IEEE STANDARDS ASSOCIATION

IEEE

P1752 Metadata Subgroup Group Meeting

Sponsored by IEEE Engineering in Medicine & Biology (EMB) Standards Committee

14 January 2020

Teleconference

Members/Attendance

Subgroup chair: Ida Sim, Open mHealth / UCSF
Subgroup secretary: Anand Nandugudi, U Memphis
Call out your name in the following order if you're here (so we can get familiar with your voice)

- Pradeep Balachandran
- Jakob Bardram
- Daniela Brunner
- Christina Caraballo
- Simona Carini
- Paul Harris
- Shivayogi Hiremath
- Sean McConnell
- Leonard Njeru Njiru
- Henry Ogoe
- Paul Petronelli
- Udi Rubin
- Anna T

Action Items
From Last Meeting

Action Items from Dec 17

- F/U with Brian Page on DatapointID [Ida, Sean]
- Investigate the difference between URI and IRI [Simona]
- Model sampling rate so that it can be represented in either Hz or a frequency-unit-value object [Simona]
- AMA BP use case example [Ida]

Drafting Metadata Schema

header.json

schema-id.json

References to External Datasheets

Metadata Landscape

Open mHealth

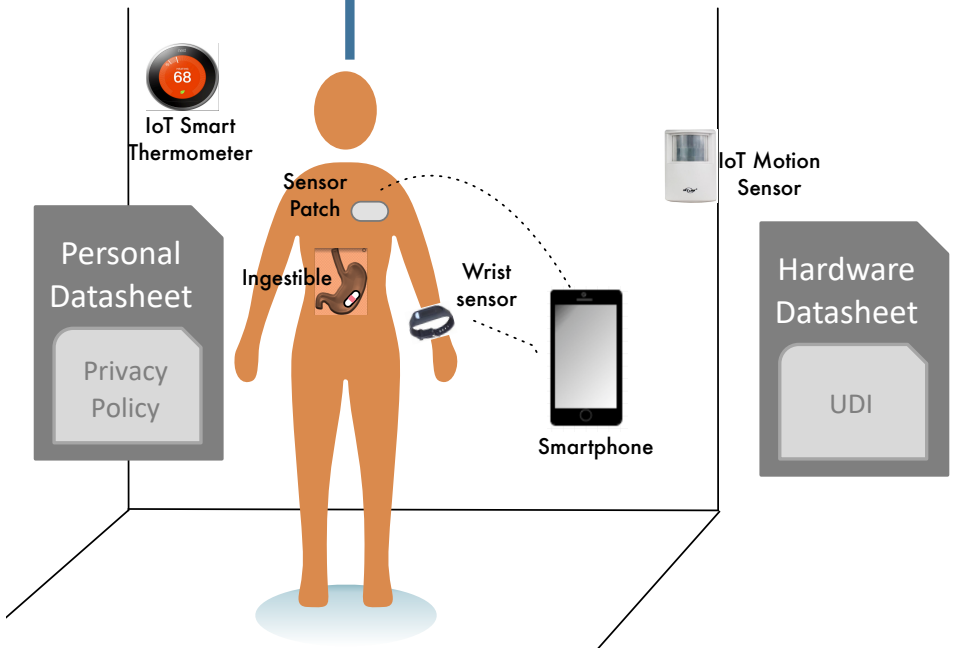
data and metadata standards

Software Datasheet

Dataset & ML Details

Data stream(s) → Firmware/ algorithm → { Header/min metadata } { Body: datapoint }

