnector/pipeconnectortype-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute pipeconnectorType MAY be populated, or its corresponding bound attribute pipeconnetorBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pipeconnector/pipeconnectortype-exclusivity** |

**pipeconnectorMaterial**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the material used in the connection

|  |
| --- |
| 1. This optional attribute pipeconnectorMaterial SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pipeconnectormaterial.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/pipeconnector/pipeconnectormaterial-codelistURI** |

**pipeconnectorNumber**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the number assigned to this connection

**pipeconnectorBoundType**: An optional value that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: pipeconnectorType; pipeconnectorSubType.

|  |
| --- |
| 1. This optional bound attribute pipeconnectorBoundType SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/pipeconnector/pipeconnectorboundtype-boundtype** |

|  |
| --- |
| 1. This optional bound attribute pipeconnectorBoundType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pipeconnectorboundtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/pipeconnector/pipeconnectorboundtype-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute pipeconnectorBoundType MAY be populated, or its corresponding set of unbound attributes (pipeconnectorType, and pipeconnectorSubType) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pipeconnector/pipeconnectorboundtype-exclusivity** |

**pipeconnectorSubType**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the subtype of connector

|  |
| --- |
| 1. This optional unbound attribute pipeconnectorSubType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pipeconnectorsubtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/pipeconnector/pipeconnectorsubtype-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute pipeconnectorType MAY be populated, or its corresponding bound attribute pipeconnectorBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pipeconnector/pipeconnectorsubtype-exclusivity** |

#### Virtualconnector Class Requirements Class

The Virtualconnector Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 14.

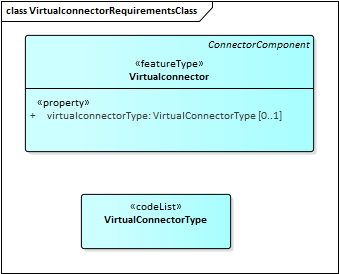


Figure 14. Virtualconnector Requirements Class

The Virtualconnector Class contains attributes and relationships to other classes:

**virtualconnectorType**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of virtual connector

#### CoterminousComponent Class Requirements Class

The CoterminousComponent Class is an abstract class and therefore SHALL NOT be encoded into a PipelineML dataset. This class provides an inheritance mechanism for passing attributes on to other classes that inherit from it. This class also establishes relationships with other classes in the model. This class contains classes and attributes as shown in Figure 15.

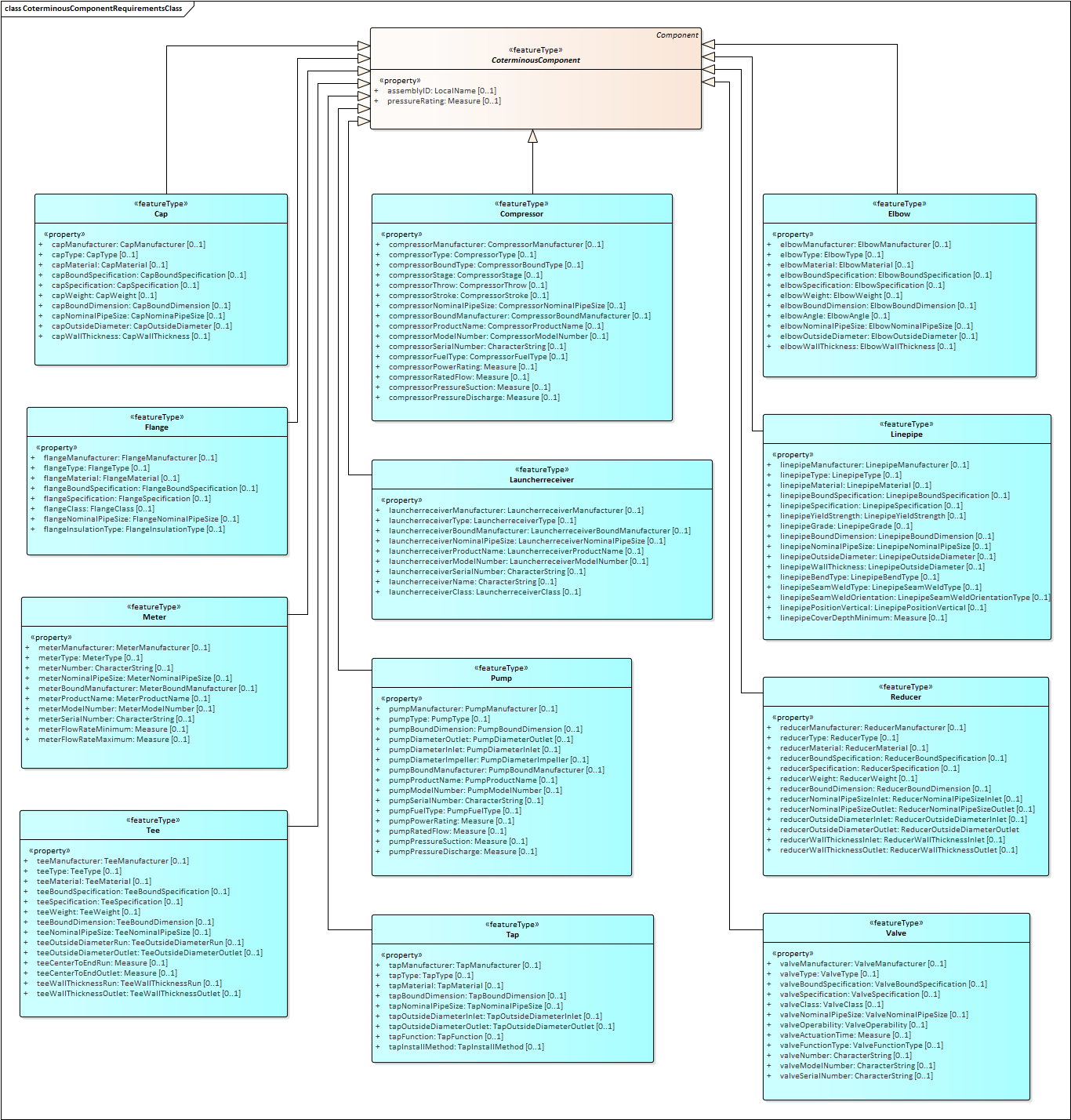


Figure 15. CoterminousComponent Requirements Class

The CoterminousComponent Class contains attributes and relationships to other classes:

**assemblyID**: A required machine-readable globally unique instance identifier for this feature. This identifier is unique within a PipelineML data package and is retained for the same piece of componentry throughout all subsequent PipelineML versions being distributed.

**pressureRating**: An optional integer value is used to define the pressure rating of the pipeline component. This measurement SHALL include a units of measure reference.

**cap**: any number of Caps

**compressor**: any number of Compressors

**elbow**: any number of Elbows

**flange**: any number of Flanges

**launcherreceiver**: any number of Launcherreceivers

**linepipe**: any number of Linepipes

**meter**: any number of Meters

**pump**: any number of Pumps

**reducer**: any number of Reducers

**tap**: any number of Taps

**tee**: any number of Tees

**valve**: any number of Valves

#### Cap Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/cap** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Cap |
| Dependency | **/req/pmlcomponent** |
| **Req 57** | **/req/cap/capmanufacturer-codelistURIexception** |
| **Req 58** | **/req/cap/captype-codelistURI** |
| **Req 59** | **/req/cap/captype-exclusivity** |
| **Req 60** | **/req/cap/capmaterial-codelistURI** |
| **Req 61** | **/req/cap/capboundspecification-boundtype** |
| **Req 62** | **/req/cap/capboundspecification-codelistURI** |
| **Req 63** | **/req/cap/capboundspecification-exclusivity** |
| **Req 64** | **/req/cap/capspecification-codelistURI** |
| **Req 65** | **/req/cap/capspecification-exclusivity** |
| **Req 66** | **/req/cap/capweight-codelistURI** |
| **Req 67** | **/req/cap/capweight-exclusivity** |
| **Req 68** | **/req/cap/capbounddimension-boundtype** |
| **Req 69** | **/req/cap/capbounddimension-codelistURI** |
| **Req 70** | **/req/cap/capbounddimension-exclusivity** |
| **Req 71** | **/req/cap/capnominalpipesize-codelistURI** |
| **Req 72** | **/req/cap/capnominalpipesize-exclusivity** |
| **Req 73** | **/req/cap/capoutsidediameter-codelistURI** |
| **Req 74** | **/req/cap/capoutsidediameter-exclusivity** |
| **Req 75** | **/req/cap/capwallthickness-codelistURI** |
| **Req 76** | **/req/cap/capwallthickness-exclusivity** |

The Cap Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 16.

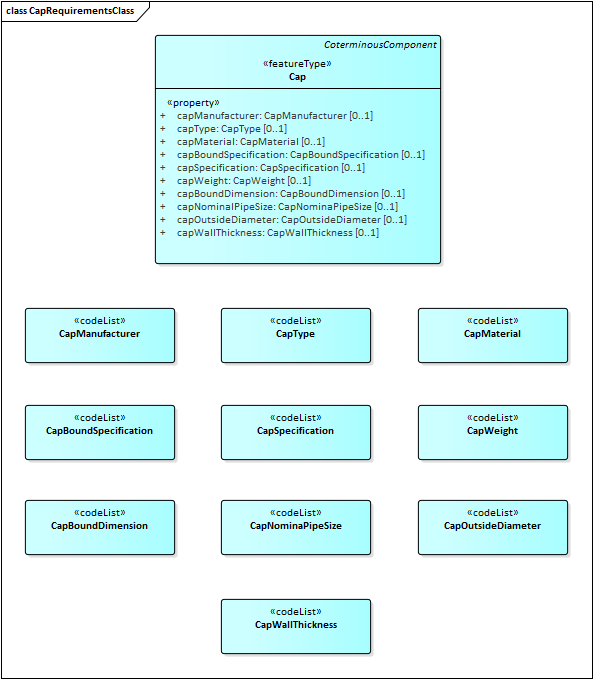


Figure 16. Cap Requirements Class

The Cap Class contains attributes and relationships to other classes:

**capManufacturer**: An optional attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute capManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/capmanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/cap/capmanufacturer-codelistURIexception** |

**capType**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of cap component

|  |
| --- |
| 1. This optional unbound attribute capType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/captype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/cap/captype-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute capType MAY be populated, or its corresponding bound attribute capBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/cap/captype-exclusivity** |

**capMaterial**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of material used to manufacture the component

|  |
| --- |
| 1. This optional attribute capMaterial SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/capmaterial.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/cap/capmaterial-codelistURI** |

**capBoundSpecification**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: capSpecification; capWeight.

|  |
| --- |
| 1. This optional bound attribute capBoundSpecification SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/cap/capboundspecification-boundtype** |

|  |
| --- |
| 1. This optional bound attribute capBoundSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/capboundspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/cap/capboundspecification-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute capBoundSpecification MAY be populated, or its corresponding set of unbound attributes (capSpecification, and capWeight) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/cap/capboundspecification-exclusivity** |

**capSpecification**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value used to define the specification of the component. This attribute SHALL NOT be provided if the attribute capBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute capSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/capspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/cap/capspecification-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute capSpecification MAY be populated, or its corresponding bound attribute capBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/cap/capspecification-exclusivity** |

**capWeight**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the weight characteristics of the component. This attribute SHALL NOT be provided if the attribute capBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute capWeight SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/capweight.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/cap/capweight-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute capWeight MAY be populated, or its corresponding bound attribute capBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/cap/capweight-exclusivity** |

**capBoundDimension**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: capNominalPipeSize; capOutsideDiameter; capWallThickness.

|  |
| --- |
| 1. This optional bound attribute capBoundDimension SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/cap/capbounddimension-boundtype** |

|  |
| --- |
| 1. This optional bound attribute capBoundDimension SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/capbounddimension.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/cap/capbounddimension-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute capBoundDimension MAY be populated, or its corresponding set of unbound attributes (capNominalPipeSize, capOutsideDiameter, capWallThickness, and capLength) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/cap/capbounddimension-exclusivity** |

**capNominalPipeSize**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the nominal pipe size of the component. This attribute SHALL NOT be provided if the attribute capBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute capNominalPipeSize SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/capnominalpipesize.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/cap/capnominalpipesize-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute capNominalPipeSize MAY be populated, or its corresponding bound attribute capBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/cap/capnominalpipesize-exclusivity** |

**capOutsideDiameter**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the outside diameter of the component. This attribute SHALL NOT be provided if the attribute capBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute capOutsideDiameter SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/capoutsidediameter.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/cap/capoutsidediameter-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute cap capOutsideDiameter MAY be populated, or its corresponding bound attribute capBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/cap/ capoutsidediameter-exclusivity** |

**capWallThickness**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the wall thickness of the component. This attribute SHALL NOT be provided if the attribute capBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute capWallThickness SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/capwallthickness.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/cap/capwallthickness-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute cap capWallThickness MAY be populated, or its corresponding bound attribute capBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/cap/capwallthickness-exclusivity** |

#### Compressor Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/compressor** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Compressor |
| Dependency | **/req/pmlcomponent** |
| **Req 77** | **/req/compressor/compressormanufacturer-codelistURIexception** |
| **Req 78** | **/req/compressor/compressormanufacturer-exclusivity** |
| **Req 79** | **/req/compressor/compressortype-codelistURI** |
| **Req 80** | **/req/compressor/compressortype-exclusivity** |
| **Req 81** | **/req/compressor/compressorboundtype-boundtype** |
| **Req 82** | **/req/compressor/compressorboundtype-codelistURI** |
| **Req 83** | **/req/compressor/compressorboundtype-exclusivity** |
| **Req 84** | **/req/compressor/compressorstage-codelistURI** |
| **Req 85** | **/req/compressor/compressorstage-exclusivity** |
| **Req 86** | **/req/compressor/compressorthrow-codelistURI** |
| **Req 87** | **/req/compressor/compressorthrow-exclusivity** |
| **Req 88** | **/req/compressor/compressorstroke-codelistURI** |
| **Req 89** | **/req/compressor/compressorstroke-exclusivity** |
| **Req 90** | **/req/compressor/compressornominalpipesize-codelistURI** |
| **Req 91** | **/req/compressor/compressorboundmanufacturer-boundtype** |
| **Req 92** | **/req/compressor/compressorboundmanufacturer-codelistURI** |
| **Req 93** | **/req/compressor/compressorboundmanufacturer-exclusivity** |
| **Req 94** | **/req/compressor/compressorproductname-codelistURIexception** |
| **Req 95** | **/req/compressor/compressorproductname-exclusivity** |
| **Req 96** | **/req/compressor/compressormodelnumber-codelistURIexception** |
| **Req 97** | **/req/compressor/compressormodelnumber-exclusivity** |
| **Req 98** | **/req/compressor/compressorfueltype-codelistURI** |
| **Req 99** | **/req/compressor/compressorpowerrating-uom** |
| **Req 100** | **/req/compressor/compressorratedflow-uom** |
| **Req 101** | **/req/compressor/compressorpressuresuction-uom** |
| **Req 102** | **/req/compressor/compressorpressuredischarge-uom** |

The Compressor Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 17.

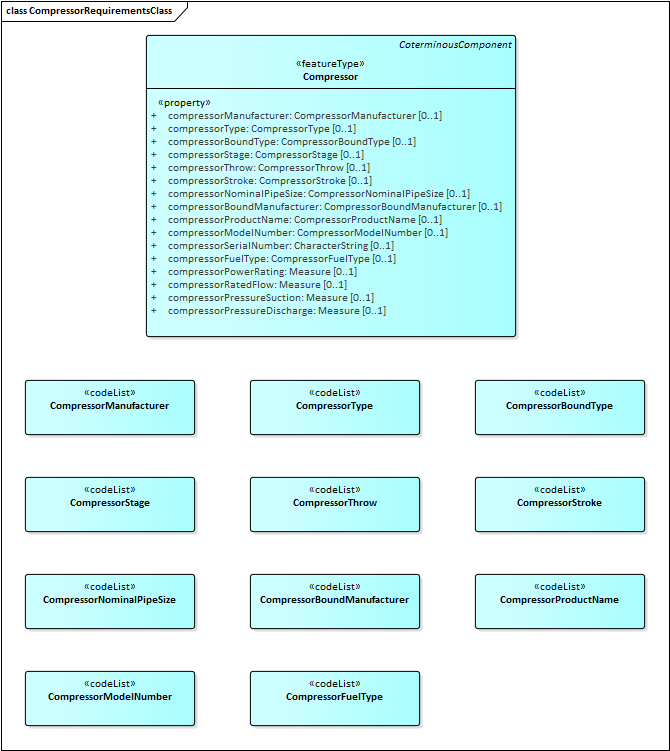


Figure 17. Compressor Requirements Class

The Compressor Class contains attributes and relationships to other classes:

**compressorManufacturer**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute compressorManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/compressormanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/compressor/compressormanufacturer-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute compressorManufacturer MAY be populated, or its corresponding bound attribute compressorBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/compressor/compressormanufacturer-exclusivity** |

**compressorType**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of compressor component

|  |
| --- |
| 1. This optional unbound attribute compressorType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/compressortype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/compressor/compressortype-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute compressorType MAY be populated, or its corresponding bound attribute compressorBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/compressor/compressortype-exclusivity** |

**compressorBoundType**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: compressorType; compressorStage; compressorThrow; compressorStroke.

|  |
| --- |
| 1. This optional bound attribute compressorBoundType SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/compressor/compressorboundtype-boundtype** |

|  |
| --- |
| 1. This optional bound attribute compressorBoundType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/compressorboundtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/compressor/compressorboundtype-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute compressorBoundType MAY be populated, or its corresponding set of unbound attributes (compressorType, compressorStage, compressorThrow, and compressorStroke) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/compressor/compressorboundtype-exclusivity** |

**compressorStage**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value used to define the number of stages in use by the component. This attribute SHALL NOT be provided if the attribute compressorBoundType is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute compressorStage SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/compressorstage.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/compressor/compressorstage-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute compressorStage MAY be populated, or its corresponding bound attribute compressorBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/compressor/compressorstage-exclusivity** |

**compressorThrow**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value used to define the throw of the component. This attribute SHALL NOT be provided if the attribute compressorBoundType is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute compressorThrow SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/compressorthrow.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/compressor/compressorthrow-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute compressorThrow MAY be populated, or its corresponding bound attribute compressorBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/compressor/compressorthrow-exclusivity** |

**compressorStroke**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value used to define the stroke type of the component. This attribute SHALL NOT be provided if the attribute compressorBoundType is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute compressorStroke SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/compressorstroke.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/compressor/compressorstroke-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute compressorStroke MAY be populated, or its corresponding bound attribute compressorBoundType MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/compressor/compressorstroke-exclusivity** |

**compressorNominalPipeSize**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the nominal pipe size of the component

|  |
| --- |
| 1. This optional attribute compressorNominalPipeSize SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/compressornominapipesize.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/compressor/compressornominalpipesize-codelistURI** |

**compressorBoundManufacturer**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: compressorManufacturer; compressorProductName; compressorModelNumber.

|  |
| --- |
| 1. This optional bound attribute compressorBoundManufacturer SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/compressor/compressorboundmanufacturer-boundtype** |

|  |
| --- |
| 1. This optional bound attribute compressorBoundManufacturer SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/compressorboundmanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/compressor/compressorboundmanufacturer-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute compressorBoundManufacturer MAY be populated, or its corresponding set of unbound attributes (compressorManufacturer, compressorProductName, and compressorModelNumber) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/compressor/compressorboundmanufacturer-exclusivity** |

**compressorProductName**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the product name of the component. This attribute SHALL NOT be provided if the attribute compressorBoundManufacturer is populated for reasons of disambiguation. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute compressorProductName SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/compressorproductname.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/compressor/compressorproductname-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute compressorProductName MAY be populated, or its corresponding bound attribute compressorBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/compressor/compressorproductname-exclusivity** |

**compressorModelNumber**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the model number of the component. This attribute SHALL NOT be provided if the attribute compressorBoundManufacturer is populated for reasons of disambiguation. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute compressorModelNumber SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/compressormodelnumber.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/compressor/compressormodelnumber-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute compressorModelNumber MAY be populated, or its corresponding bound attribute compressorBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/compressor/compressormodelnumber-exclusivity** |

**compressorSerialNumber**: An optional string value that defines a unique identification number established by the component manufacturer

**compressorFuelType**: An optional value that, if populated, SHALL be dereferenceable to define a string value describing the type of fuel used to operate the component.

|  |
| --- |
| 1. This optional attribute compressorFuelType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/compressorfueltype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/compressor/compressorfueltype-codelistURI** |

**compressorPowerRating**: An optional measurement made by the manufacturer that defines the amount of power the compressor is rated to produce. If this attribute is populated with a value, a units-of-measure sub-attribute SHALL also be provided.

|  |
| --- |
| 1. If this optional attribute compressorPowerRating is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/compressor/compressorpowerrating-uom** |

**compressorRatedFlow**: An optional measurement made by the manufacturer that defines the rate of product flow the compressor is rated to produce. If populated is populated with a value, a units-of-measure sub-attribute SHALL also be provided.

|  |
| --- |
| 1. If this optional attribute compressorRatedFlow is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/compressor/compressorratedflow-uom** |

**compressorPressureSuction**: An optional measurement made by the manufacturer that defines the amount of suction the compressor is rated to produce. If populated is populated with a value, a units-of-measure sub-attribute SHALL also be provided.

|  |
| --- |
| 1. If this optional attribute compressorPressureSuction is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/compressor/compressorpressuresuction-uom** |

**compressorPressureDischarge**: An optional measurement made by the manufacturer that defines the amount of pressure the compressor is rated to discharge. If populated is populated with a value, a units-of-measure sub-attribute SHALL also be provided.

|  |
| --- |
| 1. If this optional attribute compressorPressureDischarge is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/compressor/compressorpressuredischarge-uom** |

#### Elbow Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/elbow** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Elbow |
| Dependency | **/req/pmlcomponent** |
| **Req 103** | **/req/elbow/elbowmanufacturer-codelistURIexception** |
| **Req 104** | **/req/elbow/elbowtype-codelistURI** |
| **Req 105** | **/req/elbow/elbowmaterial-codelistURI** |
| **Req 106** | **/req/elbow/elbowboundspecification-boundtype** |
| **Req 107** | **/req/elbow/elbowboundspecification-codelistURI** |
| **Req 108** | **/req/elbow/elbowboundspecification-exclusivity** |
| **Req 109** | **/req/elbow/elbowspecification-codelistURI** |
| **Req 110** | **/req/elbow/elbowspecification-exclusivity** |
| **Req 111** | **/req/elbow/elbowweight-codelistURI** |
| **Req 112** | **/req/elbow/elbowweight-exclusivity** |
| **Req 113** | **/req/elbow/elbowbounddimension-boundtype** |
| **Req 114** | **/req/elbow/elbowbounddimension-codelistURI** |
| **Req 115** | **/req/elbow/elbowbounddimension-exclusivity** |
| **Req 116** | **/req/elbow/elbowangle-codelistURI** |
| **Req 117** | **/req/elbow/elbowangle-exclusivity** |
| **Req 118** | **/req/elbow/elbownominalpipesize-codelistURI** |
| **Req 119** | **/req/elbow/elbownominalpipesize-exclusivity** |
| **Req 120** | **/req/elbow/elbowoutsidediameter-codelistURI** |
| **Req 121** | **/req/elbow/elbowoutsidediameter-exclusivity** |
| **Req 122** | **/req/elbow/elbowwallthickness-codelistURI** |
| **Req 123** | **/req/elbow/elbowwallthickness-exclusivity** |

The Elbow Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 18.



