Dictionary of Data *XDI* Metadata

XDI Working Group

- Matthew NEWVILLE (University of Chicago Center for Advanced Radiation Sources, APS)
- Bruce RAVEL (NIST) bravel AT bnl DOT gov
- V. Armando SOLÉ (ESRF)
- Gerd WELLENREUTHER (DESY)
- mailing list - http://millenia.cars.aps.anl.gov/mailman/listinfo/xasformat
- GitHub organization - https://github.com/XraySpectroscopy

Contents

1 Overview

This is **version 1.0** of the dictionary of metadata to be used with the XAS Data Interchange (XDI) format. Each item definition includes:

1. The name representing the datum
2. The meaning of the datum
3. Thw units of the datum
4. The format of representing its value

Words used to signify the requirements in the specification **shall** follow the practice of RFC 2119.

A use of this dictionary is not compliant if it fails to satisfy one or more of the **must** or **required** level requirements presented herein.

1.1 The meaning of metadata

The purpose of this dictionary is to identify a set of metadata to be encoded in the specification of the *XDI* format and to assign names to each meaningful concept. This effort must take a broad view, capturing metadata concepts as broadly as they are used in the community. This effort must also be open ended in that there must be a mechanism for providing new forms of metadata not considered up front. This effort is intended to serve as the XAS metadata dictionary for other data format types, for instance a database format for libraries of XAS spectra or a hierarchical format for multi-spectral datasets.

1.2 The *XDI* syntax

This dictionary has been developed along with the *XDI* specification. Any examples given in this dictionary use the **recommended** *XDI* syntax. The metadata name consists of the capitalized namespace, followed by a dot, followed by a tag. Here is an example: `Element.symbol`. When appearing in an *XDI* file to convey a metadata value, the line begins with a comment token and end with an end-of-line token. A colon is the delimiting token between the metadata name and its value. Here is an example:

```
# Element.symbol: Cu
```

1.3 The format of the value

Some of the tags in this dictionary have formatted values as part of their definitions.

- *string*: A string is specifically an ASCII string representable by characters in the the lower 128 of the ASCII set. This **must** the English-language representation of the value. For example, the string representing `Facility.name` for the Thai synchrotron **must** be `SLRI` rather than a sequence of characters in the Thai script.

- *free-format string*: This is a string which can contain any character (save end-of-line characters) in any encoding system. A free-sormat string need not be ASCII and need not be English. Because applications using *XDI* may not be capable of handling some encoding systems, it is **recommended** that free-format strings be ASCII.

- *string + units*: This is a string as defined above, followed by white space, followed by a string denoting the units of the previous string. As an example, a value for `Column.1` might be `energy eV`, which identifies the contents of the first column in the data table as containing energy values expressed in electron volt units.

- *float*: A float is a string which is interpretable as a floating-point number in the C programming language. An integer is permissable. Values of `NaN`, `sNAN`, `qNAN`, `inf`, `+inf`, and `-inf` are not allowed in *XDI*. That is, a float in *XDI* must be finite number. See IEEE 754-2008.

- *float + units*: This is a float as defined above, followed by white space, followed by a string identifying the units of the number. For example, a value for `Sample.temperature`, which identifies the tamperature at which an XAS measurement is made, might be `500 K`, which identifies the temperature of the measurements in Kelvin temperature units. The selection of possible units for a tag is given in the definition of the tag.

- *chemical formulas*: `Sample.stoichiometry` is intended to represent the elemental composition of the sample. To make these formulas interpretable by computer, this and extension fields which represent chemical information **must** use the IUCr definition of a chemical formula.

- *time*: Because of the wide variability of cultural standards in the representation of time, *XDI* defines a strict standard for time stamps in *XDI* files. `Scan.start_time`, `Scan_end_time`, and any extension fields dealing in time **must** use the ISO 8601 specification for combined dates and times

- *element symbols*: `Element.symbol`, `Element.reference`, and any extension fields identifying specific elements **must** use one of the recognized 1, 2, or 3 letter symbols given in Sec. 2.10

- *edge symbols*: `Element.edge`, `Element.ref_edge`, and any extension fields identifying specific absorption edges **must** use one of the recognized 1 or 2 letter symbols given in Sec. 2.10. Note that the subscript is represented as an Arabic numeral and not as a Roman numeral.

