

# **CytometryML Data List with Relationships**

## **ISAC CYTO 2012 (B84 190)**

Robert C. Leif\*<sup>a</sup> and Stephanie H. Leif<sup>a</sup>

<sup>a</sup>XML\_Med, a Division of Newport Instruments,  
3345 Hopi Place, San Diego, CA 92117  
rleif@rleif.com;

<http://www.cytometryML.org> (An extended version of this poster is available at this web site.)

### **ABSTRACT**

CytometryML is an XML schema based translation, extension, and amalgamation of the DICOM and ISAC standards. CytometryML consists of 5 major XML schemas: Relations, Series, Instance, Instrument, and Specimen; it also includes Image, and List-Mode schemas. Series metadata, which is specific for an entire collection of images and/or list-mode files produced by a single instrument and derived from a single specimen, is stored together with related metadata files in an EPUB container (ZIP) file. Each Instance container file includes binary image and/or list-mode files together with related metadata files that are specific for a single or closely related group of instrument runs from a single specimen. The ISAC Archival Cytometry Standard (ACS) proposed Table of Contents schema including its Resource Description Framework (RDF) capabilities has been extended, modified, and renamed for use in the Instance schema. The replacement of standard RDF syntax by a simple sentence (element) based format (Subject, Predicate, and Object) permits multiple relations between two file references that can be in both directions.

### **INTRODUCTION**

- The (ISAC) Data Standards Task Force (DSTF) is now finishing work on the “Archival Cytometry Standard (ACS), which has been developed to bundle data with different components that describe cytometry experiments”.
- The ISAC ACS container is a ZIP file that archives the binary data and related metadata files obtained from one or more cytometry data acquisitions of list-mode, image or similar data.
  - The binary data can include the original binary data produced by the instrument and the binary files that are part of the sequence that results in data suitable for the display of cellular populations.
  - The control data, if used in the processing of the data, can also be included.
- The ACS container file includes an XML-based Table of Contents, which specifies file locations and presentation order of files and relations among files.
- Since EPUB files have wide acceptance because they are the existing standard for e-books including scientific journals, the ACS container file design has been modified and extended to be an EPUB Open Container Format (OCF) 3.0 file. An EPUB container file includes its own table of contents, toc.ncf, which is the Navigation Control file and specifies the order of presentation of the files that are visible to the reader. To avoid confusion, the CytometryML Table of

Contents was renamed to List of Relations. EPUB files have the capacity to include multiple HTML, XML, and binary documents that contain the data in an easy to read format.

### Code Fragment 1 navMap Element from an EPUB Table of Contents (Navigation Control file)

```

1 <navPoint id="File2" playOrder="3">
2   <navLabel>
3     <text>File2 (Compensated Data)</text>
4     </navLabel>
5     <content src="http://
6       www.ACS_Image_Data/25Apr2012/File2.html"/>
7   </navPoint>

```

- Code Fragment 1 is an EPUB navPoint element of a ToC Navigation Control file. This means the html collection of pages that describe the compensated data are in the third section of the html document.
- The present design of the CytometryML relations schema is substantially based on the former CytometryML version of the ACS Table of Contents (ToC) schema. The relations schema includes some of the functionality of the Resource Description Framework (RDF) to document relations.

**Table 1 Differences between ACS and CytometryML files**

ACS	CytometryML
Is Unique	Follows and extends the DICOM Standard & EPUB
One Type of XML file. Equivalent to the Multiple CytometryML files.	One Series & 1 or more Separate Instance Files
Complex elements in the Tree Structure	Multiple relatively simple elements in the Tree Structure
Raw Data First	Most Relevant (processed) Data First
ACS container files large & monolithic	CytometryML smaller modules
Monolithic Large --Slower Transfer	Modular Small --Faster Transfer
Emphasis on data analysis	Emphasis on instrumentation and histo-cytochemistry
Table of Contents	List of Relations
Designed for cytometrists	Designed for clinicians, end-users, and cytometrists