Cloud or Edge: Runtime acquisition and privacy metadata



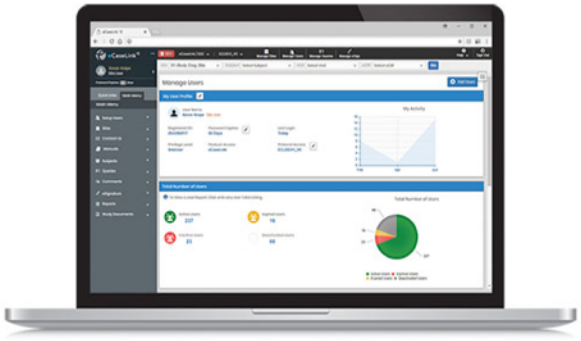
Product Datasheet

Privacy Policy

Electronic Health Record

Study Protocol

Data Sharing Policy



Clinical Trial Management System

Current Datasheet Properties in Header

```
        },
        "software_datasheet": {
    "description":"A URI to retrieve the software datasheet",
    "type":"string",
    "format":"uri"
        },
"hardware_datasheet": {
    "description":"A URI to retrieve the hardware datasheet",
    "type":"string",
    "format":"uri"
        },
"product_datasheet": {
    "description":"A URI to retrieve the product datasheet",
    "type":"string",
    "format":"uri"
        },
"personal_datasheet": {
    "description":"A URI to retrieve the personal datasheet",
    "type":"string",
    "format":"uri"
        },
"study_datasheet": {
    "description":"A URI to retrieve the study datasheet",
    "type":"string",
    "format":"uri"
        }
```

- Cardinality
  - Should assume 1-to-many? E.g., more than one hardware datasheet
- Can also model as
  - External_datasheet
    - Datasheet_type: software, hardware, product, personal, study
    - URI or IRI or unrestricted string (e.g., phone #)

URI vs IRI

Summarizing we can say:

- IRI is a superset of URI (IRI ⊃ URI)
- URI is a superset of URL (URI ⊃ URL)
- URI is a superset of URN (URI ⊃ URN)
- URL and URN are disjoint (URL ∩ URN = ∅)

- URI: The set of characters is limited to US-ASCII excluding some reserved characters. Characters outside the set of allowed characters can be represented using Percent-Encoding. A URI can be used as a locator, a name, or both. If a URI is a locator, it describes a resource's primary access mechanism. If a URI is a name, it identifies a resource by giving it a unique name. The exact specifications of syntax and semantics of a URI depend on the used Scheme that is defined by the characters before the first colon. [RFC3986]
- IRI: Defined similarly to a URI, but the character set is extended to the *Universal Coded Character Set.* Therefore, it can contain any Latin and non Latin characters except the reserved characters. Instead of extending the definition of URI, the term IRI was introduced to allow for a clear distinction and avoid incompatibilities. IRIs are meant to replace URIs in identifying resources in situations where the *Universal Coded Character Set* is supported. By definition, every URI is an IRI. Furthermore, there is a defined surjective mapping of IRIs to URIs: Every IRI can be mapped to exactly one URI, but different IRIs might map to the same URI. Therefore, the conversion back from a URI to an IRI may not produce the original IRI. [RFC3987]

Source: https://fusion.cs.uni-jena.de/fusion/blog/2016/11/18/iri-uri-url-urn-and-their-differences/

“uri/iri” vs “uri/iri-reference”

URI

- "uri": A universal resource identifier (URI), according to RFC3986.
- "uri-reference": A URI Reference (either a URI or a relative-reference), according to RFC3986, section 4.1.

IRI

- "iri": The internationalized equivalent of a “uri”, according to RFC3987.
- "iri-reference": The internationalized equivalent of a “uri-reference”, according to RFC3987

https://json-schema.org/understanding-json-schema/reference/string.html#format

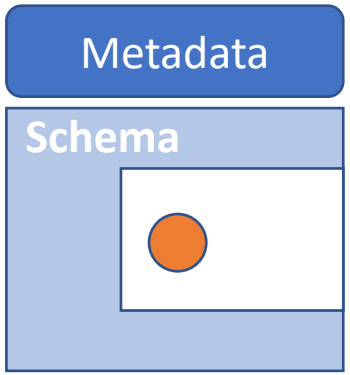
Proposal

- Will not use "uri-reference" or "iri-reference" because these allow relative references, which cannot be guaranteed to be resolvable
- We use “IRI” as preferred reference: The internationalized equivalent of a “URI” (universal resource identifier [RFC3986]), according to RFC3987
- Allow unrestricted strings? E.g., phone number (this is PII -- this creates privacy issues?)