Some additional comments:

- A tag which is in a defined family but which is not defined in this dictionary **must** be interpreted as have a free-format string as its value.

- A tag which is present in an *XDI* file but which has no value or only white space as its value (i.e. the colon is followed by zero or more spaces tokens then by an end-of-line token) **must** be interpreted as a zero-length string or as the value 0, as appropriate to the value type.

- Strings identifying facilities and beamlines **must** use whatever convention is in use at the beamline. In the case where a beamline is known both by a designation and a name (for example, beamline 13ID at the Advanced Photon Source is also known by its name, “GSECARS”), the designation is **recommended**.

2 The dictionary

2.1 Name spaces

The purpose of namespaces is to provide sensible, widely understood, semantic groupings of defined metadata tags. All tags associated with conveying information about sample preparation and the measurement environment of the sample belong in the *Sample* namespace, all tags associated with the configuration of the beamline optics belong in the *Beamline* namespace, and so on.

Namespaces are strings composed of a subset of the ASCII character set. The first character **must** be a letter. The remaining characters **must** be letters, numbers, underscores, or dashes. Letters are ASCII 65 through 90 (`A-Z`) and ASCII 97-122 (`a-z`). Numbers are ASCII 48-57 (`0-9`). Underscore (`_`) is ASCII 95 and dash (`-`) is ASCII 45. The namespace **must** be interpreted as case insentitive.

Here is a list of all defined semantic groupings:

1. `Facility`: Tags related to the synchrotron or other facility at which the measurement was made
2. `Beamline`: Tags related to the structure of the beamline and its photon delivery system
3. `Mono`: Tags related to the monochromator
4. `Detector`: Tags related to the details of the photon detection system
5. `Sample`: Tags related to the details of sample preparation and measurement
6. `Scan`: Tags related to the parameters of the scan
7. `Element`: Tags related to the absorbing atom
8. `Column`: Tags used for identifying the data columns and their units

Below, specific members of these namespaces are defined. The definitons are not exclusive. Other metadata can be placed in these namespaces as needed. Of course, undefined metadata are unlikely to be interpreted correctly by applications using this dictionary. Metadata added to a defined namespace **must not** use a defined tag. The defined namespaces and tags **shall** be interpreted without sensitivity to case.

When defined metadata are present, the units and formatting specified below **must** be observed.

2.2 Tags

Tags are the words used to denote a specific entry in a namespace.

Tags are strings composed of a subset of the ASCII character set. All characters **must** be letters (ASCII 65 through 90, `A-Z` and ASCII 97-122, `a-z`), numbers (ASCII 48-57, `0-9`), underscore (ASCII 95, `_`), or dash (ASCII 45, `-`).

The tag **must** be interpreted as case insentitive.

2.3 Required metadata

Three items are essential to the interchange and successful interpretation of XAS data. These are **required** in all files using the *XDI* specification.

- `Element.symbol`: The element of the absorbing atom. The periodic table is replete with examples of atoms that have absorption edges with very similar edge energies. For example, the tabulated values of the Cr K edge and the Ba L1 edge are both 5989 eV. Without identification of the species of the absorbing atom and of the absorption edge measured, some data cannot cannot be unambiguously identified.
- `Element.edge`: The absorption edge measured. See above.
- `Mono.d_spacing`: The d-spacing of the monochromator. This is **required** when the abscissa is expressed in angle or encoder steps. It is required to convert that abscissa into energy. Also a correction to the energy axis of measured data, which may be required in the case of a miscalibration due to inaccuracies in the translation from angular position of the monochromator to energy, would need the d-spacing.

Most other metadata definitions that follow are **optional** for use with *XDI*. Some are **recommended** for use with all *XDI* files. The **recommended** metadata convey information that is of substantive value to the interpretation of the data.