Figure 18. Elbow Requirements Class

The Elbow Class contains attributes and relationships to other classes:

**elbowManufacturer**: An optional attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute elbowManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/elbowmanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/elbow/elbowmanufacturer-codelistURIexception** |

**elbowType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of elbow component

|  |
| --- |
| 1. This optional attribute elbowType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/elbowtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/elbow/elbowtype-codelistURI** |

**elbowMaterial**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of material use in the manufacture of the component

|  |
| --- |
| 1. This optional attribute elbowMaterial SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/elbowmaterial.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/elbow/elbowmaterial-codelistURI** |

**elbowBoundSpecification**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: elbowSpecification; elbowWeight.

|  |
| --- |
| 1. This optional bound attribute elbowBoundSpecification SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/elbow/elbowboundspecification-boundtype** |

|  |
| --- |
| 1. This optional bound attribute elbowBoundSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/elbowboundspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/elbow/elbowboundspecification-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute elbowBoundSpecification MAY be populated, or its corresponding set of unbound attributes (elbowSpecification, and elbowWeight) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/elbow/elbowboundspecification-exclusivity** |

**elbowSpecification**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value used to define the specification of the component. This attribute SHALL NOT be provided if the attribute elbowBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute elbowSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/elbowspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/elbow/elbowspecification-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute elbowSpecification MAY be populated, or its corresponding bound attribute elbowBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/elbow/elbowspecification-exclusivity** |

**elbowWeight**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the weight characteristics of the component. This attribute SHALL NOT be provided if the attribute elbowBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute elbowWeight SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/elbowweight.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/elbow/elbowweight-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute elbowWeight MAY be populated, or its corresponding bound attribute elbowBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/elbow/elbowweight-exclusivity** |

**elbowBoundDimension**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: elbowAngle; elbowNominalPipeSize; elbowOutsideDiameter; elbowWallThickness.

|  |
| --- |
| 1. This optional bound attribute elbowBoundDimension SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/elbow/elbowbounddimension-boundtype** |

|  |
| --- |
| 1. This optional bound attribute elbowBoundDimension SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/elbowbounddimension.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/elbow/elbowbounddimension-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute elbowBoundDimension MAY be populated, or its corresponding set of unbound attributes (elbowAngle, elbowNominalPipeSize, elbowOutsideDiameter, and elbowWallThickness) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/elbow/elbowbounddimension-exclusivity** |

**elbowAngle**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the angle of the bend in the component. This attribute SHALL NOT be provided if the attribute elbowBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute elbowAngle SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/elbowangle.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/elbow/elbowangle-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute elbowAngle MAY be populated, or its corresponding bound attribute elbowBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/elbow/elbowangle-exclusivity** |

**elbowNominalPipeSize**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the nominal pipe size of the component. This attribute SHALL NOT be provided if the attribute elbowBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute elbowNominalPipeSize SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/elbownominalpipesize.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/elbow/elbownominalpipesize-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute elbowNominalPipeSize MAY be populated, or its corresponding bound attribute elbowBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/elbow/elbownominalpipesize-exclusivity** |

**elbowOutsideDiameter**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the outside diameter of the component. This attribute SHALL NOT be provided if the attribute elbowBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute elbowOutsideDiameter SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/elbowoutsidediameter.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/elbow/elbowoutsidediameter-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute elbowOutsideDiameter MAY be populated, or its corresponding bound attribute elbowBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/elbow/elbowoutsidediameter-exclusivity** |

**elbowWallThickness**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the wall thickness of the component. This attribute SHALL NOT be provided if the attribute elbowBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute elbowWallThickness SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/elbowwallthickness.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/elbow/elbowwallthickness-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute elbowWallThickness MAY be populated, or its corresponding bound attribute elbowBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/elbow/elbowwallthickness-exclusivity** |

#### Flange Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/flange** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Flange |
| Dependency | **/req/pmlcomponent** |
| **Req 124** | **/req/flange/flangemanufacturer-codelistURIexception** |
| **Req 125** | **/req/flange/flangetype-codelistURI** |
| **Req 126** | **/req/flange/flangetype-exclusivity** |
| **Req 127** | **/req/flange/flangematerial-codelistURI** |
| **Req 128** | **/req/flange/flangeboundspecification-boundtype** |
| **Req 129** | **/req/flange/flangeboundspecification-codelistURI** |
| **Req 130** | **/req/flange/flangeboundspecification-exclusivity** |
| **Req 131** | **/req/flange/flangespecification-codelistURI** |
| **Req 132** | **/req/flange/flangespecification-exclusivity** |
| **Req 133** | **/req/flange/flangeclass-codelistURI** |
| **Req 134** | **/req/flange/flangeclass-exclusivity** |
| **Req 135** | **/req/flange/flangenominalpipesize-codelistURI** |
| **Req 136** | **/req/flange/flangeinsulationtype-codelistURI** |

The Flange Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 19.

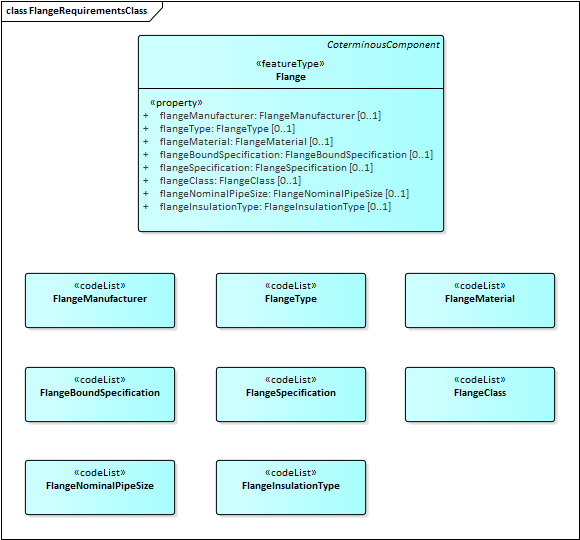


Figure 19. Flange Requirements Class

The Flange Class contains attributes and relationships to other classes:

**flangeManufacturer**: An optional attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute flangeManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/flangemanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/flange/flangemanufacturer-codelistURIexception** |

**flangeType**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of flange component

|  |
| --- |
| 1. This optional unbound attribute flangeType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/flangetype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/flange/flangetype-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute flangeType MAY be populated, or its corresponding bound attribute flangeBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/flange/flangetype-exclusivity** |

**flangeMaterial**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of material use in the manufacture of the component

|  |
| --- |
| 1. This optional attribute flangeMaterial SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/flangematerial.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/flange/flangematerial-codelistURI** |

**flangeBoundSpecification**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: flangeType; flangeSpecification; flangeClass.

|  |
| --- |
| 1. This optional bound attribute flangeBoundSpecification SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/flange/flangeboundspecification-boundtype** |

|  |
| --- |
| 1. This optional bound attribute flangeBoundSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/flangeboundspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/flange/flangeboundspecification-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute flangeBoundSpecification MAY be populated, or its corresponding set of unbound attributes (flangeType, flangeSpecification, and flangeClass) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/flange/flangeboundspecification-exclusivity** |

**flangeSpecification**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value used to define the specification of the component. This attribute SHALL NOT be provided if the attribute flangeBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute flangeSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/flangespecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/flange/flangespecification-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute flangeSpecification MAY be populated, or its corresponding bound attribute flangeBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/flange/flangespecification-exclusivity** |

**flangeClass**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the class of the flange component. This attribute SHALL NOT be provided if the attribute flangeBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute flangeClass SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/flangeclass.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/flange/flangeclass-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute flangeClass MAY be populated, or its corresponding bound attribute flangeBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/flange/flangeclass-exclusivity** |

**flangeNominalPipeSize**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the nominal pipe size of the component

|  |
| --- |
| 1. This optional attribute flangeNominalPipeSize SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/flangenominalpipesize.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/flange/flangenominalpipesize-codelistURI** |

**flangeInsulationType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of insulation used in the component

|  |
| --- |
| 1. This optional attribute flangeInsulationType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/flangeInsulationtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/flange/flangeinsulationtype-codelistURI** |

#### Launcherreceiver Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/launcherreceiver** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Launcherreceiver |
| Dependency | **/req/pmlcomponent** |
| **Req 137** | **/req/launcherreceiver/launcherreceivermanufacturer-codelistURIexception** |
| **Req 138** | **/req/launcherreceiver/launcherreceivermanufacturer-exclusivity** |
| **Req 139** | **/req/launcherreceiver/launcherreceivertype-codelistURI** |
| **Req 140** | **/req/launcherreceiver/launcherreceiverboundmanufacturer-boundtype** |
| **Req 141** | **/req/launcherreceiver/launcherreceiverboundmanufacturer-codelistURI** |
| **Req 142** | **/req/launcherreceiver/launcherreceiverboundmanufacturer-exclusivity** |
| **Req 143** | **/req/launcherreceiver/launcherreceiverproductname-codelistURIexception** |
| **Req 144** | **/req/launcherreceiver/launcherreceiverproductname-exclusivity** |
| **Req 145** | **/req/launcherreceiver/launcherreceivermodelnumber-codelistURIexception** |
| **Req 146** | **/req/launcherreceiver/launcherreceivermodelnumber-exclusivity** |
| **Req 147** | **/req/launcherreceiver/launcherreceiverclass-codelistURI** |

The Launcherreceiver Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 20.

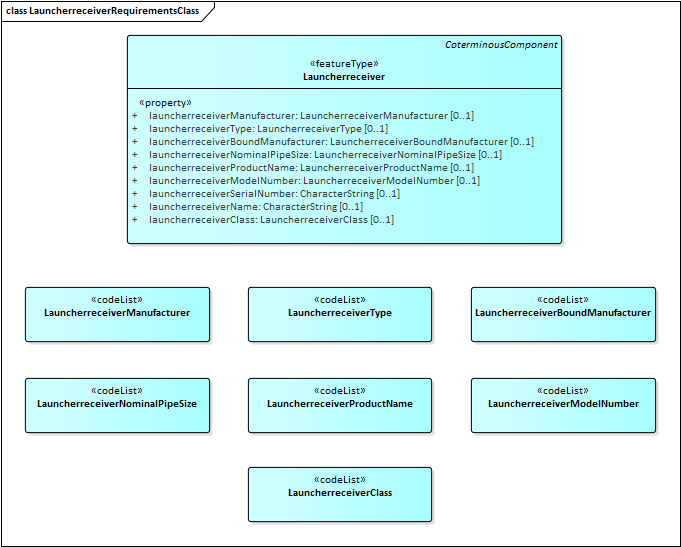


Figure 20. Launcherreceiver Requirements Class

The Launcherreceiver Class contains attributes and relationships to other classes:

**launcherreceiverManufacturer**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute launcherreceiverManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/launcherreceivermanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/launcherreceiver/launcherreceivermanufacturer-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute launcherreceiverManufacturer MAY be populated, or its corresponding bound attribute launcherreceiverBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/launcherreceiver/launcherreceivermanufacturer-exclusivity** |

**launcherreceiverType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of launcherreceiver component

|  |
| --- |
| 1. This optional attribute launcherreceiverType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/launcherreceivertype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/launcherreceiver/launcherreceivertype-codelistURI** |

**launcherreceiverBoundManufacturer**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: launcherreceiverManufacturer; launcherreceiverProductName; launcherreceiverModelNumber.

|  |
| --- |
| 1. This optional bound attribute launcherreceiverBoundMnaufacturer SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/launcherreceiver/launcherreceiverboundmanufacturer-boundtype** |

|  |
| --- |
| 1. This optional bound attribute launcherreceiverBoundManufacturer SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/launcherreceiverboundmanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/launcherreceiver/launcherreceiverboundmanufacturer-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute launcherreceiverBoundManufacturer MAY be populated, or its corresponding set of unbound attributes (launcherreceiverManufacturer, launcherreceiverProductName, and launcherreceiverModelNumber) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/launcherreceiver/launcherreceiverboundmanufacturer-exclusivity** |

**launcherreceiverProductName**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the product name of the component. This attribute SHALL NOT be provided if the attribute launcherreceiverBoundManufacturer is populated for reasons of disambiguation. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute launcherreceiverProductName SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/launcherreceiverproductname.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/launcherreceiver/launcherreceiverproductname-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute launcherreceiverProductName MAY be populated, or its corresponding bound attribute launcherreceiverBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/launcherreceiver/launcherreceiverproductname-exclusivity** |

**launcherreceiverModelNumber**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the model number of the component. This attribute SHALL NOT be provided if the attribute launcherreceiverBoundManufacturer is populated for reasons of disambiguation. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute launcherreceiverModelNumber SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/launcherreceivermodelnumber.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/launcherreceiver/launcherreceivermodelnumber-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute launcherreceiverModelNumber MAY be populated, or its corresponding bound attribute launcherreceiverBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/launcherreceiver/launcherreceivermodelnumber-exclusivity** |

**launcherreceiverSerialNumber**: An optional string attribute that defines a unique identification number established by the component manufacturer

**launcherreceiverName**: An optional string attribute that defines a unique identification number established by the component manufacturer

**launcherreceiverClass**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the class of the launcherreceiver component

|  |
| --- |
| 1. This optional attribute launcherreceiverClass SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/launcherreceiverclass.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/launcherreceiver/launcherreceiverclass-codelistURI** |

#### Linepipe Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/linepipe** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Linepipe |
| Dependency | **/req/pmlcomponent** |
| **Req 148** | **/req/linepipe/linepipemanufacturer-codelistURIexception** |
| **Req 149** | **/req/linepipe/linepipetype-codelistURI** |
| **Req 150** | **/req/linepipe/linepipematerial-codelistURI** |
| **Req 151** | **/req/linepipe/linepipeboundspecification-boundtype** |
| **Req 152** | **/req/linepipe/linepipeboundspecification-codelistURI** |
| **Req 153** | **/req/linepipe/linepipeboundspecification-exclusivity** |
| **Req 154** | **/req/linepipe/linepipespecification-codelistURI** |
| **Req 155** | **/req/linepipe/linepipespecification-exclusivity** |
| **Req 156** | **/req/linepipe/linepipeyieldstrength-codelistURI** |
| **Req 157** | **/req/linepipe/linepipeyieldstrength-exclusivity** |
| **Req 158** | **/req/linepipe/linepipegrade-codelistURI** |
| **Req 159** | **/req/linepipe/linepipegrade-exclusivity** |
| **Req 160** | **/req/linepipe/linepipebounddimension-boundtype** |
| **Req 161** | **/req/linepipe/linepipebounddimension-codelistURI** |
| **Req 162** | **/req/linepipe/linepipebounddimension-exclusivity** |
| **Req 163** | **/req/linepipe/linepipenominalpipesize-codelistURI** |
| **Req 164** | **/req/linepipe/linepipeoutsidediameter-codelistURI** |
| **Req 165** | **/req/linepipe/linepipeoutsidediameter-exclusivity** |
| **Req 166** | **/req/linepipe/linepipewallthickness-codelistURI** |
| **Req 167** | **/req/linepipe/linepipewallthickness-exclusivity** |
| **Req 168** | **/req/linepipe/linepipebendtype-codelistURI** |
| **Req 169** | **/req/linepipe/linepipeseamweldtype-codelistURI** |
| **Req 170** | **/req/linepipe/linepipeseamweldorientation-codelistURI** |
| **Req 171** | **/req/linepipe/linepipeseamweldorientation-componentmeasuredfrom** |
| **Req 172** | **/req/linepipe/linepipepositionvertical-codelistURI** |
| **Req 173** | **/req/linepipe/linepipecoverdepthminimum-uom** |

The Linepipe Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 21.

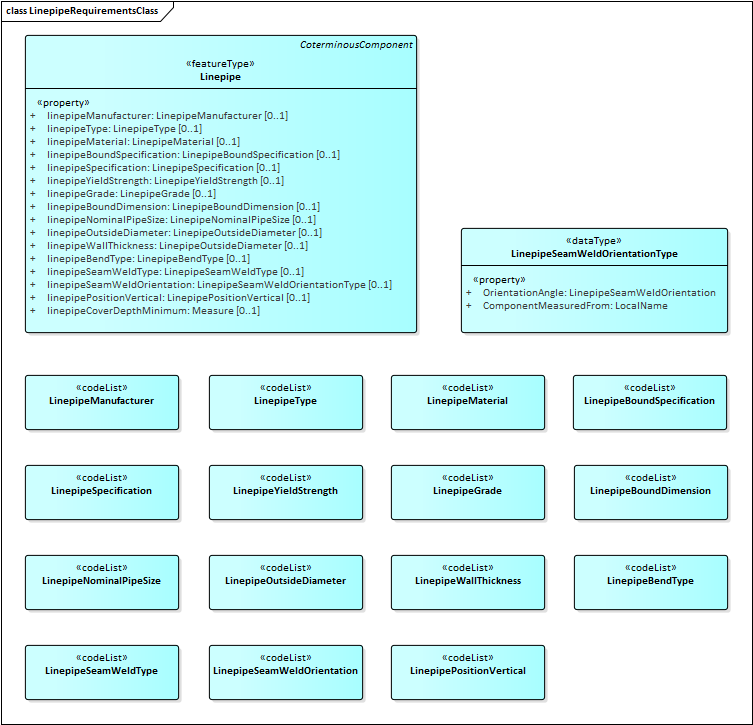


Figure 21. Linepipe Requirements Class

The Linepipe Class contains attributes and relationships to other classes:

**linepipeManufacturer**: An optional attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute linepipeManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipemanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/linepipe/linepipemanufacturer-codelistURIexception** |

**linepipeType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of linepipe component

|  |
| --- |
| 1. This optional attribute linepipeType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipetype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/linepipe/linepipetype-codelistURI** |

**linepipeMaterial**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of material use in the manufacture of the component

|  |
| --- |
| 1. This optional attribute linepipeMaterial SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipematerial.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/linepipe/linepipematerial-codelistURI** |

**linepipeBoundSpecification**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: linepipeSpecification; linepipeYieldStrength; linepipeGrade.

|  |
| --- |
| 1. This optional bound attribute linepipeBoundSpecification SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/linepipe/linepipeboundspecification-boundtype** |

|  |
| --- |
| 1. This optional bound attribute linepipeBoundSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipeboundspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/linepipe/linepipeboundspecification-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute linepipeBoundSpecification MAY be populated, or its corresponding set of unbound attributes (linepipeSpecification, linepipeYieldStrength, and linepipeGrade) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/linepipe/linepipeboundspecification-exclusivity** |

**linepipeSpecification**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value used to define the specification of the component. This attribute SHALL NOT be provided if the attribute linepipeBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute linepipeSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipespecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/linepipe/linepipespecification-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute linepipeSpecification MAY be populated, or its corresponding bound attribute linepipeBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/linepipe/linepipespecification-exclusivity** |

**linepipeYieldStrength**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value that defines the rated yield strength of the component. This attribute SHALL NOT be provided if the attribute linepipeBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute linepipeYieldStrength SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipeyieldstrength.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/linepipe/linepipeyieldstrength-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute linepipeYieldStrength MAY be populated, or its corresponding bound attribute linepipeBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/linepipe/linepipeyieldstrength-exclusivity** |

**linepipeGrade**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value that defines the grade specification of the component. This attribute SHALL NOT be provided if the attribute linepipeBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute linepipeGrade SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipegrade.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/linepipe/linepipegrade-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute linepipeGrade MAY be populated, or its corresponding bound attribute linepipeBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/linepipe/linepipegrade-exclusivity** |

**linepipeBoundDimension**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: linepipeNominalPipeSize; linepipeOutsideDiameter; linepipeWallThickness.

|  |
| --- |
| 1. This optional bound attribute linepipeBoundDimension SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/linepipe/linepipebounddimension-boundtype** |

|  |
| --- |
| 1. This optional bound attribute linepipeBoundDimension SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipebounddimension.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/linepipe/linepipebounddimension-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute linepipeBoundDimension MAY be populated, or its corresponding set of unbound attributes (linepipeNominalPipeSize, linepipeOutsideDiameter, and linepipeWallThickness) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/linepipe/linepipebounddimension-exclusivity** |

**linepipeNominalPipeSize**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the nominal pipe size of the component. This attribute SHALL NOT be provided if the attribute linepipeBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute linepipeNominalPipeSize SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipenominalpipesize.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/linepipe/linepipenominalpipesize-codelistURI** |

**linepipeOutsideDiameter**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the outside diameter of the component. This attribute SHALL NOT be provided if the attribute linepipeBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute linepipeOutsideDiameter SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipeoutsidediameter.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/linepipe/linepipeoutsidediameter-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute linepipeOutsideDiameter MAY be populated, or its corresponding bound attribute linepipeBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/linepipe/linepipeoutsidediameter-exclusivity** |

**linepipeWallThickness**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the wall thickness of the component. This attribute SHALL NOT be provided if the attribute linepipeBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute linepipeWallThickness SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipewallthickness.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/linepipe/linepipewallthickness-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute linepipeWallThickness MAY be populated, or its corresponding bound attribute linepipeBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/linepipe/linepipewallthickness-exclusivity** |

**linepipeBendType**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of bend (if any) in the component

|  |
| --- |
| 1. This optional attribute linepipeBendType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipebendtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/linepipe/linepipebendtype-codelistURI** |

**linepipeSeamWeldType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of seam weld used in the manufacturer of the component

|  |
| --- |
| 1. This optional unbound attribute linepipeSeamWeldType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipeseamweldtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/linepipe/linepipeseamweldtype-codelistURI** |

**linepipeSeamWeldOrientation**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the orientation angle of seam weld of the component when installed. If this attribute is populated, it shall provide a ComponentMeasuredFrom value that contains the unique identification of the component whose vantage point is used as a reference to indicate the direction of the orientation angle.

|  |
| --- |
| 1. This optional attribute linepipeSeamWeldOrientation SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipeseamweldorientation.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/linepipe/linepipeseamweldorientation-codelistURI** |

|  |
| --- |
| 1. This optional attribute utilizes a complex data type that supports the inclusion of an “componentmeasuredfrom” sub-attribute value that contains the unique identification of the component whose vantage point is used as a reference point to indicate the direction of the orientation angle. If this attribute is populated, it SHALL also populate the ComponentMeasuredFrom sub-attribute.   **/req/linepipe/linepipeseamweldorientation-componentmeasuredfrom** |

**linepipePositionVertical**: An optional attribute that, if populated, SHALL be dereferenceable to a string value containing the vertical position of the component relative to natural ground as it was installed (such as On Ground, Below Surface, and Elevated or Suspended)

|  |
| --- |
| 1. This optional attribute linepipePositionVertical SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/linepipepositionvertical.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/linepipe/linepipepositionvertical-codelistURI** |

**linepipeCoverDepthMinimum**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the minimum depth of ground cover required by the manufacturer of the component. If this attribute is populated, a units of measure sub-attribute value SHALL also be populated.

|  |
| --- |
| 1. If this optional attribute linepipeCoverDepthMinimum is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/linepipe/linepipecoverdepthminimum-uom** |

#### Meter Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/meter** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Meter |
| Dependency | **/req/pmlcomponent** |
| **Req 174** | **/req/meter/metermanufacturer-codelistURIexception** |
| **Req 175** | **/req/meter/metermanufacturer-exclusivity** |
| **Req 176** | **/req/meter/metertype-codelistURI** |
| **Req 177** | **/req/meter/meternominalpipesize-codelistURI** |
| **Req 178** | **/req/meter/meterboundmanufacturer-boundtype** |
| **Req 179** | **/req/meter/meterboundmanufacturer-codelistURI** |
| **Req 180** | **/req/meter/meterboundmanufacturer-exclusivity** |
| **Req 181** | **/req/meter/meterproductname-codelistURIexception** |
| **Req 182** | **/req/meter/meterproductname-exclusivity** |
| **Req 183** | **/req/meter/metermodelnumber-codelistURIexception** |
| **Req 184** | **/req/meter/metermodelnumber-exclusivity** |
| **Req 185** | **/req/meter/meterflowrateminimum-uom** |
| **Req 186** | **/req/meter/meterflowratemaximum-uom** |

The Meter Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 22.

