Specifications for Machine-Readable Calibration and Measurement Capabilities

NCSL International 141 MII and Automation Committee

2023-02-09

Contents

List of Figures

List of Tables

1 Digital CMC Construction

Digital CMCs[1] should unambiguously describe laboratory services for machine consumption. A CMC comprises a measurand specification together with a measurement uncertainty. Sections 1.1 and 1.2, adapted from [2] respectively detail the measurand construction and the CMC uncertainty. The approved CMC taxonomy, comprising uniquely identified CMCs together with templates of their measurand specifications, resides at https://somewhere.there.

1.1 MII Measurand Structure

Simply stated, the measurand identifies what we intend to measure [3]. A measurement will fit its purpose to the degree that the measurand specification unambiguously and accurately describes the intent. Likewise, locating and selecting the correct measuring instrument to perform a measurement or a laboratory with the appropriate CMC to calibrate the instrument depends heavily on measurand specification's clarity and completeness. This certainly applies to manual operations aided by human judgment, but becomes all the more critical for unaided machine interpretation. We first discuss measurand names and then the full measurand schema.

1.1.1 Measurand Name

The measurand name, the measurand's specific output quantity, matters most for clarity. Therefore, MII documents tag their machine-readable MII measurands with clear, unique, and fully descriptive taxons from a defined measurand taxonomy. Taxons as measurand names apply only to internal document encoding and not to human-readable documents. The individual MII taxons adhere to the following structure and naming rules:

1. Every MII document identifies a given measurand by the same unique taxon string.

2. Each taxon may have aliases, such as commonly used equivalents (from *ISO-IEC 80000* [4], the KCDB, an AB's[2] conventions, etc.). These aliases may appear in human-readable documents generated from the digital document as the user prefers.

[1]calibration and measurement capabilities
[2]accreditation body

3. Each taxon comprises a series of tokens separated by the period (.) character.
4. Each token uses the UpperCamelCase[3] naming convention, e.g., `FrequencyModulation`.
5. A taxon's first token represents the process type, taking either the value `Measure` or `Source` to identify an input- or output-quantity measurement, respectively.[4]
6. The taxon's remaining tokens indicate the measured quantity.
7. The measured quantity's first token identifies the quantity kind [3], which shall unambiguously link to an M-Layer aspect.
8. Any further tokens after the quantity-kind token hierarchically qualify the quantity, proceeding from more general toward more specific quantity descriptors.
9. The string data format encourages concise tokens and widely recognized acronyms, e.g., `DC`, `RF`, `PRT`, `CMM`.
10. Parameters substitute for additional tokens to distinguish details within the same measurement process. `Source.Temperature.Simulated.Thermocouple`, for example, covers all thermocouple types via a type parameter, whereas a separate taxon (`Source.Temperature.Simulated.PRT`) covers platinum resistance thermometers (PRTs) because the measurement process changes (sourcing resistance instead of voltage).
11. Special tokens with their own syntax identify common measurement scenarios.
    a) The `Ratio` quantity token precedes the quantity-kind token to identify a quotient of two like-kind quantities.
    b) The `Coefficient` quantity token precedes two successive and differing quantity-kind tokens to identify a quotient of two unlike quantity kinds.
    c) The `Delta` token follows the quantity(ies) to identify a further quantity that differs when measuring the quotient's numerator and denominator.
    d) The `Model` token after a quantity introduces a standard instrument model.

Table 1.1 lists some taxonomy examples and their KCDB equivalents that illustrate the MII measurand structure and typical qualifier detail. Note that the taxon *as a whole* serves as a metadata tag to identify MII measurands. Other than distinguishing `Measure` or `Source` processes, a taxon's syntax and individual tokens do not encode meaning for machine processing; the taxon structure simply facilitates and standardizes taxonomy construction and organization.

[3] also known as Pascal or Capitalized

[4] Regardless of whether the measurement process uses a direct, common source, or comparator measurement method [5]. A token to capture both options might seem useful, but source and measure uncertainties usually if not always differ and therefore require separate CMCs.

MII Taxon	Closest KCDB Alias
`Measure.MassDensity.Solid`[a]	Density of solid
`Measure.Pressure.Pneumatic.Absolute.Static`	Absolute pressure, Gas medium
`Source.Current.AC.Sinewave.3Phase`	AC Current, Meters
`Source.Mass.Conventional`[b]	—

[a]Mass, as against flux or other densities; solid as against gas or liquid
[b]As against (true) mass

Table 1.1 Taxon examples [1].