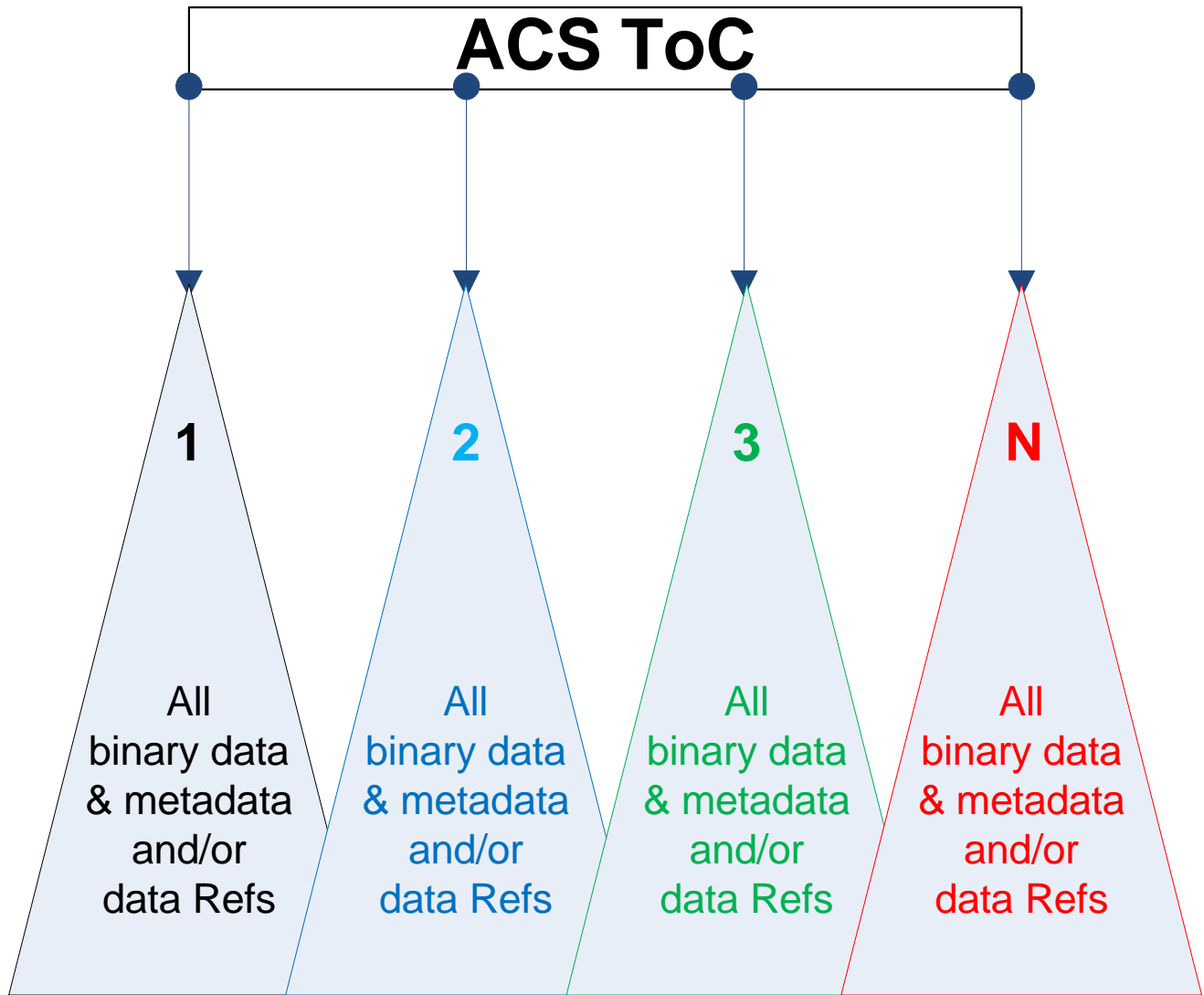


Figure 1, Structure of an ISAC Archival Cytometry Standard (ACS) table of contents. One or more XML ToC files is located at the apex of the ACS ZIP file. In this case, N ToC files are shown. The triangles represent references to trees of file descriptions. Each of these file description trees can consist of multiple file description trees.