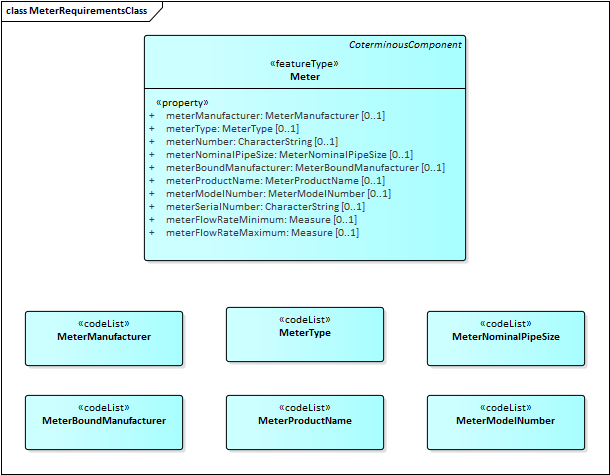


Figure 22. Meter Requirements Class

The Meter Class contains attributes and relationships to other classes:

**meterManufacturer**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute meterManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/metermanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/meter/metermanufacturer-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute meterManufacturer MAY be populated, or its corresponding bound attribute meterBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/meter/metermanufacturer-exclusivity** |

**meterType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of meter component

|  |
| --- |
| 1. This optional unbound attribute meterType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/metertype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/meter/metertype-codelistURI** |

**meterNumber**: An optional string attribute that defines a unique identification number for the component established by the pipeline operator

**meterNominalPipeSize**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the nominal pipe size of the component

|  |
| --- |
| 1. This optional unbound attribute meterNominalPipeSize SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/meternominalpipesize.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/meter/meternominalpipesize-codelistURI** |

**meterBoundManufacturer**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: meterManufacturer; meterProductName; meterModelNumber.

|  |
| --- |
| 1. This optional bound attribute meterBoundMnaufacturer SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/meter/meterboundmanufacturer-boundtype** |

|  |
| --- |
| 1. This optional bound attribute meterBoundManufacturer SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/meterboundmanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/meter/meterboundmanufacturer-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute meterBoundManufacturer MAY be populated, or its corresponding set of unbound attributes (meterManufacturer, meterProductName, and meterModelNumber) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/meter/meterboundmanufacturer-exclusivity** |

**meterProductName**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the product name of the component. This attribute SHALL NOT be provided if the attribute meterBoundManufacturer is populated for reasons of disambiguation. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute meterProductName SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/meterproductname.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/meter/meterproductname-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute meterProductName MAY be populated, or its corresponding bound attribute meterBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/meter/meterproductname-exclusivity** |

**meterModelNumber**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the model number of the component. This attribute SHALL NOT be provided if the attribute meterBoundManufacturer is populated for reasons of disambiguation. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute meterModelNumber SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/metermodelnumber.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/meter/metermodelnumber-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute meterModelNumber MAY be populated, or its corresponding bound attribute meterBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/meter/metermodelnumber-exclusivity** |

**meterSerialNumber**: An optional string attribute that defines a unique identification number established by the component manufacturer

**meterFlowRateMinimum**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the minimum product flow rate as determined by the manufacturer of the component. If this attribute is populated, a units of measure sub-attribute value SHALL also be populated.

|  |
| --- |
| 1. If this optional attribute meterFlowRateMinimum is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/meter/meterflowrateminimum-uom** |

**meterFlowRateMaximum**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the maximum product flow rate as determined by the manufacturer of the component. If this attribute is populated, a units of measure sub-attribute value SHALL also be populated.

|  |
| --- |
| 1. If this optional attribute meterFlowRateMaximum is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/meter/meterflowratemaximum-uom** |

#### Pump Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/pump** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Pump |
| Dependency | **/req/pmlcomponent** |
| **Req 187** | **/req/pump/pumpmanufacturer-codelistURIexception** |
| **Req 188** | **/req/pump/pumpmanufacturer-exclusivity** |
| **Req 189** | **/req/pump/pumptype-codelistURI** |
| **Req 190** | **/req/pump/pumpbounddimension-boundtype** |
| **Req 191** | **/req/pump/pumpbounddimension-codelistURI** |
| **Req 192** | **/req/pump/pumpbounddimension-exclusivity** |
| **Req 193** | **/req/pump/pumpdiameteroutlet-codelistURI** |
| **Req 194** | **/req/pump/pumpdiameteroutlet-exclusivity** |
| **Req 195** | **/req/pump/pumpdiameterinlet-codelistURI** |
| **Req 196** | **/req/pump/pumpdiameterinlet-exclusivity** |
| **Req 197** | **/req/pump/pumpdiameterimpeller-codelistURI** |
| **Req 198** | **/req/pump/pumpdiameterimpeller-exclusivity** |
| **Req 199** | **/req/pump/pumpboundmanufacturer-boundtype** |
| **Req 200** | **/req/pump/pumpboundmanufacturer-codelistURI** |
| **Req 201** | **/req/pump/pumpboundmanufacturer-exclusivity** |
| **Req 202** | **/req/pump/pumpproductname-codelistURIexception** |
| **Req 203** | **/req/pump/pumpproductname-exclusivity** |
| **Req 204** | **/req/pump/pumpmodelnumber-codelistURIexception** |
| **Req 205** | **/req/pump/pumpmodelnumber-exclusivity** |
| **Req 206** | **/req/pump/pumpfueltype-codelistURI** |
| **Req 207** | **/req/pump/pumppowerrating-uom** |
| **Req 208** | **/req/pump/pumpratedflow-uom** |
| **Req 209** | **/req/pump/pumppressuresuction-uom** |
| **Req 210** | **/req/pump/pumppressuredischarge-uom** |

The Pump Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 23.

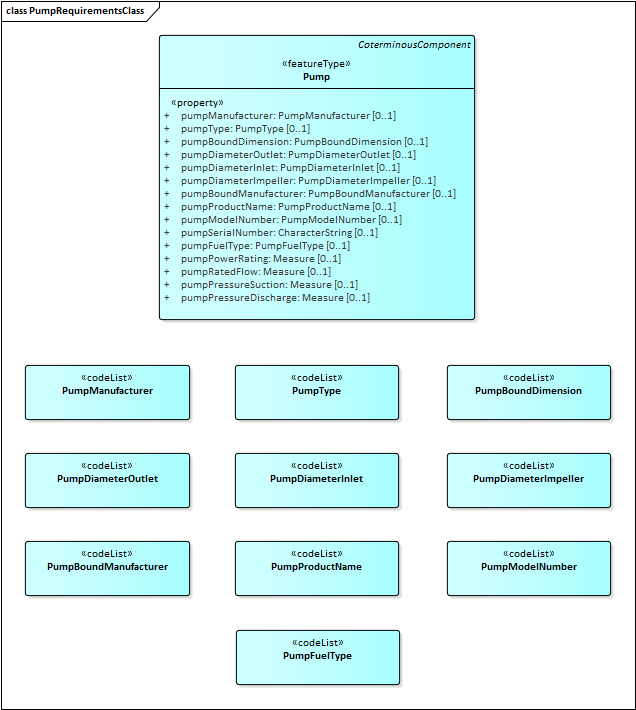


Figure 23. Pump Requirements Class

The Pump Class contains attributes and relationships to other classes:

**pumpManufacturer**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute pumpManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pumpmanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/pump/pumpmanufacturer-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute pumpManufacturer MAY be populated, or its corresponding bound attribute pumpBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pump/pumpmanufacturer-exclusivity** |

**pumpType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of pump component

|  |
| --- |
| 1. This optional unbound attribute pumpType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pumpmetertype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/pump/pumptype-codelistURI** |

**pumpBoundDimension**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: pumpDiameterOutlet; pumpDiameterInlet; pumpDiameterImpeller.

|  |
| --- |
| 1. This optional bound attribute pumpBoundDimension SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/pump/pumpbounddimension-boundtype** |

|  |
| --- |
| 1. This optional bound attribute pumpBoundDimension SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pumpbounddimension.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/pump/pumpbounddimension-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute pumpBoundDimension MAY be populated, or its corresponding set of unbound attributes (pumpDiameterOutlet, pumpDiameterInlet, pumpDiameterImpeller) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pump/pumpbounddimension-exclusivity** |

**pumpDiameterOutlet**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the outlet diameter size of the component. This attribute SHALL NOT be provided if the attribute pumpBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute pumpDiameterOutlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pumpdiameteroutlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/pump/pumpdiameteroutlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute pumpDiameterOutlet MAY be populated, or its corresponding bound attribute pumpBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pump/pumpdiameteroutlet-exclusivity** |

**pumpDiameterInlet**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the inlet diameter size of the component. This attribute SHALL NOT be provided if the attribute pumpBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute pumpDiameterInlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pumpdiameterinlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/pump/pumpdiameterinlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute pumpDiameterInlet MAY be populated, or its corresponding bound attribute pumpBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pump/pumpdiameterinlet-exclusivity** |

**pumpDiameterImpeller**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the impeller diameter size of the component. This attribute SHALL NOT be provided if the attribute pumpBoundDimension is populated for reasons of disambiguation..

|  |
| --- |
| 1. This optional unbound attribute pumpDiameterImpeller SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pumpdiameterimpeller.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/pump/pumpdiameterimpeller-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute pumpDiameterImpeller MAY be populated, or its corresponding bound attribute pumpBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pump/pumpdiameterimpeller-exclusivity** |

**pumpBoundManufacturer**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: pumpManufacturer; pumpProductName; pumpModelNumber.

|  |
| --- |
| 1. This optional bound attribute pumpBoundManufacturer SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/pump/pumpboundmanufacturer-boundtype** |

|  |
| --- |
| 1. This optional bound attribute pumpBoundManufacturer SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pumpboundmanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/pump/pumpboundmanufacturer-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute pumpBoundManufacturer MAY be populated, or its corresponding set of unbound attributes (pumpManufacturer, pumpProductName, and pumpModelNumber) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pump/pumpboundmanufacturer-exclusivity** |

**pumpProductName**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the product name of the component. This attribute SHALL NOT be provided if the attribute pumpBoundManufacturer is populated for reasons of disambiguation. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute pumpProductName SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pumpproductname.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/pump/pumpproductname-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute pumpProductName MAY be populated, or its corresponding bound attribute pumpBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pump/pumpproductname-exclusivity** |

**pumpModelNumber**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the model number of the component. This attribute SHALL NOT be provided if the attribute pumpBoundManufacturer is populated for reasons of disambiguation. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute pumpModelNumber SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pumpmodelnumber.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/pump/pumpmodelnumber-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute pumpModelNumber MAY be populated, or its corresponding bound attribute pumpBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/pump/pumpmodelnumber-exclusivity** |

**pumpSerialNumber**: An optional string attribute that defines a unique identification number established by the component manufacturer

**pumpFuelType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value describing the type of fuel used to operate the component

|  |
| --- |
| 1. This optional attribute pumpFuelType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/pumpfueltype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/pump/pumpfueltype-codelistURI** |

**pumpPowerRating**: An optional measurement made by the manufacturer that defines the amount of power the pump is rated to produce. If this attribute is populated with a value, a units-of-measure sub-attribute SHALL also be provided.

|  |
| --- |
| 1. If this optional attribute pumpPowerRating is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/pump/pumppowerrating-uom** |

**pumpRatedFlow**: An optional measurement made by the manufacturer that defines the rate of product flow the pump is rated to produce. If populated is populated with a value, a units-of-measure sub-attribute SHALL also be provided.

|  |
| --- |
| 1. If this optional attribute pumpRatedFlow is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/pump/pumpratedflow-uom** |

**pumpPressureSuction**: An optional measurement made by the manufacturer that defines the amount of suction the pump is rated to produce. If populated is populated with a value, a units-of-measure sub-attribute SHALL also be provided.

|  |
| --- |
| 1. If this optional attribute pumpPressureSuction is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/pump/pumppressuresuction-uom** |

**pumpPressureDischarge**: An optional measurement made by the manufacturer that defines the amount of pressure the pump is rated to discharge. If populated is populated with a value, a units-of-measure sub-attribute SHALL also be provided.

|  |
| --- |
| 1. If this optional attribute pumpPressureDischarge is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/pump/pumppressuredischarge-uom** |

#### Reducer Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/reducer** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Reducer |
| Dependency | **/req/pmlcomponent** |
| **Req 211** | **/req/reducer/reducermanufacturer-codelistURIexception** |
| **Req 212** | **/req/reducer/reducertype-codelistURI** |
| **Req 213** | **/req/reducer/reducermaterial-codelistURI** |
| **Req 214** | **/req/reducer/reducerboundspecification-boundtype** |
| **Req 215** | **/req/reducer/reducerboundspecification-codelistURI** |
| **Req 216** | **/req/reducer/reducerboundspecification-exclusivity** |
| **Req 217** | **/req/reducer/reducerspecification-codelistURI** |
| **Req 218** | **/req/reducer/reducerspecification-exclusivity** |
| **Req 219** | **/req/reducer/reducerweight-codelistURI** |
| **Req 220** | **/req/reducer/reducerweight-exclusivity** |
| **Req 221** | **/req/reducer/reducerbounddimension-boundtype** |
| **Req 222** | **/req/reducer/reducerbounddimension-codelistURI** |
| **Req 223** | **/req/reducer/reducerbounddimension-exclusivity** |
| **Req 224** | **/req/reducer/reducernominalpipesizeinlet-codelistURI** |
| **Req 225** | **/req/reducer/reducernominalpipesizeinlet-exclusivity** |
| **Req 226** | **/req/reducer/reducernominalpipesizeoutlet-codelistURI** |
| **Req 227** | **/req/reducer/reducernominalpipesizeoutlet-exclusivity** |
| **Req 228** | **/req/reducer/reduceroutsidediameterinlet-codelistURI** |
| **Req 229** | **/req/reducer/reduceroutsidediameterinlet-exclusivity** |
| **Req 230** | **/req/reducer/reduceroutsidediameteroutlet-codelistURI** |
| **Req 231** | **/req/reducer/reduceroutsidediameteroutlet-exclusivity** |
| **Req 232** | **/req/reducer/reducerwallthicknessinlet-codelistURI** |
| **Req 233** | **/req/reducer/reducerwallthicknessinlet-exclusivity** |
| **Req 234** | **/req/reducer/reducerwallthicknessoutlet-codelistURI** |
| **Req 235** | **/req/reducer/reducerwallthicknessoutlet-exclusivity** |

The Reducer Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 24.

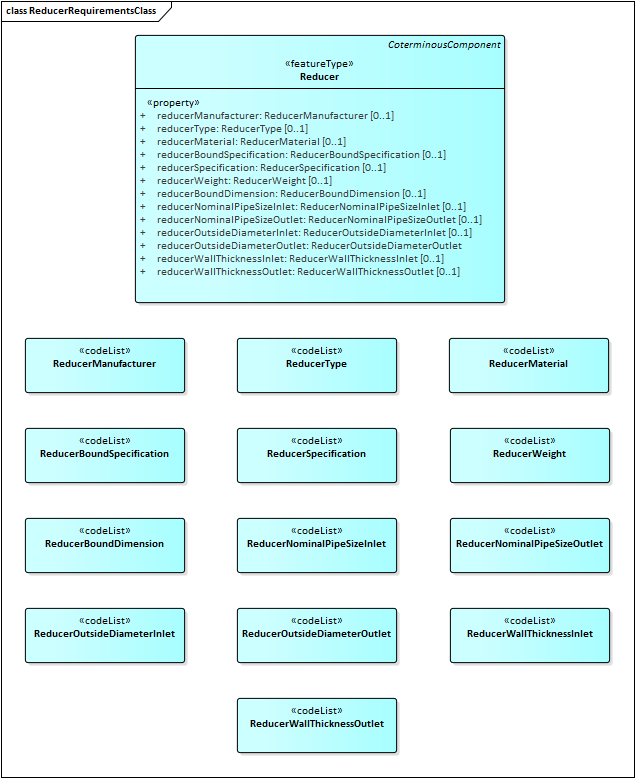


Figure 24. Reducer Requirements Class

The Reducer Class contains attributes and relationships to other classes:

**reducerManufacturer**: An optional attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute reducerManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reducermanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/reducer/reducermanufacturer-codelistURIexception** |

**reducerType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of reducer component.

|  |
| --- |
| 1. This optional attribute reducerType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reducertype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/reducer/reducertype-codelistURI** |

**reducerMaterial**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of material use in the manufacture of the component

|  |
| --- |
| 1. This optional attribute reducerMaterial SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reducermaterial.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/reducer/reducermaterial-codelistURI** |

**reducerBoundSpecification**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: reducerSpecification; reducerWeight.

|  |
| --- |
| 1. This optional bound attribute reducerBoundSpecification SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/reducer/reducerboundspecification-boundtype** |

|  |
| --- |
| 1. This optional bound attribute reducerBoundSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reducerboundspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/reducer/reducerboundspecification-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute reducerBoundSpecification MAY be populated, or its corresponding set of unbound attributes (reducerSpecification, and reducerWeight) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/reducer/reducerboundspecification-exclusivity** |

**reducerSpecification**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value used to define the specification of the component. This attribute SHALL NOT be provided if the attribute reducerBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute reducerSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reducerspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/reducer/reducerspecification-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute reducerSpecification MAY be populated, or its corresponding bound attribute reducerBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/reducer/reducerspecification-exclusivity** |

**reducerWeight**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the weight characteristics of the component. This attribute SHALL NOT be provided if the attribute reducerBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute reducerWeight SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reducerweight.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/reducer/reducerweight-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute reducerWeight MAY be populated, or its corresponding bound attribute reducerBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/reducer/reducerweight-exclusivity** |

**reducerBoundDimension**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: reducerNominalPipeSizeInlet; reducerNominalPipeSizeOutlet; reducerOutsideDiameterInlet; reducerOutsideDiameterOutlet; reducerWallThicknessInlet; reducerWallThicknessOutlet.

|  |
| --- |
| 1. This optional bound attribute reducerBoundDimension SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/reducer/reducerbounddimension-boundtype** |

|  |
| --- |
| 1. This optional bound attribute reducerBoundDimension SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reducerbounddimension.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/reducer/reducerbounddimension-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute reducerBoundDimension MAY be populated, or its corresponding set of unbound attributes (reducerNominalPipeSizeInlet, and reducerNominalPipeSizeOutlet) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/reducer/reducerbounddimension-exclusivity** |

**reducerNominalPipeSizeInlet**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the nominal pipe size on the inlet of the component

|  |
| --- |
| 1. This optional unbound attribute reducerNominalPipeSizeInlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reducernominalpipesizeinlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/reducer/reducernominalpipesizeinlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute reducerNominalPipeSizeInlet MAY be populated, or its corresponding bound attribute reducerBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/reducer/reducernominalpipesizeinlet-exclusivity** |

**reducerNominalPipeSizeOutlet**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the nominal pipe size on the outlet of the component

|  |
| --- |
| 1. This optional unbound attribute reducerNominalPipeSizeOutlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reducernominalpipesizeoutlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/reducer/reducernominalpipesizeoutlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute reducerNominalPipeSizeOutlet MAY be populated, or its corresponding bound attribute reducerBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/reducer/reducernominalpipesizeoutlet-exclusivity** |

**reducerOutsideDiameterInlet**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the outside diameter size of the component inlet

|  |
| --- |
| 1. This optional unbound attribute reducerNominalOutsideDiameterInlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reduceroutsidediameterinlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/reducer/reduceroutsidediameterinlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute reducerNominalOutsideDiameterInlet MAY be populated, or its corresponding bound attribute reducerBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/reducer/reduceroutsidediameterinlet-exclusivity** |

**reducerOutsideDiameterOutlet**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the outside diameter size of the component outlet

|  |
| --- |
| 1. This optional unbound attribute reducerNominalOutsideDiameterOutlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reduceroutsidediameteroutlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/reducer/reduceroutsidediameteroutlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute reducerNominalOutsideDiameterOutlet MAY be populated, or its corresponding bound attribute reducerBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/reducer/reduceroutsidediameteroutlet-exclusivity** |

**reducerWallThicknessInlet**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the wall thickness size of the component inlet

|  |
| --- |
| 1. This optional unbound attribute reducerWallThicknessInlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reducerwallthicknessinlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/reducer/reducerwallthicknessinlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute reducerWallThicknessInlet MAY be populated, or its corresponding bound attribute reducerBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/reducer/reducerwallthicknessinlet-exclusivity** |

**reducerWallThicknessOutlet**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the wall thickness size of the component outlet

|  |
| --- |
| 1. This optional unbound attribute reducerWallThicknessOutlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/reducerwallthicknessoutlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/reducer/reducerwallthicknessoutlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute reducerWallThicknessOutlet MAY be populated, or its corresponding bound attribute reducerBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/reducer/reducerwallthicknessoutlet-exclusivity** |

#### Tap Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/tap** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Tap |
| Dependency | **/req/pmlcomponent** |
| **Req 236** | **/req/tap/tapmanufacturer-codelistURIexception** |
| **Req 237** | **/req/tap/taptype-codelistURI** |
| **Req 238** | **/req/tap/tapmaterial-codelistURI** |
| **Req 239** | **/req/tap/tapbounddimension-boundtype** |
| **Req 240** | **/req/tap/tapbounddimension-codelistURI** |
| **Req 241** | **/req/tap/tapbounddimension-exclusivity** |
| **Req 242** | **/req/tap/tapnominalpipesize-codelistURI** |
| **Req 243** | **/req/tap/tapnominalpipesize-exclusivity** |
| **Req 244** | **/req/tap/tapoutsidediameterinlet-codelistURI** |
| **Req 245** | **/req/tap/tapoutsidediameterinlet-exclusivity** |
| **Req 246** | **/req/tap/tapoutsidediameteroutlet-codelistURI** |
| **Req 247** | **/req/tap/tapoutsidediameteroutlet-exclusivity** |
| **Req 248** | **/req/tap/tapfunction-codelistURI** |
| **Req 249** | **/req/tap/tapinstallmethod-codelistURI** |

The Tap Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 25.

