M-Layer Registry Prototype

Mark Kuster Blair Hall Ryan White

v0.04-draft (2022.5.10)

Version History

Version	Date	Changes
0.04	2022.0?.??	
0.03	2022.05.09	Extensive revision by BH, RW, MK
0.02	2022.04.25	Corrected the notation in the Usage section
0.01	2022.04.07	Added versioning and effective dates to the quantity systems and unit systems models. Added a table of Prefixes. Added Prefixable field to Units data model and corrected the conversion examples. Added UnitAspects. Added to the text throughout. Added an interface use case. Added a table of Aliases.
0.00	2022.03.15	Initial draft, adapted from discussions around [1–3] and MII working meetings

1 Introduction

This document sketches a high-level data model for the elements comprising a prototype M-Layer registry without regard to any particular data technology. As a draft design, all the information herein remains open to change.

1.1 M-Layer overview

The traditional expression of a quantity comprises a numeric value and the name or symbol for a unit, such as 10 kg. A notation is often adopted where, for the expression of a general quantity q,

$$q = \{q\}\ [q]$$

with $\{q\}$ representing the value and $[q]$ the unit (also called a reference).

This way of expressing quantities has some limitations, which the M-Layer would address. The M-Layer adds two main components:

1. A component called 'aspect', to capture the kind of quantity (and also generalising the idea of kind of quantity);

2. A more general notion of unit, or reference, which includes the type of scale.

For example, instead of the simple expression $x = 10\,\mathrm{kg}$, the M-Layer would capture an aspect (mass), a value (10), and a scale (extended unit) composed of the unit symbol (kg) and the scale-type (ratio scale). In plain English, the mass aspect of x is 10 when expressed in kilograms on a ratio scale.

A notation for M-Layer with the three components is

$$q = \langle q \rangle\, \{q\}\, [\![q]\!]\ ,$$

where $\langle q \rangle$ is the aspect, $\{q\}$ is the value, and $[\![q]\!]$ is the extended reference, or extended unit (we will find it convenient to use the term 'scale' instead of 'extended reference' or 'extended unit').

An M-Layer expression is a digital format that is not necessarily displayed to people using a system. Each aspect shall have a unique digital identity and each extended reference must also be uniquely identified. An aspect and scale must be combined to form an expression. However, the aspect does not represent the measurand (the quantity intended to be measured). Rather, the aspect identifies a set of alternative scales (e.g., ratio scales for the imperial pound or the SI kilogram to express mass).

So, the M-Layer serves to capture the expression of a particular data and also to facilitate a conversion to alternative expressions. It is also intended that the M-Layer could facilitate displays of data in units that may be understood by people working in a well-defined context but which could easily become ambiguous otherwise (so, conversion to such units would not be appropriate). These objectives lead to distinct operations that could be applied:

Convert an M-Layer expression from one scale to another without changing the scale type (e.g., convert between an expression of temperature in degrees Celsius on an interval scale to degrees Fahrenheit on an interval scale)

Cast an M-Layer expression from one scale to another with a change of scale type (e.g., cast an expression of temperature in degrees Celsius on an interval scale to kelvin on a ratio scale)

2 Implementation

To implement the M-Layer, a registry model is envisaged. The following sections indicate the essential features. Note, use of the terms 'table' and 'field' does not imply any particular data technology, such as SQL tables and columns. In addition to the details that follow, the overall registry would require administrative support, such as version control.

2.1 Aspect table

The aspect table would hold the registered aspects, according to the data model in Table 1. The AspectID is a unique key.

Table 1. The aspect data model.

Data Element	Description	Example
AspectID	unique identifier, or index, representing the aspect $\langle q \rangle$ in machine-readable documents and data	⟨length⟩
Name[1]	registered name	length
Symbol	mathematical symbol markup (e.g., LaTeX, MathML [4])	l
Definition[1]	textual description or external reference	PID to an ontology definition for length

2.2 Scales table

The scales table would register hold the registered scales, according to the data model in Table 2. The ScaleID is a unique key. This essentially combines an existing external unit, or reference, and a specific type of scale.

Table 2. The scale data model.

Data Element	Description	Example
ScaleID	unique identifier, or index, representing the scale $[\![q]\!]$ in machine-readable documents and data	⟦metre⟧
Name	the name is for human readers of the table	ml-metre
Unit or reference	UnitID	???
Scale type	ScaleTypeID	???

2.3 Units table

The units table (Table 3) indexes traditional measurement units. Systems would use these to render data for people. The same unit may be incorporated in more than one scale. For example, the degree Celsius might associate with both a ratio scale and an interval scale, distinct entries in the Scales register.

Table 3. The unit data model.

Data Element	Description	Example
UnitID	unique identifier-index	[in]
Name	unit name	inch
Symbol	unit symbol†	in
Definition	textual description or external reference	???

†Simple text may suffice, with the understanding that client systems render quantity values per applicable guidance documents [6, 7].