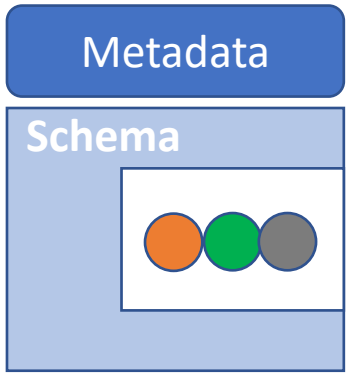
Datapoint ID

Datapoint versus Datapoint series: IDs

- Schema can be used for instances of arrays of observations (i.e. a series) not only a single datapoint
- Metadata must be identical for every data point in the series.
- Is a unique ID assigned to the Datapoint or each observation in the Datapoint series?



Datapoint



Datapoint series

JSON arrays are ordered

Unique ID Options

- UUID (16 bytes; 32-char string)
  - At least 5 different standard versions, some including timestamp and MAC address.
  - Another implementation is GUID, which is still RFC 4122 compliant from Micro$oft.
  - It seems that v-5 is frequently preferred, since it uses SHA-1.
  - Can include a hashed namespace, which could perhaps help with Datapoint series.

  Example: AA97B177-9383-4934-8543-0F91A7A02836

- ULID (16 bytes; 26-char string)

  Example: 01BX5ZZKBKACTAV9WEVGEMMVS0

- Autoincrement-type IDs (often 8 bytes; integer)

  Example: 18446744073709551615

ULID Approach

- 128-bit compatibility with UUID
- 1.21e+24 unique ULIDs per millisecond
- Lexicographically sortable!
- Canonically encoded as a 26 character string, as opposed to the 36 character UUID
- Uses Crockford's base32 for better efficiency and readability (5 bits per character)
- Case insensitive
- No special characters (URL safe)
- Monotonic sort order (correctly detects and handles the same millisecond)

Considerations/Principles?

- How important is security, and the chance (however small) of being able to guess a key / ID?
- This extends to any need for lack of duplication or to avoid potential collisions –
- To what extent does the ID need to be unique, e.g., as a key, for merging data across files or datasets?
- What about the need for sorting the IDs, and time involved in storage/retrieval?
- Do the IDs (ever) need to be URL-safe?

Modeling Acquisition Rate

frequency-unit-value-1.0.json

- Assumptions: Acquisition rate is uniform/regular
- Acquisition rate often but not always in Hz: how to model?
  - Force all rates into Hz equivalents, or
  - Allow other frequencies, e.g., daily, weekly, every 42 seconds

Mininum Metadata: Proposal

Metadata Elements: Datapoint

Needs	Property (bold = required)	Example
Which datapoint is this?	**UUID (datapoint, datapoint series?)**	**Generate using RFC 4122 approach**
What does this value represent?	**schema ID and schema metadata**	**Pointer to the stress datapoint schema**
When is the effective time of this data?	[in the datapoint itself]	

Metadata Elements: Acquisition

Needs	Properties (bold = required)	Example
When was this datapoint first created at the (sensor) source? Recorded or packaged time.	**source_creation_datetime** date-time schema represents a point in time (ISO8601). Timezone is UTC unless otherwise specified	**2019-08-01T07:01:00Z**
Was the datapoint sensed or self-reported?	**modality**	**sensed**
If data was acquired with a periodic rate, what was the rate?	**acquisition_rate**	**Value : 100** **Unit : Hz.**

Metadata Elements: Source

Needs	Properties (bold = required)	Example
What firmware/algorithm? What hardware? What app/product? Which person? Which study?	Pointer(s) to *Software Datasheet, Hardware Datasheet (UDI), Product Datasheet, Personal Datasheet (User ID), Study Datasheet (Study ID)*	Datasheet type {software, hardware, product, personal, study} Pointer: URI

Future Work

Outstanding Items

- Datapoint UUID – Ida & Brian Page
- Draft metadata sample data examples
- Items from Schema Review Calls
  - Filtering
  - Flag identifying raw data

Future Meetings

Upcoming Meetings

- Metadata WG
  - Tuesday, February 4: **9:00 – 10:00** AM Pacific

Adjournment