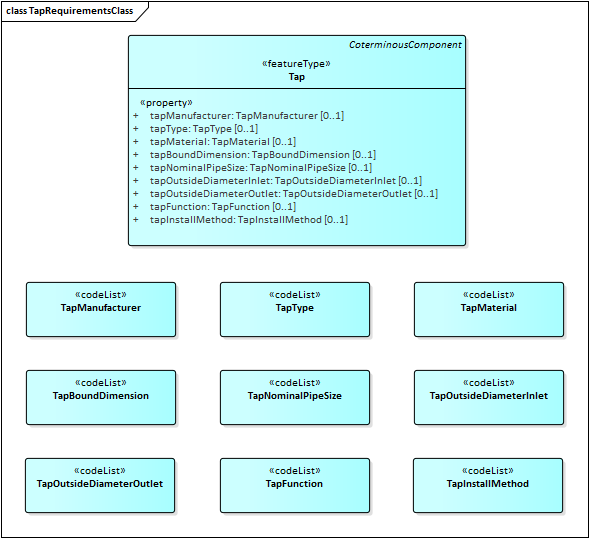


Figure 25. Tap Requirements Class

The Tap Class contains attributes and relationships to other classes:

**tapManufacturer**: An optional attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute tapManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/tapmanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/tap/tapmanufacturer-codelistURIexception** |

**tapType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of tap component

|  |
| --- |
| 1. This optional attribute tapType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/taptype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tap/taptype-codelistURI** |

**tapMaterial**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of material use in the manufacture of the component

|  |
| --- |
| 1. This optional attribute tapMaterial SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/tapmaterial.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tap/tapmaterial-codelistURI** |

**tapBoundDimension**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: tapNominalPipeSize; tapOutsideDiameterInlet; tapOutsideDiameterOutlet.

|  |
| --- |
| 1. This optional bound attribute tapBoundDimension SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/tap/tapbounddimension-boundtype** |

|  |
| --- |
| 1. This optional bound attribute tapBoundDimension SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/tapbounddimension.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/tap/tapbounddimension-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute tapBoundDimension MAY be populated, or its corresponding set of unbound attributes (tapNominalPipeSize, tapOutsideDiameterInlet, and tapOutsideDiameterOutlet) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tap/tapbounddimension-exclusivity** |

**tapNominalPipeSize**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the nominal pipe size of the component

|  |
| --- |
| 1. This optional unbound attribute tapNominalPipeSize SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/tapnominalpipesize.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tap/tapnominalpipesize-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute tapNominalPipeSize MAY be populated, or its corresponding bound attribute tapBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tap/tapnominalpipesize-exclusivity** |

**tapOutsideDiameterInlet**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the outside diameter size of the component inlet

|  |
| --- |
| 1. This optional unbound attribute tapOutsideDiameterInlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/tapoutsidediameterinlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tap/tapoutsidediameterinlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute tapOutsideDiameterInlet MAY be populated, or its corresponding bound attribute tapBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tap/tapoutsidediameterinlet-exclusivity** |

**tapOutsideDiameterOutlet**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the outside diameter size of the component outlet

|  |
| --- |
| 1. This optional unbound attribute tapOutsideDiameterOutlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/tapoutsidediameteroutlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.s   **/req/tap/tapoutsidediameteroutlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute tapOutsideDiameterOutlet MAY be populated, or its corresponding bound attribute tapBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tap/tapoutsidediameteroutlet-exclusivity** |

**tapFunction**: An optional attribute that, if populated, SHALL be dereferenceable to a string value containing the function this tap performs

|  |
| --- |
| 1. This optional attribute tapFunction SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/tapfunction.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.s   **/req/tap/tapfunction-codelistURI** |

**tapInsallMethod**: An optional attribute that, if populated, SHALL be dereferenceable to a string value containing the description of the method used to install the component

|  |
| --- |
| 1. This optional attribute tapInsallMethod SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/tapinsallmethod.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.s   **/req/tap/tapinstallmethod-codelistURI** |

#### Tee Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/tee** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Tee |
| Dependency | **/req/pmlcomponent** |
| **Req 250** | **/req/tee/teemanufacturer-codelistURIexception** |
| **Req 251** | **/req/tee/teetype-codelistURI** |
| **Req 252** | **/req/tee/teematerial-codelistURI** |
| **Req 253** | **/req/tee/teeboundspecification-boundtype** |
| **Req 254** | **/req/tee/teeboundspecification-codelistURI** |
| **Req 255** | **/req/tee/teeboundspecification-exclusivity** |
| **Req 256** | **/req/tee/teespecification-codelistURI** |
| **Req 257** | **/req/tee/teespecification-exclusivity** |
| **Req 258** | **/req/tee/teeweight-codelistURI** |
| **Req 259** | **/req/tee/teeweight-exclusivity** |
| **Req 260** | **/req/tee/teebounddimension-boundtype** |
| **Req 261** | **/req/tee/teebounddimension-codelistURI** |
| **Req 262** | **/req/tee/teebounddimension-exclusivity** |
| **Req 263** | **/req/tee/teenominalpipesize-codelistURI** |
| **Req 264** | **/req/tee/teenominalpipesize-exclusivity** |
| **Req 265** | **/req/tee/teeoutsidediameterrun-codelistURI** |
| **Req 266** | **/req/tee/teeoutsidediameterrun-exclusivity** |
| **Req 267** | **/req/tee/teeoutsidediameteroutlet-codelistURI** |
| **Req 268** | **/req/tee/teeoutsidediameteroutlet-exclusivity** |
| **Req 269** | **/req/tee/teecentertoendrun-codelistURI** |
| **Req 270** | **/req/tee/teecentertoendrun-exclusivity** |
| **Req 271** | **/req/tee/teecentertoendoutlet-codelistURI** |
| **Req 272** | **/req/tee/teecentertoendoutlet-exclusivity** |
| **Req 273** | **/req/tee/teewallthicknessrun-codelistURI** |
| **Req 274** | **/req/tee/teewallthicknessrun-exclusivity** |
| **Req 275** | **/req/tee/teewallthicknessoutlet-codelistURI** |
| **Req 276** | **/req/tee/teewallthicknessoutlet-exclusivity** |

The Tee Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 26.

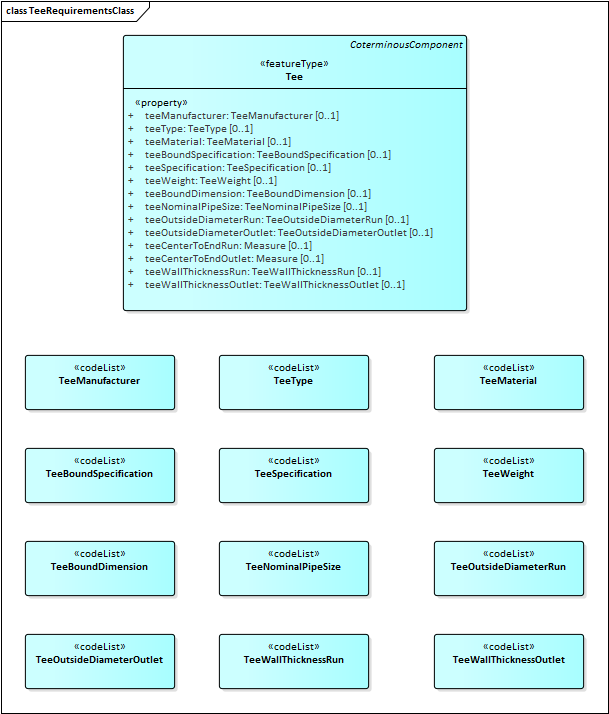


Figure 26. Tee Requirements Class

The Tee Class contains attributes and relationships to other classes:

**teeManufacturer**: An optional attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute teeManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teemanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/tee/teemanufacturer-codelistURIexception** |

**teeType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of tee component

|  |
| --- |
| 1. This optional attribute teeType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teetype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tee/teetype-codelistURI** |

**teeMaterial**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of material use in the manufacture of the component

|  |
| --- |
| 1. This optional attribute teeMaterial SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teematerial.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tee/teematerial-codelistURI** |

**teeBoundSpecification**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: teeSpecification; teeWeight.

|  |
| --- |
| 1. This optional bound attribute teeBoundSpecification SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/tee/teeboundspecification-boundtype** |

|  |
| --- |
| 1. This optional bound attribute teeBoundSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teeboundspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/tee/teeboundspecification-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute teeBoundSpecification MAY be populated, or its corresponding set of unbound attributes (teeSpecification, and teeWeight) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tee/teeboundspecification-exclusivity** |

**teeSpecification**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value used to define the specification of the component. This attribute SHALL NOT be provided if the attribute teeBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute teeSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teespecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tee/teespecification-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute teeSpecification MAY be populated, or its corresponding bound attribute teeBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tee/teespecification-exclusivity** |

**teeWeight**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the weight characteristics of the component. This attribute SHALL NOT be provided if the attribute teeBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute teeWeight SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teeweight.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tee/teeweight-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute teeWeight MAY be populated, or its corresponding bound attribute teeBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tee/teeweight-exclusivity** |

**teeBoundDimension**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: teeNominalPipeSize; teeNominalOutsideDiameterRun; teeNominalOutsideDiameterOutlet; teeCenterToEndRun; teeCenterToEndOutlet; teeWallThicknessRun; teeWallThicknessOutlet.

|  |
| --- |
| 1. This optional bound attribute teeBoundDimension SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/tee/teebounddimension-boundtype** |

|  |
| --- |
| 1. This optional bound attribute teeBoundDimension SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teebounddimension.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/tee/teebounddimension-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute teeBoundDimension MAY be populated, or its corresponding set of unbound attributes (teeNominalPipeSize, teeNominalOutsideDiameterRun, and teeNominalOutsideDiameterOutlet) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tee/teebounddimension-exclusivity** |

**teeNominalPipeSize**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the nominal pipe size of the component. This attribute SHALL NOT be provided if the attribute teeBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute teeNominalPipeSize SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teenominalpipesize.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tee/teenominalpipesize-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute teeNominalPipeSize MAY be populated, or its corresponding bound attribute teeBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tee/teenominalpipesize-exclusivity** |

**teeOutsideDiameterRun**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the outside diameter size of the component. This attribute SHALL NOT be provided if the attribute teeBoundDimension is populated for reasons of disambiguation. This attribute is illustrated in Figure 27.

Figure 27. TeeOutsideDiameterRun

|  |
| --- |
| 1. This optional unbound attribute teeOutsideDiameterRun SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teeoutsidediameterrun.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tee/teeoutsidediameterrun-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute teeOutsideDiameterRun MAY be populated, or its corresponding bound attribute teeBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tee/teeoutsidediameterrun-exclusivity** |

**teeOutsideDiameterOutlet**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the outside diameter size of the component outlet. This attribute SHALL NOT be provided if the attribute teeBoundDimension is populated for reasons of disambiguation. This attribute is illustrated in Figure 28.

Figure 28. TeeOutsideDiameterOutlet

|  |
| --- |
| 1. This optional unbound attribute teeOutsideDiameterOutlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teeoutsidediameteroutlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tee/teeoutsidediameteroutlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute teeOutsideDiameterOutlet MAY be populated, or its corresponding bound attribute teeBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tee/teeoutsidediameterrun-exclusivity** |

**teeCenterToEndRun**: An optional unbound attribute describing the length of the component from the center to the end of the run. If this attribute is populated, the units-of-measure sub-attribute SHALL also be populated. This attribute is illustrated in Figure 29.

Figure 29. TeeCenterToEndRun

|  |
| --- |
| 1. This optional unbound attribute teeCenterToEndRun SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teecentertoendrun.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tee/teecentertoendrun-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute teeCenterToEndRun MAY be populated, or its corresponding bound attribute teeBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tee/teecentertoendrun-exclusivity** |

**teeCenterToEndOutlet**: An optional unbound attribute describing the length of the component from the center to the end of the outlet. If this attribute is populated, the units-of-measure sub-attribute SHALL also be populated. This attribute is illustrated in Figure 30.

Figure 30. TeeCenterToEndOutlet

|  |
| --- |
| 1. This optional unbound attribute teeCenterToEndOutlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teecentertoendoutlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tee/teecentertoendoutlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute teeCenterToEndOutlet MAY be populated, or its corresponding bound attribute teeBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tee/teecentertoendoutlet-exclusivity** |

**teeWallThicknessRun**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the wall thickness size of the component run. This attribute is illustrated in Figure 31.

Figure 31. TeeWallThicknessRun

|  |
| --- |
| 1. This optional unbound attribute teeWallThicknessRun SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teewallthicknessrun.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tee/teewallthicknessrun-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute teeWallThicknessRun MAY be populated, or its corresponding bound attribute teeBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tee/teewallthicknessrun-exclusivity** |

**teeWallThicknessOutlet**: An optional unbound value that, if populated, SHALL be dereferenceable to define a string value containing the wall thickness size of the component outlet. This attribute is illustrated in Figure 32.

Figure 32. TeeWallThicknessOutlet

|  |
| --- |
| 1. This optional unbound attribute teeWallThicknessOutlet SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/teewallthicknessoutlet.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/tee/teewallthicknessoutlet-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute teeWallThicknessOutlet MAY be populated, or its corresponding bound attribute teeBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/tee/teewallthicknessoutlet-exclusivity** |

#### Valve Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/valve** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Valve |
| Dependency | **/req/pmlcomponent** |
| **Req 277** | **req/valve/valvemanufacturer-codelistURIexception** |
| **Req 278** | **/req/valve/valvetype-codelistURI** |
| **Req 279** | **/req/valve/valveboundspecification-boundtype** |
| **Req 280** | **/req/valve/valveboundspecification-codelistURI** |
| **Req 281** | **/req/valve/valveboundspecification-exclusivity** |
| **Req 282** | **/req/valve/valvespecification-codelistURI** |
| **Req 283** | **/req/valve/valvespecification-exclusivity** |
| **Req 284** | **/req/valve/valveclass-codelistURI** |
| **Req 285** | **/req/valve/valveclass-exclusivity** |
| **Req 286** | **/req/valve/valvenominalpipesize-codelistURI** |
| **Req 287** | **/req/valve/valveoperability-codelistURI** |
| **Req 288** | **/req/valve/valveactuationTime-uom** |
| **Req 289** | **/req/valve/valvefunctiontype-codelistURI** |
| **Req 290** | **/req/valve/valvemodelnumber-codelistURIexception** |

The Valve Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 33.

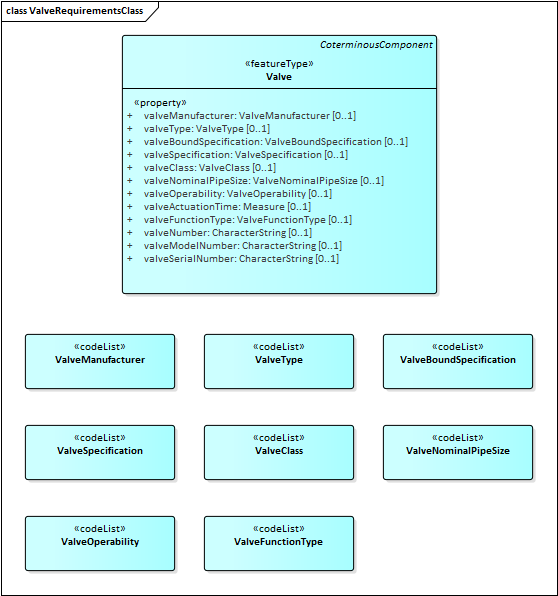


Figure 33. Valve Requirements Class

The Valve Class contains attributes and relationships to other classes:

**valveManufacturer**: An optional value that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute valveManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/valvemanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/valve/valvemanufacturer-codelistURIexception** |

**valveType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of valve component

|  |
| --- |
| 1. This optional attribute valveType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/valvetype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/valve/valvetype-codelistURI** |

**valveBoundSpecification**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: valveSpecification; valveClass.

|  |
| --- |
| 1. This optional bound attribute valveBoundSpecification SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/valve/valveboundspecification-boundtype** |

|  |
| --- |
| 1. This optional bound attribute valveBoundSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/valveboundspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/valve/valveboundspecification-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute valveBoundSpecification MAY be populated, or its corresponding set of unbound attributes (valveSpecification, and valveClass) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/valve/valveboundspecification-exclusivity** |

**valveSpecification**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value used to define the specification of the component. This attribute SHALL NOT be provided if the attribute valveBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute valveSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/valvespecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/valve/valvespecification-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute valveSpecification MAY be populated, or its corresponding bound attribute valveBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/valve/valvespecification-exclusivity** |

**valveClass**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the class of the valve component. This attribute SHALL NOT be provided if the attribute valveBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute valveClass SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/valveclass.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/valve/valveclass-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute valveClass MAY be populated, or its corresponding bound attribute valveBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/valve/valveclass-exclusivity** |

**valveNominalPipeSize**: An optional value that, if populated, SHALL be dereferenceable to a string value containing the nominal pipe size of the component

|  |
| --- |
| 1. This optional attribute valveNominalPipeSize SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/valvenominalpipesize.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/valve/valvenominalpipesize-codelistURI** |

**valveOperability**: An optional value that, if populated, SHALL be deferenceable to a string value describing the method of operability of the component (i.e. actuator type, and remote or onsite operability)

|  |
| --- |
| 1. This optional attribute valveOperability SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/valveoperability.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/valve/valveoperability-codelistURI** |

**valveActuationTime**: An optional measurement attribute that indicates the actuation time (time required to fully open or close the valve) of the component as determined by the manufacturercomponent outlet. This attribute is illustrated in Figure 33.

|  |
| --- |
| 1. If this optional attribute valveActuationTime is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/valve/valveactuationTime-uom** |

**valveFunctionType**: An optional attribute that, if populated, SHALL be dereferenceable to a string value describing the primary function the component is intended to fulfill

|  |
| --- |
| 1. This optional attribute valveFunctionType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/valvefunctiontype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/valve/valvefunctiontype-codelistURI** |

**valveNumber**: An optional string attribute that defines a unique identification number for the component established by the pipeline operator

**valveModelNumber**: An optional string attribute containing the model number of the component as established by the component manufacturer

|  |
| --- |
| 1. **[Recommendation]** This optional attribute valveModelNumber SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/valvemodelnumber.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/valve/valvemodelnumber-codelistURIexception** |

**valveSerialNumber**: An optional string attribute containing the serial number of the component as established by the component manufacturer

#### AppurtenantComponent Class Requirements Class (Normative)

The AppurtenantComponent Class is an abstract class and therefore SHALL NOT be encoded into a PipelineML dataset. This class provides an inheritance mechanism for passing attributes on to other classes that inherit from it. This class also establishes relationships with other classes in the model. This class contains classes and attributes as shown in Figure 34.

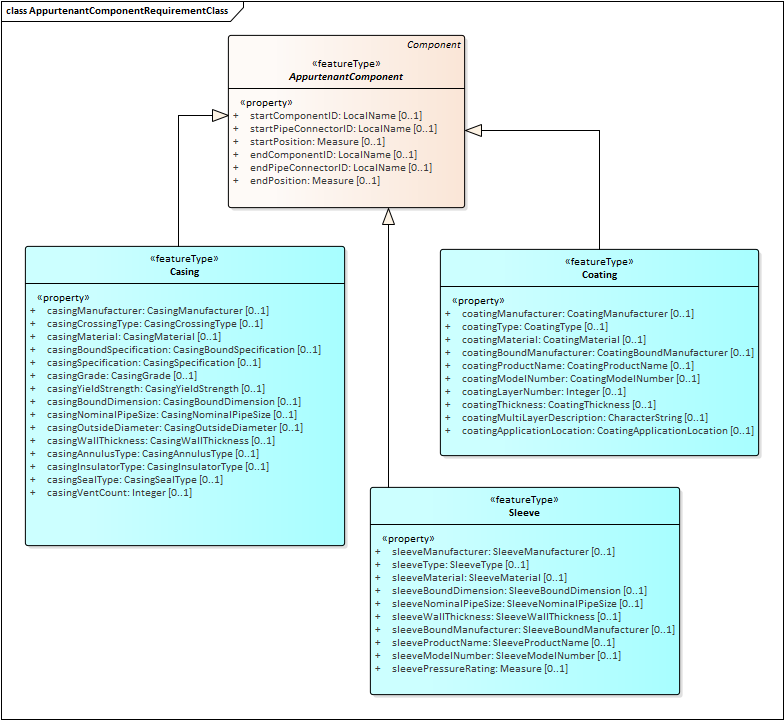


Figure 34. AppurtenantComponent Requirements Class

The AppurtenantComponent Class contains attributes and relationships to other classes. All attributes of the abstract AppurtenantComponent Class are optional due to the various methods of location that are made available to this class (see informative section below):

**startComponentID**: An optional unique identifier associated with a concrete CoterminousComponent feature to which the start of this appurtenant component is attached

**startPipeConnectorID**: An optional unique identifier associated with a concrete PipeConnector feature to which the start of this appurtenant component is referenced.

**startPosition**: An optional measurement of the distance from the start of the coterminous component or pipe connector to which this appurtenant component is attached to the start of the appurtenant component (see Figures 35-39 in informative section)

**endComponentID**: An optional unique identifier associated with a concrete CoterminousComponent feature to which the end of this appurtenant component is attached

**endPipeConnectorID**: An optional unique identifier associated with a concrete PipeConnector feature to which the end of this appurtenant component is referenced.

**endPosition**: An optional measurement of the distance from the start of the coterminous component or pipe connector to which this appurtenant component is attached to the end of the appurtenant component (see Figures 35-39 in informative section)

**casing**: Any number of Casings

**coating**: Any number of Coatings

**sleeve**: Any number of Sleeves

##### Appurtenant Component Location Methods (Informative)

PipelineML defines multiple methods for defining the location of appurtenant components. Each method enables unambiguous definition of the position of appurtenances. Method selection SHALL be determined based on the business use case being addressed by the data interchange and which location method best services the needs of that use case.

