SysML 1.4: Quantities, Units, Dimensions & Values (QUDV) & ISO 80000 Library

Nicolas Rouquette
Jet Propulsion Laboratory
California Institute of Technology
March 2014

Copyright 2013, 2014, California Institute of Technology ("Caltech")
U.S. Government sponsorship acknowledged. All rights reserved.

What is ISO/IEC 80000?

Standards-Development Organizations (the OMG is a much smaller SDO compared to ISO/IEC):

ISO:	International Standards Organization
IEC:	International Electrotechnical Commission

ISO/IEC 80000 is a collection of 14 standards (called parts):

Part 1:	General information & definitions about quantities & units (covers a subset of VIM chapter 1 “quantities & units” – see below)
Part 2:	Mathematical symbols & notation
Parts 3-14:	Tables of domain-specific definitions of quantities & units

Brief history:

1875:	International Bureau of Weights and Measures (BIPM) formed to “establish fundamental standards & scales for the measurement of the principal physical quantities & maintain the international prototypes”
1970:	First publication of the “International System of Units” (SI brochure) http://www.bipm.org/en/si/si_brochure/
1997:	First publication of “Guide of expression of Uncertainty in Measurement” (GUM) and “Vocabulary of International Metrology” (VIM) http://www.bipm.org/en/publications/guides/gum.html http://www.bipm.org/en/publications/guides/vim.html
2009:	All 14 parts in ISO/IEC 80000 published

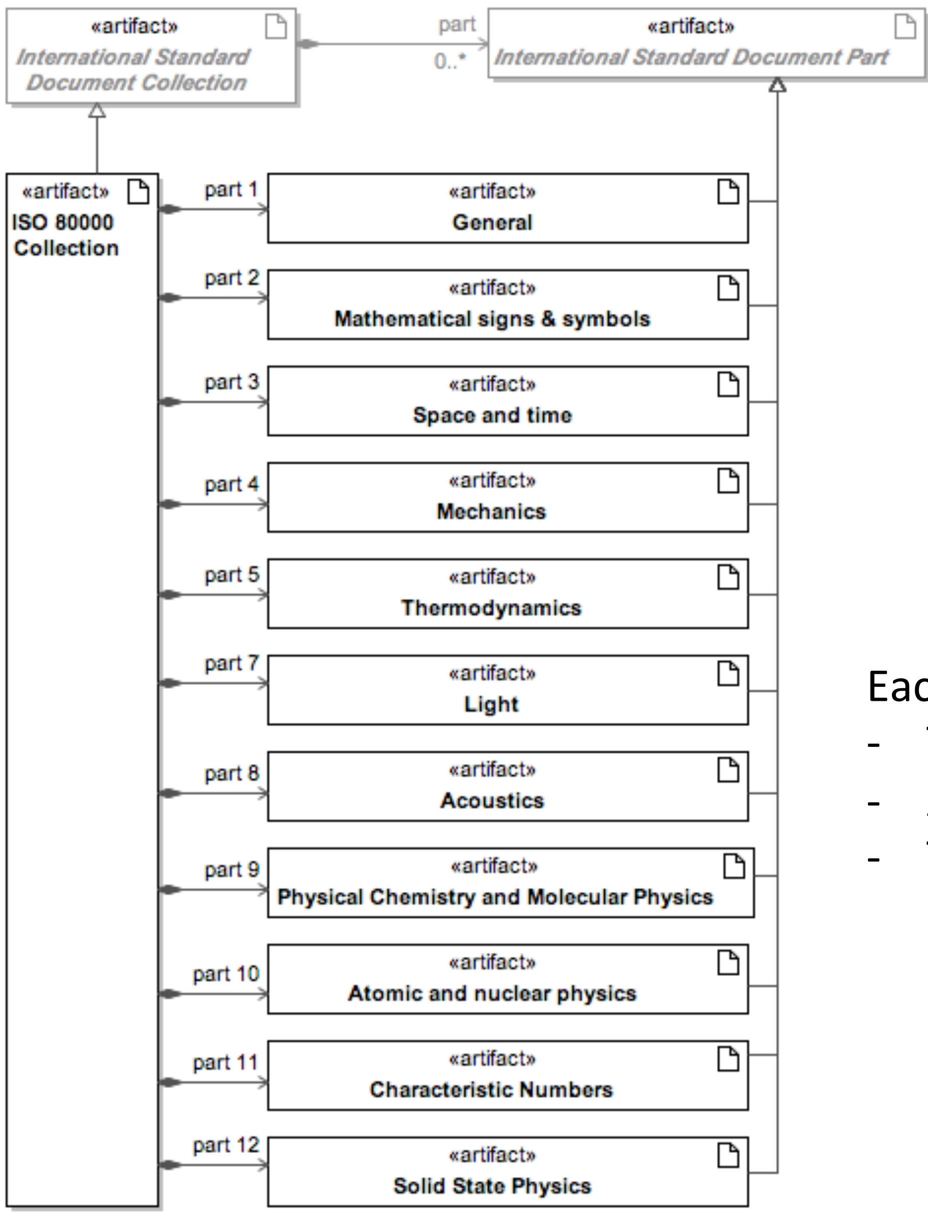
Topics

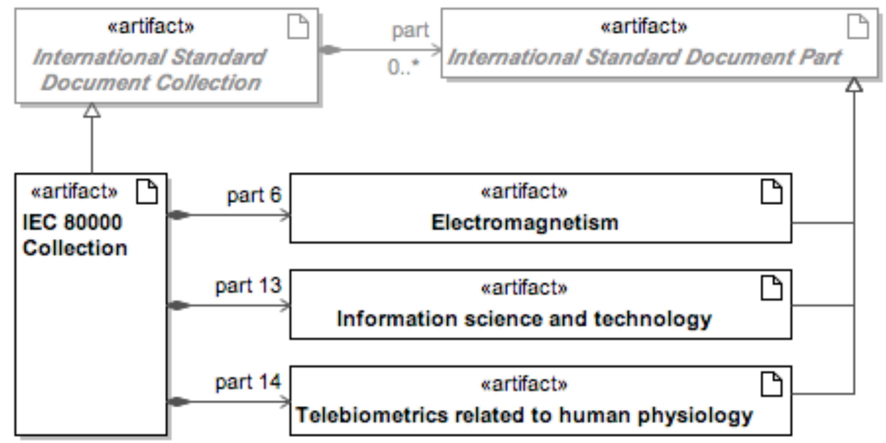
- What is ISO/IEC 80000?
- Coverage of ISO/IEC 80000 in SysML 1.4
- Brief Introduction to QUDV
- Organization of the library
  1. Using the library
  2. Modeling Values of Scalar & Structured ValueTypes
  3. Extending the library
  4. Dimensional Analysis

Backup Material / Technical Details

- History; • Technical Issues;
- Differences between SysML & QUDV Vocabularies;
- Modeling ValueTypes & ValueSpecifications;
- Example of Structured vs. Scalar DataType Modeling

What is ISO/IEC 80000 ?

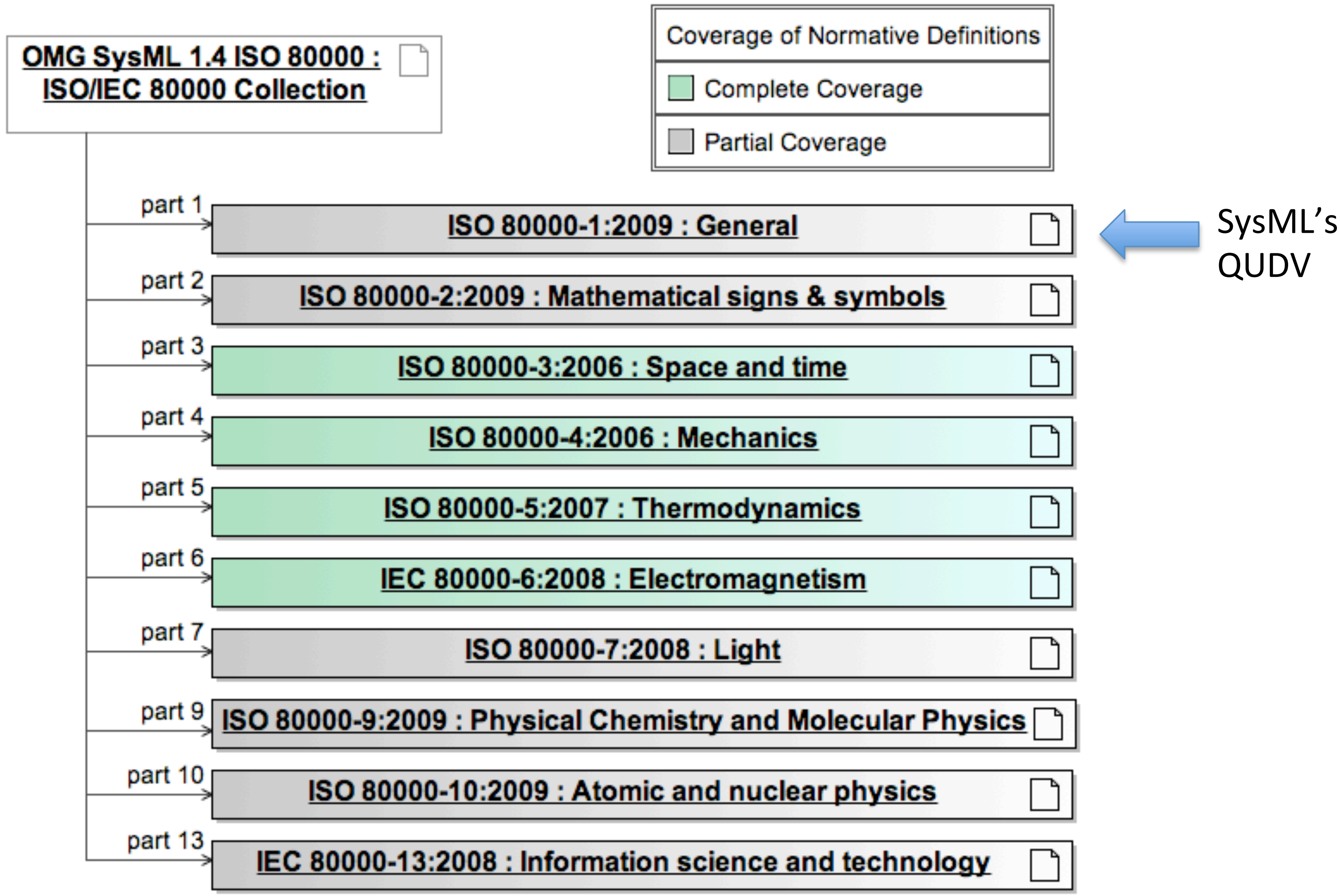




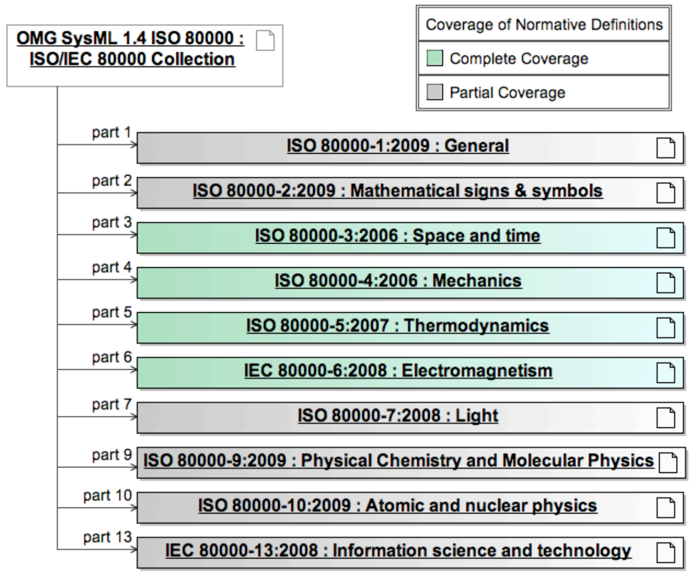
Each document part is identified by:

- The publication date (e.g., ISO 80000-1:2009)
- An edition number (e.g., 1st edition)
- Technical Corrigendum (as applicable)

Coverage of ISO/IEC 80000 in SysML 1.4 (**non-normative** Annex E.6)



Scope of Coverage in SysML 1.4 (**<u>non-normative</u>** Annex E.6)



No coverage in SysML 1.4 for:

- Part 8 Acoustics
- Part 11 Characteristic Numbers
- Part 12 Solid-state physics
- Part 14 Telebiometrics related to human physiology

(contributions welcome!)

Notes about coverage of ISO/IEC 80000 in SysML 1.4:

1. Schedule (SysML 1.4 RTF ended December 2013)
2. Base quantities & units (parts 3,4,5,6,7,9)
3. Improving QUDV to encode ISO/IEC 80000 definitions
4. Domain-specific knowledge required

=> Needs peer review and contributions!

Brief Introduction to QUDV (Quantities, Units, Dimensions & Values) (**non-normative** Annex E.5)

- Why QUDV?
  - SysML has the concept of 'quantity' [VIM 3-1.1]

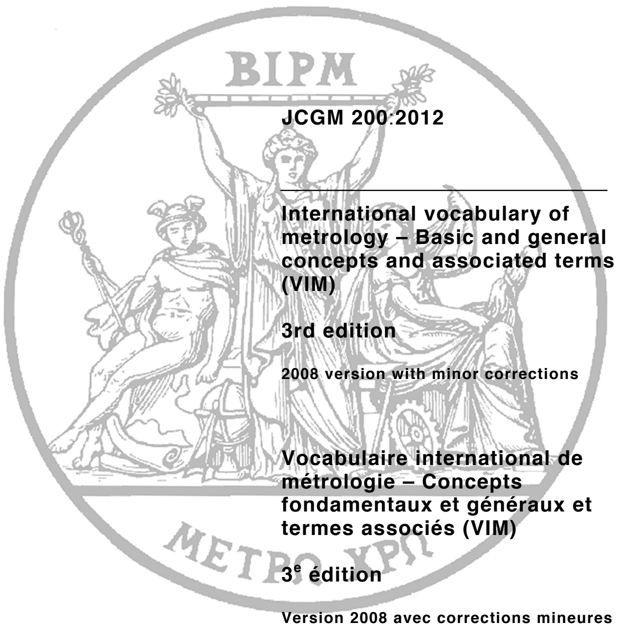
    "property of a phenomenon, body or substance, where the property has a magnitude that can be expressed as a number and a reference"
  - SysML 1.1 had limited vocabulary: Unit & Dimension
    - SysML lacked the concept of 'kind of quantity' [VIM 3-1.2]

      aspect common to mutually comparable **quantities**
  - QUDV first introduced in SysML 1.2 & subsequently revised
- SysML 1.4 & QUDV:
  - Improved alignment with VIM 3rd edition, JCGM 200:2012
  - Adequate for modeling a significant scope of ISO/IEC 80000
  - Semantics allow for computer-based Dimensional Analysis!

See #48 for an overview of the scope of the SysML & QUDV vocabularies

Using VIM 3rd edition in SysML 1.4 & QUDV

http://www.bipm.org/en/publications/guides/vim.html



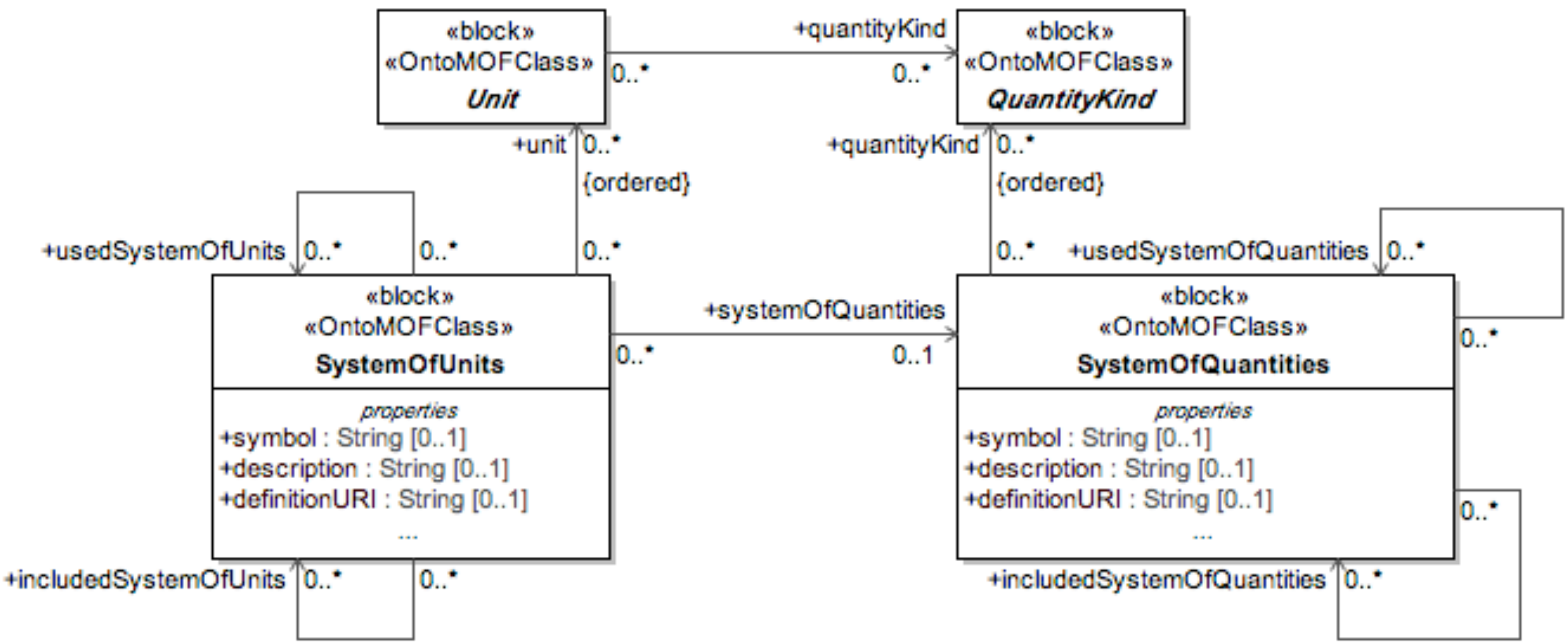
		Normative SysML		Non-normative QUDV & ISO-80000		
1	Quantities and units (30 definitions)	language concept	modeling pattern	language concept	modeling pattern	ISO-80000-1 Library
1.1	quantity	✔				
1.2	kind of quantity			✔		
1.3	system of quantities			✔		
1.4	base quantity			✔		
1.5	derived quantity			✔		
1.6	International System of Quantities (ISQ)					✔
1.7	quantity dimension			✔		
1.8	quantity of dimension one			✔		
1.9	measurement unit	✔				
1.10	base unit			✔		
1.11	derived unit			✔		
1.12	coherent derived unit				✔	
1.13	system of units			✔		
1.14	coherent system of units				✔	
1.15	off-system measurement unit				✔	
1.16	International System of Units (SI)					✔
1.17	multiple of a unit			✔		
1.18	submultiple of a unit			✔		
1.19	quantity value		✔			
1.20	numerical quantity value		✔			
1.21	quantity calculus				✔	
1.22	quantity equation			✔		
1.23	unit equation			✔		
1.24	conversion factor between units			✔		
1.25	numerical value equation		✔			
1.26	ordinal quantity		✔			
1.27	quantity-value scale					
1.28	ordinal quantity-value scale		✔		✔	
1.29	conventional reference scale		✔		✔	
1.30	nominal property				✔	

		No Coverage in QUDV or SysML
2	Measurement	(53 definitions)
3	Devices for measurement	(12 definitions)
4	Properties of measuring devices	(31 definitions)
5	Measurement standards (etalons)	(18 definitions)

