**FLAT IMPLICIT DICOM JSON INSTANCE (FIDJI)**

**Background**

The DICOM metadata format improves the basic key/value paradigma both in the axis of multivalued attributes and the axis of recursivity of composition. A traduction of the model into XML opened up a discrete access to specific metadata thanks to XPATH. Yet another traduction of the model into JSON simplified the parsing of subsets of metadata in QIDO responses.

Both the XML and JSON traductions were derived from the explicit binary syntaxis, wich implies:

1. encapsulated datasets
2. multivalued attributes
3. comunication of the datatype of each attribute

(1) and (2) are implemented with XML sequences and JSON arrays. (3) is implemented with XML attributes and an additional layer of JSON object description. These constraints are counter productive for data analysis tasks (for instance machine learning), which are performed more easily on flat datasets.

**Description of the FIDJI format**

This is why we designed yet another DICOM JSON format we called FIDJI (FLAT IMPLICIT DICOM JSON INSTANCE) :

* FLAT means the use of one array of key/value only
* IMPLICIT means that attribute's keyword and datatype are left outside of the format. The Dictionary found in Part 6 of the DICOM standard, with eventual private extensions deal with these specifics.
* DICOM refers the same modelization shared by the already existing binary, xml and json representations, with the corollary that it is posible to translate FIDJI from and to any of the other representations.
* JSON is the core grammar used. JSON is easier to parse than XML and easy to search when the objects are only one level deep, because most languages contain functions to work on maps (key value lists)
* INSTANCE (singular) indicates that the format gathers metadata for a unique SOP Instance (fully or partially represented). A DICOM Study would be represented by a set of FIDJI objects.

**Specifics**

* encapsulated attributes are treated in accordance to the DICOM QIDO parameter format, which defines aaaaaaaa.bbbbbbbb as the attribute designed by tag bbbbbbbb included in the (first) item of sequence designed by tag aaaaaaaa.
  + in qido syntaxis, tags and corresponding keyword are interchangeable. In fidji, as in dicom json tags only are used.
* aaaaaaaa.bbbbbbbb represents the complex concept of the attribute, but not its place within an item o a list of values. We append the latter information with a \_ (underscore) after the conceptual part. For instance aaaaaaaa.bbbbbbbb\_2.1 means the first value of attribute bbbbbbbb of item 2 of the sequence aaaaaaaa.
* in many cases the sequence contains a unique item containing a monovalued attribute. In theses cases, the appendice can be omitted. For instance aaaaaaaa.bbbbbbbb equals aaaaaaaa.bbbbbbbb\_1.1
* value formats follow Table F.2.3-1. DICOM VR to JSON Data Type Mapping of the DICOM standard, with two exceptions:
  + PN, which components are represented in only one string made of segments separated by ^ and =
  + the only SQ attribute to be represented is the empty one, using json null as its value.
* when it is better to access binary fields by reference than coded base64, we use a json object as value. This object contains a "url" key and eventually other keys usefull for a request.

**Example**

The first item of the example of QIDO-RS Search For Studies found in <http://dicom.nema.org/medical/dicom/current/output/html/part18.html#sect_F.2.4> would be written in fidji as follows:

[

{

"00080005": "ISO\_IR 192",

"00080020": "20130409",

"00080030": "131600.0000",

"00080050": "11235813",

"00080056": "ONLINE",

**"00080061\_1": "CT",**

**"00080061\_2": "PET",**

"00080090": "^Bob^^Dr.",

"00081190": "<http://wado.nema.org/studies/1.2.392.200036.9116.2.2.2.1762893313.1029997326.945873>",

"00090010": "Vendor A",

"00091002": "z0x9c8v7",

**"00100010": "Wang^XiaoDong=王^小東",**

"00100020": "12345",

"00100021": "Hospital A",

"00100030": "19670701",

"00100040": "M",

**"00101002.00100020\_1.1": "54321",**

**"00101002.00100020\_2.1": "24680",**

**"00101002.00100021\_1.1": "Hospital B",**

**"00101002.00100021\_2.1": "Hospital C",**

"0020000D": "1.2.392.200036.9116.2.2.2.1762893313.1029997326.945873",

"00200010": "11235813",

"00201206": 4,

"00201208": 942

}

]

**Evaluation**

* more compact than dicom json
* easier to parse than dicom json
* more searchable than dicom json
* could be used as storage format, eventually in nosql databases in order to avoid the duplication of the information with a copy of the raw objects into a repository and a copy of selected metadata into a registry
* usefull for optimized machine learning analysis