##### Method 1 – Absolute Geospatial Location

This method provides absolute geospatial location using one of two variants. This method provides the most accurate location of the appurtenance in the most succinct manner possible. This location method can be combined with methods 2 and 3. In this case, the geometry of method 1 (absolute position) has precedence on the geometry of methods 2 or 3.

##### Method 1a - Absolute Geospatial, Point or Linestring (2D or 3D point or centerline)

The following diagram illustrates this method variant.



Figure 35. Method 1a - Absolute Geospatial, Point or Linestring

The exact geospatial location of the appurtenance is described in the ‘Location’ tag. For linear appurtenances, this is a 2D or 3D gml:LineString representing the centerline of the appurtenance. For point appurtenances, this is a 2D or 3D gml:Point representing the center of the appurtenance.

There is no direct link between the appurtenance and the coterminous componentry, although this method can be combined with relative positioning methods (methods 2-3) if such a relation is required.

The order on which the appurtenance geometry vertices are entered is arbitrary.

This method is recommended for datasets in which no direct relationship between appurtenances and coterminous componentry exists and where the exact position of the appurtenances has been independently measured.

##### Method 1b – Absolute Geospatial, Polygon 2D

The following diagram illustrates this method variant.



Figure 36. Method 1b – Absolute Geospatial, Polygon 2D

The exact geospatial location of the appurtenance is described in the ‘Location’ tag. For both linear and point appurtenances this is described by a 2D gml:Polygon that represents the outline of the appurtenance in a top-down view.

No 3D polygons are supported in this release of the PipelineML Standard.

There is no direct link between the appurtenance and the coterminous componentry, although this method can be combined with relative positioning methods (methods 2-3) if such a relation is required.

This method is recommended if the 2D perimeter of appurtenances is known and is important. It cannot be used when 3D position of appurtenances is required.

##### Method 2 – Relative Directionality-based Location

This method provides relative geospatial location based on the relationship of the appurtenance to its underlying coterminous components. This method supports directionality and should be preferred where business use cases specify pipeline or component directionality. It features two variants.

##### Method 2a – Relative, Girth Welds, Directionality

This variant is intended for cases in which directionality needs to be supported and the location of girth welds are known. The following diagram illustrates 2 examples of this method variant.



Figure 37. Method 2a – Relative, Girth Welds, Directionality

The geospatial location of the appurtenance is described relative to the position of coterminous components that relate to these appurtenances. For linear appurtenances these are the first and last component of the relation. For point appurtenances this is the closest component. For referencing to girth welds, the appurtenance is described by two connectors: one at the start of the appurtenance and one at the end.

By referencing both connectors, a local direction is defined along which all measures are taken: positive measures are along the local direction, negative measures are against the local direction.

The start position is measured along the local direction, offset at the start connector. The end position is offset at the end connector and is also measured along the local direction. 2D and 3D appurtenances are supported as they inherit the geometry of the coterminous components they relate to.

This method is recommended if appurtenance positions are not separately measured, but the position relative to componentry is known. This method requires girth welds to be explicitly modeled.

##### Method 2b – Relative, Componentry, Directionality

This variant is intended for cases in which directionality needs to be supported and the location of components are known. The following diagram illustrates 3 examples of this method variant.



Figure 38. Method 2b – Relative, Componentry, Directionality

This is a variation on method 2a. The appurtenances are now described relative to components instead of girth welds. This method is recommended if appurtenance positions are not separately measured, but the position relative to componentry is known. This method does not require girth welds to be explicitly modeled.

##### Method 3 – Relative Non-directionality Location

This method provides relative geospatial location based on the relationship of the appurtenance to its underlying coterminous components. This method does not require directionality of the pipeline to be known to correctly reference the start and end points of the appurtenance. It requires 3 references to note the beginning point of an appurtenance and 3 references to indicate the ending point. The following diagram illustrates 3 examples of using this method to reference the location of an appurtenance. When possible, this method is considered a best practice over method 2 as it eliminates any potential confusion regarding pipeline or component directionality.



Figure 39. Method 3 – Relative Non-directionality Location

* The geospatial location of the appurtenance is described relative to the position of coterminous components that relate to these appurtenances.
* For linear appurtenances these are:
* the first component of the relation
* the last component of the relation
* a first girth weld connecting to the first component
* a second girth weld connecting to the last component
* For point appurtenances these are:
* the closest component
* a girth weld connecting to this component
* The start of the appurtenance is defined by measuring the distance from the first girth weld in the direction of the first component. The end is defined from the second girth weld measuring in the direction of the last component.
* Only positive measures are used.
* This method is recommended if appurtenance positions are not separately measured, but the position relative to componentry is known. This method requires girth welds to be explicitly modeled.

#### Casing Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/casing** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Casing |
| Dependency | **/req/pmlcomponent** |
| **Req 291** | **/req/casing/casingmanufacturer-codelistURIexception** |
| **Req 292** | **/req/casing/casingcrossingtype-codelistURI** |
| **Req 293** | **/req/casing/casingmaterial-codelistURI** |
| **Req 294** | **/req/casing/casingboundspecification-boundtype** |
| **Req 295** | **/req/casing/casingboundspecification-codelistURI** |
| **Req 296** | **/req/casing/casingboundspecification-exclusivity** |
| **Req 297** | **/req/casing/casingspecification-codelistURI** |
| **Req 298** | **/req/casing/casingspecification-exclusivity** |
| **Req 299** | **/req/casing/casingspecification-codelistURI** |
| **Req 300** | **/req/casing/casingspecification-exclusivity** |
| **Req 301** | **/req/casing/casingyieldstrength-codelistURI** |
| **Req 302** | **/req/casing/casingyieldstrength-exclusivity** |
| **Req 303** | **/req/casing/casingbounddimension-boundtype** |
| **Req 304** | **/req/casing/casingbounddimension-codelistURI** |
| **Req 305** | **/req/casing/casingbounddimension-exclusivity** |
| **Req 306** | **/req/casing/casingnominalpipesize-codelistURI** |
| **Req 307** | **/req/casing/casingnominalpipesize-exclusivity** |
| **Req 308** | **/req/casing/casingoutsidediameter-codelistURI** |
| **Req 309** | **/req/casing/casingoutsidediameter-exclusivity** |
| **Req 310** | **/req/casing/casingwallthickness-codelistURI** |
| **Req 311** | **/req/casing/casingwallthickness-exclusivity** |
| **Req 312** | **/req/casing/casingannulustype-codelistURI** |
| **Req 313** | **/req/casing/casinginsulatortype-codelistURI** |
| **Req 314** | **/req/casing/casingsealtype-codelistURI** |
| **Req 315** | **/req/casing/casingventcount-codelistURI** |

The Casing Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 40.

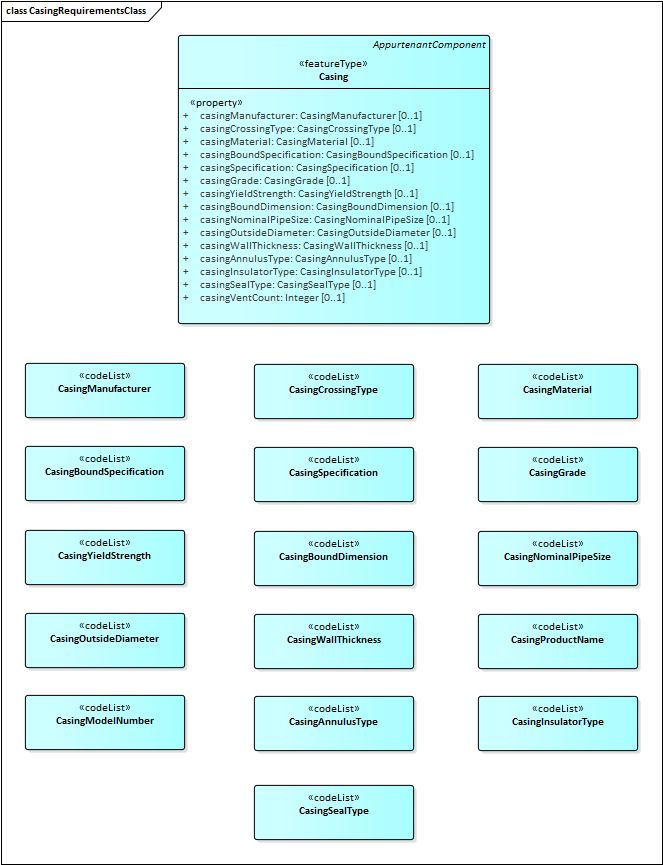


Figure 40. Casing Requirements Class

The Casing Class contains attributes and relationships to other classes:

**casingManufacturer**: An optional attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute casingManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingmanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/casing/casingmanufacturer-codelistURIexception** |

**casingCrossingType**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of crossing

|  |
| --- |
| 1. This optional attribute casingCrossingType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingcrossingtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/casing/casingcrossingtype-codelistURI** |

**casingMaterial**: An optional attribute that, if populated, SHALL be dereferenceable to define a string value containing the type of material used in the manufacture of the component

|  |
| --- |
| 1. This optional attribute casingMaterial SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingmaterial.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/casing/casingmaterial-codelistURI** |

**casingBoundSpecification**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: casingSpecification; casingGrade; casingYieldStrength.

|  |
| --- |
| 1. This optional bound attribute casingBoundSpecification SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/casing/casingboundspecification-boundtype** |

|  |
| --- |
| 1. This optional bound attribute casingBoundSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingboundspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/casing/casingboundspecification-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute casingBoundSpecification MAY be populated, or its corresponding set of unbound attributes (casingSpecification, casingGrade, and casingYieldStrength) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/casing/casingboundspecification-exclusivity** |

**casingSpecification**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value used to define the specification of the component. This attribute SHALL NOT be provided if the attribute casingBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute valveSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/casing/casingspecification-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute casingSpecification MAY be populated, or its corresponding bound attribute casingBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/casing/casingspecification-exclusivity** |

**casingGrade**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value that defines the grade specification of the component. This attribute SHALL NOT be provided if the attribute casingBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute casingSpecification SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingspecification.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/casing/casingspecification-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute casingSpecification MAY be populated, or its corresponding bound attribute casingBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/casing/casingspecification-exclusivity** |

**casingYieldStrength**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value that defines the rated yield strength of the component. This attribute SHALL NOT be provided if the attribute casingBoundSpecification is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute casingYieldStrength SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingyieldstrength.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/casing/casingyieldstrength-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute casingYieldStrength MAY be populated, or its corresponding bound attribute casingBoundSpecification MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/casing/casingyieldstrength-exclusivity** |

**casingBoundDimension**: An optional bound attribute that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: casingNominalPipeSize; casingOutsideDiameter; casingWallThickness.

|  |
| --- |
| 1. This optional bound attribute casingBoundDimension SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/casing/casingbounddimension-boundtype** |

|  |
| --- |
| 1. This optional bound attribute casingBoundDimension SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingbounddimension.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/casing/casingbounddimension-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute casingBoundDimension MAY be populated, or its corresponding set of unbound attributes (casingNominalPipeSize, casingOutsideDiameter, and casingWallThickness) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/casing/casingbounddimension-exclusivity** |

**casingNominalPipeSize**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the nominal pipe size of the component. This attribute SHALL NOT be provided if the attribute casingBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute casingNominalPipeSize SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingnominalpipesize.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/casing/casingnominalpipesize-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute casingNominalPipeSize MAY be populated, or its corresponding bound attribute casingBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/casing/casingnominalpipesize-exclusivity** |

**casingOutsideDiameter**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the outside diameter size of the component. This attribute SHALL NOT be provided if the attribute casingBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute casingOutsideDiameter SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingoutsidediameter.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/casing/casingoutsidediameter-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute casingOutsideDiameter MAY be populated, or its corresponding bound attribute casingBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/casing/casingoutsidediameter-exclusivity** |

**casingWallThickness**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the wall thickness size of the component. This attribute SHALL NOT be provided if the attribute casingBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute casingWallThickness SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingwallthickness.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/casing/casingwallthickness-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute casingWallThickness MAY be populated, or its corresponding bound attribute casingBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/casing/casingwallthickness-exclusivity** |

**casingAnnulusType**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of annulus used in the installation of the component

|  |
| --- |
| 1. This optional attribute casingAnnulusType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingannulustype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/casing/casingannulustype-codelistURI** |

**casingInsulatorType**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of insulator used in the installation of the component

|  |
| --- |
| 1. This optional attribute casingInsulatorType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casinginsulatortype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/casing/casinginsulatortype-codelistURI** |

**casingSealType**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of seal used in the installation of the component

|  |
| --- |
| 1. This optional attribute casingSealType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingsealtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/casing/casingsealtype-codelistURI** |

**casingVentCount**: An optional value that indicates the number of vents used in the installation of the component

|  |
| --- |
| 1. This optional attribute casingVentCount SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/casingventcount.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/casing/casingventcount-codelistURI** |

#### Coating Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/coating** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Coating |
| Dependency | **/req/pmlcomponent** |
| **Req 316** | **/req/coating/coatingmanufacturer-codelistURIexception** |
| **Req 317** | **/req/coating/coatingmanufacturer-exclusivity** |
| **Req 318** | **/req/coating/coatingtype-codelistURI** |
| **Req 319** | **/req/coating/coatingmaterial-codelistURI** |
| **Req 320** | **/req/coating/coatingboundmanufacturer-boundtype** |
| **Req 321** | **/req/coating/coatingboundmanufacturer-codelistURI** |
| **Req 322** | **/req/coating/coatingboundmanufacturer-exclusivity** |
| **Req 323** | **/req/coating/coatingproductname-codelistURIexception** |
| **Req 324** | **/req/coating/coatingproductname-exclusivity** |
| **Req 325** | **/req/coating/coatingmodelnumber-codelistURIexception** |
| **Req 326** | **/req/coating/coatingmodelnumber-exclusivity** |
| **Req 327** | **/req/coating/coatinglayernumber-exclusivity** |
| **Req 328** | **/req/coating/coatingthickness-codelistURI** |
| **Req 329** | **/req/coating/coatingmultilayerdescription-exclusivity** |
| **Req 330** | **/req/coating/coatingapplicationlocation-codelistURI** |

The Coating Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 41.

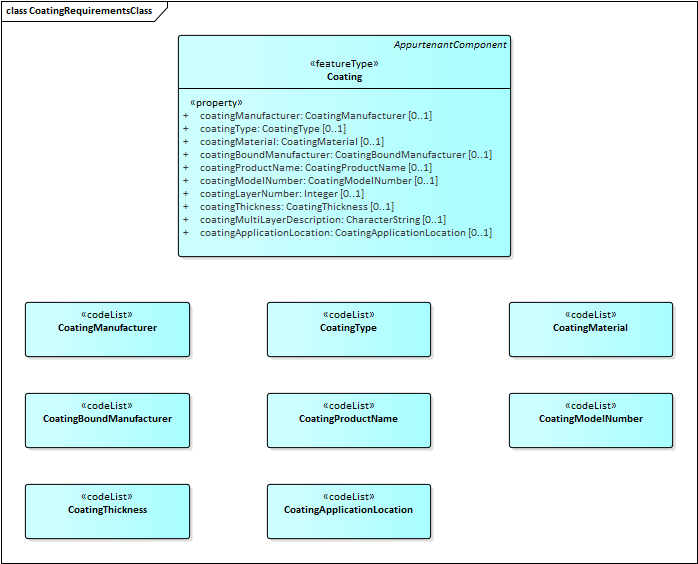


Figure 41. Coating Requirements Class

The Coating Class contains attributes and relationships to other classes:

**coatingManufacturer**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute coatingManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/coatingmanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/coating/coatingmanufacturer-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute coatingManufacturer MAY be populated, or its corresponding bound attribute coatingBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/coating/coatingmanufacturer-exclusivity** |

**coatingType**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of coating

|  |
| --- |
| 1. This optional attribute coatingType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/coatingtype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/coating/coatingtype-codelistURI** |

**coatingMaterial**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of material used in the manufacture of the component

|  |
| --- |
| 1. This optional attribute coatingMaterial SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/coatingmaterial.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/coating/coatingmaterial-codelistURI** |

**coatingBoundManufacturer**: An optional value that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: coatingManufacturer; coatingProductName; coatingModelNumber.

|  |
| --- |
| 1. This optional bound attribute coatingBoundManufacturer SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/coating/coatingboundmanufacturer-boundtype** |

|  |
| --- |
| 1. This optional bound attribute coatingBoundManufacturer SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/coatingboundmanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/coating/coatingboundmanufacturer-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute coatingBoundManufacturer MAY be populated, or its corresponding set of unbound attributes (coatingManufacturer, coatingProductName, and coatingModelNumber) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/coating/coatingboundmanufacturer-exclusivity** |

**coatingProductName**: An optional unbound attribute that, if populated, SHALL be dereferenceable to define a string value containing the product name of the component. This attribute SHALL NOT be provided if the attribute coatingBoundManufacturer is populated for reasons of disambiguation. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute coatingProductName SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/coatingproductname.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/coating/coatingproductname-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute coatingProductName MAY be populated, or its corresponding bound attribute coatingBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/coating/coatingproductname-exclusivity** |

**coatingModelNumber**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the model number of the component. This attribute SHALL NOT be provided if the attribute coatingBoundManufacturer is populated for reasons of disambiguation. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute coatingModelNumber SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/coatingmodelnumber.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/coating/coatingmodelnumber-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute coatingModelNumber MAY be populated, or its corresponding bound attribute coatingBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/coating/coatingmodelnumber-exclusivity** |

**coatingLayerNumber**: An optional integer value containing the sequential number of the coating layer being defined (the value of 1 defines the innermost coating layer and each additional layer increments by 1). This attribute supports the use case of multiple coating attributes for a single component. The practice of defining individual coating layer records is not currently a standardized practice within the pipeline industry; however, this mechanism provides a way forward to defining better granularity and more accuracy in describing the various appurtenant coating layers applied to pipeline components. The attribute coatingMultiLayerDescription supports legacy coating management practices where a concatenated string describes the composite coating layers. This attribute SHALL NOT be provided if the attribute coatingMultiLayerDescription is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional attribute coatingLayerlNumber MAY be populated, or its corresponding attribute coatingMultiLayerDescription MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/coating/coatinglayernumber-exclusivity** |

**coatingThickness**: An optional measurement describing the thickness of the coating. If this attribute is populated, the units-of-measure sub-attribute SHALL also be populated.

|  |
| --- |
| 1. This optional unbound attribute coatingThickness SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/coatingthickness.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/coating/coatingthickness-codelistURI** |

**coatingMultiLayerDescription**: An optional string description defining the various layers of protective appurtenant coating applied to the thickness of the coterminous component. If this attribute is populated, only a single coating record SHALL be defined for a given pipeline component and the coatingLayerNumber attribute SHALL NOT be populated.

|  |
| --- |
| 1. This optional attribute coatingMultiLayerDescription MAY be populated, or its corresponding attribute coatingLayerNumber MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/coating/coatingmultilayerdescription-exclusivity** |

**coatingApplicationLocation**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the location where the coating was applied to the component (i.e. manufacturer facility, field, etc.).

|  |
| --- |
| 1. This optional unbound attribute coatingApplicationLocationSHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/coatingapplicationlocation.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/coating/coatingapplicationlocation-codelistURI** |

#### Sleeve Class Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/sleeve** | |
| Target type | Encoding of conceptual model |
| Name | PMLComponent Sleeve |
| Dependency | **/req/pmlcomponent** |
| **Req 331** | **/req/sleeve/sleevemanufacturer-codelistURIexception** |
| **Req 332** | **/req/sleeve/sleevemanufacturer-exclusivity** |
| **Req 333** | **/req/sleeve/sleevetype-codelistURI** |
| **Req 334** | **/req/sleeve/sleevematerial-codelistURI** |
| **Req 335** | **/req/sleeve/sleevebounddimension-boundtype** |
| **Req 336** | **/req/sleeve/sleevebounddimension-codelistURI** |
| **Req 337** | **/req/sleeve/sleevebounddimension-exclusivity** |
| **Req 338** | **/req/sleeve/sleevenominalpipesize-codelistURI** |
| **Req 339** | **/req/sleeve/sleevenominalpipesize-exclusivity** |
| **Req 340** | **/req/sleeve/sleevewallthickness-codelistURI** |
| **Req 341** | **/req/sleeve/sleevewallthickness-exclusivity** |
| **Req 342** | **/req/sleeve/sleeveboundmanufacturer-boundtype** |
| **Req 343** | **/req/sleeve/sleeveboundmanufacturer-codelistURI** |
| **Req 344** | **/req/sleeve/sleeveboundmanufacturer-exclusivity** |
| **Req 345** | **/req/sleeve/sleeveproductname-codelistURIexception** |
| **Req 346** | **/req/sleeve/sleeveproductname-exclusivity** |
| **Req 347** | **/req/sleeve/sleevemodelnumber-codelistURIexception** |
| **Req 348** | **/req/sleeve/sleevemodelnumber-exclusivity** |
| **Req 349** | **/req/sleeve/sleevepressurerating-uom** |

The Sleeve Class is a concrete class and therefore may be encoded into a PipelineML dataset. This class contains classes and attributes as shown in Figure 42.