See #46 for references

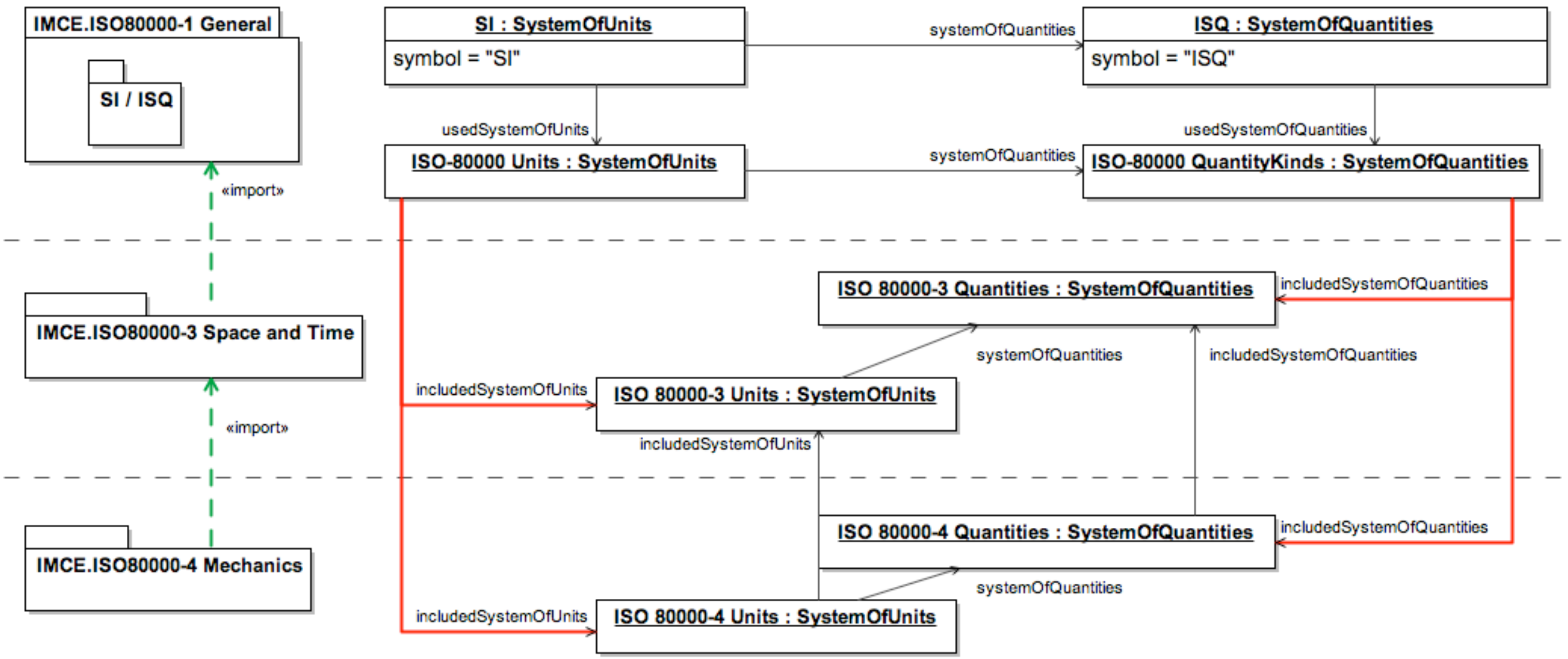
Systems of Units & Quantity Kinds

		Normative SysML		Non-normative QUDV & ISO-80000		
Def.	Quantities and units (30 definitions)	language concept	modeling pattern	language concept	modeling pattern	ISO-80000-1 Library
1.3	system of quantities			✔		
1.6	International System of Quantities (ISQ)					✔
1.13	system of units			✔		
1.16	International System of Units (SI)					✔



Systems of Units & Quantity Kinds

		Normative SysML		Non-normative QUDV & ISO-80000		
Def.	Quantities and units (30 definitions)	language concept	modeling pattern	language concept	modeling pattern	ISO-80000-1 Library
1.3	system of quantities			✔		
1.6	International System of Quantities (ISQ)					✔
1.13	system of units			✔		
1.16	International System of Units (SI)					✔

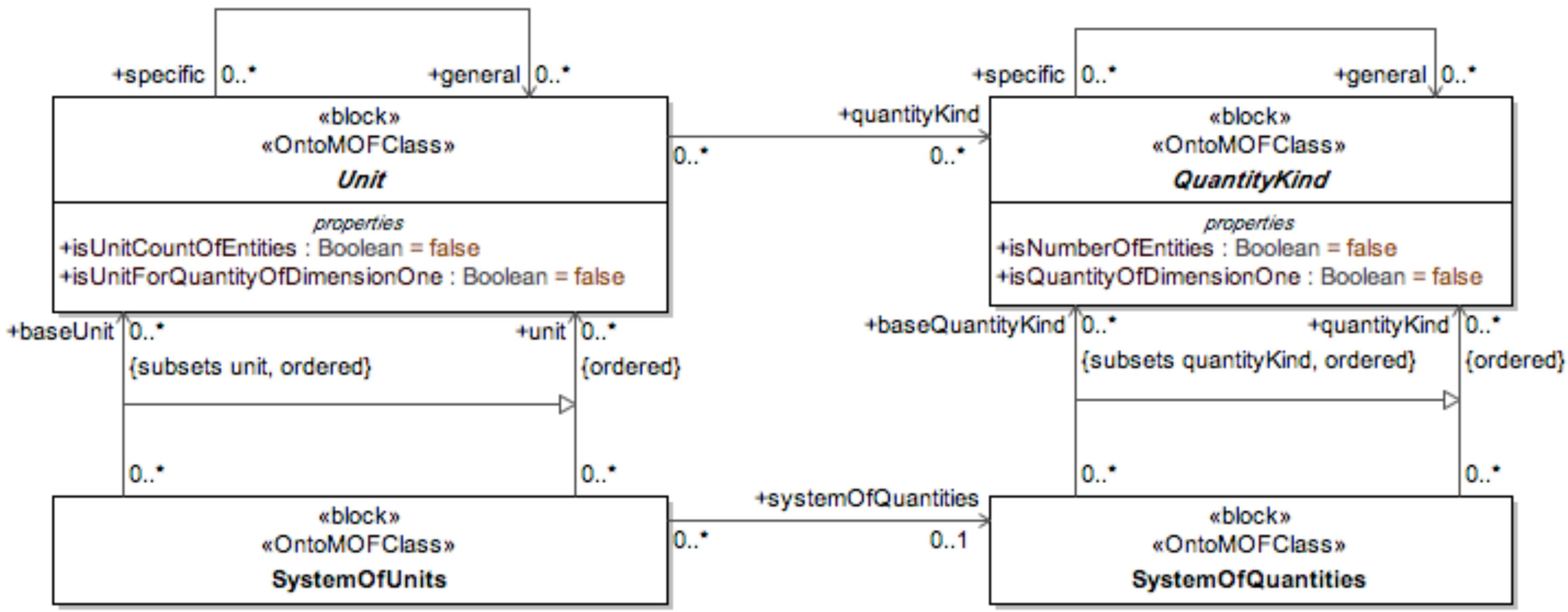


See [#44](#) for SysML issue #16263 preventing modularizing OMG SysML 1.4 ISO-80000 library

See [#45](#) for JPL's modularization where <<import>> has a strong separation semantics

base quantity kinds (ISQ) & base units (SI)

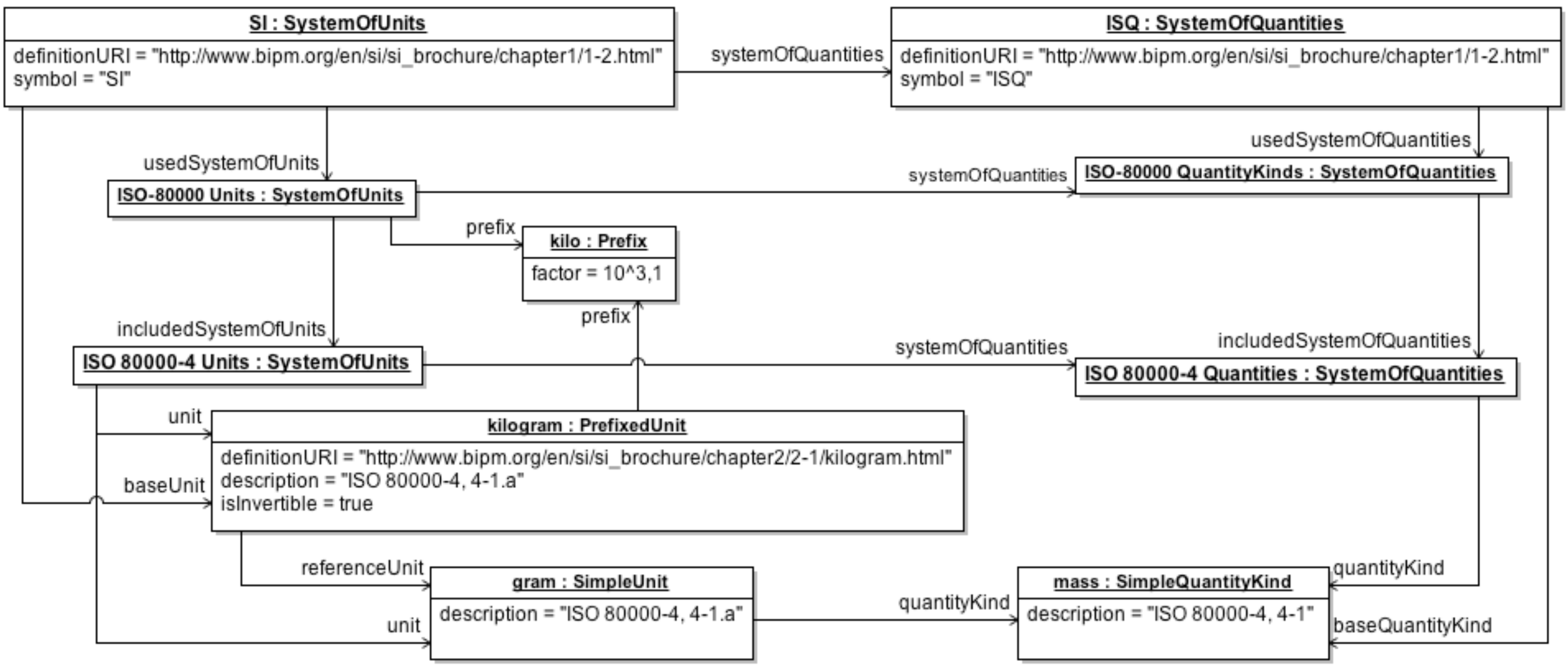
		Normative SysML		Non-normative QUDV & ISO-80000		
1	Quantities and units (30 definitions)	language concept	modeling pattern	language concept	modeling pattern	ISO-80000-1 Library
1.2	kind of quantity			✔		
1.4	base quantity			✔		
1.9	measurement unit	✔				
1.10	base unit			✔		
1.17	multiple of a unit			✔		



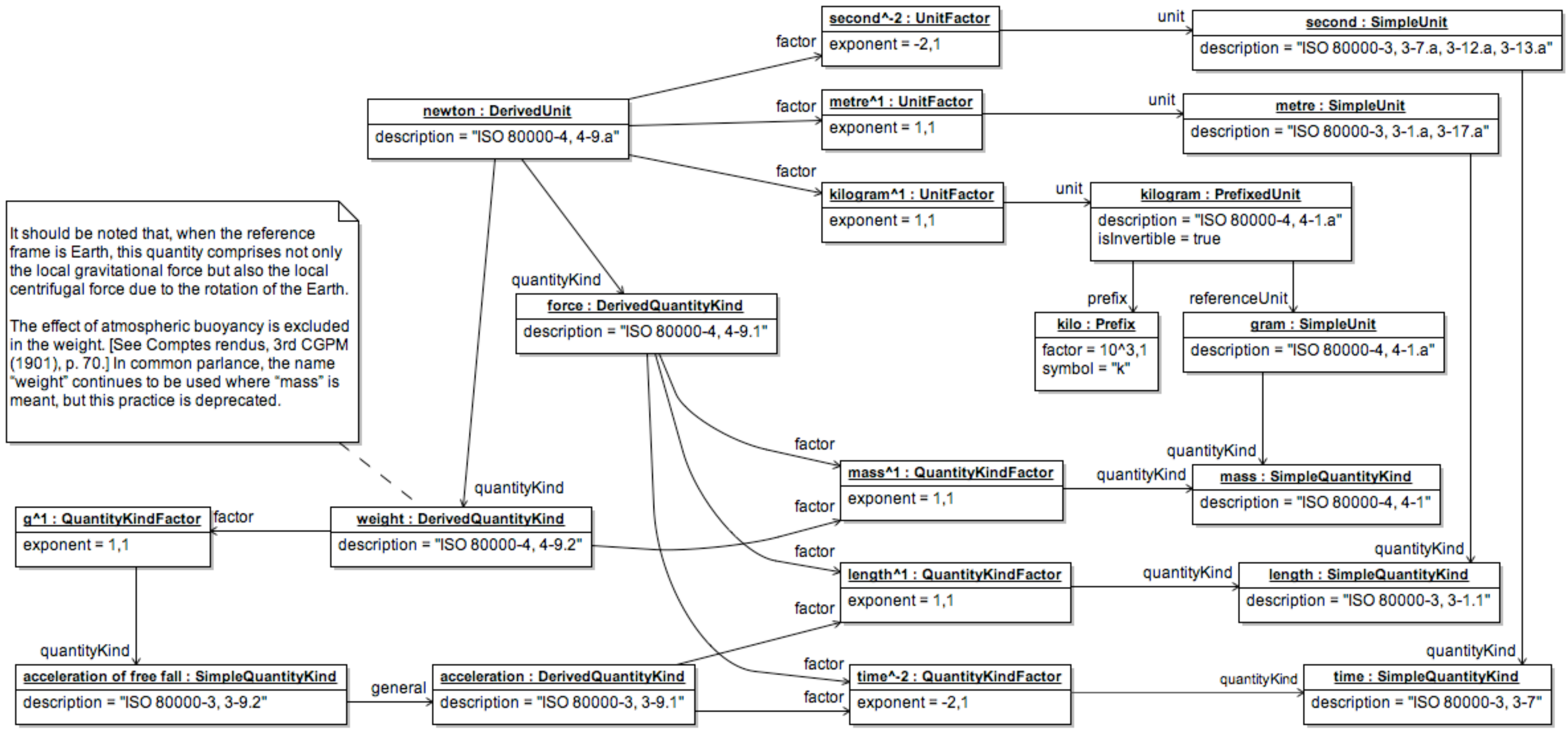
See [#47](#) for more information about taxonomic relationships in metrology (i.e., general/specific units/quantity kinds)

base quantity kinds (ISQ) & base units (SI)

		Normative SysML		Non-normative QUDV & ISO-80000		
1	Quantities and units (30 definitions)	language concept	modeling pattern	language concept	modeling pattern	ISO-80000-1 Library
1.2	kind of quantity			✔		
1.4	base quantity			✔		
1.9	measurement unit	✔				
1.10	base unit			✔		
1.17	multiple of a unit			✔		



Example of Derived Definitions



Definitions illustrated in this figure		Count
1.5	**derived quantity**	3
1.11	**derived unit**	1
1.22	**quantity equation**	3
1.23	**unit equation**	2

In SysML 1.4, ISO 80000 has definitions for:
- 325 quantity kinds
- 191 non-prefixed units; 2564 prefixed units

Organization of the ISO 80000 Library

Practical Perspective:

1. Using the library
2. Extending the library
3. Dimensional Analysis

Notes:

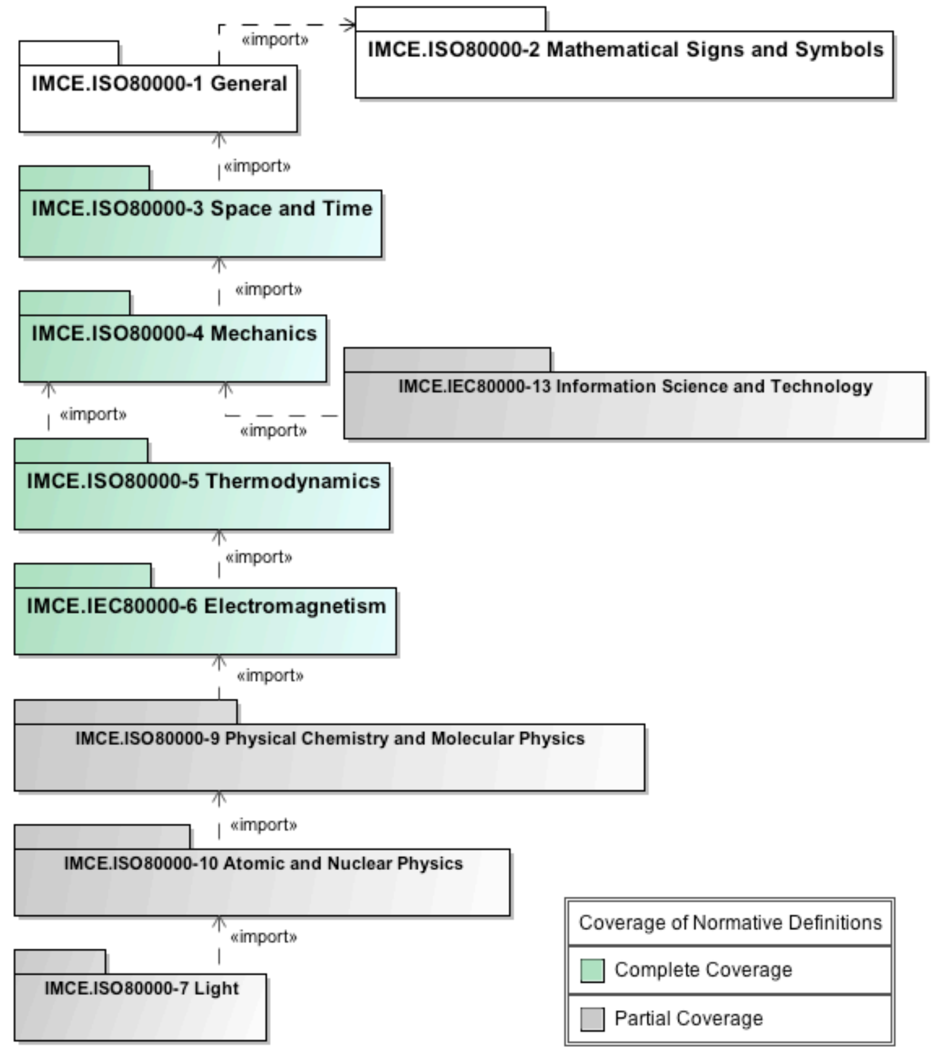
OMG SysML 1.4 ISO 80000 uses <<import>> with a stronger modularity semantics than defined in UML

In OMG SysML 1.4 ISO 80000, [X] <<import>> [Y] means:

1) Some definition in [X] depends on some definition in [Y]
2) No definition in the <<import>> closure of [Y] depends on any definition in [X]
3) In most cases: ISO/IEC document part [X] has a normative reference to ISO/IEC document part [Y]

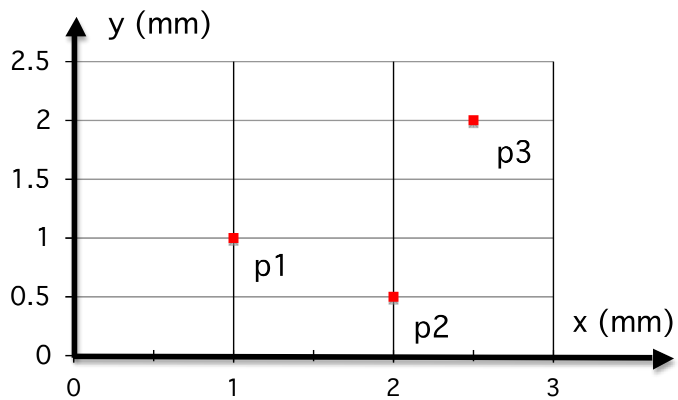
(A stereotype extending PackageImport would have been appropriate to indicate the above semantics)

See #46 for Normative References vs. Semantic Dependencies



1) Using the ISO-80000 Library

Example: Modeling the position of a point on a 2D Cartesian plane

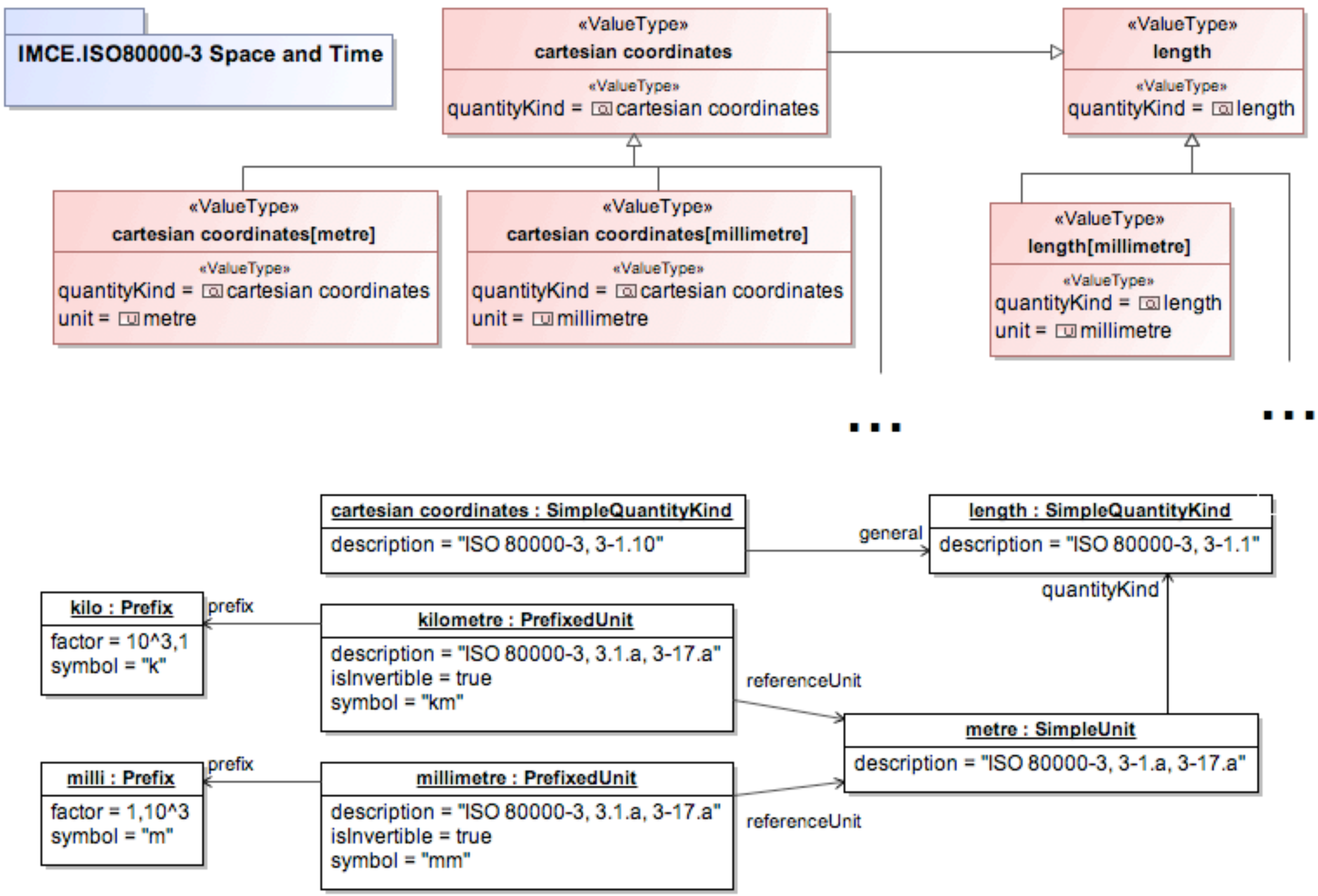


Goal: Represent in SysML the following points in the 2D plane:

- Origin (x=0mm, y=0mm)
- p1 (x=1 mm, y=1 mm)
- p2 (x=2 mm, y=0.5 mm)
- p3 (x=2.5 mm, y=2 mm)

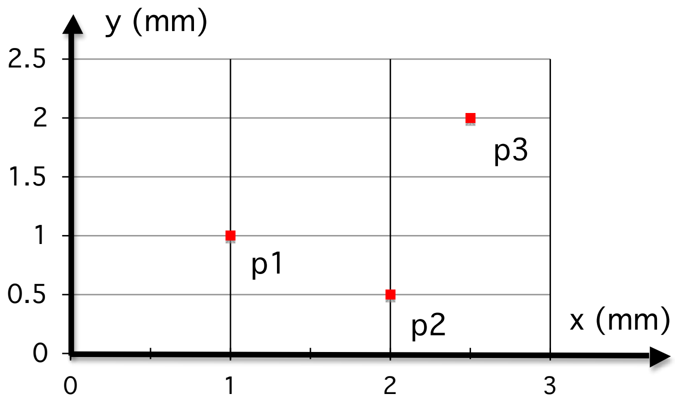
Modularity: Import ISO 80000-3 (space & time)
(this part suffices for this application)

Relevant Subset of ISO 80000-3 for 2D Cartesian geometry example



See #54 for practical considerations about naming ValueTypes

2D Cartesian Geometry Example (Scalar & Structured Values)



«ValueType»
cartesian coordinates[millimetre]
{quantityKind = cartesian coordinates, unit = millimetre}

0.5 : cartesian coordinates[millimetre]
SysML::Libraries::PrimitiveTypes::Real(0.5)

1 : cartesian coordinates[millimetre]
SysML::Libraries::PrimitiveTypes::Real(1)

2 : cartesian coordinates[millimetre]
SysML::Libraries::PrimitiveTypes::Real(1)

2.5 : cartesian coordinates[millimetre]
SysML::Libraries::PrimitiveTypes::Real(2.5)

«ValueType»
2D Cartesian Position[mm]
x : cartesian coordinates[millimetre]
y : cartesian coordinates[millimetre]

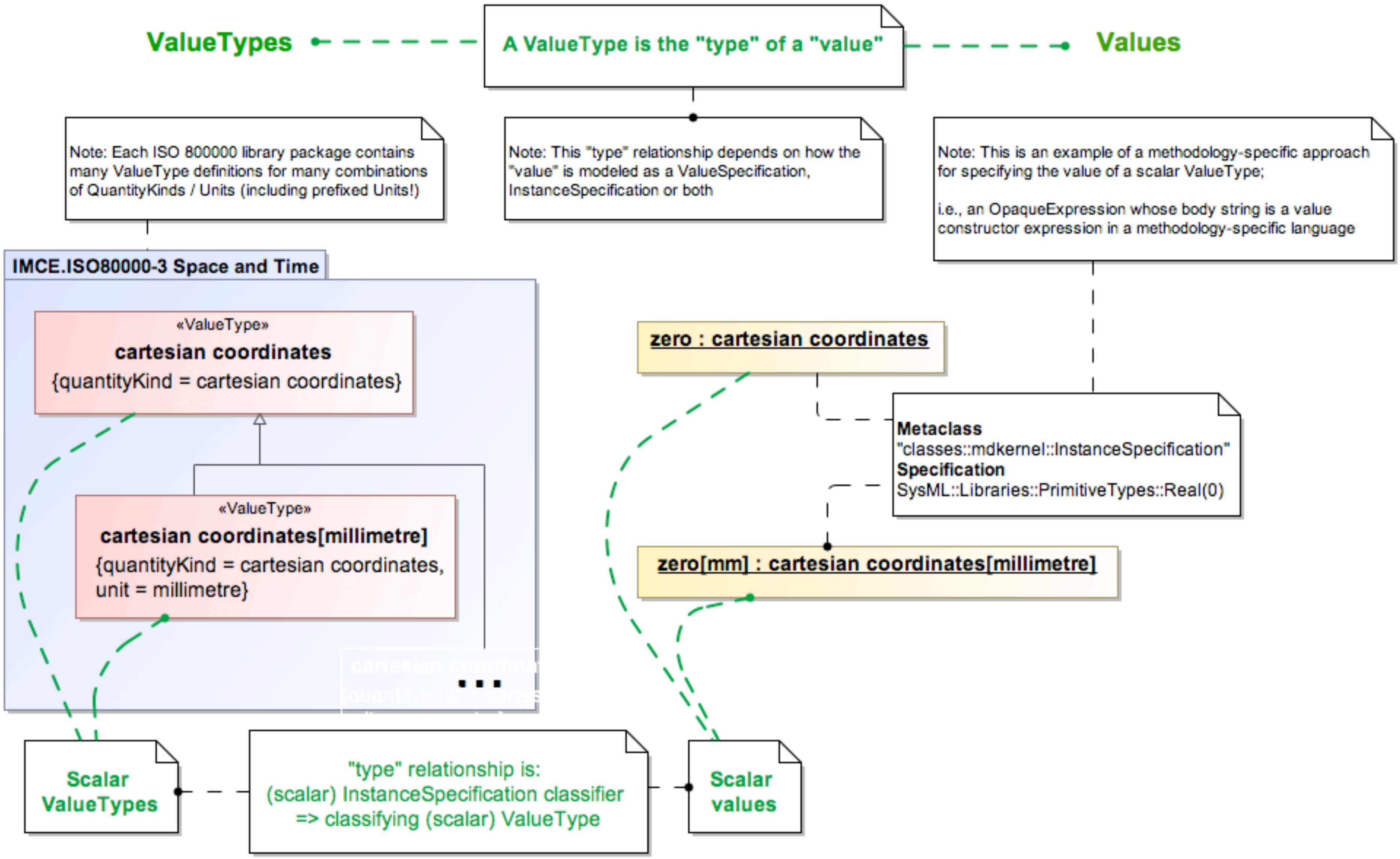
p1 : 2D Cartesian Position[mm]
x = 1
y = 1

p3 : 2D Cartesian Position[mm]
x = 2.5
y = 2

p2 : 2D Cartesian Position[mm]
x = 2
y = 0.5

See #55 for an alternative paradigm for modeling Values & ValueTypes

2) Modeling Values of Scalar ValueTypes

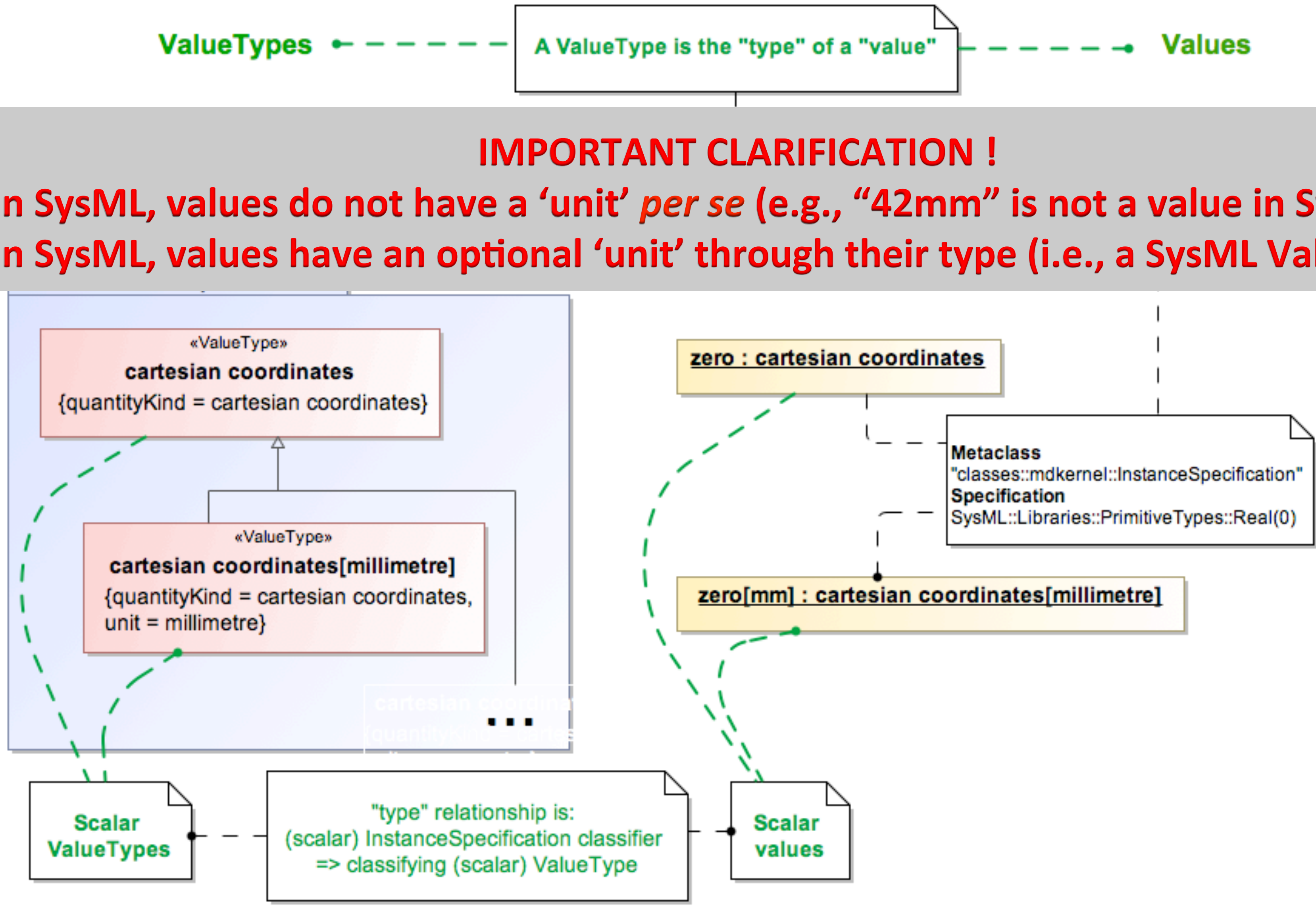


Modeling Values of Scalar ValueTypes

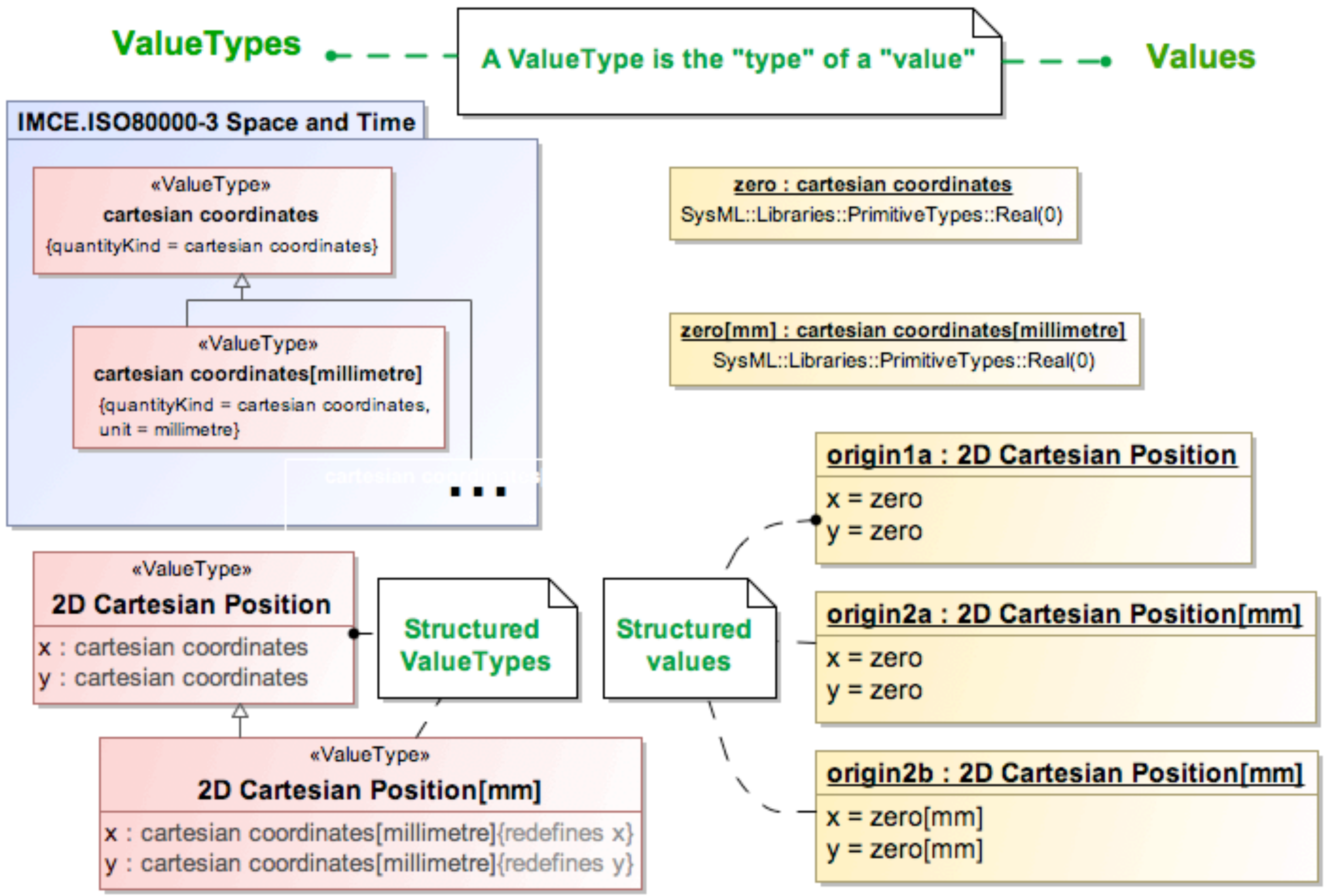
IMPORTANT CLARIFICATION !

In SysML, values do not have a 'unit' *per se* (e.g., "42mm" is not a value in SysML!)

In SysML, values have an optional 'unit' through their type (i.e., a SysML ValueType)

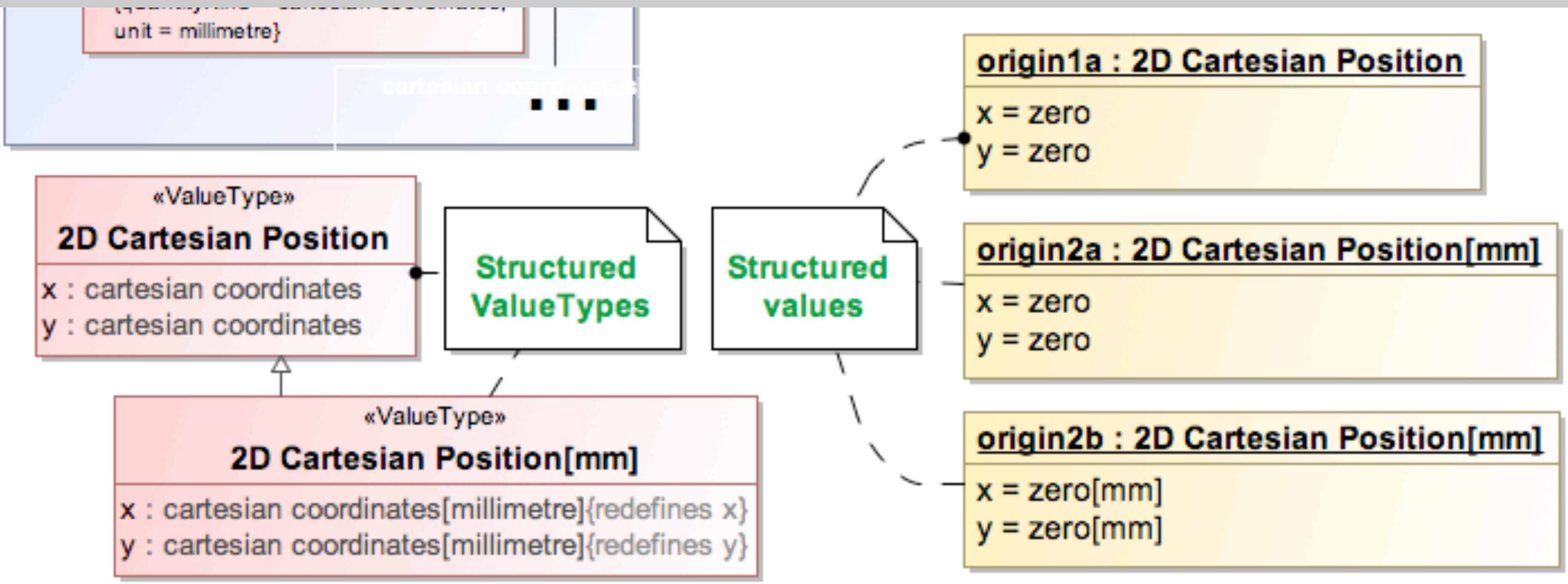


Modeling Values of Structured ValueTypes

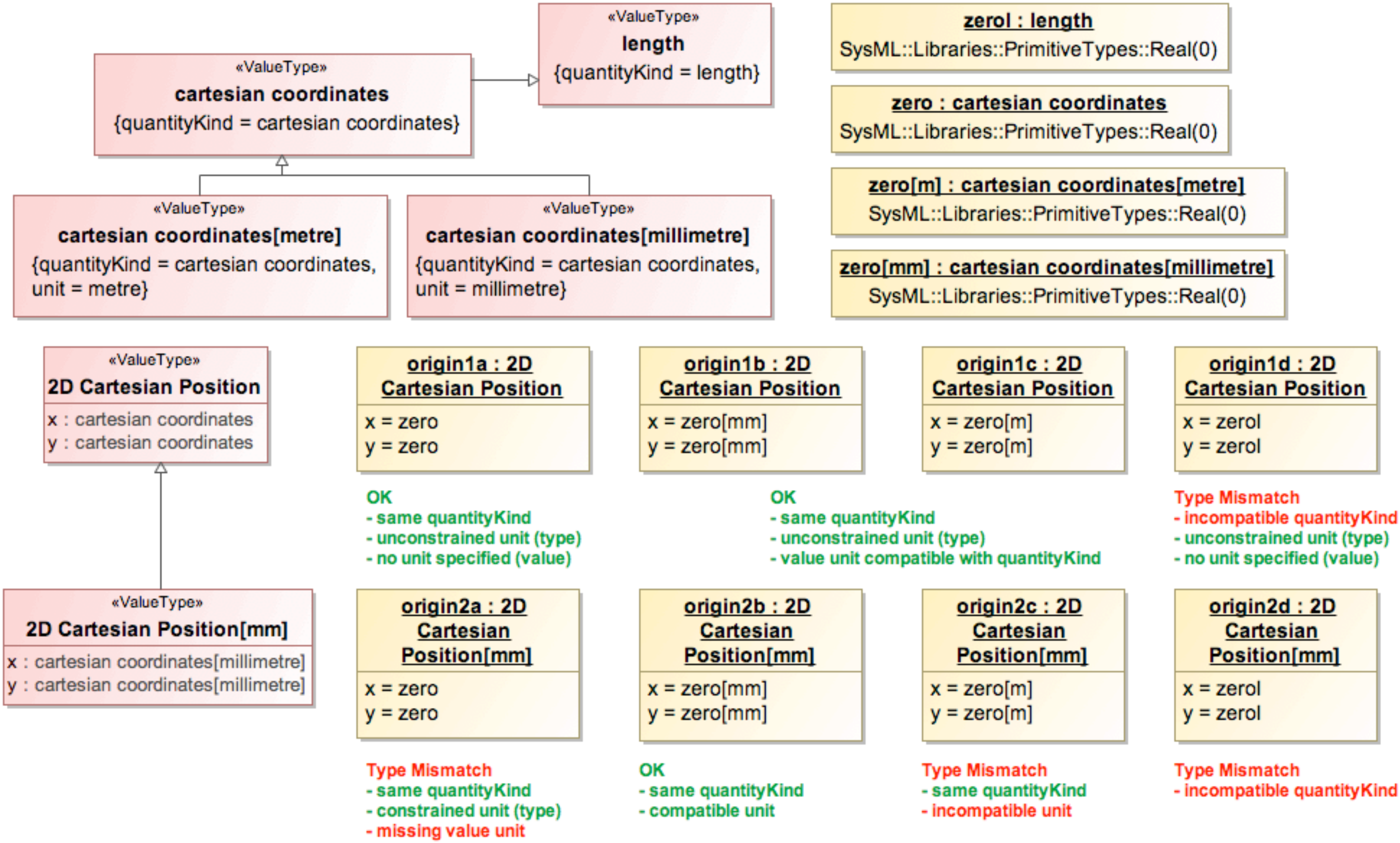


Modeling Values of Structured ValueTypes

IMPORTANT CLARIFICATION !
Although SysML values do not have a ‘unit’ per se,
Value / ValueType Conformance is essential
(otherwise a value has no meaning in SysML)



