**Requirements for new File Format for storing whole study**

1. Group Structure
   1. /archive/patient/ study/series/instance/value
      1. /archive/patient/dataset
      2. /archive/patient/study/dataset
      3. /archive/patient/study/series/dataset
      4. …
   2. Value: element/sequence
   3. Element: id (number/keyword), type, value
   4. Sequence id (number/keyword), Dataset List
   5. Dataset: ordered list of elements & sequences
2. Datatypes (User)
   1. Element
   2. Sequence
   3. Uid/Uuid
   4. ??
3. It must be possible to add one SOP Instance at a time
   1. Metadata
   2. Bulkdata
   3. Pixel data
4. Partial load of metadata only
   1. Retrieve all metadata for patient or study at once
   2. Selective load of bulkdata (large object)
      1. By Study
      2. By Series
      3. By Instance
      4. Pixel Data only
      5. Report only
         1. Simple
         2. Structured (XML)
5. Distinguish between simple element and sequence
   1. Simple Element
      1. tag, type, value list
   2. Sequences
      1. Currently nested
      2. Would like syntax for flat sequences
         1. Id = sq.sq.sq.item.tag

Questions

1. How to make a schema/ data model? dataspace
2. Sequence of arrays of different dimensions
3. How to write an interface for Dart

C Library

File

One-half

Deep Learning/HP Computing use HDF5

Competitor of HADOOP

Other-half

PDF of Science

Self-describing data

Hybrid files – dicom followed by HDF5

Design of HDF5 DICOM

**Terminology**

**Element** Equivalent to Simple Element

**Element Type**

A non-Sequence Element. Simple Elements have the following properties:

Index A non-negative integer that identifies the Element

Tag A Uint32 value that uniquely identifies the Element

Keyword A UTF-8 String that uniquely identifies the Element ([A-Za-z] [A-Za-z0-9]\*)

Name A UTF-8 String that allows whitespace

Data Type A combination of the DICOM VR and VM. The Data Type include Shape information

Presence Type 1, 1C, 2, 2C, 3

IsRetired true/false

In HDF5 elements are encoded as an index/value pair where the index specifies the Type including Data Type of the Element.

**Simple Elements**

Element Type Identifier

Value – 0 or more values depending on the Data Type

**Sequence Element**

Element Type Identifier

Items – 0 or more Items

**Dataset**

A Map<identifier, value>

**Item extends Dataset**

Parent Sequence

**Issue: How to encode Private Elements** (i.e. Manufacturer defined Elements)

**Issue**: Terminology

* Dataset?

Questions:

* How efficient are groups?
* How to implement data structures

Exercise: Create and Instance Representation

1. Define a Simple Element Data Type: tag, vrIndex, values[]
2. Define a Simple Dataset Data Type: array of element: Element []
   1. Each tag can only appear once
3. Create Elements and add to Dataset
4. Define an Item which is an extension of Dataset the include a reference to:
   1. The Parent Sequence
   2. The Parent Dataset
5. Define a Root Dataset where Parent Sequence and Parent Dataset are Null
6. How to define a variable length array with type and shape,
   1. Shape has a width and a min and max number of elements. Sometimes max is 2^32
   2. That is there are 1, 2 and 3 dimensional arrays.
7. Define an Element Type (index, code, keyword, name, datatype, …)
   1. Public Elements
   2. Private Elements
      1. Definer Type (name, manufacturer, equipment, …)
      2. Define Private Dataset
8. Update Dataset to include an array of 0 or more Private Datasets

Questions:

* Are there canonical null, true and false values?
* How to represent a simple record
  + Index Uint16
  + Value array of DataType
* Shape:
  + (1, min:max)
  + (2, min:max)
  + (3, min:max)
* Hierarchy
  + Study
    - Metadata
    - Series []
  + Series
    - Metadata
    - Instance []
  + Instance
    - Metadata
    - Data (image or document)

Study

Study Metadata

Series[]

Series

Series Metadata

Instance Metadata[]

Image[]

Report []