1.1.1.1 Special Tokens

To aid in naming taxons, the MII measurand taxonomy treats two common quantities specially: ratios and coefficients. The term "ratio" indicates a dimensionless quotient [4], such as strain (length per length), amplifier voltage gain (voltage per voltage), or refraction index (light speed per light speed). "Coefficient" on the other hand, indicates a quotient of two different quantities [4], such as a transducer calibration correction (voltage per pressure). A ratio takes the name "factor" when used as a dimensionless proportionality constant [4]. In practice, some common measurand names ignore this convention, e.g., "reflection coefficient", "index of reflection", both of which we compute as ratios and use as factors. Both ratios and coefficients play into CMCs.

Ratios We structure ratio taxons as `...Ratio.Q`[5], where `Q` names both ratioed quantities. `Q`'s structure follows the taxon rules—first a token for the quantity kind representing an M-Layer aspect, then successively more specific descriptors. So, `Measure.Ratio.Pressure...` would identify a ratio of two particular pressures and `Source.Ratio.Power.RF...` would represent a ratio of two microwave powers. The M-Layer would have aspect entries for `Pressure` and `Power`.

Coefficients Coefficients relate an instrument's input and output quantities. Unconditioned piezoelectric accelerometers, for example, output an electric charge that varies with sensed acceleration, a response requiring quantification. Manufacturers therefore specify a nominal coefficient value that users wish to calibrate in order to correct the transducer output, and so we want a CMC to describe a laboratory's compatible service. The MII taxon structure therefore includes the syntax `Measure.Coefficient.QOut.QIn...`, where the two quantities listed after `Coefficient` have quantity-register entries and the coefficient equals $Q_{\mathrm{out}}/Q_{\mathrm{in}}$. Accelerometer sensitivity would look like `Measure.Coefficient.Charge.Ac` When the two quantities require different descriptor tokens, the numerator's descriptor tokens appear directly after the two quantity names, and the denominator's descriptor tokens thereafter. So we would name a coefficient of DC voltage to absolute pressure `Measure.Coefficient.Voltage.Pressure.DC.Absolute`.

[5]used for both ratios and factors since both require only one quantity kind

Delta The two quantities involved in ratios and coefficients often have an influence quantity that differs between them. For example, we might measure a frequency response by first measuring an signal amplitude V_{ref} at a reference frequency, then changing the frequency and measuring the new amplitude V. The ratio quantity (V/V_{ref}) represents the frequency response between the two frequencies. After the main quantity, the special tokens `...Delta.QInf` flags an influence quantity `QInf` (with a quantity-register entry) that changes during the measurement. So using AC RMS amplitudes in this example, we would name their ratio `Ratio.Voltage.AC.RMS.Delta.Frequency`.

Instrument Models So far, we've discussed ratios and coefficients only in a point-measurement context—calibrating a device at one or more measurement points and determining a *separate* bias-correction coefficient value at each point. Coefficients also arise in a separate but related context though: the coefficients of a mathematical model (function) that corrects instrument indications *over a range*. Examples include ITS-90[6] range and subrange functions for PRTs, quadratic or cubic curve fits for force transducers, Callendar-Van Dusen (CVD) equations for RTDs[7], and many others. In theory, we may assign any measuring instrument a correction model and determine the model's coefficients from measurement results. Whether done at the calibration-point level or at the range, function, or instrument level, such a correction function with coefficient values raises the service from verification (that the instrument meets tolerances) to true calibration [3].

Though either the calibrating laboratory or the customer may have software to calculate modeling coefficients from the point-by-point calibration results, the laboratory more likely has the expertise, and for smart instruments, customers may prefer turnkey calibrations that load coefficients into the instrument. This might drive CMC taxons for identifying such measurement services. The MII tokens `...Model.M` serves this purpose, where `Model` signals an immediately following defined model type `M`. So if an instrument's instrument specification tagged a measuring function with `Measure.Temperature.PRT.Model.ITS90`, then `Source.Temperature.PRT.Model.ITS90` would identify the CMC to calibrate that function. In general though, the MII instrument specification schema will provide for calibration models of any form for which calibration services may assign coefficient values for smart instruments and digital calibration certificates [6].