Type Checking: Value / ValueType Conformance



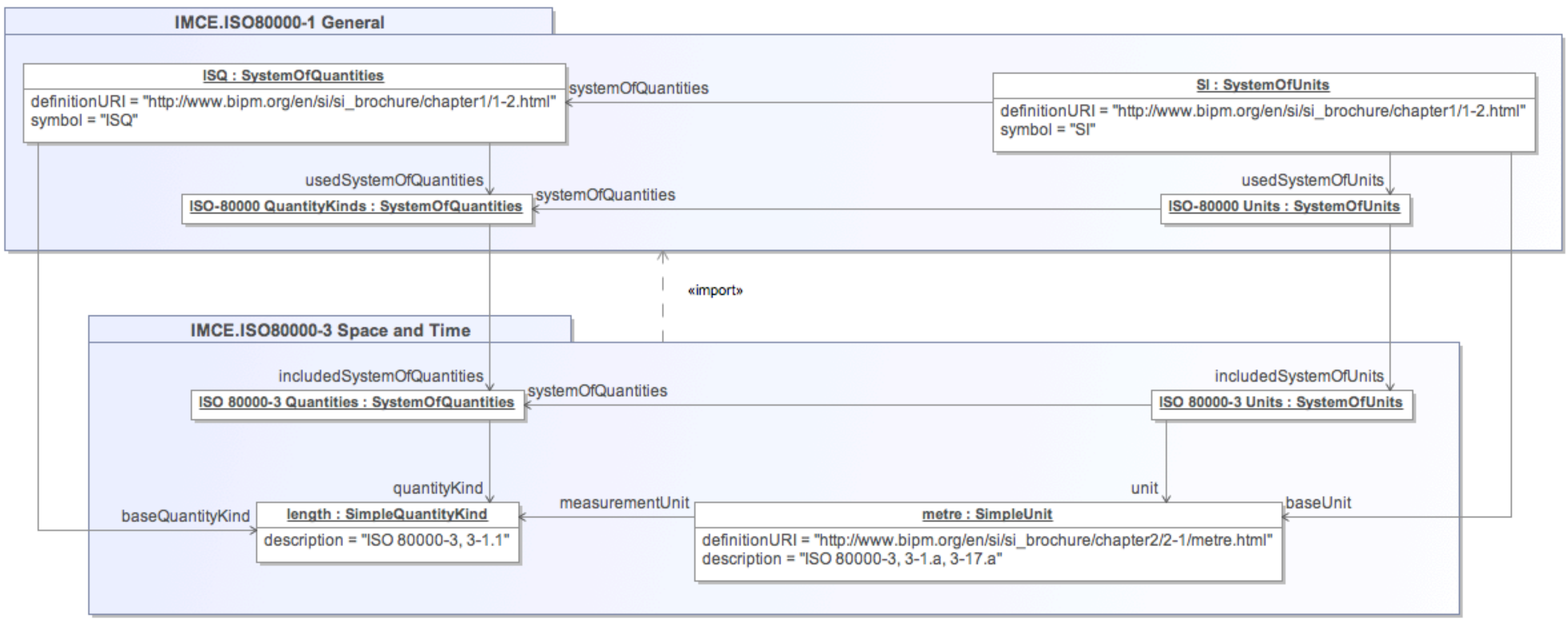
See #51 for practical considerations about modeling values

See #55 for a comparison between structured & monadic ValueType modeling

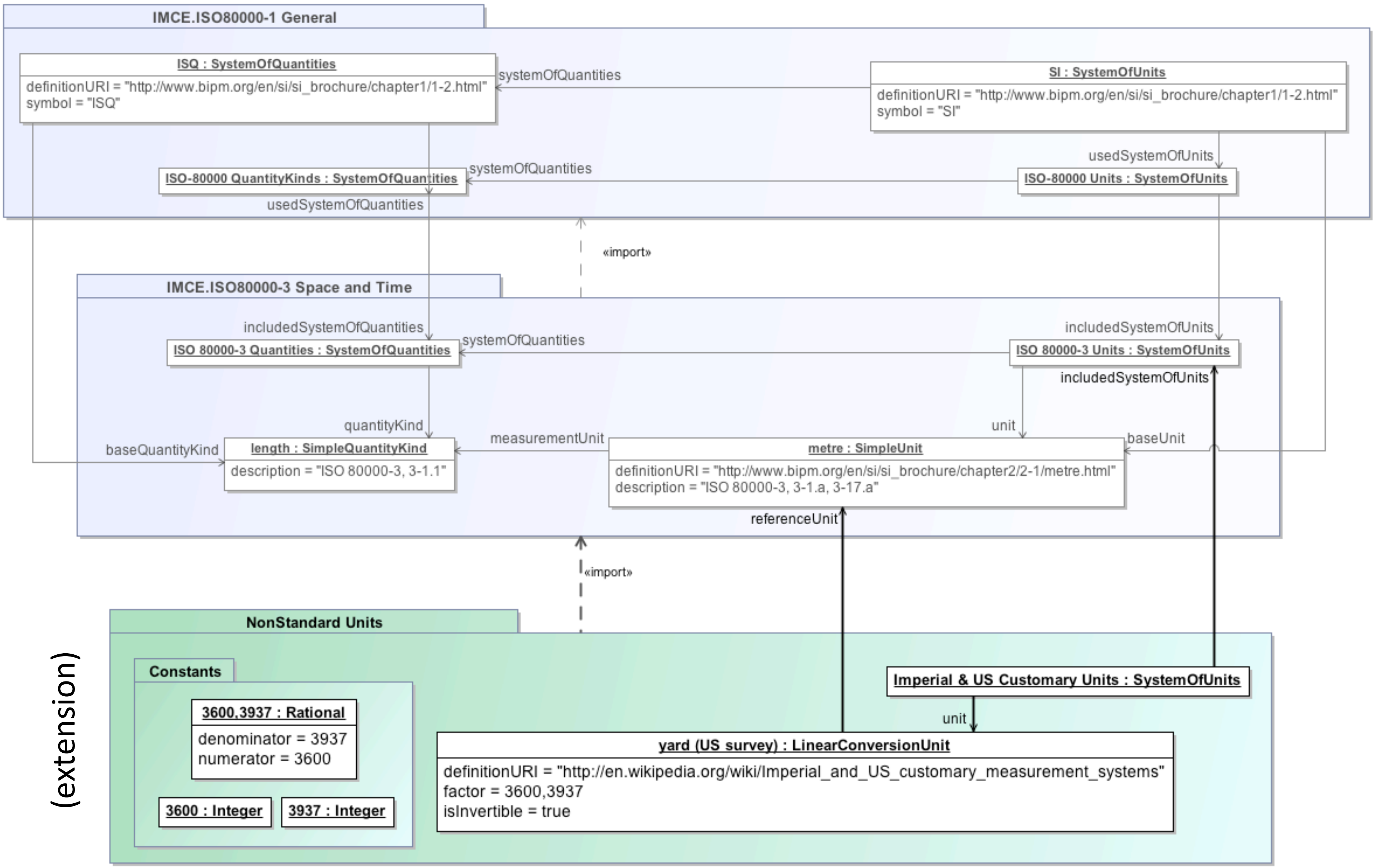
3) Extending the ISO-80000 Library

Goal: Define 'yard (US survey)' → Extend ISO 80000-3 (Space & Time)

(Identify the relevant ISO 80000 library)



Extending the ISO-80000 Library (cont)



4) Dimensional Analysis with QUDV

- What is Dimensional Analysis useful for?
  - Checking consistency of derivations between quantities and corresponding units
  - Calculating the dimension of derived quantities as a product of the dimensions of other quantities
  - Checking the derivation of quantities of dimension one

    Converting *scale-dependent models* (with dimensioned quantities) to equivalent *scale-independent models* (with quantities of dimension 1)
- For more information, see:
  - http://www.bipm.org/en/si/si_brochure/chapter1/1-3.html
  - http://en.wikipedia.org/wiki/Dimensional_analysis

Dimensional Analysis Example: Checking the derivation of quantities of dimension one

- Reynolds number is well-known ‘dimensionless’ quantity used for developing scale models of fluid dynamic systems
  - If a model is expressed in terms of ‘dimensionless’ quantities only, then the model can be scaled
  - Scaled models are useful for simulating non-linear phenomena in the lab (e.g. supersonic flight, …)
- Use SysML 1.4 QUDV for dimensional analysis!

QUDV-based Dimensional Analysis: Reynolds number

- Typically, a dimensionless quantity has multiple formulations
- Redundancy is a *good thing* for dimensional analysis!
  - Check that all *equivalent* formulations have *equivalent* dimensions
- Reynolds number can be formulated in 2 ways:

Reynolds number (*Re*)
(ISO 80000-11:11-4.1)

This is a "dimensionless" quantity defined as:

Re = $\rho \cdot \upsilon \cdot l / \eta = \upsilon \cdot l / \nu$

ρ	= ISO 80000-4:4-2	mass density
υ	= ISO 80000-3:3-8-1	speed
l	= ISO 80000-3:3-1.1	length
η	= ISO 80000-4:4-23	dynamic viscosity
ν	= ISO 80000-4:4-24	kinematic viscosity

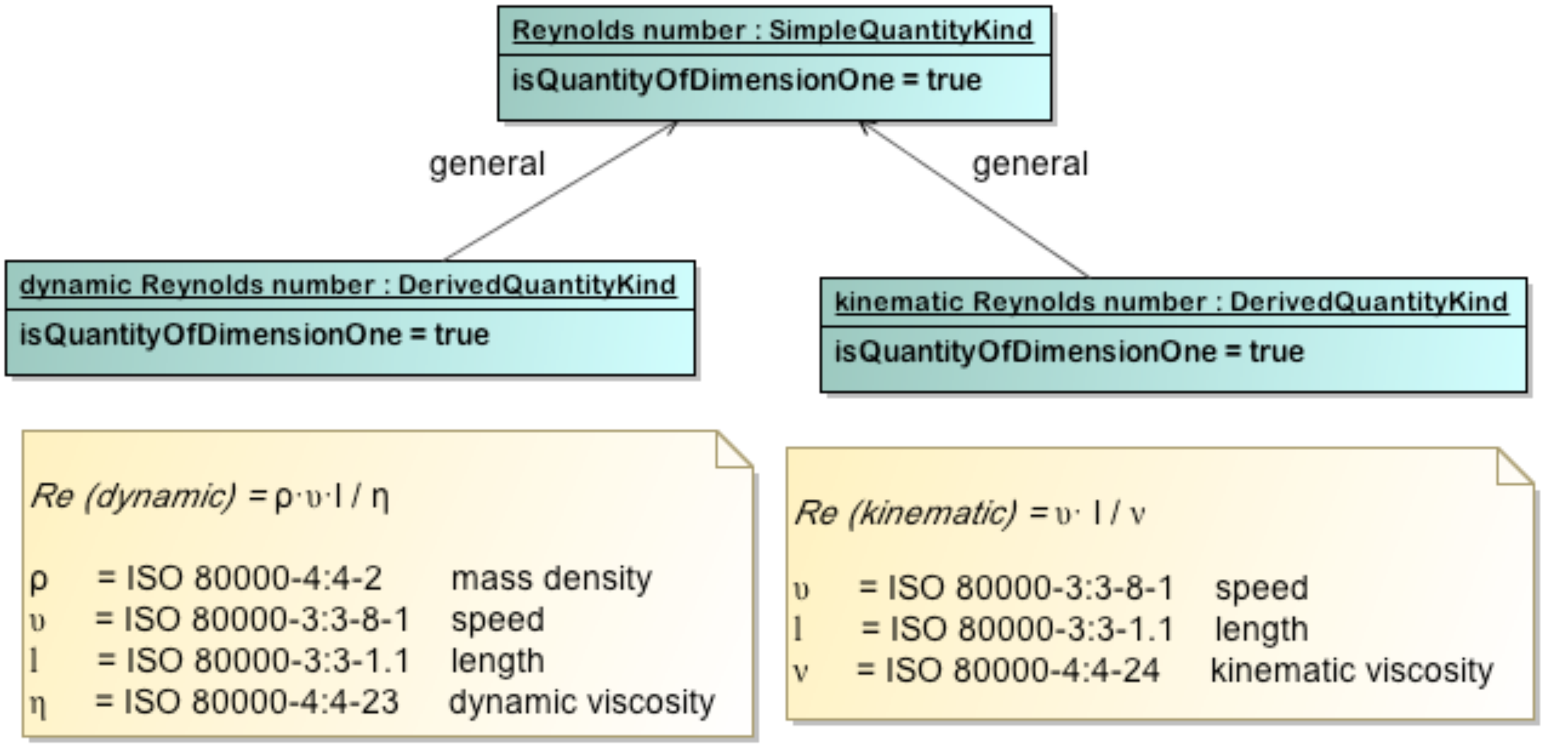
Re (dynamic) = $\rho \cdot \upsilon \cdot l / \eta$

ρ	= ISO 80000-4:4-2	mass density
υ	= ISO 80000-3:3-8-1	speed
l	= ISO 80000-3:3-1.1	length
η	= ISO 80000-4:4-23	dynamic viscosity

Re (kinematic) = $\upsilon \cdot l / \nu$

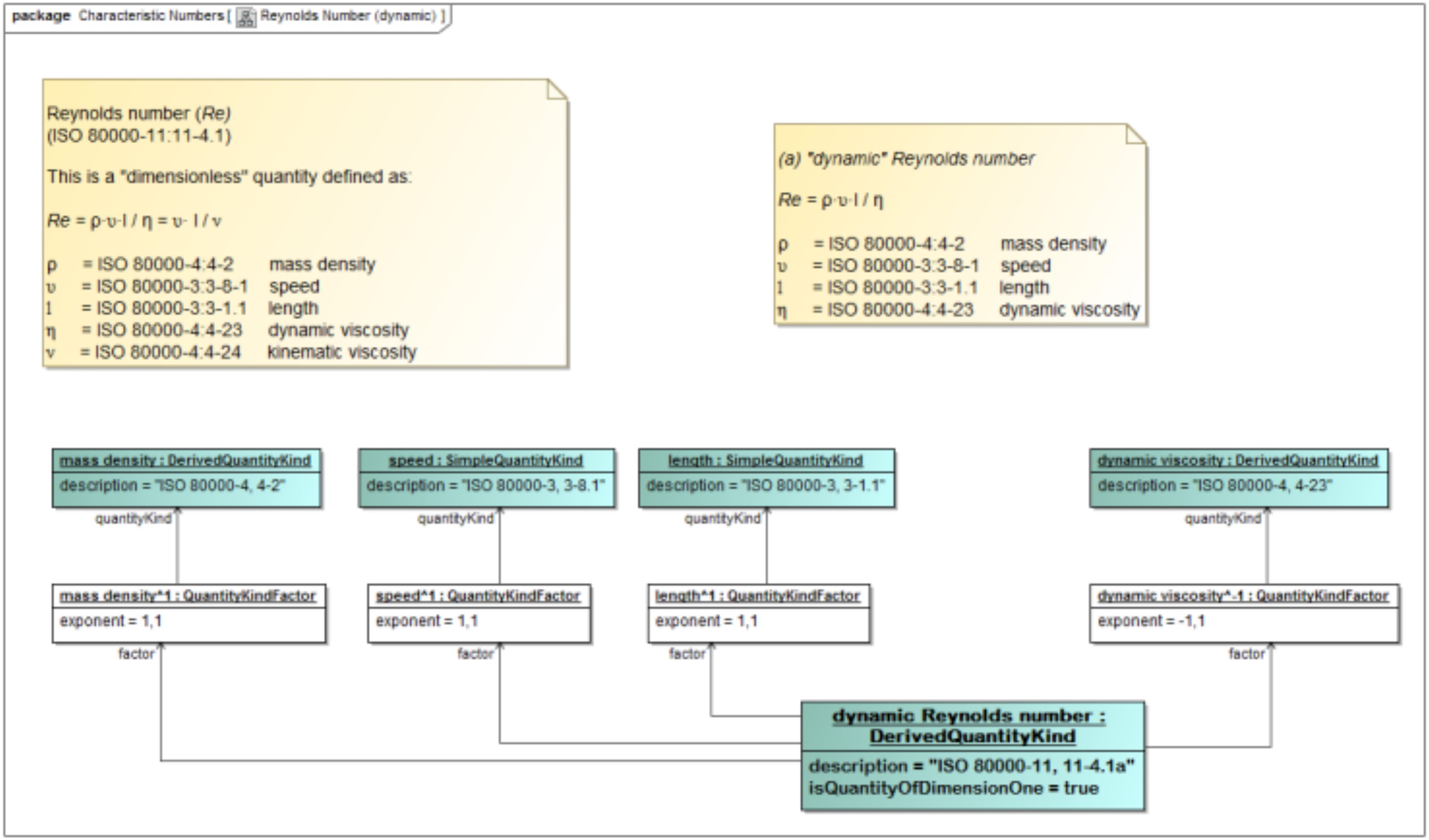
υ	= ISO 80000-3:3-8-1	speed
l	= ISO 80000-3:3-1.1	length
ν	= ISO 80000-4:4-24	kinematic viscosity

Redundant Modeling of Dimensionless Quantities

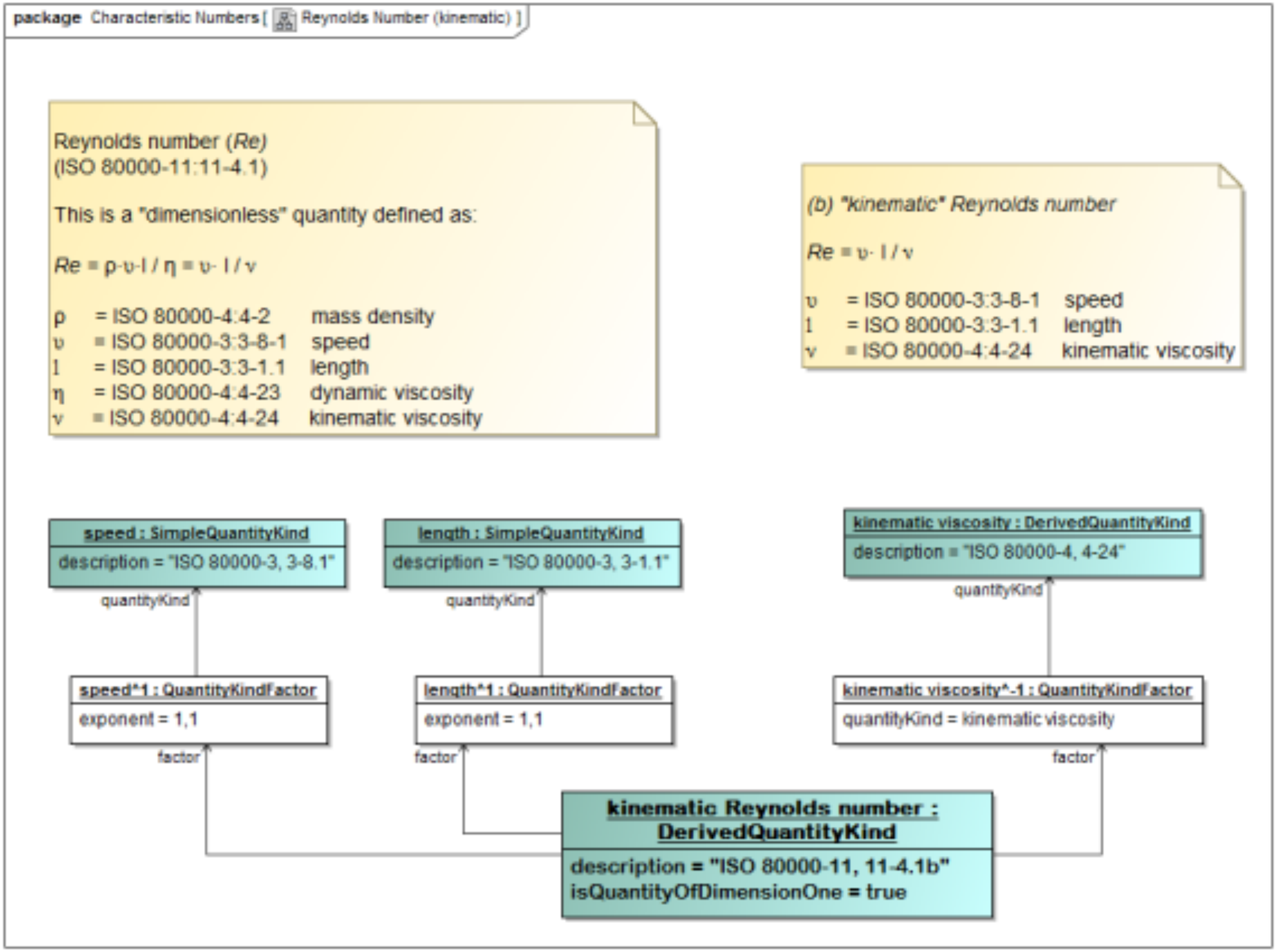


- QUDV support for VIM-based taxonomic relationships among quantities is essential!
  - dynamic & kinematic formulations ***must*** be equivalent
  - Therefore, they must have equivalent dimensions!