## CYTOMETRYML ARCHITECTURE

- This poster will describe the elements of the new relations schema, which are imported into (made visible) by the Instance and Series schemas.
- Figure 2 is an extension of a previously described model. This separation into Series and Instances limits the number of trees in any container. A Series file does not contain any binary data. In many cases, the binary data files in an Instance container file can be limited to one measurement, its controls, and processed data files.

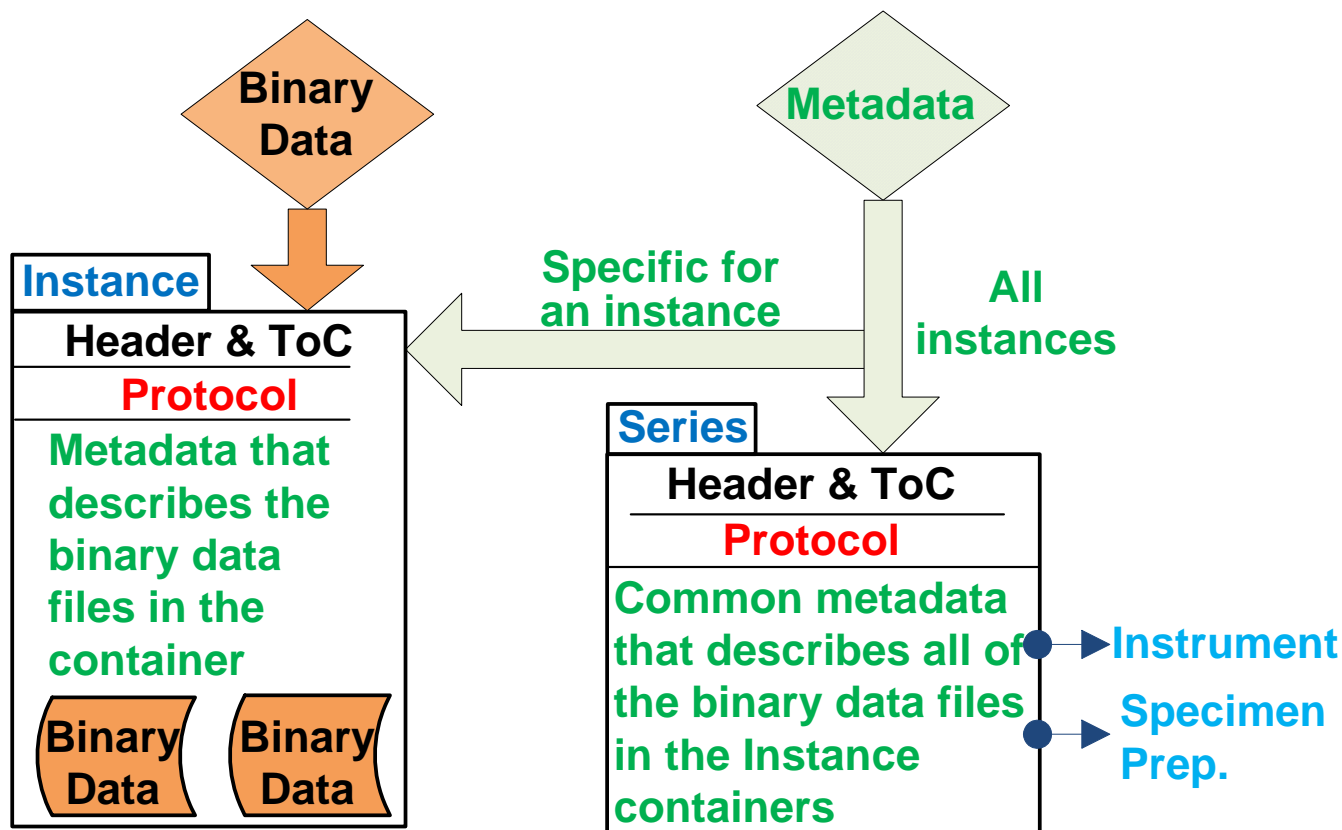


Figure 2 Diagram showing the placement of binary data and metadata into the Instance and Series containers. The Instance\_Data\_Type (left) and Series\_Data\_Type (right) and their corresponding elements each contain Header Information, a list of relations, Relation\_List, and a description of the Protocol that contains the metadata necessary to analyse the data and eventually to repeat the measurement.