1.1.1.2 Formal Taxon Syntax

The following BNF[8] grammar defines the measurand taxon syntax

where the subscripts "n" and "d" represent numerator and denominator, respectively, and RQK means registered quantity kind (M-Layer aspect).

[6] International Temperature Scale, 1990

[7] resistance temperature detectors (or devices)

[8] Backus-Naur form: "—" separates alternatives, "*" means zero or more consecutive instances, angle brackets enclose descriptive text, parentheses group tokens

Taxon	::=	ProcessType . (Quantity — Ratio — Coefficient) [. Model]
ProcessType	::=	`Measure` — `Source`
Quantity	::=	RQK (. Descriptor)*
RQK	::=	⟨any name in the quantity kind registry⟩
Descriptor	::=	⟨any measurand-qualifying term⟩
Ratio	::=	`Ratio` . Quantity
Coefficient	::=	`Coefficient` . RQK_n . RQK_d (. $\mathrm{Descriptor}_\mathrm{n}$)* (. $\mathrm{Descriptor}_\mathrm{d}$)*
Model	::=	`Model` . ModelName
ModelName	::=	⟨any instrument-model name⟩

1.1.2 Supporting Information

The measurand name identifies the measurement and disambiguates it from other measurements but does not specify the (critical) process or capability details. Here we discuss the further measurand detail required to clarify a CMC.

1.1.2.1 Definition

Regardless of care taken in naming taxons, a clear, human-readable definition helps disambiguate one measurand from another. This helps the metrologist select the correct measurand from a list, for example, when building an SoA. Also, since new measurands continually arise with new technology or measurement techniques, we will never have a complete measurand taxonomy. A definition thus helps determine whether the measurand of interest appears in the taxonomy or requires a new entry. Finally, an extensible taxonomy exposes both its taxons and their definitions to change reflecting the current state of knowledge. For example, if we everyone only measured static pressure, then the taxon `Source.Pressure` and its definition would require changes to differentiate `Pressure.Static` and `Pressure.Dynamic` once a demand for dynamic-pressure measurement arose.

1.1.2.2 Parameters

We devalue calibration without fully knowing and stating the measurement conditions, the measurand's state. Specifying the measurand's full state restricts its *definitional* uncertainty [3], the range of (true) values that match the measurand; failing to do so may inflate definitional uncertainty beyond other uncertainty components, or even beyond the instrument MPE[9] specification, essentially making the calibration worthless. We should define our measurands such that definitional uncertainty remains insignificant relative to other uncertainty components and include those definitions as metadata in instrument specifications, calibration certificates and SoAs.

The measurand state includes input quantities, influence quantities, and instrument operating conditions. Input quantities affect the measured (calculated) value and usually

[9] maximum permissible error

the CMC uncertainty. Influence quantities do not affect the measured value's calculation but may affect the CMC uncertainty. Both input and influence quantities determine the measurand's state and thus affect the measurement result, so CMCs and their representative taxons should specify the applicable quantities. Examples include dew- or frost-point temperature in chilled-mirror relative-humidity measurements, frequency in AC measurements, acceleration in accelerometer sensitivity measurements, temperature in dimensional and many other measurements. For some measurands, a non-numeric property such as a thermocouple type (J, K, S, T, ...) may apply.

The MII measurand structure refers to these quantities and properties as "parameters" and defines both required and optional parameters. An automated CMC search, a CMC uncertainty calculation and the actual measurement will not all succeed without the required parameters' values. Required parameters usually include the primary measured (output) quantity. Optional parameters, however may remain uncontrolled, perhaps invoking a higher uncertainty, or defaulting to a specified (nominal) value. Taxons in the taxonomy catalog designate parameters as required or optional as seems most appropriate, but when used in a CMC, the laboratory will choose which parameters to require, which to make optional with default values such as a $50\,\Omega$ input impedance or a $20\,°\mathrm{C}$ reference temperature, and which to omit entirely as immaterial to the measurement process.

A complete instrument specification includes the (rated, limiting, and reference) operating conditions [3] for which its specifications apply. However, an MII measurand taxon's parameters include instrument operating conditions only when they overlap with laboratory capabilities. For instance, a voltage reference standard may require battery operation (at a minimum voltage) for specified accuracy, but this procedural detail does not distinguish one laboratory's capability from another. In contrast, some rated or reference operating conditions may limit influence quantity values to ranges that some laboratories may not achieve, such as a tight ambient-temperature tolerance. In some cases, the CMC uncertainty would reflect the relevant capability, but not for all instruments.