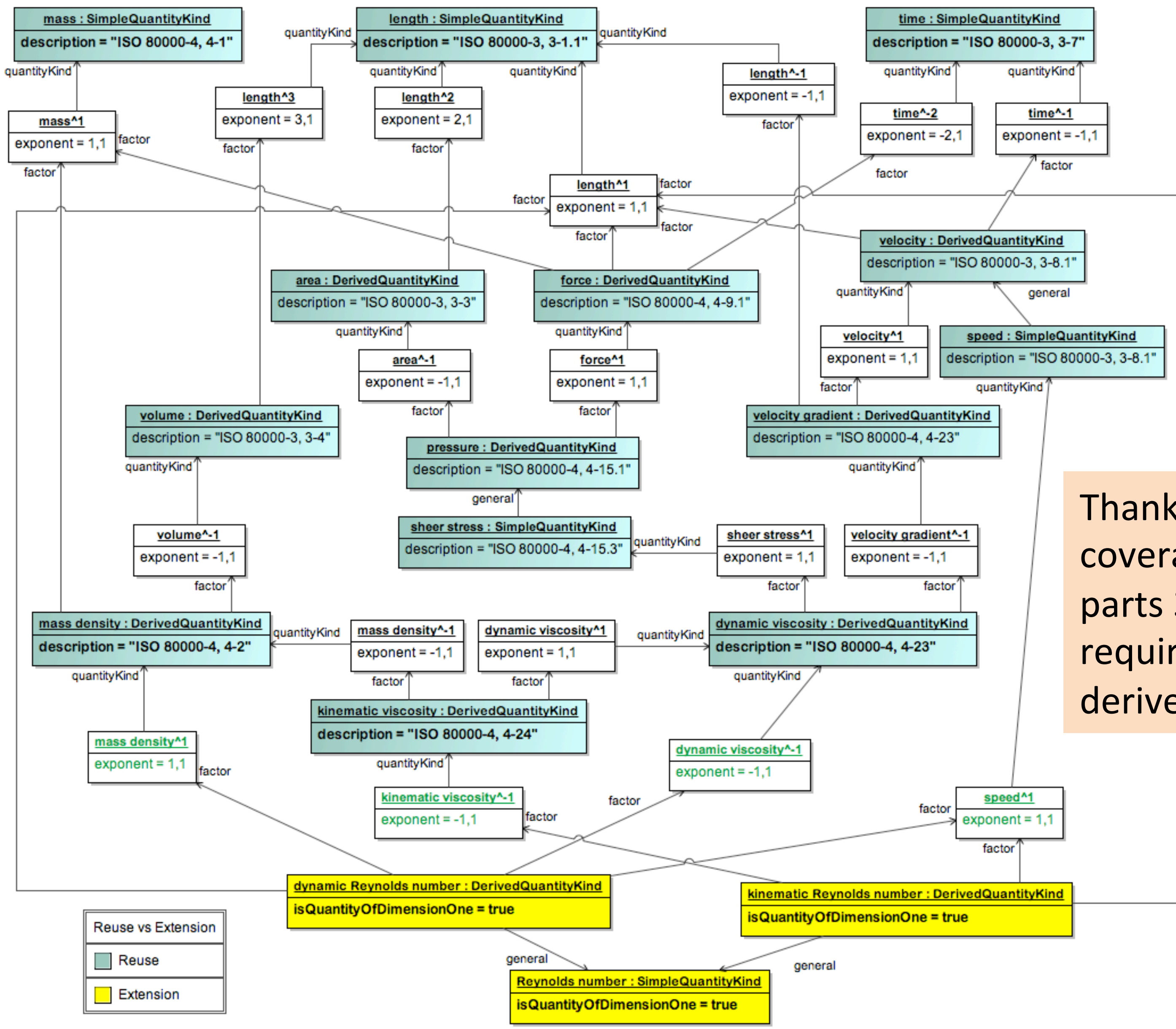
Elaborate the Redundant Formulations (1/2)



Elaborate the Redundant Formulations (2/2)

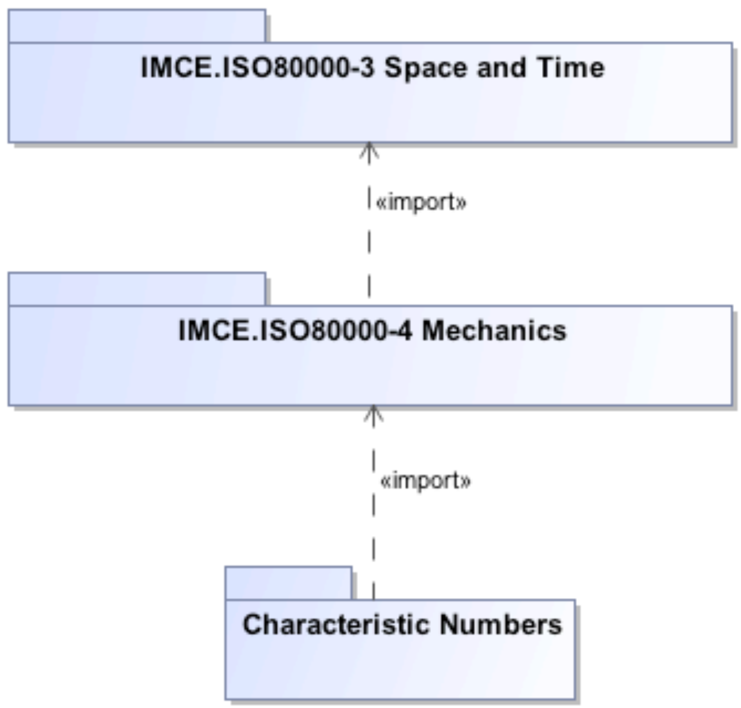


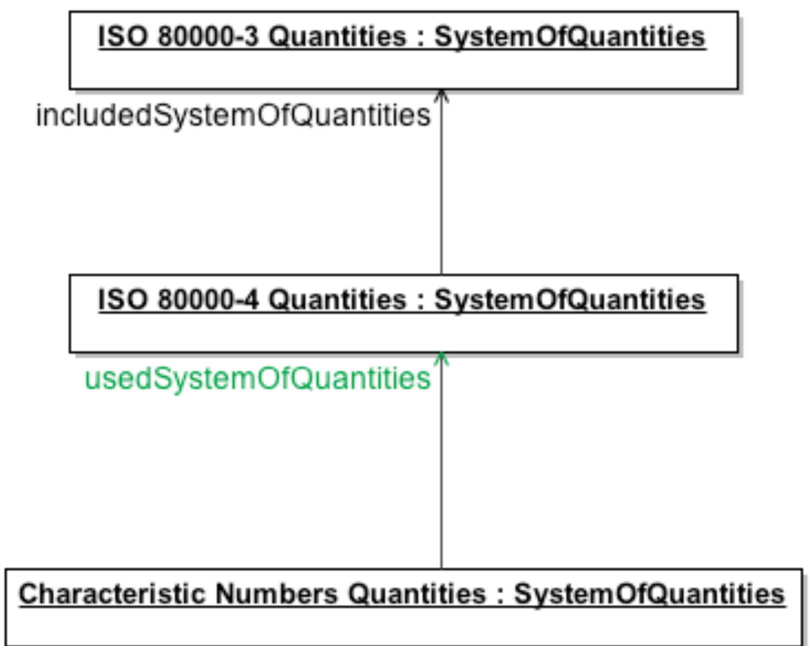
Reuse and/or Extend the ISO 80000 library to include derivations for all quantities relevant to the analysis



Thanks to comprehensive coverage of ISO 80000 parts 3,4,5,6, this example requires defining only 3 derived quantity kinds!

Dimensional Analysis in context of a System of Quantities

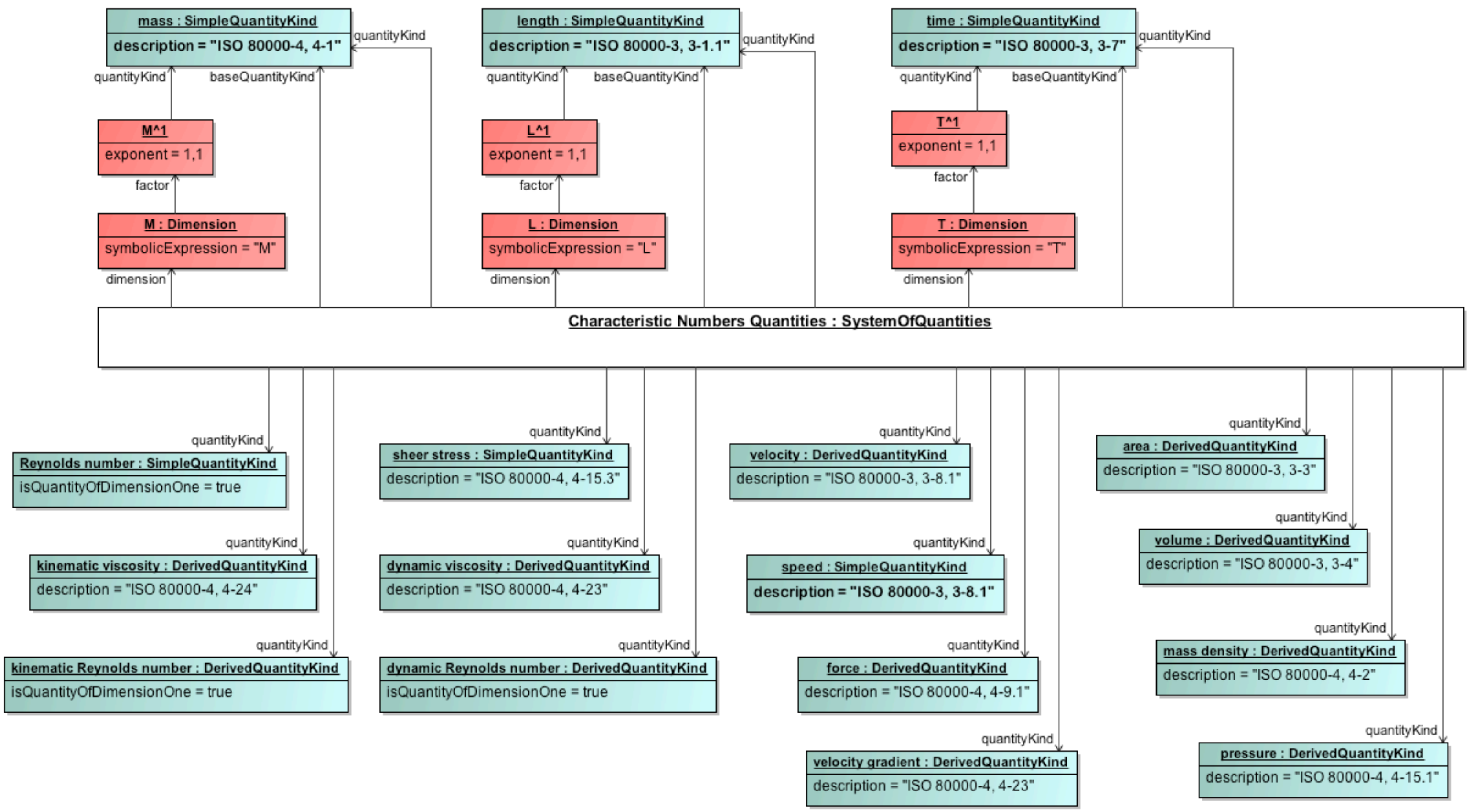




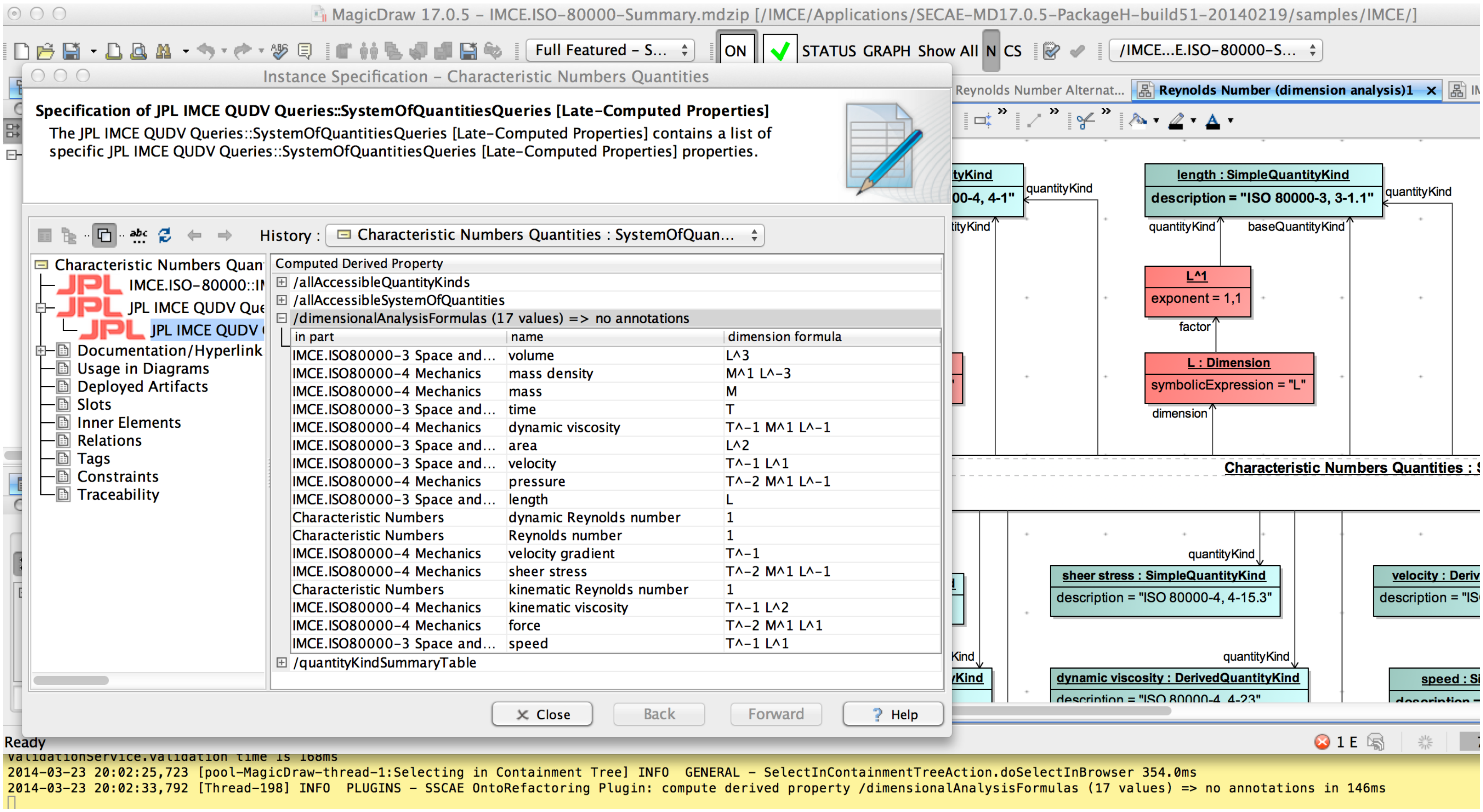
- Expressing a derived quantity as a product of the powers of base quantities requires specifying the context for this analysis, i.e., a SystemOfQuantities
  - In this example, context = 'Characteristic Numbers Quantities'

Context for the Dimension Analysis of Reynolds number redundant formulations

- Dimension Analysis requires choosing a set of base quantities
  - For this example: Mass (M), Length (L), Time (T)



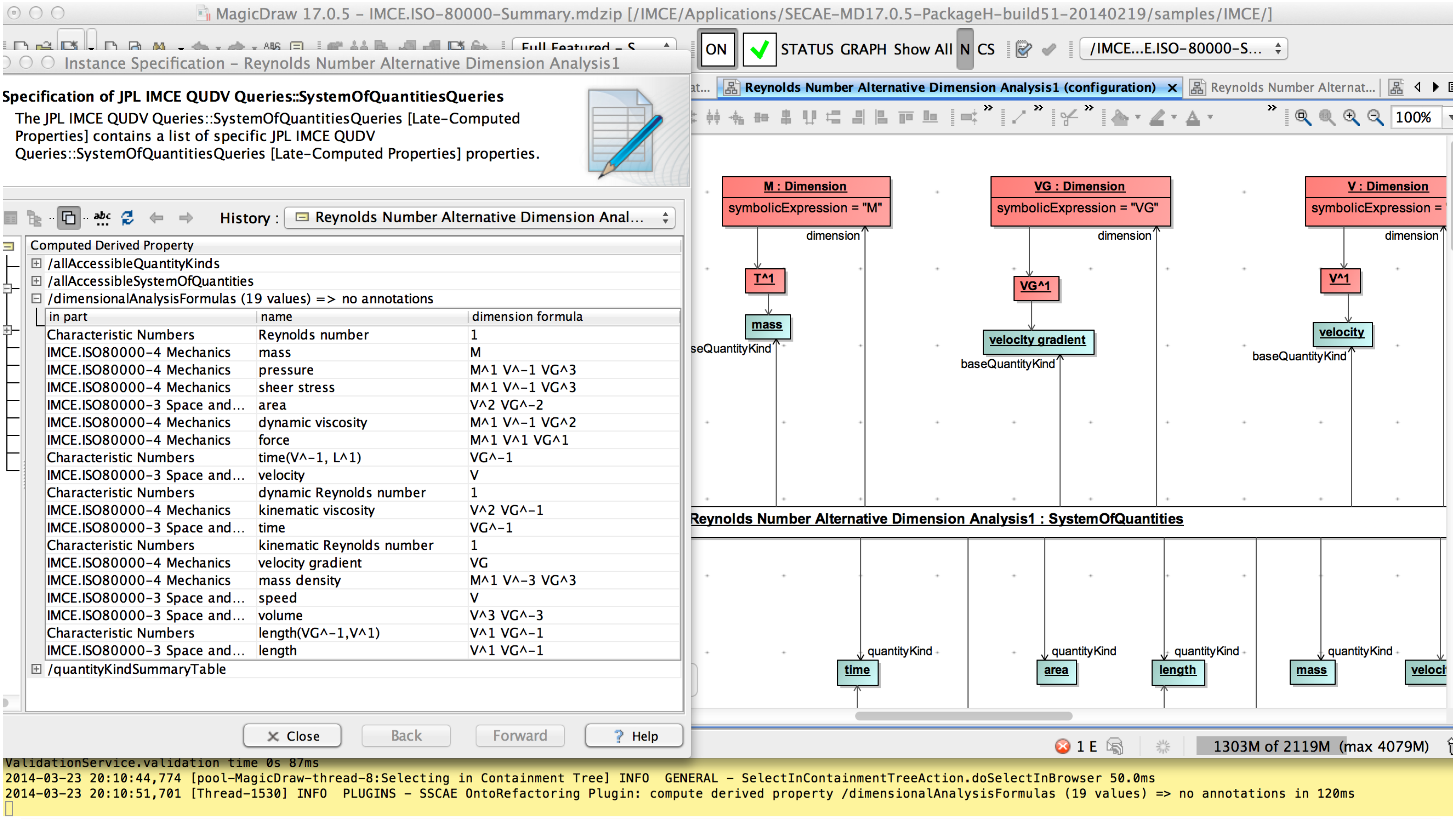
Calculating the Dimension of all Derived Quantities



Example from JPL's SysML 1.4 QUDV tooling for MagicDraw 17.0.5 SP1:

Dimensional Analysis is calculated for all quantity kinds in scope of a selected SystemOfQuantities ('Characteristic Numbers Quantities' in this example) *on demand!)*

Alternative Dimensional Analysis Formulation



"Base Quantities" is intrinsically a choice made in a System Of Quantities. This example uses a non-standard set of base quantities: mass (M), Velocity Gradient (VG) and Velocity (V)

The rules for Dimensional Analysis are generic and can be applied to *any* System of Quantities

Summary of SysML 1.4 QUDV & ISO 80000 Library

- SysML 1.4 represents a significant leap in capability
  - QUDV's expressiveness is adequate for *all* normative definitions in ISO/IEC 80000 parts 3,4,5,6
  - QUDV's semantics is adequate for automated, computer-assisted Dimensional Analysis
- In practice, working with QUDV & ISO 80000 emphasizes the need for good tool support for instance & value specification modeling!
  - SysML 1.4 ISO 80000 (as of December 2013) is significantly affected by SysML issue #16263:
    - Slots for class-owned association end properties are duplicated (links of such associations & instance specifications of such classes)
    - Slot duplication leads to circular references that make the ISO 80000 monolithic
    - Slot duplication is a significant source of inconsistencies and ambiguities!
  - JPL IMCE ISO 80000 uses the suggested approach in SysML issue #16263:
    - All JPL IMCE QUDV associations own both association end properties
    - No slot duplication avoids problems with inconsistencies & ambiguities and enables modularity!
  - Current OMG paradigms for specifying semantics (e.g. OCL queries, invariants, derived properties) are oblivious to the *computational complexity* of such semantics
    - Dimensional Analysis is *not cheap!* It is a "whole model" graph analysis
- The ISO/IEC 80000 library is incomplete and needs careful peer review!
  - Constructive review feedback and expert contributions welcome!