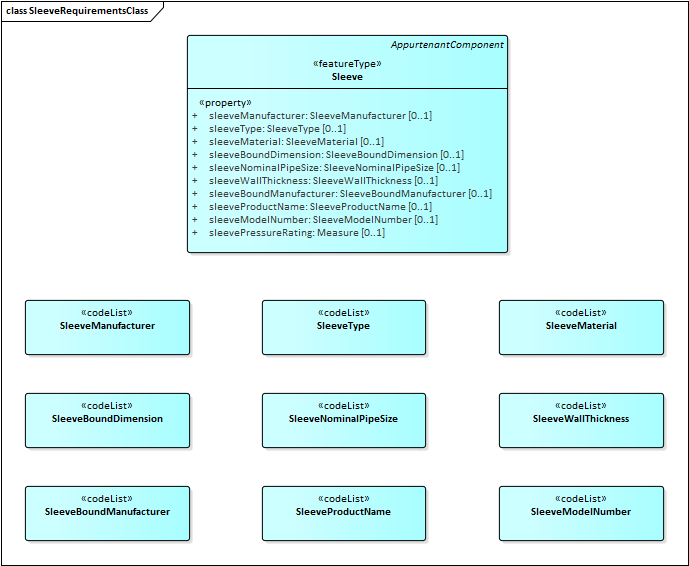


Figure 42. Sleeve Requirements Class

The Sleeve Class contains attributes and relationships to other classes:

**sleeveManufacturer**: An optional unbound attribute that, if populated, SHALL be dereferenceable to a string value containing the name of the company that manufactured the component. The use of an external reference code list (controlled vocabulary) is intended to provide authoritative data that enhances application interoperability. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute sleeveManufacturer SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/sleevemanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/sleeve/sleevemanufacturer-codelistURIexception** |

|  |
| --- |
| 1. This optional unbound attribute sleeveManufacturer MAY be populated, or its corresponding bound attribute sleeveBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/sleeve/sleevemanufacturer-exclusivity** |

**sleeveType**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of sleeve

|  |
| --- |
| 1. This optional attribute sleeveType SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/sleevetype.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/sleeve/sleevetype-codelistURI** |

**sleeveMaterial**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the type of material used in the manufacture of the component

|  |
| --- |
| 1. This optional attribute sleeveMaterial SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/sleevematerial.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/sleeve/sleevematerial-codelistURI** |

**sleeveBoundDimension**: An optional value that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: sleeveNominalPipeSize; casingWallThickness.

|  |
| --- |
| 1. This optional bound attribute sleeveBoundDimension SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/sleeve/sleevebounddimension-boundtype** |

|  |
| --- |
| 1. This optional bound attribute sleeveBoundDimension SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/sleevebounddimension.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/sleeve/sleevebounddimension-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute sleeveBoundDimension MAY be populated, or its corresponding set of unbound attributes (sleeveNominalPipeSize, and sleeveWallThickness) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/sleeve/sleevebounddimension-exclusivity** |

**sleeveNominalPipeSize**: An optional value that, if populated, SHALL be dereferenceable to a string value containing the nominal pipe size of the component. This attribute SHALL NOT be provided if the attribute sleeveBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute sleeveNominalPipeSize SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/sleevenominalpipesize.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/sleeve/sleevenominalpipesize-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute sleeveNominalPipeSize MAY be populated, or its corresponding bound attribute sleeveBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/sleeve/sleevenominalpipesize-exclusivity** |

**sleeveWallThickness**: An optional value that, if populated, SHALL be dereferenceable to a string value containing the wall thickness size of the component. This attribute SHALL NOT be provided if the attribute sleeveBoundDimension is populated for reasons of disambiguation.

|  |
| --- |
| 1. This optional unbound attribute sleeveWallThickness SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/sleevewallthickness.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHALL be dereferencable to a string attribute.   **/req/sleeve/sleevewallthickness-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute sleeveWallThickness MAY be populated, or its corresponding bound attribute sleeveBoundDimension MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/sleeve/sleevewallthickness-exclusivity** |

**sleeveBoundManufacturer**: An optional value that, if populated, SHALL be dereferenceable to define a bound combination of attributes as a concatenated string with each attribute being separated by semicolons (;). An alternate method is provided (below) for defining each of these attributes in unbound form. Only the bound type or unbound types may be populated, but NOT BOTH. This bound form of attributes represents the highest pedigree of data standardization and is considered best practice, as it provides a mechanism for identifying valid combinations of attributes. This bound set of values includes the following attributes: sleeveManufacturer; sleeveProductName; sleeveModelNumber.

|  |
| --- |
| 1. This optional bound attribute sleeveBoundManufacturer SHALL utilize the PipelineML boundType construct. This construct support a combination of attributes that are tightly bound together for the purpose of defining valid combinations of attributes using a single attribute value. A bound data type SHALL be dereferencable to a concatenated string of attributes separated by semicolons.   **/req/sleeve/sleeveboundmanufacturer-boundtype** |

|  |
| --- |
| 1. This optional bound attribute sleeveBoundManufacturer SHALL utilize the codelistURI construct. Any value populated for this attribute SHALL utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/sleeveboundmanufacturer.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value shall be dereferencable to a set of strongly bound string attributes separated by semicolons.   **/req/sleeve/sleeveboundmanufacturer-codelistURI** |

|  |
| --- |
| 1. This optional bound attribute sleeveBoundManufacturer MAY be populated, or its corresponding set of unbound attributes (sleeveManufacturersleeveProductName, and sleeveModelNumber) MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/sleeve/sleeveboundmanufacturer-exclusivity** |

**sleeveProductName**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the product name of the component. This attribute SHALL NOT be provided if the attribute sleeveBoundManufacturer is populated for reasons of disambiguation. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute sleeveProductName SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/sleeveproductname.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/sleeve/sleeveproductname-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute sleeveProductName MAY be populated, or its corresponding bound attribute sleeveBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/sleeve/sleeveproductname-exclusivity** |

**sleeveModelNumber**: An optional value that, if populated, SHALL be dereferenceable to define a string value containing the model number of the component. This attribute SHALL NOT be provided if the attribute sleeveBoundManufacturer is populated for reasons of disambiguation. This attribute supports the use of substandard (non-authoritative) data as defined in the othervalue requirement.

|  |
| --- |
| 1. **[Recommendation]** This optional attribute sleeveModelNumber SHOULD utilize the codelistURIexception construct. Any value populated for this attribute SHOULD utlize the external reference code list (controlled vocabulary that represents authoritative data) located at <http://registry.pipelineml.org/codelist/sleevemodelnumber.xml> [This URI is subject to change following OGC Naming Authority allocation]. This value SHOULD be dereferencable to a string attribute.   This attribute supports the use of substandard data by using an alternate source to the PipelineML Authoritative Data Governance Body for the unique code value (machine readable value) and the title (human readable value). If a value needs to be represented that is not contained in the PipelineML authoritative code list, an alternative code value MAY be used as a temporary solution, so long as it complies with the following guidelines:   1. A non-authoritative code value may not be used if an existing authoritative code exists that means the same thing with a variance in wording/phrasing, spelling, word sequence, capitalization, parentheses, or abbreviation. If the meaning of a value closely matches that of an authoritative value, the authoritative value should be used and not an alternate. For example, the value “A.O. Smith” exists as a LinepipeManufacturer value. If someone preferred the value “AO Smith”, this would be considered a spelling variation of an existing code value, and the existing authoritative value should be used. 2. A non-authoritative code identifier must not match or conflict with an existing authoritative code identifier. For example, the code identifier 3390 is allocated to the meaning “Axial; Dual Stage; 15 Throw; 2 Stroke” in the CompressorBoundType code list. Hence, the code identifier 3390 may not be used to identify a substandard/non-authoritative value. The utilization of a UUID or a database GUID that begins with a non-numeric value (such as “id”, id2673c08c-642f-e711-80ea-0050569E3087) provides a safe mechanism for encoding a unique code identifier. 3. A non-authoritative code value must be unique in the dataset (a given code identifier can only have a single meaning within a dataset). That is, if a non-authoritative code identifier is used more than once a dataset, its meaning must remain consistent throughout the dataset. 4. The utilization of this sub-standard construct SHOULD be promptly followed by the submission of the missing value/s to the PipelineML Authoritative Data Governance Body for consideration of inclusion in the authoritative code domain. 5. If the PipelineML Authoritative Data Governance Body determines that a proposed value is too similar to an existing code value in meaning, all future encodings should utilize the matching existing authoritative code value instead of continuing to encode the proposed alternate.   **/req/sleeve/sleevemodelnumber-codelistURI** |

|  |
| --- |
| 1. This optional unbound attribute sleeveModelNumber MAY be populated, or its corresponding bound attribute sleeveBoundManufacturer MAY be populated, but NOT BOTH. This exclusivity requirement is intended to promote disambiquity.   **/req/sleeve/sleevemodelnumber-exclusivity** |

**sleevePressureRating**: An optional value used to define the pressure rating of the pipeline component as determined by the component manufacturer. If this attribute is populated, the units-of-measure sub-attribute SHALL be populated.

|  |
| --- |
| 1. If this optional attribute sleevePressureRating is provided, a units-of-measure sub-attribute (uom) SHALL be provided that indicates the appropriate units of measure used for the provided value.   **/req/sleeve/sleevepressurerating-uom** |

# XML Encoding Requirement Class (Normative)

XSD schemas were derived from the UML model following GML 3.3 encoding (OGC ISO19136-2, OGC 10-129r1) that extends and supersedes some of ISO 19136-2007. The normative artefacts for XML encoding are the W3C XSD documents provided online with this specification. Those documents explicitly provide the requirements that must be met by any XML instance claiming compliance to this specification. Any requirements that cannot be expressed in XSD are described in the relevant XML encoding section of this document. Therefore, compliant XML instances shall:

* validate with XSD schemas and then
* pass compliance tests listed in relevant compliance sections.

## PipelineML Core XML Abstract Requirements Class (Normative)

|  |  |
| --- | --- |
| **Requirements Class** | |
| **/req/pipelineml-xsd** | |
| Target type | Data instance |
| Name | PipelineML Core XML Abstract Requirements Class |
| Dependency | **/req/pipelineml-abstract** |
| Dependency | **W3C XML Schema Definition Language 1.1** |
| Dependency | **ISO19118 Encoding** |
| Dependency | **ISO 19136:2007 Geography Markup Language (GML)** |
| Dependency | **ISO19136-2:2015 Geography Markup Language (GML) Part 2: Extended schemas and encoding rules** |
| **Req 350** | **/req/pipelineml-xsd/xsd** |
| **Req 351** | **/req/pipelineml-xsd/codelist** |
| **Req 352** | **/req/pipelineml-xsd/identifier-uri** |
| **Req 353** | **/req/pipelineml-xsd/iso8601-time** |
| **Req 354** | **/req/pipelineml-xsd/time-zone** |

This requirements class is shared by all GML/XML PipelineML instances.

### XML Document Validation

An XML instance shall validate to the XSD rules provided by this specification for each of the XML requirements classes.

|  |
| --- |
| 1. An XML instance SHALL validate against XSD schema.   **/req/pipelineml-xsd/xsd** |

### Codelist

Open code lists are encoded as gml:ReferenceType which is a sequence of gml:OwnershipAttributeGroup and gml:AssociationAttributeGroup, providing a series of xml attributes from W3C XLINK (http://www.w3.org/TR/xlink11/). A vocabulary term reference has mandatory xlink:href and xlink:title attributes.

<pml:compressorBoundType xlink:href="http://registry.pipelineml.org/codelists/compressorBoundType.xml#id3555" xlink:title="Centrifugal; Dual Stage; 5 Throw; 1 Stroke"/>

<pml:elbowType xlink:href="http://registry.pipelineml.org/codelists/elbowType.xml#id5683" xlink:title="Long Radius"/>

<pml:meterNominalPipeSize xlink:href="http://registry.pipelineml.org/codelists/meterNominalPipeSize.xml#id6470" xlink:title="14.000"/>

The xlink:href contains an absolute HTTP URI that should resolve to a resource representation. The resource can have multiple representations and it is not guaranteed that an XML parsable document can be obtained from the vocabulary service.

|  |
| --- |
| 1. Vocabulary term SHALL be encoded with HTTP URI in @xlink:href, provide a human readable description in @xlink:title, and where applicable @xlink:uom (when units-of-measure are a designated requirement).   **/req/pipelineml-xsd/codelist** |

### Identifiers

The PipelineML community has developed a best practice of using http://www.ietf.org/rfc/rfc2616 as a codespace for string with authority to designate string that are resolvable HTTP URI. gml:identifier’s flagged with the specific codeSpace “http://www.ietf.org/rfc/rfc2616” should be resolvable HTTP URI that return an instance of itself.

|  |
| --- |
| 1. Feature identifiers (unique name) provided in gml:identifier and @codeSpace = “http://www.ietf.org/rfc/rfc2616” SHOULD be URI of resource using Linked Open Data principles.   **/req/pipelineml-xsd/identifier-uri** |

### Date Encoding

Date-time values shall conform to ISO 8601 standards. Although this is already a GML 3.2 encoding rule (clause 14.2.2.7), this format should also be used in any string that should contain a date, or date and time.

|  |
| --- |
| 1. All date-time elements occurrences SHOULD be encoded using ISO8601 extended time format.   **/req/pipelineml-xsd/iso8601-time** |

Note that this precludes the use of time-coordinate systems such as UNIX time. This is specified in order to be maximally consistent with TimeSeriesML requirements. The time zone shall be included in the time element.

|  |
| --- |
| 1. The value of each time element SHALL include a time zone definition using a signed 4 digit character or a ‘Z’ to represent Zulu or Greenwich Mean Time (GMT). This is defined by the following regular expression: (Z|[+-]HH:MM).   **/req/pipelineml-xsd/time-zone** |

<pml:manufactureDate>2007-11-12T04:14:09-05:00</pml:manufactureDate>

<pml:installDate>2012-11-12T09:20:41.6875000-05:00</pml:installDate>

# Media Types for any data encoding(s)

PipelineML data conforming to clause 8 is encoded in GML-conformant XML documents. The standard MIME-type and sub-type for GML data should be used to indicate the encoding in internet exchange, as specified in MIME Media Types for GML, namely.

application/gml+xml

## Abbreviations

In this document, the following abbreviations and acronyms are used or introduced:

* API – American Petroleum Institute
* ERA – Entity, Relation, Attribute (pre-object modeling technique)
* GML – Geography Markup Language
* ISO – International Organization for Standardization (from Greek for “same”)
* OGC – Open Geospatial Consortium (<http://www.opengeospatial.org/>)
* UML – Unified Modeling Language

# Annex A: Conformance Classes (Normative)

PipelineML Conceptual Conformance Class

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/pipelineml-conceptual** | | |
| **Requirements** | **/req/pipelineml-conceptual** | |
| **Test** | **/conf/pipelineml-conceptual/similarity** | |
| **Requirement** | **/req/pipelineml-conceptual/similarity (Req 1)** |
| **Test Purpose** | Ensure that the target encoding is compatible with the conceptual model. |
| **Test Method** | Determine semantic similarity between the encoding model and conceptual model using an established method such as: (i) visual comparison of the UML diagrams, (ii) comparison of encoding and conceptual components expressed in a common knowledge representation language such as first order logic, or (iii) comparison after mapping to a reference ontology. |
| **Test Type** | Capability |

PipelineML Abstract Conformance Class

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/pipelineml-abstract** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **Urn:iso:dis:iso:19156:clause:A.1.1** | |
| **Dependency** | **RFC 3986** | |
| **Test** | **/conf/pipelineml-abstract/uml-entity-name** | |
| **Requirement** | **/req/pipelineml-abstract/uml-entity-name (Req 2)** |
| **Test Purpose** | Ensure that the names used in the target implementation matches the names in the encoding model |
| **Test Method** | Visually inspect that, when possible, name used in the target encoding matches the names in the conceptual model. |
| **Test Type** |  |
| **Test** | **/conf/pipelineml-abstract/uml-cardinality** | |
| **Requirement** | **/req/pipelineml-abstract/uml-cardinality (Req 3)** |
| **Test Purpose** | Ensure that the cardinalities of the properties are the same as the conceptual model |
| **Test Method** | Visually inspect the target encoding and validate that all properties have the same cardinality as the one expressed in the conceptual model. |
| **Test Type** |  |
| **Test** | **/conf/pipelineml-abstract/uml-abstract** | |
| **Requirement** | **/req/pipelineml-abstract/uml-abstract (Req 4)** |
| **Test Purpose** | Ensure that encoding does not allow materialization of abstract classes |
| **Test Method** | Visually inspect the target implementation and validate that no classifiers (Feature, Type or Datatype) marked as abstract can be materialised. |
| **Test Type** | Capability |
| **Test** | **/conf/pipelineml-abstract/uml-polymorphism** | |
| **Requirement** | **/req/pipelineml-abstract/uml-polymorphism (Req 5)** |
| **Test Purpose** | Ensure that target implementation has mechanisms to allow type substitutions for property values. |
| **Test Method** | Visually inspect that the target implementation accepts all substitutable types for property values |
| **Test Type** | Capability |
| **Test** | **/conf/pipelineml-abstract/quantities-uom** | |
| **Requirement** | **/req/pipelineml-abstract/quantities-uom (Req 6)** |
| **Test Purpose** | Ensure that quantities have a governed unit of measure |
| **Test Method** | Visually inspect that the target implementation has a mechanism to enforce a unit of measure from a governed source |
| **Test Type** | Capability |
| **Test** | **/conf/pipelineml-abstract/codelist** | |
| **Requirement** | **/req/pipelineml-abstract/codelist (Req 7)** |
| **Test Purpose** | Ensure that vocabulary terms are encode as URI according to RFC3986 |
| **Test Method** | Visually inspect that the target implementation has a mechanism to enforce a unit of measure from a governed source |
| **Test Type** | Capability |

PipelineML Linked Open Data Conformance Class

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/pipelineml-lod** | | |
| **Requirements** | **/req/pipelineml-lod** | |
| **Dependency** | **RFC 3986** | |
| **Test** | **/conf/pipelineml-lod/codelistURI** | |
| **Requirement** | **/req/pipelineml-lod/codelistURI (Req 8)** |
| **Test Purpose** | Ensure that the URI matches the officially governed external reference code list defined in this PipelineML Standard definition document |
| **Test Method** | Visually inspect that, when possible, the name used in the target encoding matches the names in the conceptual model |
| **Test Type** | Capability |
| **Test** | **/conf/pipelineml-lod/codelistURIexception** | |
| **Requirement** | **/req/pipelineml-lod/codelistURIexception [Recommendation 9]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/pipelineml-lod/identifier** | |
| **Requirement** | **/req/pipelineml-lod/identifier (Req 10)** |
| **Test Purpose** | Verify that the encoding utilizes the URI identifier as specified |
| **Test Method** | Visually inspect the target implementation and check that it provides a mechanism to verify that the identifier can be dereferenced and return a representation of the resource assigned to the identifier |
| **Test Type** | Capability |
| **Test** | **/conf/pipelineml-lod/byref** | |
| **Requirement** | **/req/pipelineml-lod/byref (Req 11)** |
| **Test Purpose** | Verify that the encoding properties that reference an external resource using a HTTP URI identifier is dereferenceable. |
| **Test Method** | Visually inspect the target implementation and check that it provides a mechanism to verify that URI are dereferenceable |
| **Test Type** | Capability |

PipelineML Encoding Conformance Class

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/pipelineml-encoding** | | |
| **Requirements** | **/req/pipelineml-conceptual** | |
| **Test** | **/conf/pipelineml-encoding/similarity** | |
| **Requirement** | **/req/pipelineml-encoding/similarity (Req 12)** |
| **Test Purpose** | Verify that the encoding is compatible with the conceptual model. |
| **Test Method** | Determine semantic similarity between the encoding and conceptual model using an established method such as: (i) visual comparison of the UML diagrams, (ii) comparison of encoding and conceptual components expressed in a common knowledge representation language such as first order logic, or (iii) comparison after mapping to a reference ontology. |
| **Test Type** | Capability |

PMLComponent Class Conformance Class

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/pmlcomponent** | | |
| **Requirements** | **/req/pmlcomponent** | |
| **Dependency** | **/conf/pipelineml-conceptual** | |
| **Dependency** | **/conf/pipelineml-encoding** | |
| **Test** | **/conf/pmlcomponent/dataset** | |
| **Requirement** | **/req/pmlcomponent/dataset (Req 13)** |
| **Test Purpose** | Verify that the encoding specifies a PMLDataset in a format appropriate for that encoding. |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pmlcomponent/classes** | |
| **Requirement** | **/req/pmlcomponent/classes (Req 14)** |
| **Test Purpose** | Verify that the encoding provides the PMLComponent Requirements Class Classes shown in blue in Figure 3 in a manner consistent with the encoding |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pmlcomponent/crs** | |
| **Requirement** | **/req/pmlcomponent/crs (Req 15)** |
| **Test Purpose** | Verify that the encoding requirement for coordinate reference systems is consistent with OGC Abstract Specification Topic 2 |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pmlcomponent/19103** | |
| **Requirement** | **/req/pmlcomponent/19103 (Req 16)** |
| **Test Purpose** | Verify that the encoding provides the core data types specified in ISO 19103 Clause 7 that are appropriate to the supported subject area(s) |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pmlcomponent/topic-1** | |
| **Requirement** | **/req/pmlcomponent/topic-1 (Req 17)** |
| **Test Purpose** | Verify that the encoding provides the geometry types specified in the OGC Abstract Specification Topic 1, Feature Geometry are appropriate to the supported subject area(s). Verify that the encoding provides the additional geometry types not found in Topic 1, but required by a specific supported requirements class, are specified in that requirements class. |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pmlcomponent/19109** | |
| **Requirement** | **/req/pmlcomponent/19109 (Req 18)** |
| **Test Purpose** | Verify that the encoding is in accordance with the uml:feature requirement in ISO 19109:2015 Clause 8.2.6 and shown in Figure 6, as follows: each instance of FeatureType shall be implemented by the encoding’s equivalent of a UML Class having a generalization association with AnyFeature and with a stereotype of <<FeatureType>> |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pmlcomponent/boundtypes** | |
| **Requirement** | **/req/pmlcomponent/boundtypes (Req 19)** |
| **Test Purpose** | Ensure that encoding implements boundtypes as specified in this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Pipeline Class Conformance Class

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/pipeline** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/pipeline/pipelineboundtype-boundtype** | |
| **Requirement** | **/req/pipeline/pipelineboundtype-boundtype (Req 20)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeline/pipelineboundtype-codelistURI** | |
| **Requirement** | **/req/pipeline/pipelineboundtype-codelistURI (Req 21)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeline/pipelineboundtype-exclusivity** | |
| **Requirement** | **/req/pipeline/pipelineboundtype-exclusivity (Req 22)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeline/pipelineclassification-codelistURI** | |
| **Requirement** | **/req/pipeline/pipelineclassification-codelistURI (Req 23)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeline/pipelineclassification-exclusivity** | |
| **Requirement** | **/req/pipeline/pipelineclassification-exclusivity (Req 24)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeline/pipelinetype-codelistURI** | |
| **Requirement** | **/req/pipeline/pipelinetype-codelistURI (Req 25)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeline/pipelinetype-exclusivity** | |
| **Requirement** | **/req/pipeline/pipelinetype-exclusivity (Req 26)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeline/pipelinesubtype-codelistURI** | |
| **Requirement** | **/req/pipeline/pipelinesubtype-codelistURI (Req 27)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeline/pipelinesubtype-exclusivity** | |
| **Requirement** | **/req/pipeline/pipelinesubtype-exclusivity (Req 28)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Assembly Class Conformance Class