1.1.2.3 Measuring Intervals

The abstract measurand taxons in the taxonomy apply to any measured value, so the taxonomy's CMC templates themselves do not include measuring intervals (ranges [3] or nominal values). Concrete instances such as CMCs, however, should specify the measuring intervals over which they apply. The MII SoA structure [7] includes this element, which human-readable SoAs should show with every CMC[10]. Besides their contribution to CMC uncertainties, this allows intelligent searches for useful calibration services, whether a quantity at a single point or an entire instrument range interests us. This logic applies to not only the output quantity, but also all the input and influence quantities and operating conditions. Customers may choose to omit optional parameter values for CMC searches or calibration requests, but SoA CMCs should define ranges, if

[10]Caveat: Measuring intervals may not apply to SoAs outside the calibration field.

only single points, for all supported parameters. As with parameter defaults, all ranges represent nominal values in CMCs (and instrument specifications); measured values appear only in calibration certificates.

1.1.3 Interchangeable Quantities and Scales

Some quantities have multiple scales or derive in known ways from other quantities. For example, we may characterize a microwave reflection in terms of "reflection coefficient", "VSWR"[11], or "return loss" and we convert between them via defined equations. Also, some instruments (nominally) follow known models, such as thermocouples' voltage-temperature responses. Table 1.2 gives example values.

Search Quantity	Equivalent(s)
reflection coefficient: 0.10	VSWR: ≈ 1.2; return loss: 20.0 dB
thermocouple input temperature Δ: 10 °C	nominal type-K output voltage: ≈ 0.397 mV

Table 1.2 Equivalent quantities.

The question then arises whether CMCs should express multiple quantities or scales to facilitate searches. The short answer: no. If customers wish to search for a lab to calibrate a thermocouple over a certain temperature range, they likely will not care to search by the corresponding voltage range, even though they will want to calibrate the DC voltage measuring instrument used with the probe over that range. Software should handle such conversions where required. The same applies to the microwave-reflection example. Laboratories, however, may list multiple CMCs in their digital SoAs as they think useful. Customers may also specify calibration results in a particular format, but that pertains to calibration certificates, not CMCs.

To complete the picture, Figure 1.1 depicts the current draft MII taxonomy schema.

1.2 CMC Uncertainty

As abstract templates, the CMC taxonomy contains no uncertainty information. Any given concrete CMC instance, will however, contain that information. The uncertainty element takes the form of TBD...