Backup Material

Historical Review of SysML's Concepts of Unit & QuantityKind

- SysML 0.98 (ad/2005-11-01)
- SysML Adopted (ptc/2006-05-04)
- SysML 1.0 (formal/2007-09-01)
- SysML 1.1 (formal/2008-11-02)
- SysML 1.2 (formal/2008-11-02)
- SysML 1.3 (formal/2011-08-10)
- SysML 1.4 (beta) (ptc/2013-11-07)

SysML 0.98
(ad/2005-11-01)

SysML Adopted
(ptc/2006-05-04)

Before SysML 1.0

Systems Modeling Language (SysML) Specification

OMG document: ad/2005-11-01

version 0.98
DRAFT

OMG Systems Modeling Language (OMG SysML™) Specification

Final Adopted Specification
ptc/06-05-04

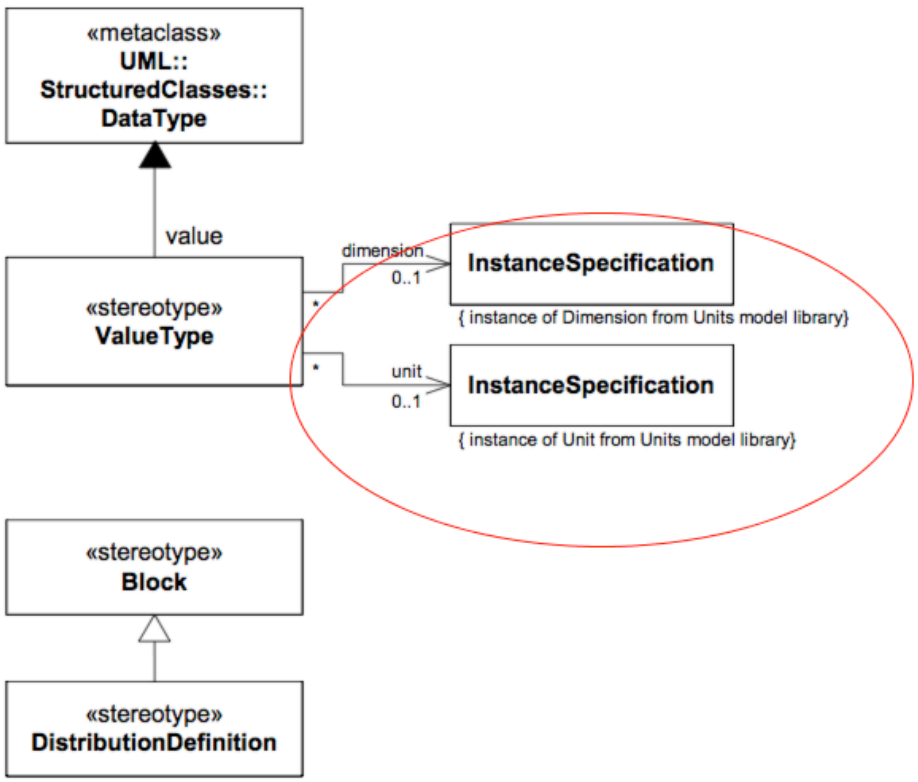


Figure 8-4. Abstract syntax extensions for SysML value types.

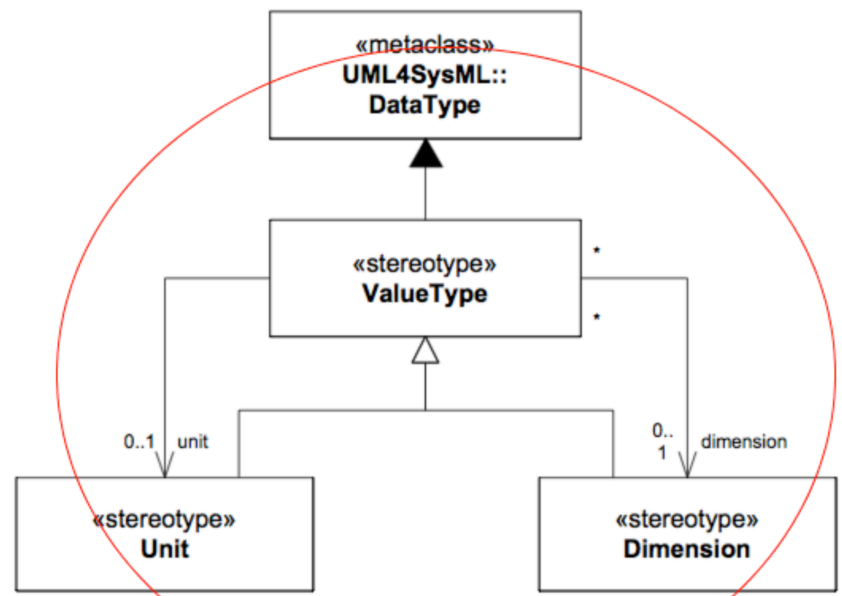


Figure 8.4 - Abstract syntax extensions for SysML value types

SysML 0.98
(ad/2005-11-01)

SysML Adopted
(ptc/2006-05-04)

SysML 1.0
(formal/2007-09-01)

SysML 1.1
(formal/2008-11-02)

SysML 1.0 & 1.1

OMG Systems Modeling Language (OMG SysML™), V1.0

OMG Available Specification

OMG Document Number: formal/2007-09-01
Standard document URL: http://www.omg.org/spec/SysML/1.0/PDF
Associated Schema Files:
http://www.omg.org/spec/SysML/20070901/SysML-profile.xmi
http://www.omg.org/spec/SysML/20070901/Activities-model.xmi
http://www.omg.org/spec/SysML/20070901/Blocks-model.xmi
http://www.omg.org/spec/SysML/20070901/UML4SysML-metamodel.xmi

---

OMG Systems Modeling Language (OMG SysML™)

Version 1.1

OMG Document Number: formal/2008-11-02
Standard document URL: http://www.omg.org/spec/SysML/1.1
Associated Schema File(s)*: http://www.omg.org/spec/SysML/20080501
http://www.omg.org/spec/SysML/20080501/SysML-profile.xmi
http://www.omg.org/spec/SysML/20080501/Activities-model.xmi
http://www.omg.org/spec/SysML/20080501/Blocks-model.xmi
http://www.omg.org/spec/SysML/20080501/UML4SysML-metamodel.xmi

---

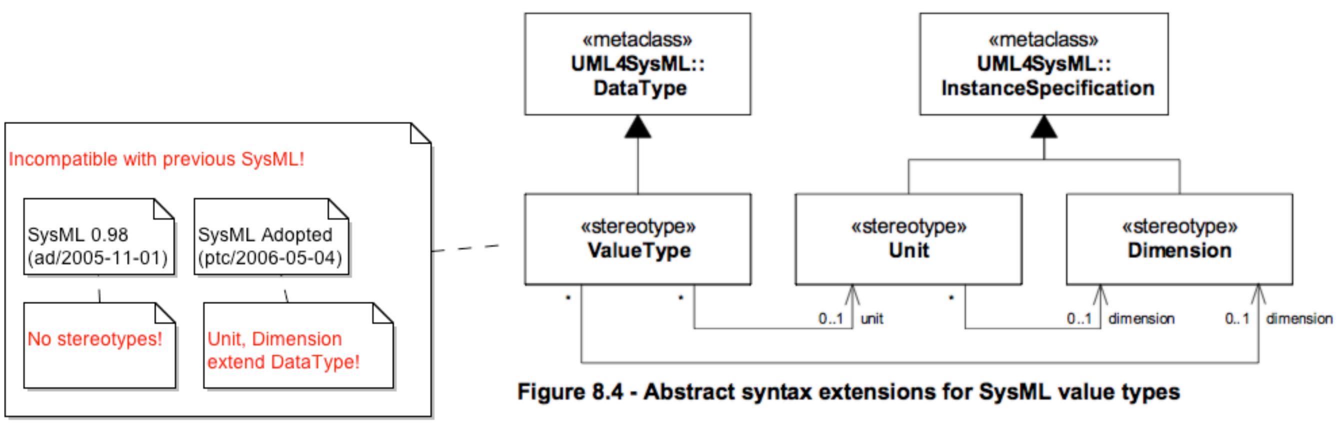


Figure 8.4 - Abstract syntax extensions for SysML value types

SysML 0.98
(ad/2005-11-01)

SysML Adopted
(ptc/2006-05-04)

SysML 1.0
(formal/2007-09-01)

SysML 1.1
(formal/2008-11-02)

SysML 1.2
(formal/2008-11-02)

SysML 1.3
(formal/2011-08-10)

SysML 1.2 & 1.3

OMG Systems Modeling Language (OMG SysML™)

Version 1.2
without change bars

OMG Document Number: formal/2010-06-01
Standard Specification URL: http://www.omg.org/spec/SysML/1.2/
Associated Schema File(s)*: http://www.omg.org/spec/SysML/20100301
http://www.omg.org/spec/SysML/20100301/SysML-profile.uml
http://www.omg.org/spec/SysML/20100301/UML4SysML-metamodel.uml
http://www.omg.org/spec/SysML/20100301/Activities-model.xmi
http://www.omg.org/spec/SysML/20100301/Blocks-model.xmi

OMG Systems Modeling Language (OMG SysML™)

Version 1.3

Normative reference: http://www.omg.org/spec/SysML/1.3/
(Original source document: ptc/2011-08-10)
Machine consumable files: http://www.omg.org/spec/SysML/20120401
Normative:
http://www.omg.org/spec/SysML/20120401/SysML.xmi
(source XMI document: ptc/2012-04-07)
Non-normative:
http://www.omg.org/spec/SysML/20120401/ISO-80000-1-QUDV.xmii
http://www.omg.org/spec/SysML/20120401/ISO-80000-1-SysML.xmi
http://www.omg.org/spec/SysML/20120401/ QUDV.xmi
(source XMI document: ptc/2012-04-08)

SysML 1.2
(formal/2008-11-02)

New conceptual model of
Quantities, Units, Dimensions & Values (QUDV)

Renamed Dimension => QuantityKind
(Dimension is a concept that depends on a SystemOfQuantities; see Dimension Analysis)

No official library of quantities & units!

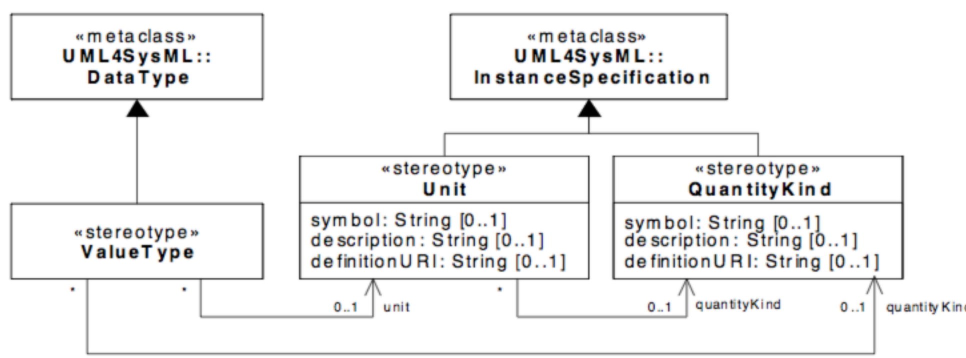


Figure 8.4 - Abstract syntax extensions for SysML value types

SysML 1.3
(formal/2011-08-10)

SysML 1.3 QUDV is identical to SysML 1.2 QUDV except for Quantity, which was incorrectly specified

Provides *2* non-normative libraries:

ISO-80000-1-SysML
(based on Unit/QuantityKind stereotypes)

ISO-80000-1-QUDV
(based on QUDV instances)

SysML 0.98
(ad/2005-11-01)

SysML Adopted
(ptc/2006-05-04)

SysML 1.0
(formal/2007-09-01)

SysML 1.1
(formal/2008-11-02)

SysML 1.2
(formal/2008-11-02)

SysML 1.3
(formal/2011-08-10)

SysML 1.4 (beta)
(ptc/2013-11-07)

SysML 1.4

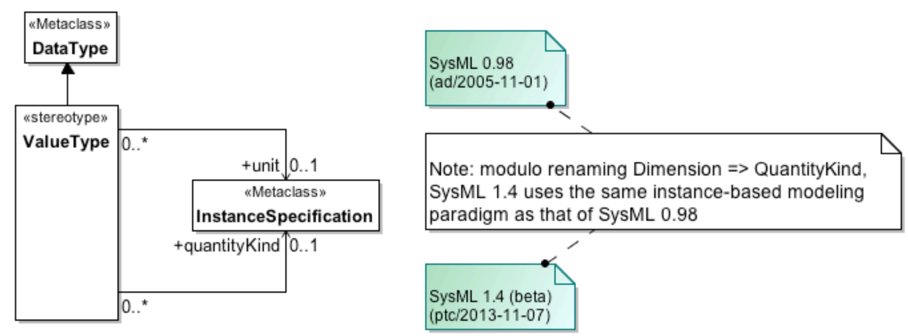
OMG Systems Modeling Language (OMG SysML™)

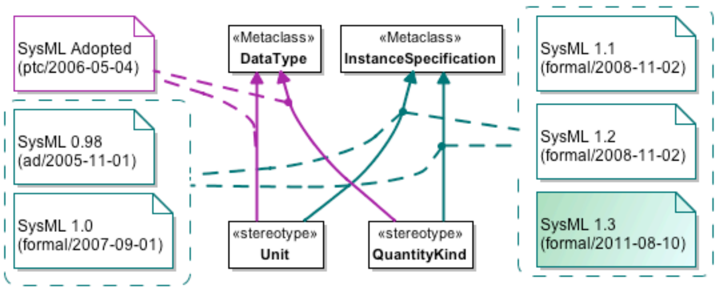
Version 1.4
(Beta Specification of the SysML 1.4 RTF, without change bars)

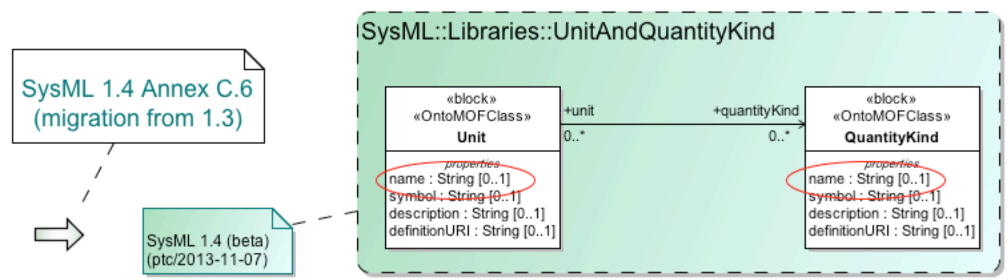
OMG Document Number: ptc/2013-11-07
Normative Reference: http://www.omg.org/spec/SysML/1.4/
Machine consumable files: http://www.omg.org/spec/SysML/20131201
Normative:
http://www.omg.org/spec/SysML/20131201/SysML.xmi
(source XMI document: ptc/2013-11-09)
Non-normative:
http://www.omg.org/spec/SysML/20131201/SysMLDI.xmii
http://www.omg.org/spec/SysML/20131201/QUDV.xmii
http://www.omg.org/spec/SysML/20131201/ISO80000.xmi
(source XMI document: ptc/2013-11-10)

SysML 1.4
- Eliminates the **duplicate ISO-80000 libraries** (SysML & QUDV)
- Eliminates the **SysML stereotypes Unit & QuantityKind** (QUDV is more expressive)
- Provides **complete coverage of ISO 80000 parts 3,4,5,6**
- Provides **partial coverage of ISO 80000 parts 7,9,10,13**
- **No coverage yet for ISO 80000 parts 8,11,12**
- **Migration from SysML 1.3 Unit/QuantityKind**

JPL's IMCE SysML 1.4 QUDV & ISO-80000 library
- **Modular, extensible organization** (impossible with OMG SysML 1.4 beta)







Technical Issues

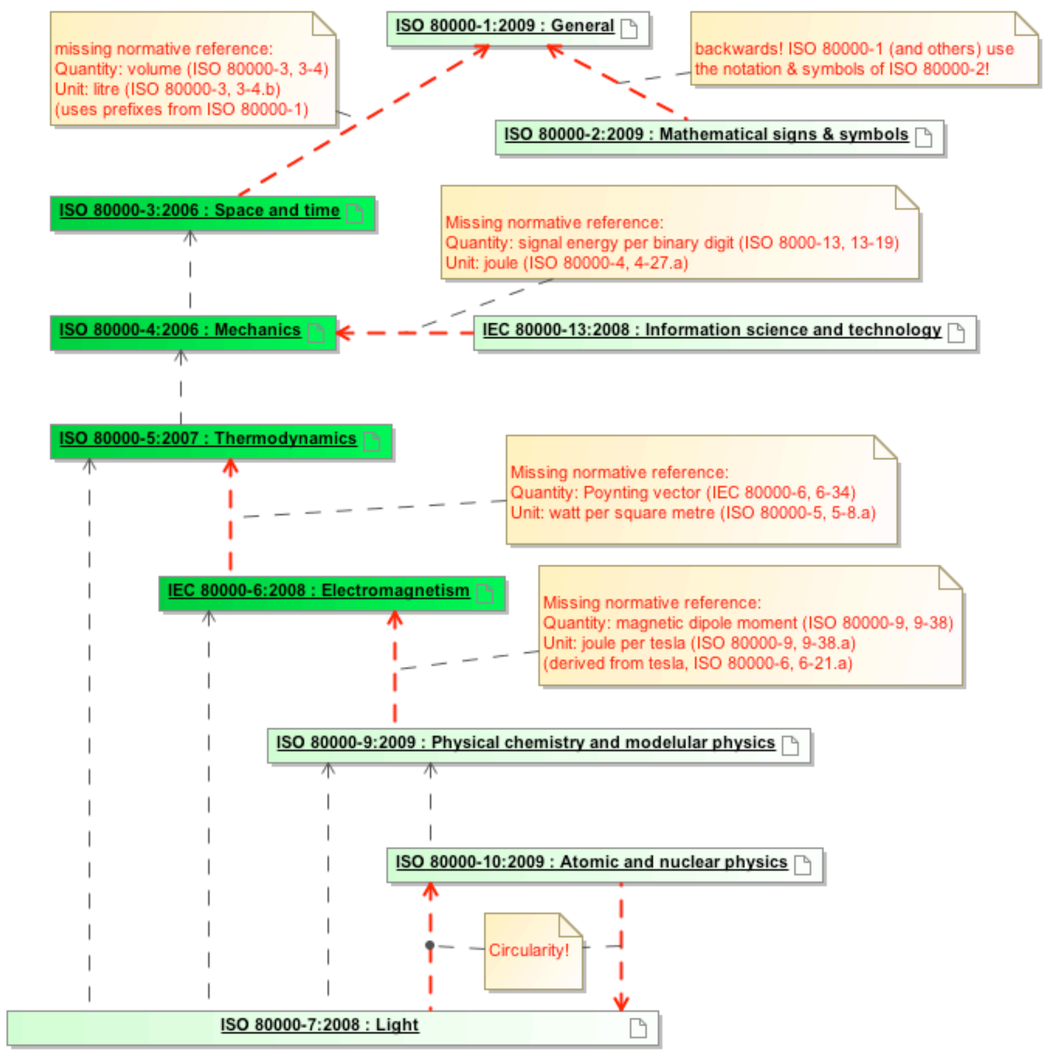
- Problems in the ISO/IEC 80000 standards
- Modularity & Extensibility
  - Onto QUDV
- Metrology Caveats

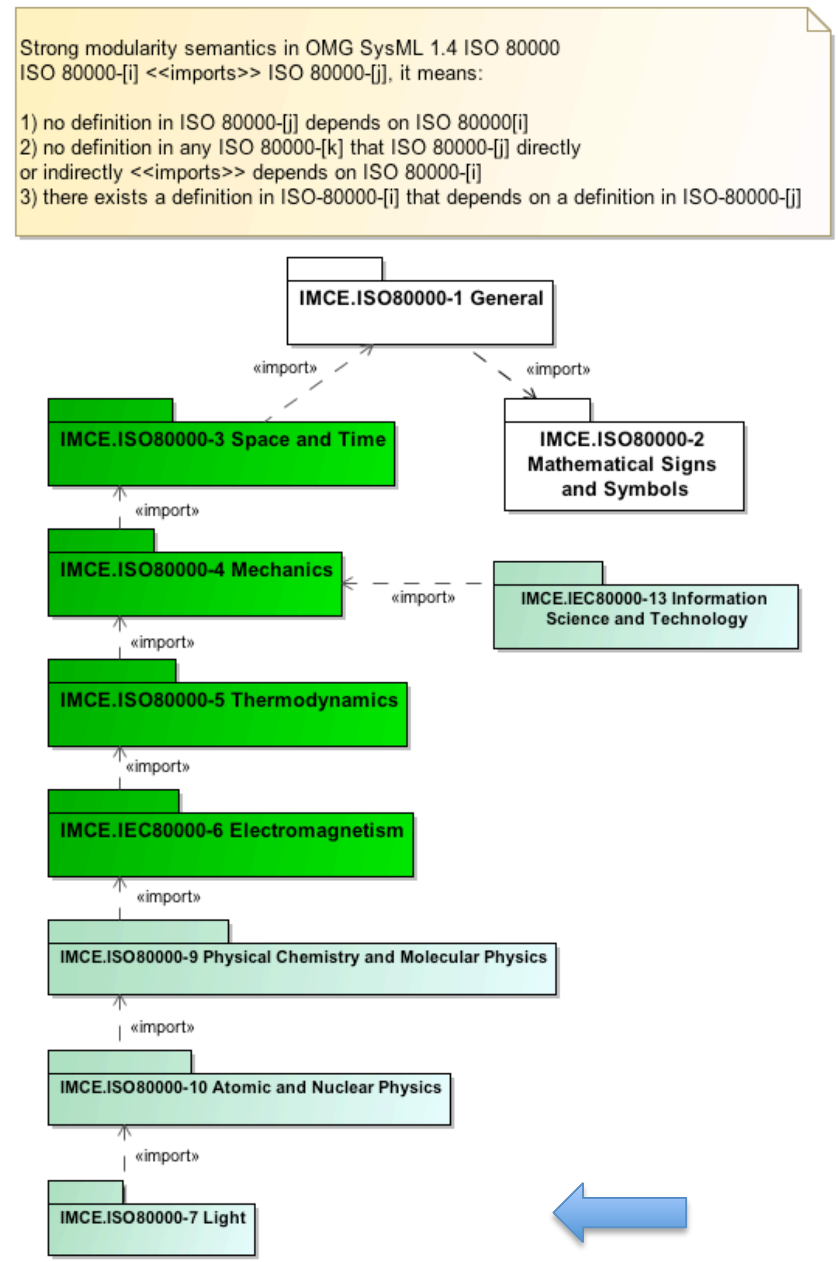
Modularity & Extensibility

SysML issue #16263 (May 25, 2011)

Associations in SysML should be able to own their ends. Otherwise modelers can't add an association between blocks in model libraries they do not have permission to modify. They also cannot create association that are non-navigable in both directions, which might be useful for directing flows across them into flows contained by the association as a block.