2.4 Defined items in the Facility namespace

- **Namespace:** `Facility` – **Tag:** `name`
  - *Description*: The name of synchrotron or other X-ray facility. This is **recommended** for use in all *XDI* files.
  - *Units*: none
  - *Format*: string
- **Namespace:** `Facility` – **Tag:** `energy`
  - *Description*: The energy of the current in the storage ring.
  - *Units*: GeV, MeV
  - *Format*: float + units
- **Namespace:** `Facility` – **Tag:** `current`
  - *Description*: The amount of stored current in the storage ring at the beginning of the scan.
  - *Units*: mA, A
  - *Format*: float + units
- **Namespace:** `Facility` – **Tag:** `source`
  - *Description*: A string identifying the source of the X-rays, such as “bend magnet”, “undulator”, or “rotating copper anode”. This is **recommended** for use in all *XDI* files.
  - *Units*: none
  - *Format*: string

2.5 Defined items in the Beamline namespace

- **Namespace:** `Beamline` – **Tag:** `name`
  - *Description*: The name by which the beamline is known. This is **recommended** for use in all *XDI* files.
  - *Units*: none
  - *Format*: free-format string
- **Namespace:** `Beamline` – **Tag:** `collimation`
  - *Description*: A concise statement of how beam collimation is provided
  - *Units*: none
  - *Format*: free-format string
- **Namespace:** `Beamline` – **Tag:** `focusing`
  - *Description*: A concise statement about how beam focusing is provided
  - *Units*: none
  - *Format*: free-format string
- **Namespace:** `Beamline` – **Tag:** `harmonic_rejection`
  - *Description*: A concise statement about how harmonic rejection is accomplished
  - *Units*: none
  - *Format*: free-format string

2.6 Defined items in the Mono namespace

- **Namespace:** `Mono` – **Tag:** `name`
  - *Description*: A string identifying the material and diffracting plane or grating spacing of the monochromator
  - *Units*: none
  - *Format*: free-format string
- **Namespace:** `Mono` – **Tag:** `d_spacing`
  - *Description*: The known d-spacing of the monochromator under operating conditions. This is a **required** parameter for use with *XDI* when data are specified as a function of angle or step count.
  - *Units*: Å
  - *Format*: float

This is the appropriate namespace for parameters of an energy dispersive polychromator. Such parameters may be defined in future versions of this dictionary.

2.7 Defined items in the Detector namespace

- **Namespace:** `Detector` – **Tag:** `i0`
  - *Description*: A description of how the incident flux was measured
  - *Units*: none
  - *Format*: free-format string
- **Namespace:** `Detector` – **Tag:** `it`
  - *Description*: A description of how the tranmission flux was measured
  - *Units*: none
  - *Format*: free-format string
- **Namespace:** `Detector` – **Tag:** `if`
  - *Description*: A description of how the fluorescence flux was measured
  - *Units*: none
  - *Format*: free-format string
- **Namespace:** `Detector` – **Tag:** `ir`
  - *Description*: A description of how the reference flux was measured
  - *Units*: none
  - *Format*: free-format string

Outstanding issues This is one of the areas for which James advocated the use of tables in order to capture a full complement of information about the detectors. For example, an ion chamber might be identified by by any or all of length, gas content, voltage, gap, gas pressure, dark current offset, and details (shaping time, amplification, etc.) about the signal chain behind the detector.

2.8 Defined items in the Sample namespace

- **Namespace:** `Sample` – **Tag:** `name`
  - *Description*: A string identifying the measured sample
  - *Units*: none
  - *Format*: free-format string
- **Namespace:** `Sample` – **Tag:** `id`
  - *Description*: A number or string uniquely identifying the measured sample. This is intended for interoperation with a database or with laboratory management software.
  - *Units*: none
  - *Format*: free-format string
- **Namespace:** `Sample` – **Tag:** `stoichiometry`
  - *Description*: The stoichiometric formula of the measured sample
  - *Units*: none
  - *Format*: see the CIF definition of chemical_formula
- **Namespace:** `Sample` – **Tag:** `prep`

  - *Description*: A string summarizing the method of sample preparation
  - *Units*: none
  - *Format*: free-format string
- **Namespace:** `Sample` – **Tag:** `experimenters`
  - *Description*: The names of the experimenters present for the measurement
  - *Units*: none
  - *Format*: free-format string
- **Namespace:** `Sample` – **Tag:** `temperature`
  - *Description*: The temperature at which the sample was measured
  - *Units*: degrees K, degrees C
  - *Format*: float + units

The Sample namespace is rather open-ended. It is probably impossible to anticipate all the kinds of sample-related metadata that may be useful to attach to data. That said, it would be useful to suggest tags for a number of common kinds of extrinsic parameters.