[11] voltage standing wave ratio

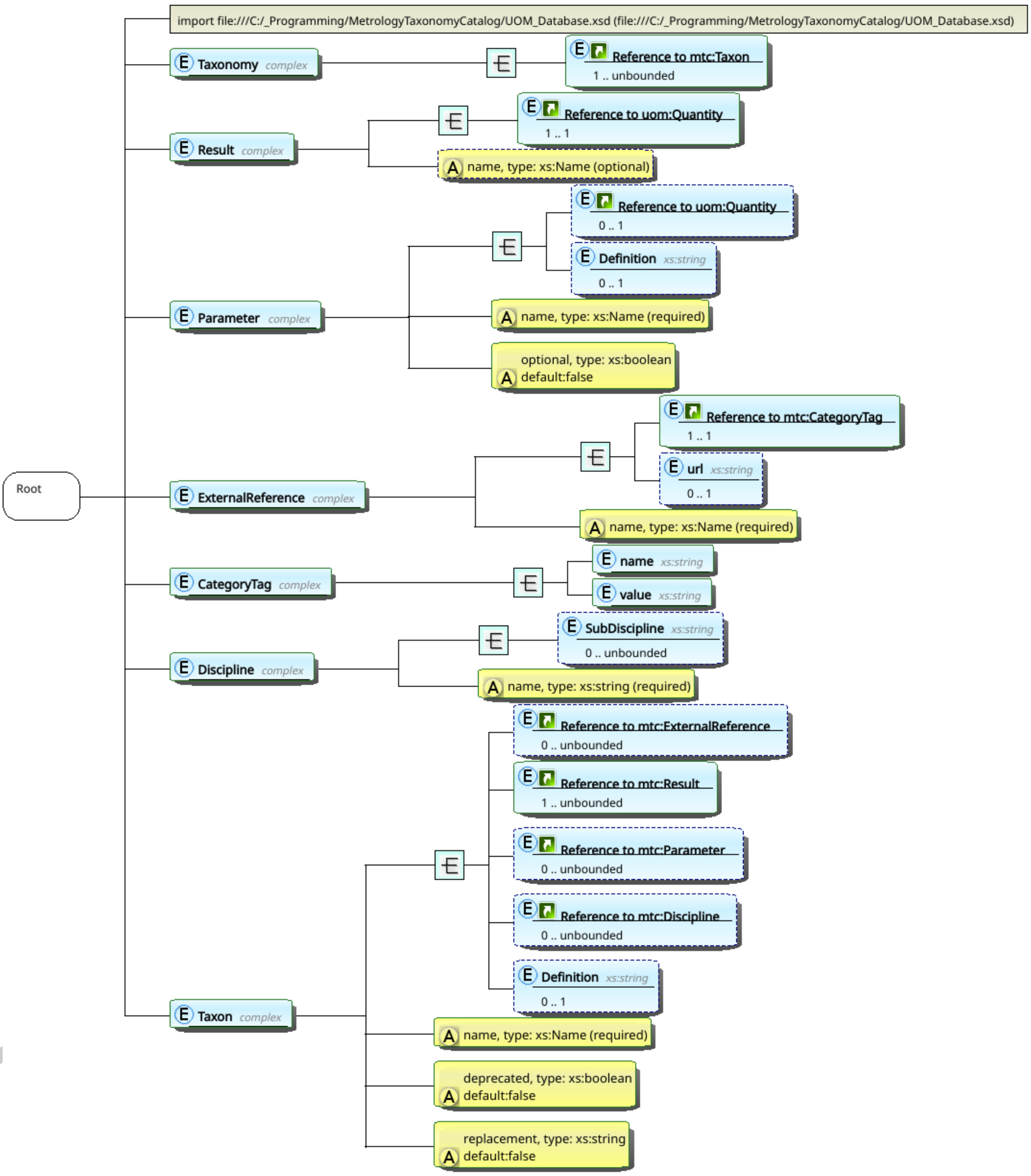

Figure 1.1 MII taxonomy schema (less the CMC-only elements). In addition to the elements previously discussed, the schema includes category and discipline tags to optionally map MII taxons to other nomenclature systems for human-readable output and interoperability.

References

[1] M. J. Kuster, "A measurement information infrastructure's benefits for industrial metrology and IoT," in *2020 IEEE International Workshop on Metrology for Industry 4.0 and IoT (MetroInd4.0&IoT)*. Rome, Italy (virtual): IEEE Press., Jun 3-5 2020, pp. 479–484. [Online]. Available: https://ieeexplore.ieee.org/xpl/conhome/9130790/proceeding

[2] ——, "A guide to standardizing digital calibration and measurement capabilities," *Metrologist*, vol. 15, no. 1, pp. 36–47, Jan. 2022. [Online]. Available: https://ncsli.org/page/mmo22

[3] *The international vocabulary of metrology—Basic and general concepts and associated terms (VIM)*, Joint Committee for Guides in Metrology (JCGM) Guidance Document JCGM 200:2012, Rev. 3rd edition, 2012. [Online]. Available: https://www.bipm.org/utils/common/documents/jcgm/JCGM_200_2012.pdf

[4] *Quantities and units—Part ⋯*, International Standardization Organization (ISO) and International Electrotechnical Commission (IEC) Std. ISO-IEC 80 000, Rev. first edition, 2006-2011.

[5] *Determining and Reporting Measurement Uncertainty*, NCSL International (NCSLI) Recommended Practice RP-12, Apr. 2013. [Online]. Available: https://ncsli.org/store/viewproduct.aspx?id=16960491

[6] M. J. Kuster, "Metrological data completeness for digital transformation," in *2021 IEEE International Workshop on Metrology for Industry 4.0 and IoT (MetroInd4.0&IoT)*. Rome, Italy (virtual): IEEE Press., Jun 7-9 2021, pp. 374–379. [Online]. Available: https://ieeexplore.ieee.org/xpl/conhome/9488326/proceeding

[7] D. Zajac, "Measurement information infrastructure: Representing ISO/IEC 17025 scope of accreditations in XML data," in *Proc. NCSL Int. Workshop & Symposium*. National Harbor, MD: NCSL International, August 12-17 2017. [Online]. Available: http://miiknowledge.wikidot.com/local--files/start/UpdatedXMLSchema-Zajak2017