- As is shown in Figure 2, the Series container includes or points to protocol information files that contain information that is relevant to all Instance files.
  - including the description of the fixed components of the Instrument and all or parts of the Specimen Preparation except those that are specific for an instance.
- There are three main types of files in an Instance container:
  1. binary data files, such as FCS, TIFF, JPEG, or DICOM image;
  2. metadata files that describe the content of the binary files and include hyperlinks to the descriptions of other binary files and other metadata files; and
  3. the other metadata files, such as the descriptions of compensation and thresholding (scene segmentation), describe processes and conditions that are relevant to the binary data files.
- Each of the Instance container files includes at least one xml file that describes and includes a hyperlink to a binary data containing file. The first of these metadata files is deemed to be of greatest interest to the end user.
- An Instance container file normally would include other binary data containing and metadata files. For example, in the case of clinical data, the top element would describe the binary Data\_File that was or will be used for performing the diagnosis and the Related\_Data\_File

could be the one that was originally produced by the instrument. In the case of research data, the top element could be the original Data\_File or the file produced by compensation of the original Data\_File. The individual Instance file containers include related binary or metadata files and a relation list that describe of their relations. For example, the relations could document the paths of the processing steps going back from the top element diagnostic file to the binary file that was initially produced by the cytometer.

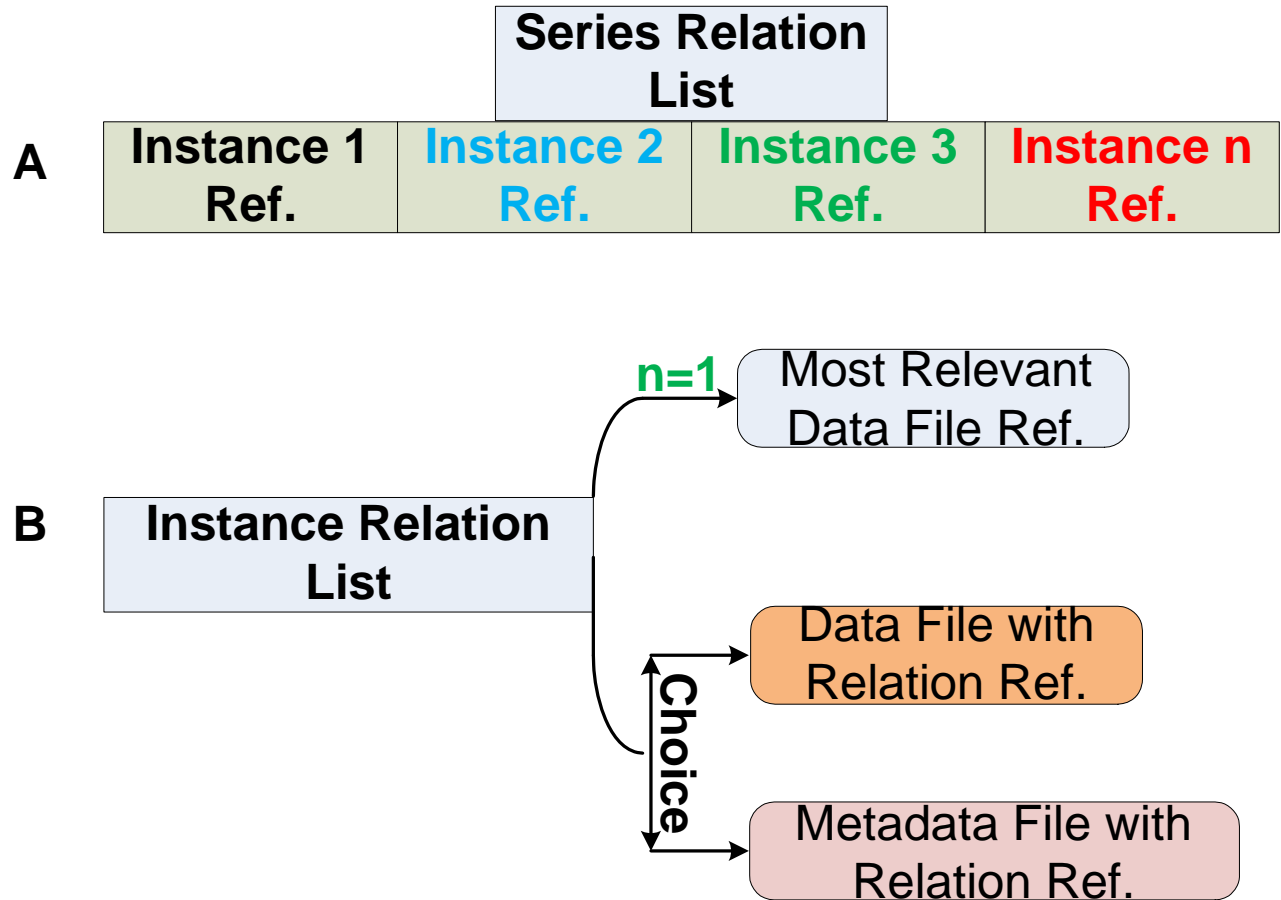


Figure 3., CytometryML Design of Relation\_List elements. As shown at the top of the figure (A), the Series Relation List consists of a list of references that include references (hyperlinks) to the locations of the Instance files. As shown in the middle and bottom of the figure (B) and described below, an Instance Relation List is a significantly more complex structure. Both binary and meta-data files can have a relation with another file