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/assembly** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/assembly/componentexclusivity** | |
| **Requirement** | **/req/assembly/componentexclusivity (Req 29)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/assembly/assemblyboundtype-boundtype** | |
| **Requirement** | **/req/assembly/assemblyboundtype-boundtype (Req 30)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/assembly/assemblyboundtype-codelistURI** | |
| **Requirement** | **/req/assembly/assemblyboundtype-codelistURI (Req 31)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/assembly/assemblyboundtype-exclusivity** | |
| **Requirement** | **/req/assembly/assemblyboundtype-exclusivity (Req 32)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/assembly/assemblyclassification-codelistURI** | |
| **Requirement** | **/req/assembly/assemblyclassification-codelistURI (Req 33)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/assembly/assemblyclassification-exclusivity** | |
| **Requirement** | **/req/assembly/assemblyclassification-exclusivity (Req 34)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/assembly/assemblytype-codelistURI** | |
| **Requirement** | **/req/assembly/assemblytype-codelistURI (Req 35)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/assembly/assemblytype-exclusivity** | |
| **Requirement** | **/req/assembly/assemblytype-exclusivity (Req 36)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/assembly/assemblysubtype-codelistURI** | |
| **Requirement** | **/req/assembly/assemblysubtype-codelistURI (Req 37)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/assembly/assemblysubtype-exclusivity** | |
| **Requirement** | **/req/assembly/assemblysubtype-exclusivity (Req 38)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Collection Class Conformance Class

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/collection** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/collection/collectionboundtype-boundtype** | |
| **Requirement** | **/req/collection/collectionboundtype-boundtype (Req 39)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/collection/collectionboundtype-codelistURI** | |
| **Requirement** | **/req/collection/collectionboundtype-codelistURI (Req 40)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/collection/collectionboundtype-exclusivity** | |
| **Requirement** | **/req/collection/collectionboundtype-exclusivity (Req 41)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/collection/collectionclassification-codelistURI** | |
| **Requirement** | **/req/collection/collectionclassification-codelistURI (Req 42)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/collection/collectionclassification-exclusivity** | |
| **Requirement** | **/req/collection/collectionclassification-exclusivity (Req 43)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/collection/collectiontype-codelistURI** | |
| **Requirement** | **/req/collection/collectiontype-codelistURI (Req 44)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/collection/collectiontype-exclusivity** | |
| **Requirement** | **/req/collection/collectiontype-exclusivity (Req 45)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/collection/collectionsubtype-codelistURI** | |
| **Requirement** | **/req/collection/collectionsubtype-codelistURI (Req 46)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/collection/collectionsubtype-exclusivity** | |
| **Requirement** | **/req/collection/collectionsubtype-exclusivity (Req 47)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Component Class Conformance Class

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/pipeconnector** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/pipeconnector/pipeconnectormanufacturer-codelistURIexception** | |
| **Requirement** | **/req/pipeconnector/pipeconnectormanufacturer-codelistURIexception [Recommendation 48]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/pipeconnector/pipeconnectortype-codelistURI** | |
| **Requirement** | **/req/pipeconnector/pipeconnectortype-codelistURI (Req 49)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeconnector/pipeconnectortype-exclusivity** | |
| **Requirement** | **/req/pipeconnector/pipeconnectortype-exclusivity (Req 50)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeconnector/pipeconnectormaterial-codelistURI** | |
| **Requirement** | **/req/pipeconnector/pipeconnectormaterial-codelistURI (Req 51)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeconnector/pipeconnectorboundtype-boundtype** | |
| **Requirement** | **/req/pipeconnector/pipeconnectorboundtype-boundtype (Req 52)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeconnector/pipeconnectorboundtype-codelistURI** | |
| **Requirement** | **/req/pipeconnector/pipeconnectorboundtype-codelistURI (Req 53)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeconnector/pipeconnectorboundtype-exclusivity** | |
| **Requirement** | **/req/pipeconnector/pipeconnectorboundtype-exclusivity (Req 54)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeconnector/pipeconnectorsubtype-codelistURI** | |
| **Requirement** | **/req/pipeconnector/pipeconnectorsubtype-codelistURI (Req 55)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipeconnector/pipeconnectorsubtype-exclusivity** | |
| **Requirement** | **/req/pipeconnector/pipeconnectorsubtype-exclusivity (Req 56)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Cap Class Conformance Class

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/cap** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/cap/capmanufacturer-codelistURIexception** | |
| **Requirement** | **/req/cap/capmanufacturer-codelistURIexception [Recommendation 57]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/cap/captype-codelistURI** | |
| **Requirement** | **/req/cap/captype-codelistURI (Req 58)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/captype-exclusivity** | |
| **Requirement** | **/req/cap/captype-exclusivity (Req 59)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capmaterial-codelistURI** | |
| **Requirement** | **/req/cap/capmaterial-codelistURI (Req 60)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capboundspecification-boundtype** | |
| **Requirement** | **/req/cap/capboundspecification-boundtype (Req 61)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capboundspecification-codelistURI** | |
| **Requirement** | **/req/cap/capboundspecification-codelistURI (Req 62)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capboundspecification-exclusivity** | |
| **Requirement** | **/req/cap/capboundspecification-exclusivity (Req 63)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capspecification-codelistURI** | |
| **Requirement** | **/req/cap/capspecification-codelistURI (Req 64)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capspecification-exclusivity** | |
| **Requirement** | **/req/cap/capspecification-exclusivity (Req 65)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capweight-codelistURI** | |
| **Requirement** | **/req/cap/capweight-codelistURI (Req 66)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capweight-exclusivity** | |
| **Requirement** | **/req/cap/capweight-exclusivity (Req 67)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capbounddimension-boundtype** | |
| **Requirement** | **/req/cap/capbounddimension-boundtype (Req 68)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capbounddimension-codelistURI** | |
| **Requirement** | **/req/cap/capbounddimension-codelistURI (Req 69)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capbounddimension-exclusivity** | |
| **Requirement** | **/req/cap/capbounddimension-exclusivity (Req 70)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capnominalpipesize-codelistURI** | |
| **Requirement** | **/req/cap/capnominalpipesize-codelistURI (Req 71)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capnominalpipesize-exclusivity** | |
| **Requirement** | **/req/cap/capnominalpipesize-exclusivity (Req 72)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capoutsidediameter-codelistURI** | |
| **Requirement** | **/req/cap/capoutsidediameter-codelistURI (Req 73)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capoutsidediameter-exclusivity** | |
| **Requirement** | **/req/cap/capoutsidediameter-exclusivity (Req 74)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capwallthickness-codelistURI** | |
| **Requirement** | **/req/cap/capwallthickness-codelistURI (Req 75)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/cap/capwallthickness-exclusivity** | |
| **Requirement** | **/req/cap/capwallthickness-exclusivity (Req 76)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Compressor Class Conformance Class

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/compressor** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/compressor/compressormanufacturer-codelistURIexception** | |
| **Requirement** | **/req/compressor/compressormanufacturer-codelistURIexception [Recommendation 77]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressormanufacturer-exclusivity** | |
| **Requirement** | **/req/compressor/compressormanufacturer-exclusivity (Req 78)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressortype-codelistURI** | |
| **Requirement** | **/req/compressor/compressortype-codelistURI (Req 79)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressortype-exclusivity** | |
| **Requirement** | **/req/compressor/compressortype-exclusivity (Req 80)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorboundtype-boundtype** | |
| **Requirement** | **/req/compressor/compressorboundtype-boundtype (Req 81)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorboundtype-codelistURI** | |
| **Requirement** | **/req/compressor/compressorboundtype-codelistURI (Req 82)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorboundtype-exclusivity** | |
| **Requirement** | **/req/compressor/compressorboundtype-exclusivity (Req 83)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorstage-codelistURI** | |
| **Requirement** | **/req/compressor/compressorstage-codelistURI (Req 84)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorstage-exclusivity** | |
| **Requirement** | **/req/compressor/compressorstage-exclusivity (Req 85)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorthrow-codelistURI** | |
| **Requirement** | **/req/compressor/compressorthrow-codelistURI (Req 86)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorthrow-exclusivity** | |
| **Requirement** | **/req/compressor/compressorthrow-exclusivity (Req 87)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorstroke-codelistURI** | |
| **Requirement** | **/req/compressor/compressorstroke-codelistURI (Req 88)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorstroke-exclusivity** | |
| **Requirement** | **/req/compressor/compressorstroke-exclusivity (Req 89)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressornominalpipesize-codelistURI** | |
| **Requirement** | **/req/compressor/compressornominalpipesize-codelistURI (Req 90)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorboundmanufacturer-boundtype** | |
| **Requirement** | **/req/compressor/compressorboundmanufacturer-boundtype (Req 91)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorboundmanufacturer-codelistURI** | |
| **Requirement** | **/req/compressor/compressorboundmanufacturer-codelistURI (Req 92)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorboundmanufacturer-exclusivity** | |
| **Requirement** | **/req/compressor/compressorboundmanufacturer-exclusivity (Req 93)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorproductname-codelistURIexception** | |
| **Requirement** | **/req/compressor/compressorproductname-codelistURIexception [Recommendation 94]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorproductname-exclusivity** | |
| **Requirement** | **/req/compressor/compressorproductname-exclusivity (Req 95)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressormodelnumber-codelistURIexception** | |
| **Requirement** | **/req/compressor/compressormodelnumber-codelistURIexception [Recommentation 96]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressormodelnumber-exclusivity** | |
| **Requirement** | **/req/compressor/compressormodelnumber-exclusivity (Req 97)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorfueltype-codelistURI** | |
| **Requirement** | **/req/compressor/compressorfueltype-codelistURI (Req 98)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorpowerrating-uom** | |
| **Requirement** | **/req/compressor/compressorpowerrating-uom (Req 99)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorratedflow-uom** | |
| **Requirement** | **/req/compressor/compressorratedflow-uom (Req 100)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorpressuresuction-uom** | |
| **Requirement** | **/req/compressor/compressorpressuresuction-uom (Req 101)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/compressor/compressorpressuredischarge-uom** | |
| **Requirement** | **/req/compressor/compressorpressuredischarge-uom (Req 102)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Elbow Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/elbow** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/elbow/elbowmanufacturer-codelistURIexception** | |
| **Requirement** | **/req/elbow/elbowmanufacturer-codelistURIexception [Recommendation 103]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowtype-codelistURI** | |
| **Requirement** | **/req/elbow/elbowtype-codelistURI (Req 104)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowmaterial-codelistURI** | |
| **Requirement** | **/req/elbow/elbowmaterial-codelistURI (Req 105)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowboundspecification-boundtype** | |
| **Requirement** | **/req/elbow/elbowboundspecification-boundtype (Req 106)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowboundspecification-codelistURI** | |
| **Requirement** | **/req/elbow/elbowboundspecification-codelistURI (Req 107)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowboundspecification-exclusivity** | |
| **Requirement** | **/req/elbow/elbowboundspecification-exclusivity (Req 108)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowspecification-codelistURI** | |
| **Requirement** | **/req/elbow/elbowspecification-codelistURI (Req 109)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowspecification-exclusivity** | |
| **Requirement** | **/req/elbow/elbowspecification-exclusivity (Req 110)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowweight-codelistURI** | |
| **Requirement** | **/req/elbow/elbowweight-codelistURI (Req 111)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowweight-exclusivity** | |
| **Requirement** | **/req/elbow/elbowweight-exclusivity (Req 112)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowbounddimension-boundtype** | |
| **Requirement** | **/req/elbow/elbowbounddimension-boundtype (Req 113)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowbounddimension-codelistURI** | |
| **Requirement** | **/req/elbow/elbowbounddimension-codelistURI (Req 114)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowbounddimension-exclusivity** | |
| **Requirement** | **/req/elbow/elbowbounddimension-exclusivity (Req 115)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowangle-codelistURI** | |
| **Requirement** | **/req/elbow/elbowangle-codelistURI (Req 116)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowangle-exclusivity** | |
| **Requirement** | **/req/elbow/elbowangle-exclusivity (Req 117)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbownominalpipesize-codelistURI** | |
| **Requirement** | **/req/elbow/elbownominalpipesize-codelistURI (Req 118)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbownominalpipesize-exclusivity** | |
| **Requirement** | **/req/elbow/elbownominalpipesize-exclusivity (Req 119)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowoutsidediameter-codelistURI** | |
| **Requirement** | **/req/elbow/elbowoutsidediameter-codelistURI (Req 120)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowoutsidediameter-exclusivity** | |
| **Requirement** | **/req/elbow/elbowoutsidediameter-exclusivity (Req 121)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowwallthickness-codelistURI** | |
| **Requirement** | **/req/elbow/elbowwallthickness-codelistURI (Req 122)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/elbow/elbowwallthickness-exclusivity** | |
| **Requirement** | **/req/elbow/elbowwallthickness-exclusivity (Req 123)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Flange Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/flange** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/flange/flangemanufacturer-codelistURIexception** | |
| **Requirement** | **/req/flange/flangemanufacturer-codelistURIexception [Recommendation 124]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/flange/flangetype-codelistURI** | |
| **Requirement** | **/req/flange/flangetype-codelistURI (Req 125)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/flange/flangetype-exclusivity** | |
| **Requirement** | **/req/flange/flangetype-exclusivity (Req 126)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/flange/flangematerial-codelistURI** | |
| **Requirement** | **/req/flange/flangematerial-codelistURI (Req 127)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/flange/flangeboundspecification-boundtype** | |
| **Requirement** | **/req/flange/flangeboundspecification-boundtype (Req 128)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/flange/flangeboundspecification-codelistURI** | |
| **Requirement** | **/req/flange/flangeboundspecification-codelistURI (Req 129)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/flange/flangeboundspecification-exclusivity** | |
| **Requirement** | **/req/flange/flangeboundspecification-exclusivity (Req 130)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/flange/flangespecification-codelistURI** | |
| **Requirement** | **/req/flange/flangespecification-codelistURI (Req 131)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/flange/flangespecification-exclusivity** | |
| **Requirement** | **/req/flange/flangespecification-exclusivity (Req 132)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/flange/flangeclass-codelistURI** | |
| **Requirement** | **/req/flange/flangeclass-codelistURI (Req 133)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/flange/flangeclass-exclusivity** | |
| **Requirement** | **/req/flange/flangeclass-exclusivity (Req 134)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/flange/flangenominalpipesize-codelistURI** | |
| **Requirement** | **/req/flange/flangenominalpipesize-codelistURI (Req 135)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/flange/flangeinsulationtype-codelistURI** | |
| **Requirement** | **/req/flange/flangeinsulationtype-codelistURI (Req 136)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Launcherreceiver Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/launcherreceiver** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/launcherreceiver/launcherreceivermanufacturer-codelistURIexception** | |
| **Requirement** | **/req/launcherreceiver/launcherreceivermanufacturer-codelistURIexception [Recommendation 137]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/launcherreceiver/launcherreceivermanufacturer-exclusivity** | |
| **Requirement** | **/req/launcherreceiver/launcherreceivermanufacturer-exclusivity (Req 138)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/launcherreceiver/launcherreceivertype-codelistURI** | |
| **Requirement** | **/req/launcherreceiver/launcherreceivertype-codelistURI (Req 139)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/launcherreceiver/launcherreceiverboundmanufacturer-boundtype** | |
| **Requirement** | **/req/launcherreceiver/launcherreceiverboundmanufacturer-boundtype (Req 140)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/launcherreceiver/launcherreceiverboundmanufacturer-codelistURI** | |
| **Requirement** | **/req/launcherreceiver/launcherreceiverboundmanufacturer-codelistURI (Req 141)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/launcherreceiver/launcherreceiverboundmanufacturer-exclusivity** | |
| **Requirement** | **/req/launcherreceiver/launcherreceiverboundmanufacturer-exclusivity (Req 142)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/launcherreceiver/launcherreceiverproductname-codelistURIexception** | |
| **Requirement** | **/req/launcherreceiver/launcherreceiverproductname-codelistURIexception [Recommendation 143]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/launcherreceiver/launcherreceiverproductname-exclusivity** | |
| **Requirement** | **/req/launcherreceiver/launcherreceiverproductname-exclusivity (Req 144)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/launcherreceiver/launcherreceivermodelnumber-codelistURIexception** | |
| **Requirement** | **/req/launcherreceiver/launcherreceivermodelnumber-codelistURIexception [Recommendation 145]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/launcherreceiver/launcherreceivermodelnumber-exclusivity** | |
| **Requirement** | **/req/launcherreceiver/launcherreceivermodelnumber-exclusivity (Req 146)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/launcherreceiver/launcherreceiverclass-codelistURI** | |
| **Requirement** | **/req/launcherreceiver/launcherreceiverclass-codelistURI (Req 147)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Linepipe Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/linepipe** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/linepipe/linepipemanufacturer-codelistURIexception** | |
| **Requirement** | **/req/linepipe/linepipemanufacturer-codelistURIexception [Recommendation 148]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipetype-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipetype-codelistURI (Req 149)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipematerial-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipematerial-codelistURI (Req 150)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipeboundspecification-boundtype** | |
| **Requirement** | **/req/linepipe/linepipeboundspecification-boundtype (Req 151)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipeboundspecification-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipeboundspecification-codelistURI (Req 152)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipeboundspecification-exclusivity** | |
| **Requirement** | **/req/linepipe/linepipeboundspecification-exclusivity (Req 153)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipespecification-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipespecification-codelistURI (Req 154)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipespecification-exclusivity** | |
| **Requirement** | **/req/linepipe/linepipespecification-exclusivity (Req 155)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipeyieldstrength-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipeyieldstrength-codelistURI (Req 156)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipeyieldstrength-exclusivity** | |
| **Requirement** | **/req/linepipe/linepipeyieldstrength-exclusivity (Req 157)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipegrade-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipegrade-codelistURI (Req 158)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipegrade-exclusivity** | |
| **Requirement** | **/req/linepipe/linepipegrade-exclusivity (Req 159)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipebounddimension-boundtype** | |
| **Requirement** | **/req/linepipe/linepipebounddimension-boundtype (Req 160)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipebounddimension-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipebounddimension-codelistURI (Req 161)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipebounddimension-exclusivity** | |
| **Requirement** | **/req/linepipe/linepipebounddimension-exclusivity (Req 162)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipenominalpipesize-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipenominalpipesize-codelistURI (Req 163)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipeoutsidediameter-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipeoutsidediameter-codelistURI (Req 164)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipeoutsidediameter-exclusivity** | |
| **Requirement** | **/req/linepipe/linepipeoutsidediameter-exclusivity (Req 165)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipewallthickness-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipewallthickness-codelistURI (Req 166)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipewallthickness-exclusivity** | |
| **Requirement** | **/req/linepipe/linepipewallthickness-exclusivity (Req 167)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipebendtype-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipebendtype-codelistURI (Req 168)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipeseamweldtype-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipeseamweldtype-codelistURI (Req 169)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipeseamweldorientation-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipeseamweldorientation-codelistURI (Req 170)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipeseamweldorientation-componentmeasuredfrom** | |
| **Requirement** | **/req/linepipe/linepipeseamweldorientation-componentmeasuredfrom (Req 171)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the componentMeasuredFrom specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipepositionvertical-codelistURI** | |
| **Requirement** | **/req/linepipe/linepipepositionvertical-codelistURI (Req 172)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/linepipe/linepipecoverdepthminimum-uom** | |
| **Requirement** | **/req/linepipe/linepipecoverdepthminimum-uom (Req 173)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Meter Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/meter** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/meter/metermanufacturer-codelistURIexception** | |
| **Requirement** | **/req/meter/metermanufacturer-codelistURIexception [Recommendation 174]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/meter/metermanufacturer-exclusivity** | |
| **Requirement** | **/req/meter/metermanufacturer-exclusivity (Req 175)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/meter/metertype-codelistURI** | |
| **Requirement** | **/req/meter/metertype-codelistURI (Req 176)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/meter/meternominalpipesize-codelistURI** | |
| **Requirement** | **/req/meter/meternominalpipesize-codelistURI (Req 177)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/meter/meterboundmanufacturer-boundtype** | |
| **Requirement** | **/req/meter/meterboundmanufacturer-boundtype (Req 178)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/meter/meterboundmanufacturer-codelistURI** | |
| **Requirement** | **/req/meter/meterboundmanufacturer-codelistURI (Req 179)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/meter/meterboundmanufacturer-exclusivity** | |
| **Requirement** | **/req/meter/meterboundmanufacturer-exclusivity (Req 180)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/meter/meterproductname-codelistURIexception** | |
| **Requirement** | **/req/meter/meterproductname-codelistURIexception [Recommendation 181]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/meter/meterproductname-exclusivity** | |
| **Requirement** | **/req/meter/meterproductname-exclusivity (Req 182)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/meter/metermodelnumber-codelistURIexception** | |
| **Requirement** | **/req/meter/metermodelnumber-codelistURIexception [Recommendation 183]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/meter/metermodelnumber-exclusivity** | |
| **Requirement** | **/req/meter/metermodelnumber-exclusivity (Req 184)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/meter/meterflowrateminimum-uom** | |
| **Requirement** | **/req/meter/meterflowrateminimum-uom (Req 185)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/meter/meterflowratemaximum-uom** | |
| **Requirement** | **/req/meter/meterflowratemaximum-uom (Req 186)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Pump Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/pump** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/pump/pumpmanufacturer-codelistURIexception** | |
| **Requirement** | **/req/pump/pumpmanufacturer-codelistURIexception [Recommendation 187]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpmanufacturer-exclusivity** | |
| **Requirement** | **/req/pump/pumpmanufacturer-exclusivity (Req 188)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumptype-codelistURI** | |
| **Requirement** | **/req/pump/pumptype-codelistURI (Req 189)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpbounddimension-boundtype** | |
| **Requirement** | **/req/pump/pumpbounddimension-boundtype (Req 190)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpbounddimension-codelistURI** | |
| **Requirement** | **/req/pump/pumpbounddimension-codelistURI (Req 191)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpbounddimension-exclusivity** | |
| **Requirement** | **/req/pump/pumpbounddimension-exclusivity (Req 192)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpdiameteroutlet-codelistURI** | |
| **Requirement** | **/req/pump/pumpdiameteroutlet-codelistURI (Req 193)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpdiameteroutlet-exclusivity** | |
| **Requirement** | **/req/pump/pumpdiameteroutlet-exclusivity (Req 194)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpdiameterinlet-codelistURI** | |
| **Requirement** | **/req/pump/pumpdiameterinlet-codelistURI (Req 195)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpdiameterinlet-exclusivity** | |
| **Requirement** | **/req/pump/pumpdiameterinlet-exclusivity (Req 196)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpdiameterimpeller-codelistURI** | |
| **Requirement** | **/req/pump/pumpdiameterimpeller-codelistURI (Req 197)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpdiameterimpeller-exclusivity** | |
| **Requirement** | **/req/pump/pumpdiameterimpeller-exclusivity (Req 198)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpboundmanufacturer-boundtype** | |
| **Requirement** | **/req/pump/pumpboundmanufacturer-boundtype (Req 199)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpboundmanufacturer-codelistURI** | |
| **Requirement** | **/req/pump/pumpboundmanufacturer-codelistURI (Req 200)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpboundmanufacturer-exclusivity** | |
| **Requirement** | **/req/pump/pumpboundmanufacturer-exclusivity (Req 201)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpproductname-codelistURIexception** | |
| **Requirement** | **/req/pump/pumpproductname-codelistURIexception [Recommendation 202]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpproductname-exclusivity** | |
| **Requirement** | **/req/pump/pumpproductname-exclusivity (Req 203)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpmodelnumber-codelistURIexception** | |
| **Requirement** | **/req/pump/pumpmodelnumber-codelistURIexception [Recommendation 204]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpmodelnumber-exclusivity** | |
| **Requirement** | **/req/pump/pumpmodelnumber-exclusivity (Req 205)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpfueltype-codelistURI** | |
| **Requirement** | **/req/pump/pumpfueltype-codelistURI (Req 206)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumppowerrating-uom** | |
| **Requirement** | **/req/pump/pumppowerrating-uom (Req 207)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumpratedflow-uom** | |
| **Requirement** | **/req/pump/pumpratedflow-uom (Req 208)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumppressuresuction-uom** | |
| **Requirement** | **/req/pump/pumppressuresuction-uom (Req 209)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pump/pumppressuredischarge-uom** | |
| **Requirement** | **/req/pump/pumppressuredischarge-uom (Req 210)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Reducer Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/reducer** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/reducer/reducermanufacturer-codelistURIexception** | |
| **Requirement** | **/req/reducer/reducermanufacturer-codelistURIexception [Recommendation 211]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducertype-codelistURI** | |
| **Requirement** | **/req/reducer/reducertype-codelistURI (Req 212)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducermaterial-codelistURI** | |
| **Requirement** | **/req/reducer/reducermaterial-codelistURI (Req 213)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerboundspecification-boundtype** | |
| **Requirement** | **/req/reducer/reducerboundspecification-boundtype (Req 214)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerboundspecification-codelistURI** | |
| **Requirement** | **/req/reducer/reducerboundspecification-codelistURI (Req 215)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerboundspecification-exclusivity** | |
| **Requirement** | **/req/reducer/reducerboundspecification-exclusivity (Req 216)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerspecification-codelistURI** | |
| **Requirement** | **/req/reducer/reducerspecification-codelistURI (Req 217)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerspecification-exclusivity** | |
| **Requirement** | **/req/reducer/reducerspecification-exclusivity (Req 218)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerweight-codelistURI** | |
| **Requirement** | **/req/reducer/reducerweight-codelistURI (Req 219)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerweight-exclusivity** | |
| **Requirement** | **/req/reducer/reducerweight-exclusivity (Req 220)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerbounddimension-boundtype** | |
| **Requirement** | **/req/reducer/reducerbounddimension-boundtype (Req 221)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerbounddimension-codelistURI** | |
| **Requirement** | **/req/reducer/reducerbounddimension-codelistURI (Req 222)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerbounddimension-exclusivity** | |
| **Requirement** | **/req/reducer/reducerbounddimension-exclusivity (Req 223)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducernominalpipesizeinlet-codelistURI** | |
| **Requirement** | **/req/reducer/reducernominalpipesizeinlet-codelistURI (Req 224)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducernominalpipesizeinlet-exclusivity** | |
| **Requirement** | **/req/reducer/reducernominalpipesizeinlet-exclusivity (Req 225)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducernominalpipesizeoutlet-codelistURI** | |
| **Requirement** | **/req/reducer/reducernominalpipesizeoutlet-codelistURI (Req 226)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducernominalpipesizeoutlet-exclusivity** | |
| **Requirement** | **/req/reducer/reducernominalpipesizeoutlet-exclusivity (Req 227)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reduceroutsidediameterinlet-codelistURI** | |
| **Requirement** | **/req/reducer/reduceroutsidediameterinlet-codelistURI (Req 228)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reduceroutsidediameterinlet-exclusivity** | |
| **Requirement** | **/req/reducer/reduceroutsidediameterinlet-exclusivity (Req 229)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reduceroutsidediameteroutlet-codelistURI** | |
| **Requirement** | **/req/reducer/reduceroutsidediameteroutlet-codelistURI (Req 230)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reduceroutsidediameteroutlet-exclusivity** | |
| **Requirement** | **/req/reducer/reduceroutsidediameteroutlet-exclusivity (Req 231)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerwallthicknessinlet-codelistURI** | |
| **Requirement** | **/req/reducer/reducerwallthicknessinlet-codelistURI (Req 232)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerwallthicknessinlet-exclusivity** | |
| **Requirement** | **/req/reducer/reducerwallthicknessinlet-exclusivity (Req 233)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerwallthicknessoutlet-codelistURI** | |
| **Requirement** | **/req/reducer/reducerwallthicknessoutlet-codelistURI (Req 234)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/reducer/reducerwallthicknessoutlet-exclusivity** | |
| **Requirement** | **/req/reducer/reducerwallthicknessoutlet-exclusivity (Req 235)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Tap Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/tap** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/tap/tapmanufacturer-codelistURIexception** | |
| **Requirement** | **/req/tap/tapmanufacturer-codelistURIexception [Recommendation 236]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/tap/taptype-codelistURI** | |
| **Requirement** | **/req/tap/taptype-codelistURI (Req 237)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tap/tapmaterial-codelistURI** | |
| **Requirement** | **/req/tap/tapmaterial-codelistURI (Req 238)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tap/tapbounddimension-boundtype** | |
| **Requirement** | **/req/tap/tapbounddimension-boundtype (Req 239)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tap/tapbounddimension-codelistURI** | |
| **Requirement** | **/req/tap/tapbounddimension-codelistURI (Req 240)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tap/tapbounddimension-exclusivity** | |
| **Requirement** | **/req/tap/tapbounddimension-exclusivity (Req 241)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tap/tapnominalpipesize-codelistURI** | |
| **Requirement** | **/req/tap/tapnominalpipesize-codelistURI (Req 242)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tap/tapnominalpipesize-exclusivity** | |
| **Requirement** | **/req/tap/tapnominalpipesize-exclusivity (Req 243)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tap/tapoutsidediameterinlet-codelistURI** | |
| **Requirement** | **/req/tap/tapoutsidediameterinlet-codelistURI (Req 244)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tap/tapoutsidediameterinlet-exclusivity** | |
| **Requirement** | **/req/tap/tapoutsidediameterinlet-exclusivity (Req 245)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tap/tapoutsidediameteroutlet-codelistURI** | |
| **Requirement** | **/req/tap/tapoutsidediameteroutlet-codelistURI (Req 246)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tap/tapoutsidediameteroutlet-exclusivity** | |
| **Requirement** | **/req/tap/tapoutsidediameteroutlet-exclusivity (Req 247)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tap/tapfunction-codelistURI** | |
| **Requirement** | **/req/tap/tapfunction-codelistURI (Req 248)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tap/tapinstallmethod-codelistURI** | |
| **Requirement** | **/req/tap/tapinstallmethod-codelistURI (Req 249)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Tee Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/tee** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/tee/teemanufacturer-codelistURIexception** | |
| **Requirement** | **/req/tee/teemanufacturer-codelistURIexception [Recommendation 250]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teetype-codelistURI** | |
| **Requirement** | **/req/tee/teetype-codelistURI (Req 251)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teematerial-codelistURI** | |
| **Requirement** | **/req/tee/teematerial-codelistURI (Req 252)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teeboundspecification-boundtype** | |
| **Requirement** | **/req/tee/teeboundspecification-boundtype (Req 253)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teeboundspecification-codelistURI** | |
| **Requirement** | **/req/tee/teeboundspecification-codelistURI (Req 254)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teeboundspecification-exclusivity** | |
| **Requirement** | **/req/tee/teeboundspecification-exclusivity (Req 255)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teespecification-codelistURI** | |
| **Requirement** | **/req/tee/teespecification-codelistURI (Req 256)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teespecification-exclusivity** | |
| **Requirement** | **/req/tee/teespecification-exclusivity (Req 257)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teeweight-codelistURI** | |
| **Requirement** | **/req/tee/teeweight-codelistURI (Req 258)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teeweight-exclusivity** | |
| **Requirement** | **/req/tee/teeweight-exclusivity (Req 259)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teebounddimension-boundtype** | |
| **Requirement** | **/req/tee/teebounddimension-boundtype (Req 260)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teebounddimension-codelistURI** | |
| **Requirement** | **/req/tee/teebounddimension-codelistURI (Req 261)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teebounddimension-exclusivity** | |
| **Requirement** | **/req/tee/teebounddimension-exclusivity (Req 262)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teenominalpipesize-codelistURI** | |
| **Requirement** | **/req/tee/teenominalpipesize-codelistURI (Req 263)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teenominalpipesize-exclusivity** | |
| **Requirement** | **/req/tee/teenominalpipesize-exclusivity (Req 264)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teeoutsidediameterrun-codelistURI** | |
| **Requirement** | **/req/tee/teeoutsidediameterrun-codelistURI (Req 265)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teeoutsidediameterrun-exclusivity** | |
| **Requirement** | **/req/tee/teeoutsidediameterrun-exclusivity (Req 266)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teeoutsidediameteroutlet-codelistURI** | |
| **Requirement** | **/req/tee/teeoutsidediameteroutlet-codelistURI (Req 267)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teeoutsidediameterrun-exclusivity** | |
| **Requirement** | **/req/tee/teeoutsidediameterrun-exclusivity (Req 268)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teecentertoendrun-codelistURI** | |
| **Requirement** | **/req/tee/teecentertoendrun-codelistURI (Req 269)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teecentertoendrun-exclusivity** | |
| **Requirement** | **/req/tee/teecentertoendrun-exclusivity (Req 270)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teecentertoendoutlet-codelistURI** | |
| **Requirement** | **/req/tee/teecentertoendoutlet-codelistURI (Req 271)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teecentertoendoutlet-exclusivity** | |
| **Requirement** | **/req/tee/teecentertoendoutlet-exclusivity (Req 272)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teewallthicknessrun-codelistURI** | |
| **Requirement** | **/req/tee/teewallthicknessrun-codelistURI (Req 273)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teewallthicknessrun-exclusivity** | |
| **Requirement** | **/req/tee/teewallthicknessrun-exclusivity (Req 274)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teewallthicknessoutlet-codelistURI** | |
| **Requirement** | **/req/tee/teewallthicknessoutlet-codelistURI (Req 275)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/tee/teewallthicknessoutlet-exclusivity** | |
| **Requirement** | **/req/tee/teewallthicknessoutlet-exclusivity (Req 276)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Valve Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/valve** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/valve/valvemanufacturer-codelistURIexception** | |
| **Requirement** | **/req/valve/valvemanufacturer-codelistURIexception [Recommendation 277]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valvetype-codelistURI** | |
| **Requirement** | **/req/valve/valvetype-codelistURI (Req 278)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valveboundspecification-boundtype** | |
| **Requirement** | **/req/valve/valveboundspecification-boundtype (Req 279)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valveboundspecification-codelistURI** | |
| **Requirement** | **/req/valve/valveboundspecification-codelistURI (Req 280)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valveboundspecification-exclusivity** | |
| **Requirement** | **/req/valve/valveboundspecification-exclusivity (Req 281)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valvespecification-codelistURI** | |
| **Requirement** | **/req/valve/valvespecification-codelistURI (Req 282)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valvespecification-exclusivity** | |
| **Requirement** | **/req/valve/valvespecification-exclusivity (Req 283)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valveclass-codelistURI** | |
| **Requirement** | **/req/valve/valveclass-codelistURI (Req 284)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valveclass-exclusivity** | |
| **Requirement** | **/req/valve/valveclass-exclusivity (Req 285)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valvenominalpipesize-codelistURI** | |
| **Requirement** | **/req/valve/valvenominalpipesize-codelistURI (Req 286)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valveoperability-codelistURI** | |
| **Requirement** | **/req/valve/valveoperability-codelistURI (Req 287)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valveactuationTime-uom** | |
| **Requirement** | **/req/valve/valveactuationTime-uom (Req 288)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valvefunctiontype-codelistURI** | |
| **Requirement** | **/req/valve/valvefunctiontype-codelistURI (Req 289)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/valve/valvemodelnumber-codelistURIexception** | |
| **Requirement** | **/req/valve/valvemodelnumber-codelistURIexception [Recommendation 290]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |

Casing Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/casing** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/casing/casingmanufacturer-codelistURIexception** | |
| **Requirement** | **/req/casing/casingmanufacturer-codelistURIexception [Recommendation 291]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingcrossingtype-codelistURI** | |
| **Requirement** | **/req/casing/casingcrossingtype-codelistURI (Req 292)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingmaterial-codelistURI** | |
| **Requirement** | **/req/casing/casingmaterial-codelistURI (Req 293)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingboundspecification-boundtype** | |
| **Requirement** | **/req/casing/casingboundspecification-boundtype (Req 294)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingboundspecification-codelistURI** | |
| **Requirement** | **/req/casing/casingboundspecification-codelistURI (Req 295)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingboundspecification-exclusivity** | |
| **Requirement** | **/req/casing/casingboundspecification-exclusivity (Req 296)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingspecification-codelistURI** | |
| **Requirement** | **/req/casing/casingspecification-codelistURI (Req 297)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingspecification-exclusivity** | |
| **Requirement** | **/req/casing/casingspecification-exclusivity (Req 298)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingspecification-codelistURI** | |
| **Requirement** | **/req/casing/casingspecification-codelistURI (Req 299)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingspecification-exclusivity** | |
| **Requirement** | **/req/casing/casingspecification-exclusivity (Req 300)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingyieldstrength-codelistURI** | |
| **Requirement** | **/req/casing/casingyieldstrength-codelistURI (Req 301)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingyieldstrength-exclusivity** | |
| **Requirement** | **/req/casing/casingyieldstrength-exclusivity (Req 302)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingbounddimension-boundtype** | |
| **Requirement** | **/req/casing/casingbounddimension-boundtype (Req 303)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingbounddimension-codelistURI** | |
| **Requirement** | **/req/casing/casingbounddimension-codelistURI (Req 304)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingbounddimension-exclusivity** | |
| **Requirement** | **/req/casing/casingbounddimension-exclusivity (Req 305)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingnominalpipesize-codelistURI** | |
| **Requirement** | **/req/casing/casingnominalpipesize-codelistURI (Req 306)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingnominalpipesize-exclusivity** | |
| **Requirement** | **/req/casing/casingnominalpipesize-exclusivity (Req 307)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingoutsidediameter-codelistURI** | |
| **Requirement** | **/req/casing/casingoutsidediameter-codelistURI (Req 308)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingoutsidediameter-exclusivity** | |
| **Requirement** | **/req/casing/casingoutsidediameter-exclusivity (Req 309)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingwallthickness-codelistURI** | |
| **Requirement** | **/req/casing/casingwallthickness-codelistURI (Req 310)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingwallthickness-exclusivity** | |
| **Requirement** | **/req/casing/casingwallthickness-exclusivity (Req 311)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingannulustype-codelistURI** | |
| **Requirement** | **/req/casing/casingannulustype-codelistURI (Req 312)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casinginsulatortype-codelistURI** | |
| **Requirement** | **/req/casing/casinginsulatortype-codelistURI (Req 313)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingsealtype-codelistURI** | |
| **Requirement** | **/req/casing/casingsealtype-codelistURI (Req 314)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/casing/casingventcount-codelistURI** | |
| **Requirement** | **/req/casing/casingventcount-codelistURI (Req 315)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Coating Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/coating** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/coating/coatingmanufacturer-codelistURIexception** | |
| **Requirement** | **/req/coating/coatingmanufacturer-codelistURIexception [Recommendation 316]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingmanufacturer-exclusivity** | |
| **Requirement** | **/req/coating/coatingmanufacturer-exclusivity (Req 317)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingtype-codelistURI** | |
| **Requirement** | **/req/coating/coatingtype-codelistURI (Req 318)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingmaterial-codelistURI** | |
| **Requirement** | **/req/coating/coatingmaterial-codelistURI (Req 319)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingboundmanufacturer-boundtype** | |
| **Requirement** | **/req/coating/coatingboundmanufacturer-boundtype (Req 320)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspects the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingboundmanufacturer-codelistURI** | |
| **Requirement** | **/req/coating/coatingboundmanufacturer-codelistURI (Req 321)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingboundmanufacturer-exclusivity** | |
| **Requirement** | **/req/coating/coatingboundmanufacturer-exclusivity (Req 322)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingproductname-codelistURIexception** | |
| **Requirement** | **/req/coating/coatingproductname-codelistURIexception [Recommendation 323]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingproductname-exclusivity** | |
| **Requirement** | **/req/coating/coatingproductname-exclusivity (Req 324)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingmodelnumber-codelistURIexception** | |
| **Requirement** | **/req/coating/coatingmodelnumber-codelistURIexception [Recommendation 325]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingmodelnumber-exclusivity** | |
| **Requirement** | **/req/coating/coatingmodelnumber-exclusivity (Req 326)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatinglayernumber-exclusivity** | |
| **Requirement** | **/req/coating/coatinglayernumber-exclusivity (Req 327)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingthickness-codelistURI** | |
| **Requirement** | **/req/coating/coatingthickness-codelistURI (Req 328)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingmultilayerdescription-exclusivity** | |
| **Requirement** | **/req/coating/coatingmultilayerdescription-exclusivity (Req 329)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/coating/coatingapplicationlocation-codelistURI** | |
| **Requirement** | **/req/coating/coatingapplicationlocation-codelistURI (Req 330)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Sleeve Class Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/sleeve** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Dependency** | **/req/pmlcomponent** | |
| **Test** | **/conf/sleeve/sleevemanufacturer-codelistURIexception** | |
| **Requirement** | **/req/sleeve/sleevemanufacturer-codelistURIexception [Recommendation 331]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleevemanufacturer-exclusivity** | |
| **Requirement** | **/req/sleeve/sleevemanufacturer-exclusivity (Req 332)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve//sleevetype-codelistURI** | |
| **Requirement** | **/req/sleeve//sleevetype-codelistURI (Req 333)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleevematerial-codelistURI** | |
| **Requirement** | **/req/sleeve/sleevematerial-codelistURI (Req 334)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleevebounddimension-boundtype** | |
| **Requirement** | **/req/sleeve/sleevebounddimension-boundtype (Req 335)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspects the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleevebounddimension-codelistURI** | |
| **Requirement** | **/req/sleeve/sleevebounddimension-codelistURI (Req 336)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleevebounddimension-exclusivity** | |
| **Requirement** | **/req/sleeve/sleevebounddimension-exclusivity (Req 337)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleevenominalpipesize-codelistURI** | |
| **Requirement** | **/req/sleeve/sleevenominalpipesize-codelistURI (Req 338)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleevenominalpipesize-exclusivity** | |
| **Requirement** | **/req/sleeve/sleevenominalpipesize-exclusivity (Req 339)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleevewallthickness-codelistURI** | |
| **Requirement** | **/req/sleeve/sleevewallthickness-codelistURI (Req 340)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleevewallthickness-exclusivity** | |
| **Requirement** | **/req/sleeve/sleevewallthickness-exclusivity (Req 341)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleeveboundmanufacturer-boundtype** | |
| **Requirement** | **/req/sleeve/sleeveboundmanufacturer-boundtype (Req 342)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the boundType specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleeveboundmanufacturer-codelistURI** | |
| **Requirement** | **/req/sleeve/sleeveboundmanufacturer-codelistURI (Req 343)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the codelistURI specifications defined within this requirement and that all URI references can be fully resolved to a GML 3.3 dictionary compliant code list |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleeveboundmanufacturer-exclusivity** | |
| **Requirement** | **/req/sleeve/sleeveboundmanufacturer-exclusivity (Req 344)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleeveproductname-codelistURIexception** | |
| **Requirement** | **/req/sleeve/sleeveproductname-codelistURIexception [Recommendation 345]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleeveproductname-exclusivity** | |
| **Requirement** | **/req/sleeve/sleeveproductname-exclusivity (Req 346)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleevemodelnumber-codelistURIexception** | |
| **Requirement** | **/req/sleeve/sleevemodelnumber-codelistURIexception [Recommendation 347]** |
| **Test Purpose** | **As this is a recommendation and not a requirement, no testing is required.** This placeholder is provided as a reminder to assess whether the encoding follows the best practices recommended, and to submit non-authoritative data to the PipelineML Data Governance Body for consideration of inclusion into authoritative code list. |
| **Test Method** | N/A |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleevemodelnumber-exclusivity** | |
| **Requirement** | **/req/sleeve/sleevemodelnumber-exclusivity (Req 348)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the exclusivity specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/sleeve/sleevepressurerating-uom** | |
| **Requirement** | **/req/sleeve/sleevepressurerating-uom (Req 349)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the units-of-measure (uom) specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

PipelineML Core XML Abstract Requirements Class (Normative)

|  |  |  |
| --- | --- | --- |
| **Conformance Class** | | |
| **/conf/pipelineml-xsd** | | |
| **Requirements** | **/req/pipelineml-abstract** | |
| **Test** | **/conf/pipelineml-xsd/xsd** | |
| **Requirement** | **/req/pipelineml-xsd/xsd (Req 350)** |
| **Test Purpose** | Ensure that the XML data instance validates against the XSD schema located at http://schemas.opengis.net/pipelineml/1.0/PMLComponent.xsd |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipelineml-xsd/codelist** | |
| **Requirement** | **/req/pipelineml-xsd/codelist (Req 351)** |
| **Test Purpose** | Ensure that the XML data instance vocabulary term is encoded with HTTP URI in @xlink:href, provide a human readable description in @xlink:title, and where applicable @xlink:uom (when units-of-measure are a designated requirement) |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipelineml-xsd/identifier-uri** | |
| **Requirement** | **/req/pipelineml-xsd/identifier-uri (Req 352)** |
| **Test Purpose** | Ensure that the XML data instance feature identifiers (unique name) are provided in gml:identifier and @codeSpace = “http://www.ietf.org/rfc/rfc2616” SHOULD be URI of resource using Linked Open Data principles |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipelineml-xsd/iso8601-time** | |
| **Requirement** | **/req/pipelineml-xsd/iso8601-time (Req 353)** |
| **Test Purpose** | Ensure that the XML data instance date-time elements occurrences are encoded using ISO8601 extended time format |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |
| **Test** | **/conf/pipelineml-xsd/time-zone** | |
| **Requirement** | **/req/pipelineml-xsd/time-zone (Req 354)** |
| **Test Purpose** | Ensure that the encoding fully adheres to the otherValue specifications defined within this requirement |
| **Test Method** | Visually inspect the encoding to verify the above requirement is met |
| **Test Type** | Capability |

Annex B: Revision history

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Release | Author | Paragraph modified | Description |
| 6/2/2018 | 1.0 | John Tisdale |  | Initial Draft |
| 8/20/2018 | 1.1 | John Tisdale | Numerous | Changed all content related to substandard/non-authoritative data |
|  |  |  |  |  |

Annex C: Bibliography

TBD

<A Bibliography, if present, shall appear as the last annex. >

1. [www.opengeospatial.org/cite](http://www.opengeospatial.org/cite) [↑](#footnote-ref-1)