[1] Indicates fields that potentially, in place of the actual data, contain PIDs or some other link to external data such as an ontology (for example, [5]) where adequate existing systems exist. The first field listed in each table represents its unique key unless otherwise stated and most name fields should have uniqueness constraints. The term 'unique identifier' herein does not intend to limit the data type to UUIDs, GUIDs, etc. DOIs look promising in that they persist and have structure such that the ID prefix may point to an entity with further components delving deeper; measurement software might then just carry lightweight suffixes to identify the aspect, scale, etc.

Note, the definition of external units will allow for changing definitions over time, e.g., coherent SI9 or SI10 units. That would establish a reference for past, current, and future unit definitions.

Note also that some familiar notions like unit systems, base units, etc., relate to the units concerned. The unit data model would hold references to that information if desired. For example, the kilogram is an SI unit and a base unit of the SI system, etc. Such details matter little to the M-Layer except in relation to the display of data.

2.4 Scale-types table

ScaleTypes (Table 4) registers different types of scale. Initially, the following types should be considered: ratio, interval, log-interval, bounded-interval, ordinal, and nominal.

Table 4. Scale-type data model.

Data Element	Description	Example
ScaleTypeID	unique identifier-index	ratio-scale ID
Name	scale type	ratio†

2.5 Conversions and casting tables

Permitted conversions and casts between scales could be defined in separate tables, or combined in one by using a field to distinguish between the two cases. The data models are essentially the same. However, the implications of casting and conversion for data are different, so the distinction is important. For example, casting from a ratio scale to a log-interval scale is non-linear, which will affect the distribution of values.[1]

Table 5. Conversion and casting data model.

Data Element	Description	Example
ConversionID	unique identifier-index	(SourceScaleID, DestinationScaleID)
Operation	mathematical conversion operation	???

Note, conversion here is intended to occur directly but it may be possible to convert (or cast) though a common intermediate value in two steps. If the later option is chosen (an implementation detail), then the scale data model could define a pair of conversion operations, to and from the common intermediate format. If that were done, then the operation entry in Table 5 could be composed (the conversions table would not be needed). However, that implementation would require an acceptable common format for all scales. Where no common format is available, the conversion table is needed.

2.6 Scales for aspects

The AspectScales table (Table 6) provides the (one-to-many) correspondence between an aspect and the scales that can be used to express it. For instance, the aspect length could be expressed by ratio scales in inches or centimetres, and so on.

Table 6. Aspect-to-scales data model.

Data Element	Description	Example
AspectID	aspect	⟨length⟩
ScaleID	scale identifier	⟦in⟧

Neither ScaleID nor AspectID have a unique constraint.

[1] In a language like C++, the type of an object may sometimes by quietly coerced into another type when there will be no loss of information. This would appear to correspond to our use here of 'conversion'. On the other hand, C++ has a number of formal casting operations that must be deliberately applied to change the type of an object. This is what is envisaged by casting here.

References

[1] B. D. Hall and M. J. Kuster, "Metrological support for quantities and units in digital systems," in *The proceedings of IMEKO 2021 XXIII World Congress*. Yokohama, Japan (virtual): ScienceDirect, Aug 30 - Sep 3 2021. [Online]. Available: https://www.sciencedirect.com/journal/measurement-sensors/special-issue/10CTR9ZPK1R

[2] ——, "Metrological support for quantities and units in digital systems," *Measurement: Sensors*, vol. 18, p. 100102, Dec. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2665917421000659

[3] ——, "Representing quantities and units in digital systems," 2022, unpublished.

[4] WC3. (2014, Apr.) Mathematical markup language (MathML). World Wide Web Consortium (W3C). [Online]. Available: https://www.w3.org/Math/

[5] (2022, Mar.) Semantic specifications for units of measure, quantity kind, dimensions and data types. Quantity, Unit, Dimension and Type (QUDT). [Online]. Available: http://www.qudt.org/

[6] *The International System of Units (SI)*, International Bureau of Weights and Measures (BIPM) Information Document SI Brochure, Rev. 9th edition, 2019. [Online]. Available: https://www.bipm.org/en/publications/guides/

[7] B. N. Taylor, "Guide for the use of the international system of units (SI)," NIST, Washington, DC, Special Publication 811, Apr. 1995.

[8] M. J. Kuster, "Toward a measurement information infrastructure," *Metrologist*, vol. 6-pres, no. 1, Jan 2013-pres. [Online]. Available: https://ncsli.org/page/mmo13

Terms

aspect: a generalisation of the notion of kind of quantity quantity found in the VIM

digital object identifier (DOI): a structured persistent identifier that resolves to digital information

measurement information infrastructure (MII): set of normative standards that unambiguously define data structures, taxonomies, service protocols and security for locating, communicating and sharing measurement information [8]

M-layer: registries of data related to quantities and measurement units with an access scheme such as an API [1–3]

persistent identifier (PID): a digital identifier that maintains an enduring link to the identified object regardless of URL or other location changes

Acknowledgments

NCSL International for its MII support
NCSL International 141 MII and Automation Committee participants for their development work