## Doubly Linked List

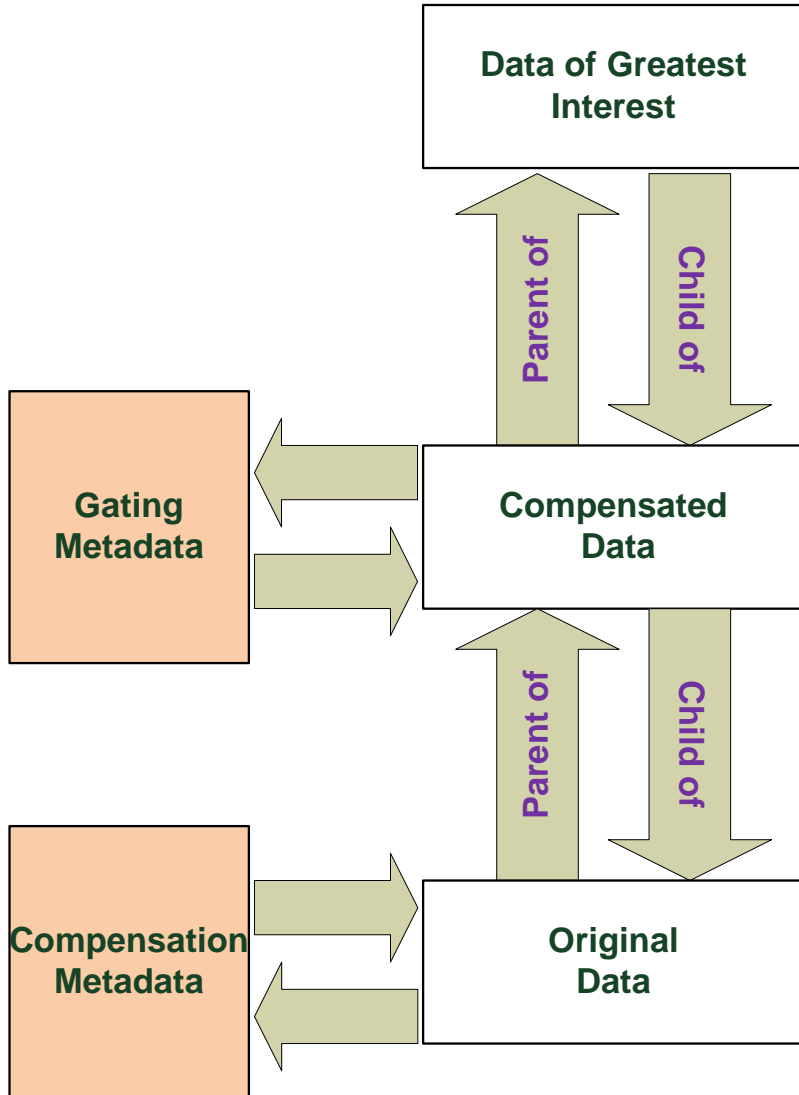


Figure 4 is a diagrammatic representation of the relations between 3 binary data files and the relations between two pairs of a metadata and two of these data files. The metadata that describes a transformation is bidirectionally joined with the data file that it transforms. The presence of multiple relations in either direction permits the binary data files to be organized as a doubly linked list, which permits the human user to sequentially navigate in both directions between related files.

Since hyperlink navigation is essentially how web pages are traversed, it is based on an efficient, reliable well developed technology that describes the various cell populations present. For each binary file except for the Data of Greatest Interest and the Original Data, links (fat arrows) to both a parent and a child data file are shown. Since the Data of Greatest Interest is the last file, it does not have any children and the Original Data file does not have any parents.

The original image binary data (Original Data file) is acted upon as described in the Compensation Metadata. Both the Compensation Metadata and the reference to the Original Data file can have references to each other. Similarly the Compensated Data reference and the Gating Metadata can have references to each other.

## MATERIALS AND METHODS

- Data-types present in the XML schemas and XML pages were those prepared by domain experts,
  - Reused from Digital Imaging and Communications in Medicine (DICOM) standard (<http://medical.nema.org/>) or Flow Cytometry Standard, FCS3.1 ([http://www.isac-net.org/images/stories/documents/Standards/fcs3.1\\_normativespecification\\_20090813.pdf](http://www.isac-net.org/images/stories/documents/Standards/fcs3.1_normativespecification_20090813.pdf)).
- The DICOM) standard is a FDA Class II device.
- The following software techniques were employed to maximize safety:
  1. Priority of readability was maximized,
  2. Adherence to naming conventions,
  3. Code modularization,
  4. Minimization of coupling between major schemas,