Here are some other possible tags denoting extrinsic parameters of the experiment along the line of `Sample.temperature`. These may be added as defined fields in future versions of the *XDI* specification.

- `Sample.pressure`
- `Sample.ph`
- `Sample.eh`
- `Sample.volume`
- `Sample.porosity`
- `Sample.density`
- `Sample.concentration`
- `Sample.resistivity`
- `Sample.viscosity`
- `Sample.electric_field`
- `Sample.magnetic_field`
- `Sample.magnetic_moment`
- `Sample.crystal_structure`
- `Sample.opacity`
- `Sample.electrochemical_potential`

Almost all of these examples **should** take a float+units as its value.

2.9 Defined items in the Scan namespace

- **Namespace:** `Scan` – **Tag:** `start_time`
  - *Description*: The beginning time of the scan. This is **recommended** for use with *XDI*.
  - *Units*: time

  - *Format*: ISO 8601 specification for combined dates and times

- **Namespace:** `Scan` – **Tag:** `end_time`
  - *Description*: The beginning time of the scan.
  - *Units*: time
  - *Format*: ISO 8601 specification for combined dates and times

- **Namespace:** `Scan` – **Tag:** `edge_energy`
  - *Description*: The absorption edge as used in the data acquisition software.
  - *Units*: eV (**recommended**), keV, inverse Å
  - *Format*: float + units

This is the appropriate namespace for any parameters associated with scan parameters, such as integration times, monochromator speed, scan boundaries, or step sizes.

An example of a combined date and time representation is `2007-04-05T14:30`, which means 2:30 in the afternoon on the day of April 5th in the year 2007.

2.10 Defined items in the Element namespace

- **Namespace:** `Element` – **Tag:** `symbol`
  - *Description*: The measured absorption edge. This is a **required** parameter for use with *XDI*.
  - *Units*: none
  - *Format*: one of these 118 1, 2, or 3 character strings for the standard atomic symbols (not case sensitive):

```
H  He Li Be B  C  N  O  F  Ne Na Mg Al Si P  S
Cl Ar K  Ca Sc Ti V  Cr Mn Fe Co Ni Cu Zn Ga Ge
As Se Br Kr Rb Sr Y  Zr Nb Mo Tc Ru Rh Pd Ag Cd
In Sn Sb Te I  Xe Cs Ba La Ce Pr Nd Pm Sm Eu Gd
Tb Dy Ho Er Tm Yb Lu Hf Ta W  Re Os Ir Pt Au Hg
Tl Pb Bi Po At Rn Fr Ra Ac Th Pa U  Np Pu Am Cm
Bk Cf Es Fm Md No Lr Rf Db Sg Bh Hs Mt Ds Rg Cn
Uut Fl Uup Lv Uus Uuo
```

  See Wikipedia's list of element symbols.

- **Namespace:** `Element` – **Tag:** `edge`
  - *Description*: The measured absorption edge. This is a **required** parameter for use with *XDI*.
  - *Units*: none
  - *Format*: one of these 28 1 or 2 character strings (not case sensitive):

```
K L L1 L2 L3  M M1 M2 M3 M4 M5 N N1 N2 N3 N4 N5 N6 N7 O O1 O2 O3 O4 O5 O6 O7
```

  See table 10.10 at IUPAC notation for X-ray absorption edges for further explanation. The use of the generic edges *L*, *M*, *N*, and *O* is **not recommended**, but **may** be used for spectra spanning multiple edges.