ISO/IEC 80000 Normative References
– vs. –
Semantic Dependencies & Modularity





References about Metrology

- Vocabulary of International Metrology (VIM)
  SysML 1.4 QUDV is aligned with VIM 3rd edition
  http://www.bipm.org/utils/common/documents/jcgm/JCGM_200_2012.pdf

- The Rationale for VIM 3 (scientific papers)
  http://www.bipm.org/en/publications/guides/rationale_vim3.html

- The Rationale for SysML 1.4 QUDV (subset of above)
  ISO terminological analysis of the VIM3 concepts 'quantity' and 'kind-of-quantity', Dybkaer R., Metrologia, 47(3), 2010, 127-143
  http://dx.doi.org/10.1088/0026-1394/47/3/003

  This paper is a small excerpt from Dybkaer's thesis:

  An ontology on Property for Physical, Chemical and Biological Systems, Dybkaer R., Copenhagen University Hospital, Denmark, 2009
  http://ontology.iupac.org/ontology.pdf

Excerpt from: “ISO terminological analysis of the VIM3 concepts ‘quantity’ and ‘kind-of-quantity’”

See #46 for references

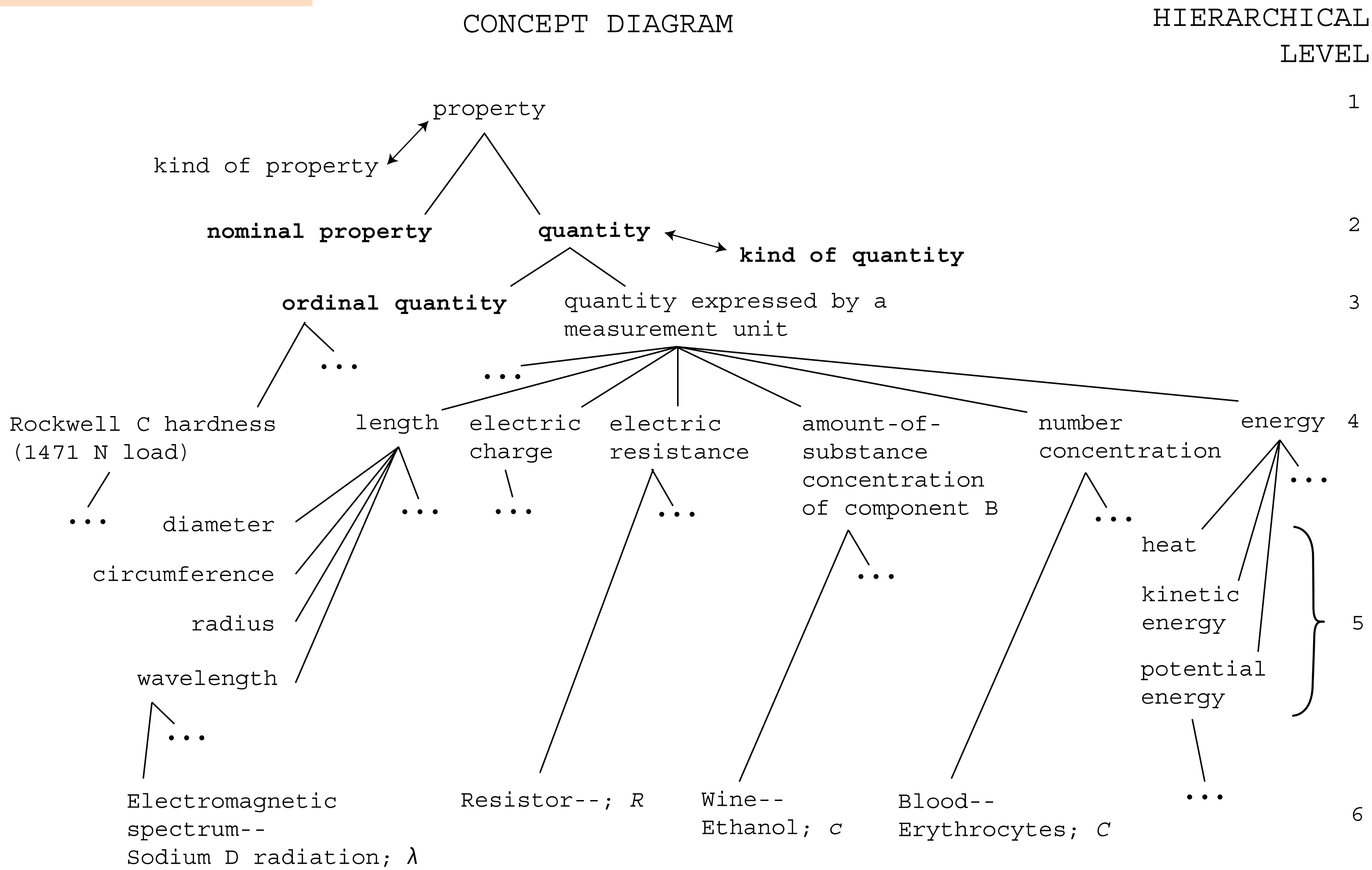
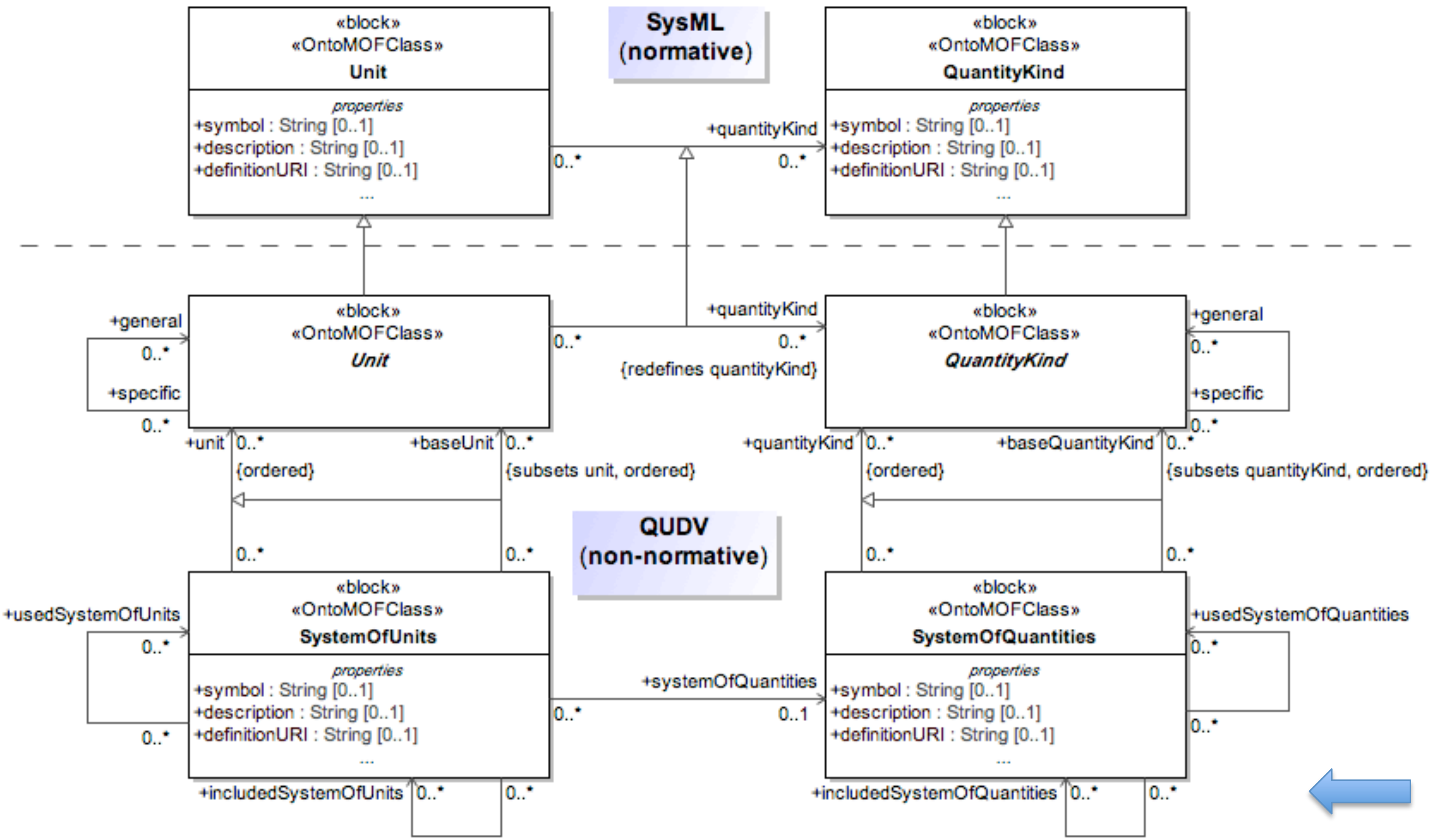
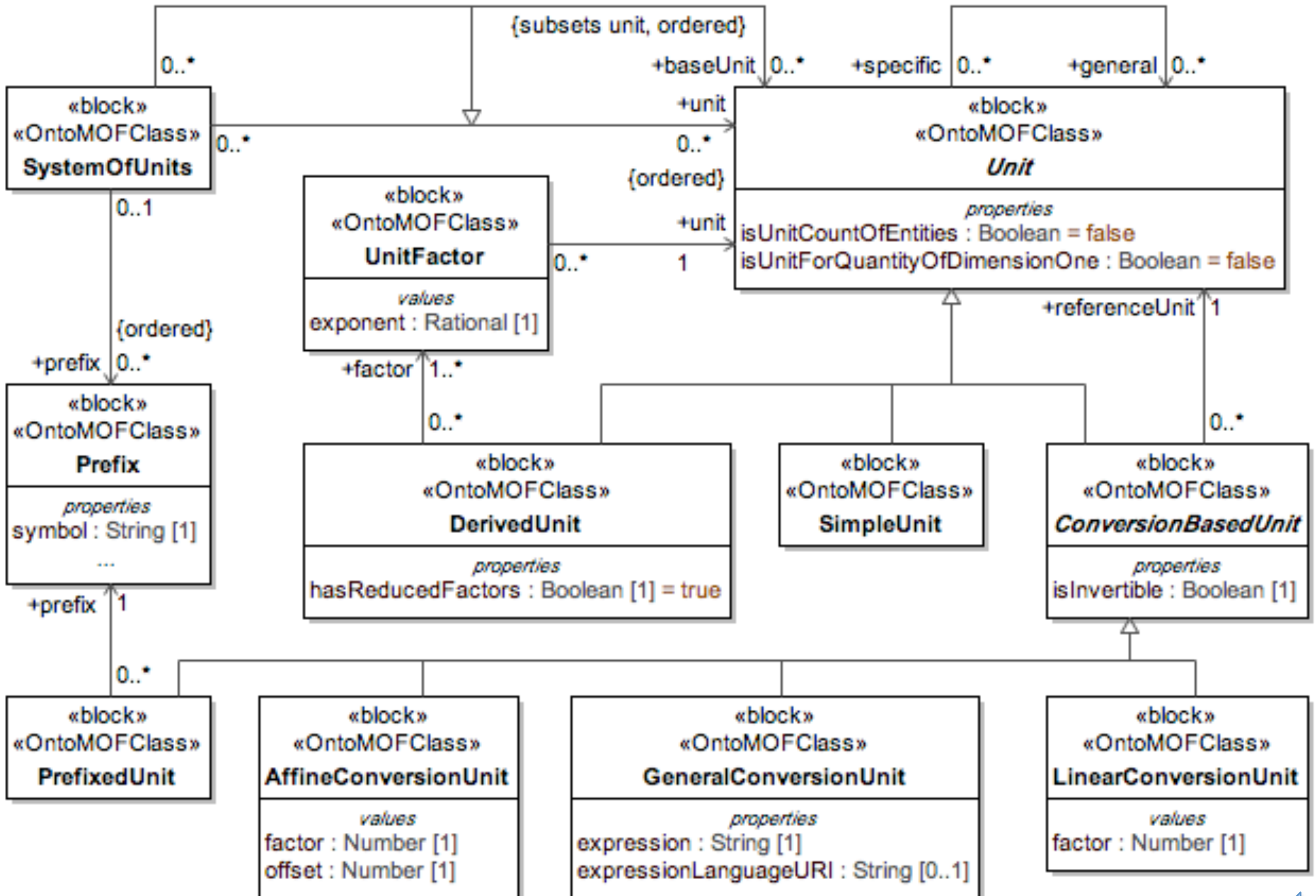


Figure 1. Mixed (generic and associative) concept diagram around ‘quantity’ as defined by text and diagram in VIM3 [1]. Bold type indicates concept defined in VIM3; ordinary type means concept only mentioned in VIM3; a straight line designates generic relation; three bullets mean unmentioned further specific concept(s); a double-headed arrow shows an associative relation. In level 6 the individualizing space–time coordinates have been omitted for simplicity.

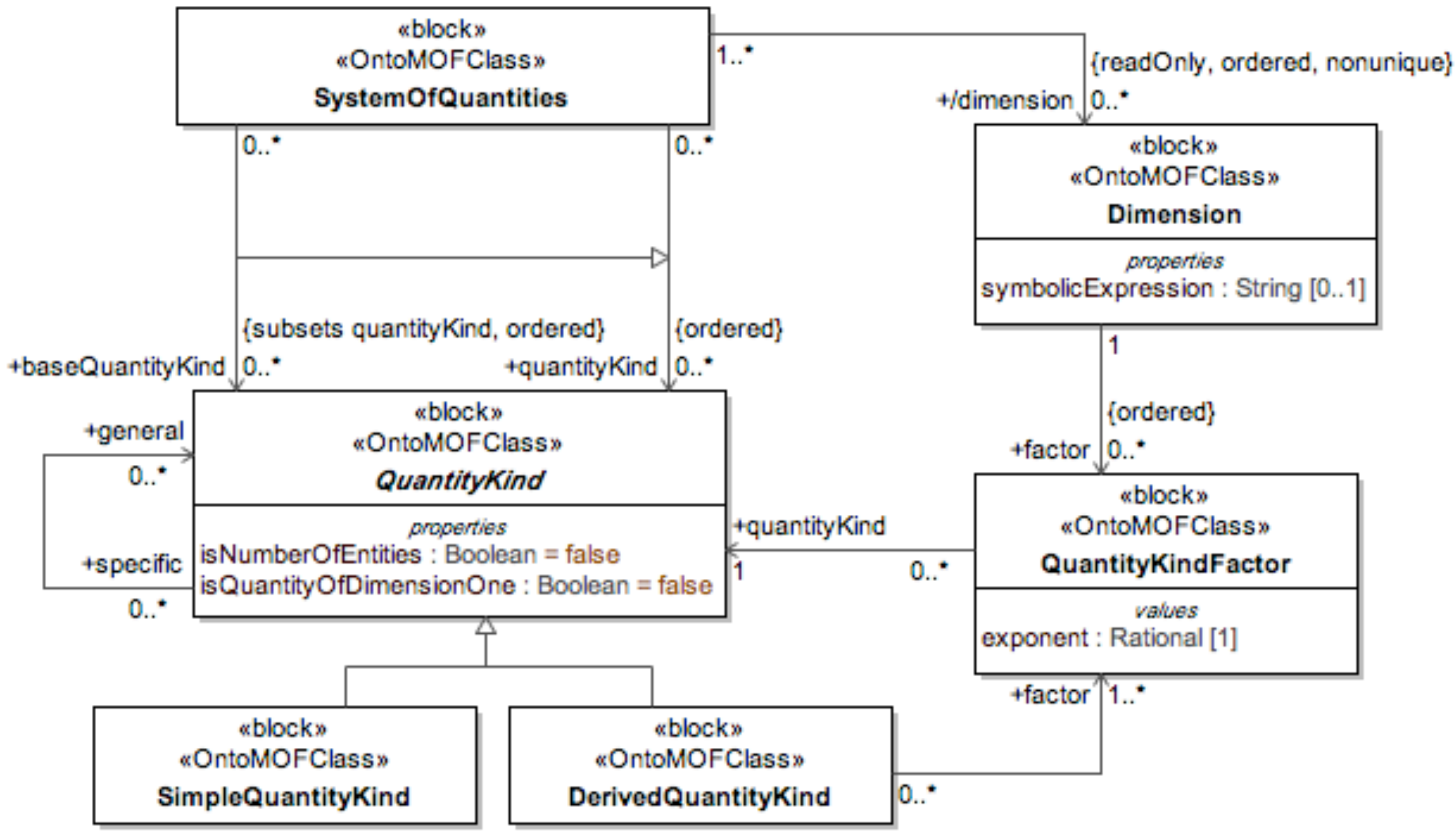
SysML 1.4 Metrology Vocabularies: SysML Profile & QUDV



Expanded Unit Vocabulary in QUDV

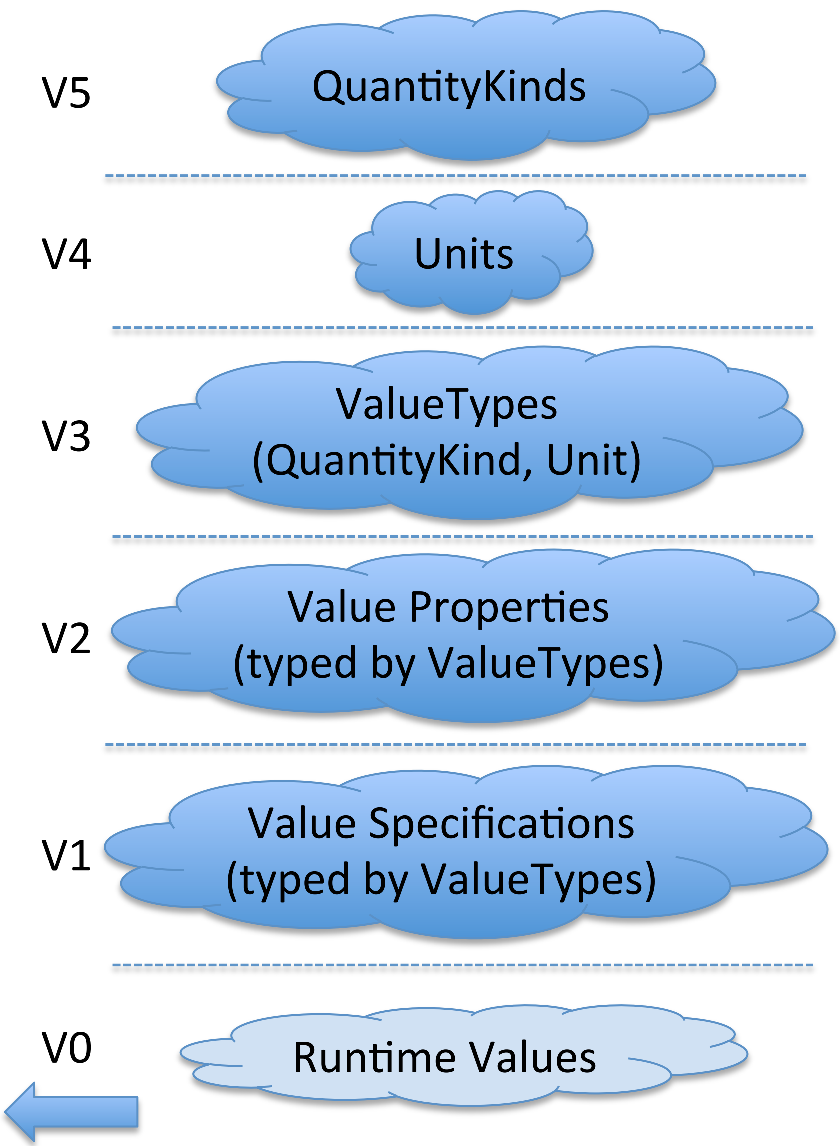


Expanded QuantityKind Vocabulary in QUDV



Practical Considerations for Modeling Values in SysML

#1: Recognize the ***distinct levels of modeling*** involved



- A Level depends on those above

  i < j => Vi depends on Vj

- V4 < V5

  See VIM3 & ISO 80000-1 foreword:

  > This first edition of ISO 80000-1 cancels and replaces ISO 31-0:1992 and ISO 1000:1992. The major technical changes from the previous standard are the following:
  >
  > — the structure has been changed to emphasize that quantities come first and units then follow;

- V3 < V4, V5

  Per SysML ValueType

- V2 < V3

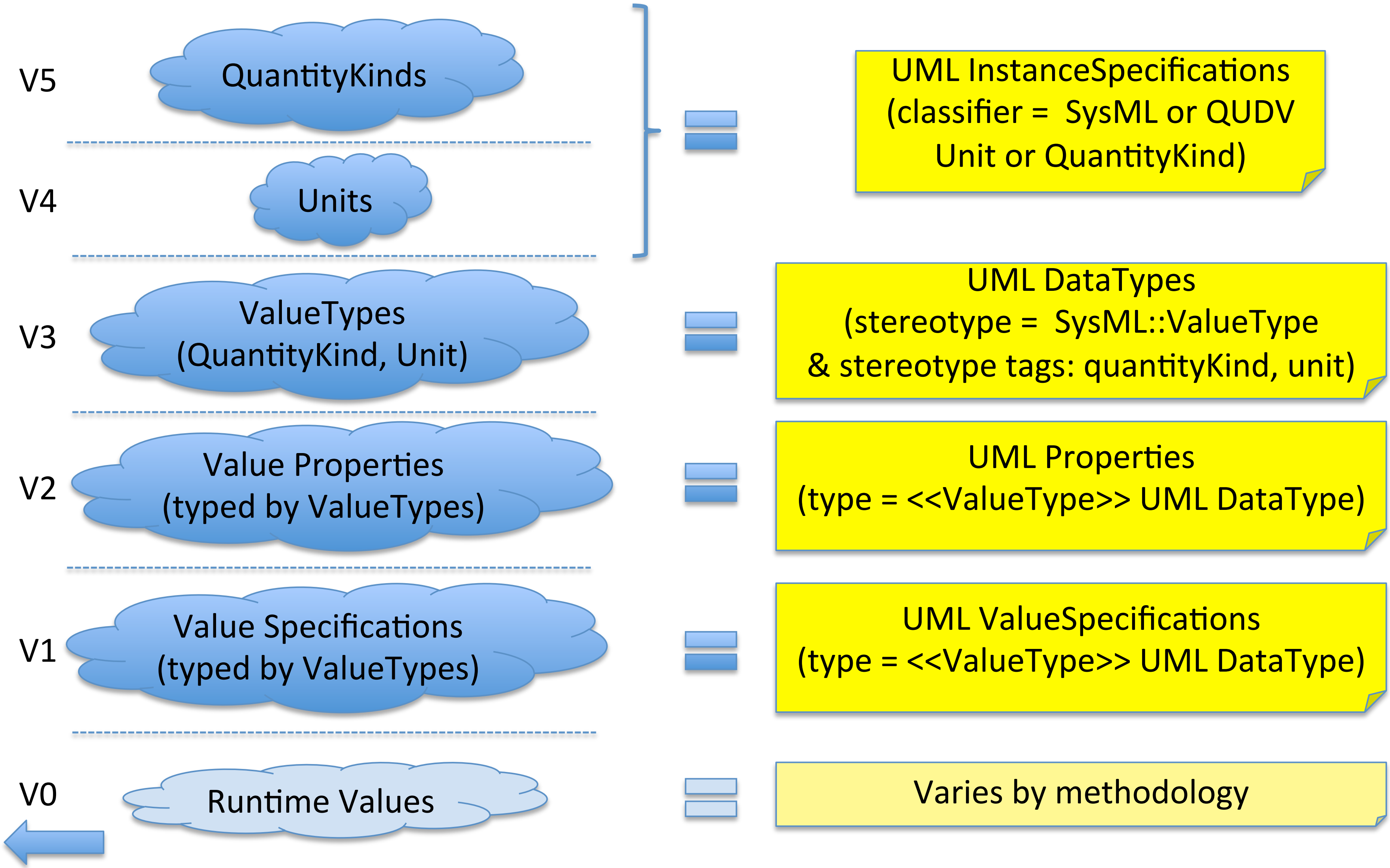
  Per UML TypedElement/Type relationship

- V1 < V2

  Per UML ValueSpecification/Slot/ StructuralFeature & defaultValue relationships

Practical Considerations for Modeling Values in SysML

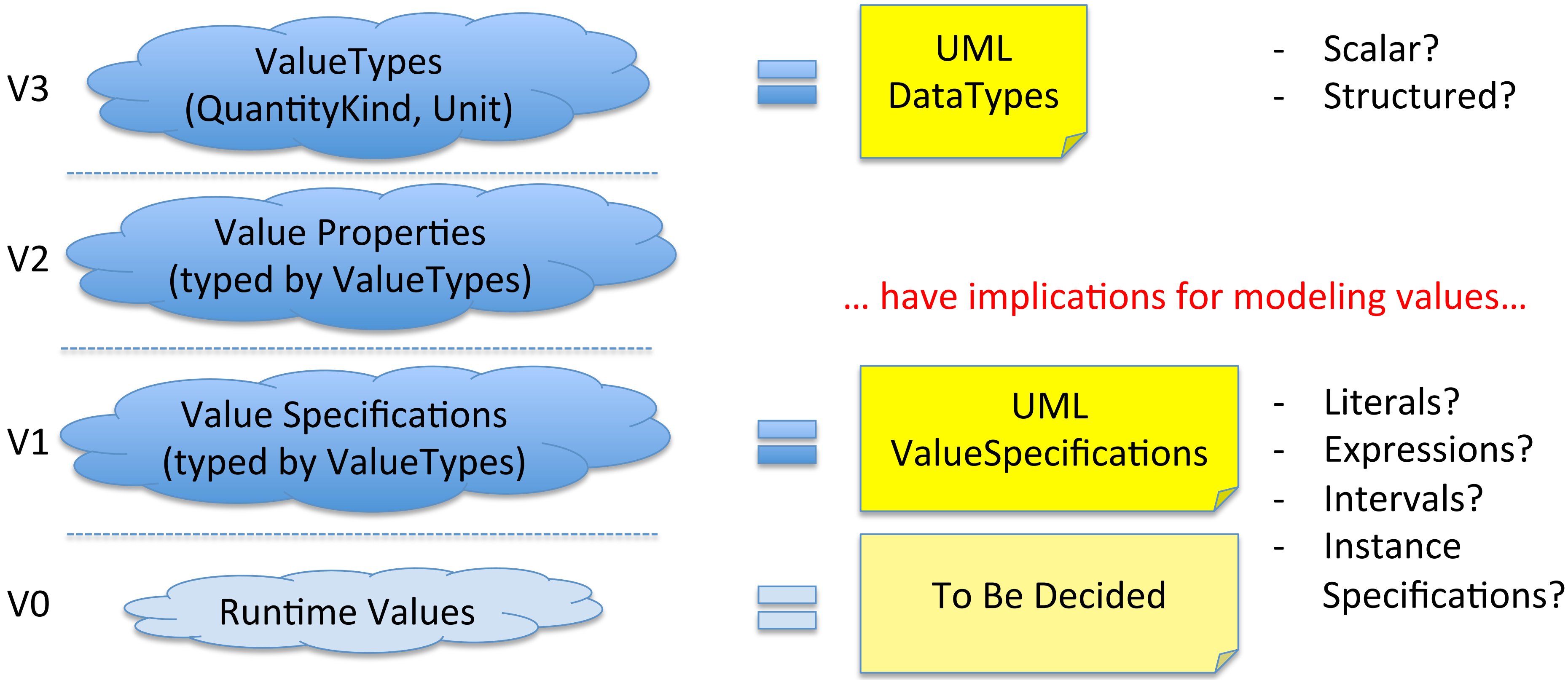
#2: Recognize the ***alignment with UML***



Practical Considerations for Modeling Values in SysML

#3: Recognize the ***implications of DataType modeling in UML***

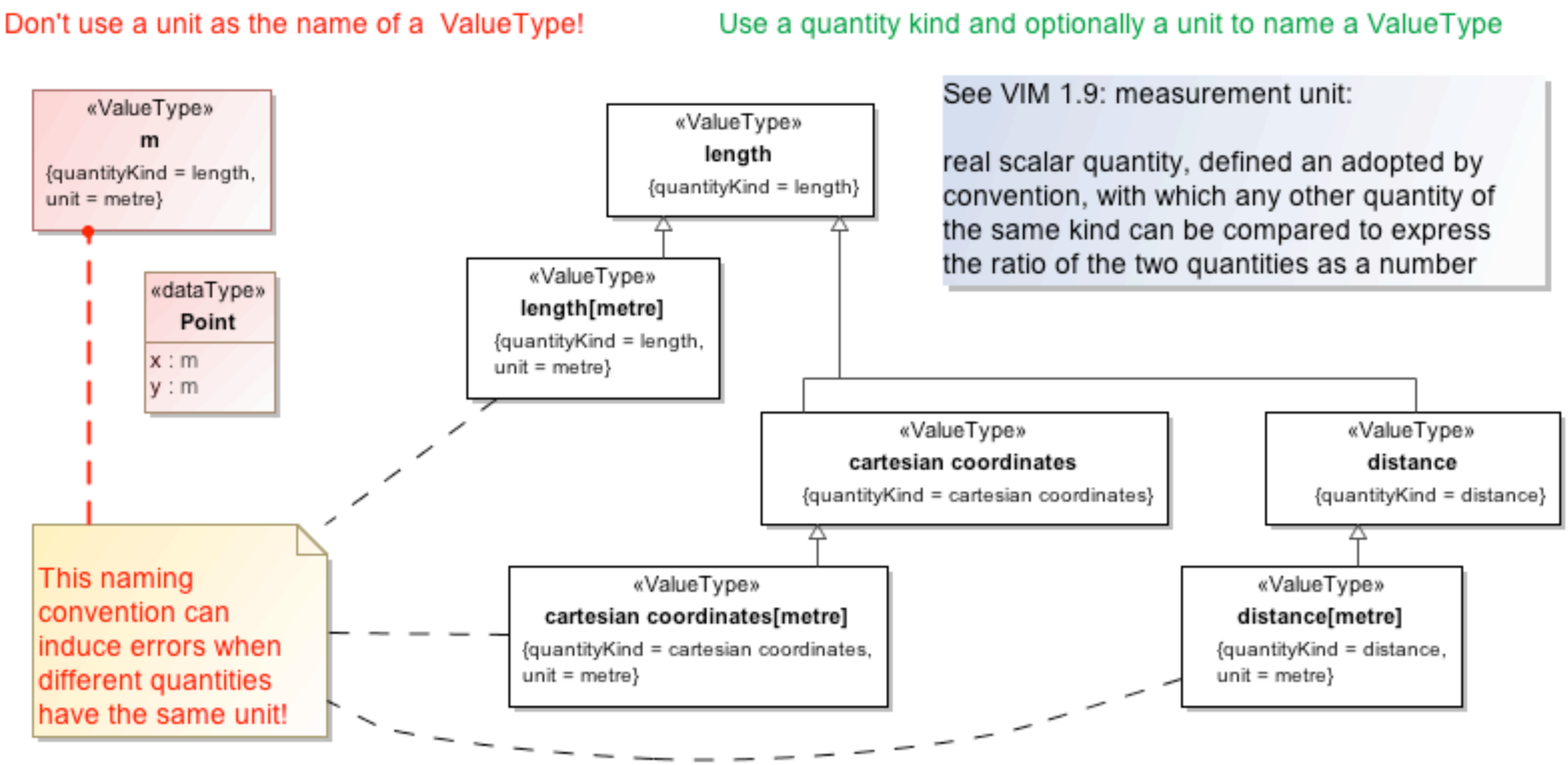
Choices made about datatype modeling…



… and type checking !

The conformance of a ValueSpecification to a ValueType depends on these choices!

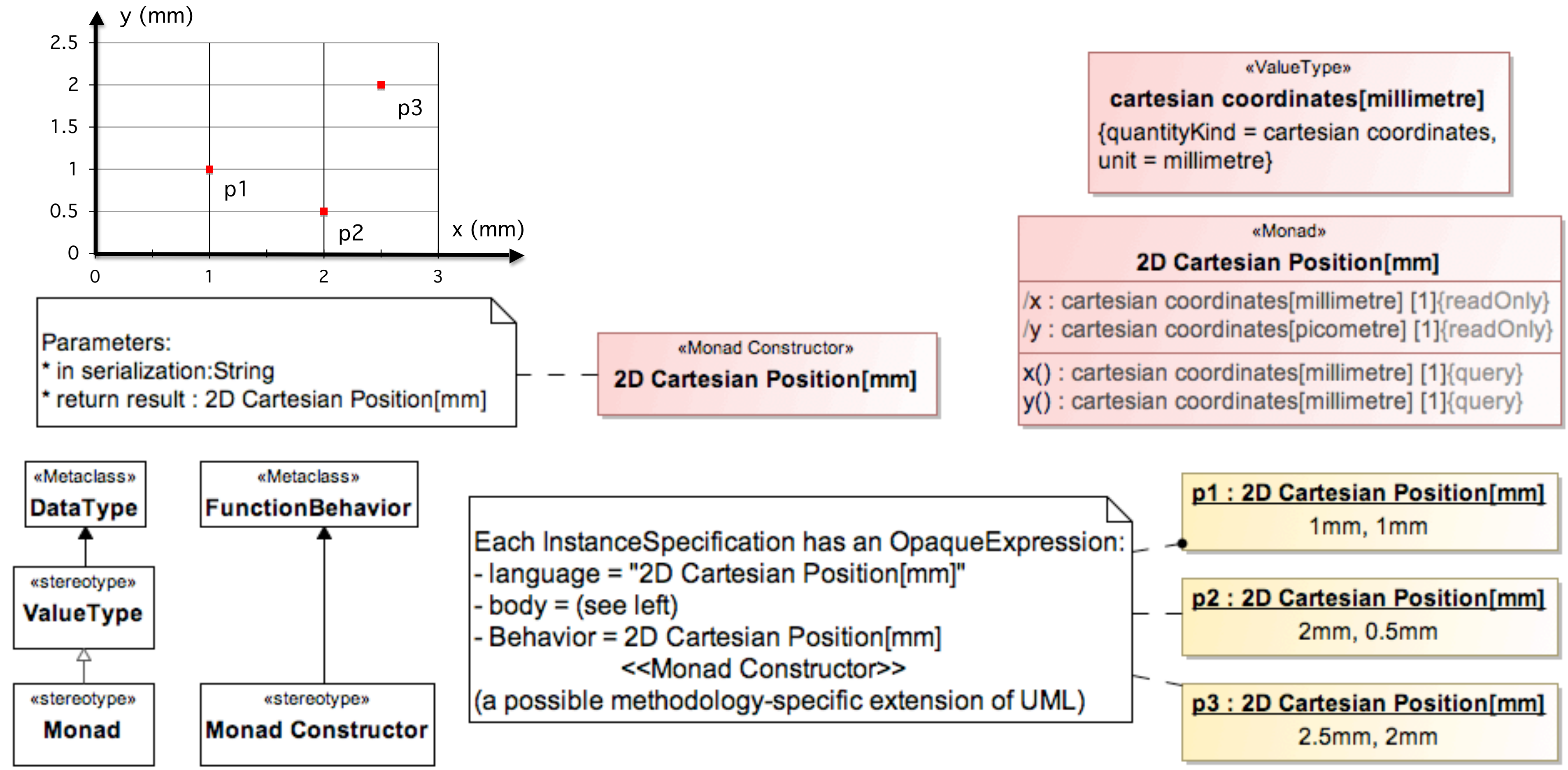
Note about naming ValueTypes



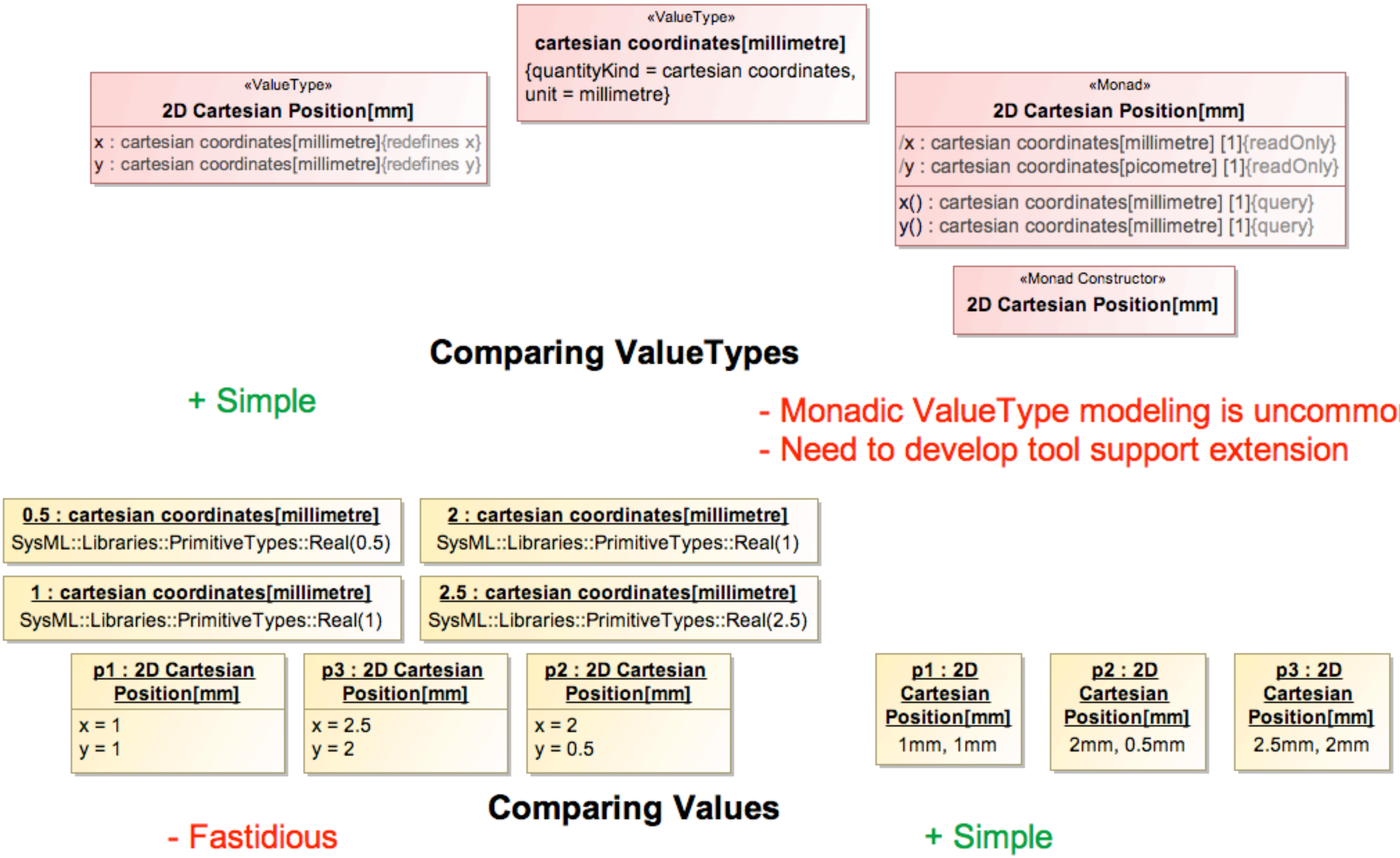
2D Cartesian Geometry Example (Alternative: Monadic ValueType)

http://en.wikipedia.org/wiki/Monad_(functional_programming)

The semantics of UML DataTypes are well-suited for “functional modeling”



Comparing Models 2D Cartesian Geometry



Comparing types & values/instances is important for assessing the pros/cons of a particular modeling methodology and for adopting one!