  5. Maximization of cohesion of individual schemas, and
  6. Reuse of existing CytometryML XML schemas.
- Since the code is presently compatible with XSD version 1.1, it can be extended to include assertions, which provide extra checks on the correctness of the code.
- The XSD schemas were validated by oXygen (<http://www.oxygenxml.com/>) and XMLSpy (<http://www.altova.com/>). Many of the schemas have also been tested with XSD 1.1. A new XML page was subsequently produced from each of the main schemas and then filled with reasonable values and validated against its schema
- The Resource Description Framework (RDF) (<http://www.w3.org/> RDF is a way of documenting the relations between objects.
  - These objects can be represented by URIs. Most embodiments of RDF technology employ schemas that are different and often incompatible with the XML Schema Definition Language (XSD), which is the basis of CytometryML and the ACS schemas.
  - RDF presently uses a special set of attributes. Since it is useful to describe relations employing RDF in standard XML schemas, such as those that comprise CytometryML, a group of relation elements has been created including a hypertext link. The change to elements, which can generate or validate unlimited element value pairs is superior to the use of attributes since it permits more than one relation between entities.

## RESULTS

- Because EPUB files already include a ToC named toc.ncx, a Relation\_List element has replaced the ACS and CytometryML ToC. This Relation\_List is part of the Relations schema and has been included in Series and Instance schemas and prototyped for an EPUB file that is essentially an extension of the ISAC standard ACS Container file. Effectively, the EPUB Container file design has been simplified from the ACS format by following the DICOM design, which divides the content of the ACS container into a Series and Instances hierarchy.
- The use of the DICOM Series and Instance objects significantly simplifies this design
  - by limiting the information in an Instance file container to that contained in one mea-

surement or a closely related group of measurements.