- **Namespace:** `Element` – **Tag:** `reference`

  - *Description*: The absorption edge of the reference spectrum. This is a **recommended** parameter for use in an *XDI* file containing a reference spectrum.
  - *Units*: none
  - *Format*: same as `Element.symbol`

- **Namespace:** `Element` – **Tag:** `ref_edge`
  - *Description*: The measured edge of the reference spectrum. This is a **recommended** parameter for use in an *XDI* file containing a reference spectrum.
  - *Units*: none
  - *Format*: same as `Element.edge`

2.11 Defined items in the Column namespace

Items in the Column namespace describe single columns of the data table. The first column **must** be the energy.

All tags in the `Column` namespace **must** be integers.

- **Namespace:** `Column` – **Tag:** `1`
  - *Description*: A description of the abscissa array for the measured data. This is **recommended** for use in an *XDI* file.
  - *Units*: eV (**recommended**), keV, pixel, angle in degrees, angle in radians, steps
  - *Format*: word + units
- **Namespace:** `Column` – **Tag:** `N`
  - *Description*: A description of the Nth column (where `N` is an integer) of the measured data. This is **recommended** for use in an *XDI* file.
  - *Units*: as needed
  - *Format*: word (+ units)

The following labels are defined for common array types. `Column.N` items **must** use these labels when appropriate. The array label line at the beginning of the data section of the *XDI* file also **must** use these labels when those columns are present.

Column label	Meaning	choice of units (if required)
`energy`	mono energy	eV / keV / pixel
`angle`	mono angle	degrees / radians / steps
`i0`	monitor intensity	
`itrans`	transmission intensity	
`ifluor`	fluorescence intensity	
`irefer`	reference intensity	
`mutrans`	mu transmission	

Column label	Meaning	choice of units (if required)
`mufluor`	mu fluorescence	
`murefer`	mu reference	
`normtrans`	normalized mu transmission	
`normfluor`	normalized mu fluorescence	
`normrefer`	normalized mu reference	
`k`	wavenumber	
`chi`	EXAFS	
`chi_mag`	magnitude of Filtered $\chi(k)$	
`chi_pha`	phase of Filtered $\chi(k)$	
`chi_re`	real part of Filtered $\chi(k)$	
`chi_im`	imaginary part of Filtered $\chi(k)$	
`r`	radial distance	
`chir_mag`	magnitude of FT[$\chi(k)$]	
`chir_pha`	phase of FT[$\chi(k)$]	
`chir_re`	real part of FT[$\chi(k)$]	
`chir_im`	imaginary part of FT[$\chi(k)$]	

A column containing some other measurement **must** be identified with units when appropriate. For example, a column counting time since the `Scan.start_time` timestamp might be labeled as

```
# Column.N: elapsed_time seconds
```

while a column containing an ongoing measure of temperature as a voltage on a themocouple might be labeled as

```
# Column.N: thermocouple millivolts
```

Outstanding issues Is a file non-compliant if the `Column` fileds are missing? Is it non-compliant if the number of `Column` fields is different from the number of columns? Is it non-compliant if the `Column` field values are different from the words in the column label line at the end of the header?

2.12 Extension fields

Metadata tags carry syntax and may carry semantics. That is, it is possible to have syntactically correct tags that have no definition. Such tags could carry information considered useful by the user or the author of software that, at some point, touches the data.

Such a tag could be an extension within an existing namespace. This has already been discussed in the context of the `Sample` and `Scan` namespaces.

Such a tag could also be part of a new namespace. One application of a new namespace would be to tie a group of metadata tags to a particular application. For example, the data processing program Athena might attach tags associated with the parameters for normalizing the data. That might look something like this:

```
# Athena.pre1: -150
# Athena.pre2: -30
# Athena.nor1: 150
# Athena.nor2: 800
```

These define the boundaries of the pre- and post-edge lines used to determine the edge step of the $\mu(E)$ spectrum.

The use of such extension tags is encouraged for authors of controls, data acquisition, data analysis, and data archiving software.

If an extension tag is not understood due its lack of defined semantics, the **recommended** behavior for software touching the data be to silently preserve the metadata.