- The elements located within a Series file container describe information that is relevant to all the Instances.
- An element located within an Instance XML page is specific for that Instance.

The XML page (Code Fragment 2 & Code Fragment 4) describes an image file that was compensated as described in the Compensation\_Metadata element, which is included in the Instance schema and consequently an instance file.

## Code Fragment 2

```
<?xml version="1.0" encoding="UTF-8"?>
1<instance:Instance_Metadata xsi:schemaLocation="http://
  www.cytometryml.org/ACS/instance instance.xsd"
  xmlns:comp="file:///cytometryML/compensation/"
  xmlns:test="file:///ACS_Image_Data/7May2012/"
  xmlns:relations="http://www.cytometryml.org/ACS/relations"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
```

- A Compact URI or CURIE consists of a prefix, such as test, which, as shown above in Code Fragment 2 can be declared as a standard XML namespace (xmlns) attribute (Element 1). As with all xmlns prefix attributes, it is followed by an equals and a reference starting with http or file. In this case the prefix, test” has been declared (xmlns:test). The prefix, test, is used below in elements 3, 6,13, of Code Fragment 3. The CURIE is replaced with a concatenation of the value represented by the prefix and the part after the colon. As is shown in Element 3 of Code Fragment 3, the reference is File3.html. Thus, the value of the URI is file:///ACS\_Image\_Data/7May2012/File3.html. Code Fragment 2 above is the beginning of the XML page shown below from which all but three of the attributes required to make visible the XML namespaces (xmlns) have been omitted.

## Code Fragment 3

```
1<relations:Most_Relevant_Image_Data_File_Ref
  mimeType="application/xhtml+xml" id="File3">
2  <relations:Curie_Link>
3    <links:a href="test:File3.html/">Instance3</links:a>
  </relations:Curie_Link>
4  <relations:Parent_Location idREF="File2">
5    <relations:Curie_Link>
6      <links:a href="test:File2.html">Instance2</links:a>
    </relations:Curie_Link>
  </relations:Parent_Location>
```



```

7   <relations:Role>
8     <relations:Data_Role_Std>Data_Of_Greatest_Interest
9     </relations:Data_Role_Std>
    </relations:Role>
10  <relations:Data_Role_Std>Gated_Img</relations:Data_Role_Std>
    </relations:Role>
11  <relations:Additional_Info>This is the image that should be
    looked at by the person who analyzes the data
    </relations:Additional_Info>
12  <relations:figure id="Image3">
13    <components:Img src="test:/images/Img3.dcm/"
        alt="Image of a europium complex bound to CD 34 positive
        cells" name="Image3" height="48" width="64"/>
14  <components:figcaption>
    Image 3: time-gated image of a Centrifugal Cytology
    preparation of cells with an anti-CD34 antibody to which
    has been bound a europium complex label and had been
    counter-stained with DAPI
    </components:figcaption>
    </relations:figure>
</relations:Most_Relevant_Image_Data_File_Ref>

```

Code Fragment 3 is the description of the reference to the binary data file that contains the most relevant data to the investigator or physician. It is the one that describes the final binary data produced by the processing steps. It is part of a simplified example of an XML page based on a schema that includes Relations\_Item element. This XML page would be included in the Instance container. The XML page example of this element includes descriptions of three binary containing files: Most\_Relevant\_Data\_File and of two Data\_File\_W\_Relation. It also includes two XML based metadata files, which describe the gating and compensation processing. Many of the values are those used in Spidlen et al.

The XML code in Code Fragment 3 starts with Element 1, which is the relations:Most\_Relevant\_Image\_Data\_File\_Ref mentioned above. The code ends with the closure of Element 14. Element 1 includes the mimeType attribute of the first Image Data File referenced in the Instance page. The value of the mimeType is given as xhtml+xml based on the hope that XML elements will be included in html5 or xhtml5, please see section Conclusions. The types of files included are documented by the value of the mimeType attribute, which does not constrain their content, which in theory could be anything. However, the files content can be constrained by employing elements based on data-types already present in the CytometryML schemas that specifically match individual file types. The URI of the most relevant binary data containing file is provided as the value of the hypertext reference (href) in Element 3. This URI ends with the file's name. The number that ends the file name is sequentially incremented as the data in the file is processed. It starts with 1 for the original (raw data) and is incremented by 1 with each processing step. Thus, the result of a process has a greater valued number (Element

3) than that of the original raw data, File1.html (not shown) and File2.html for the compensated data (, Element 4). The data structure for Code Fragment 3, Element 3 is based on an element in the XLink specification. Activation of this hypertext reference opens the data file and similar links open the metadata files described in Code Fragment 5 . The value of a links:a element, such as Element 3, should be selected to have meaning to the user; it need not have anything to do with the address of the link. It is the string that will be shown to the user as the link activation location. Code Fragment 3, Element 8 indicates that this file has the role of containing the binary data that was expected to be of greatest interest to the user and Element 10 indicates that this data file also has the role of containing the gated data, which permits classification of the cells. The addition of Role elements, which are based on the Role\_Type in relations.xsd limits permits specific descriptions of data. Specifying a unique Role for a binary data containing file or XML page provides an implicit relation (description) of itself. For example, the Role, Data\_Of\_Greatest\_Interest (Element 8), means that the Data\_File referenced by this element is of greater interest than all of the other Data\_Files. In order to permit Roles other than those enumerated in the standard, a choice between relations:Role\_Std\_Type and xlink:roleType is permitted. Element 11 provides a place for a free text input. Element 12 contains an element 13 that imports an image into the XML page, and Element 14 provides a caption for the figure. The Role and Additional\_Info elements are optional and have been included in this and the other three file descriptions. An optional Signature element is also available but because of its size has not been included in Code Fragment 3, and those that follow. The inclusion of either a classic URI, CURIE equivalent (Element 2), or idREF is mandatory. If an idREF is used, the material described in the schema can be kept together in a single XML or xhtml page.

### Code Fragment 4

```

1 <relations:Image_Data_File_W_Relation_Ref mimeType=
  "application/xhtml+xml" id="File2">
2 <relations:Curie_Link>
3   <links:a href="test:File2.html">
      Compensated_Data</links:a>
  </relations:Curie_Link>
4 <relations:Role>
5   <relations:Data_Role_Std>
      Compensated_Binary_Data
  </relations:Data_Role_Std>
  </relations:Role>
6 <relations:Relation>
7   <relations:Subject>
8     <relations:Self>Self</relations:Self>
9   </relations:Subject>
10  <relations:Predicate_Std>is parent of

```

```

11 </relations:Predicate_Std>
12 <relations:Object idREF="File3">
13   <relations:Curie_Link>
14     <links:a href="test:File3.html">Image of the segmented data
15     </links:a>
16   </relations:Curie_Link>
17 </relations:Object>
18 </relations:Relation>
19 <relations:Relation>
20   <relations:Subject>
21     <relations:Self>Self</relations:Self>
22   </relations:Subject>
23   <relations:Predicate_Std>is child of</relations:Predicate_Std>
24   <relations:Object idREF="File1">
25     <relations:Curie_Link>
26       <links:a href="test:File1.html">
27         Original data</links:a>
28     </relations:Curie_Link>
29   </relations:Object>
30 </relations:Relation>
31 <relations:Additional_Info>This compensated data is subsequently
32   segmented into different populations to produce the image of
33   Greatest interest,Img3.dcm
34 </relations:Additional_Info>
35 <relations:figure id="Fig2">
36   <components:Img src="test:/images/Img2.dcm"
37     alt="Compensated Image of europium and DAPI labeled cells"
38     name="Compensated Image" height="48" width="64"/>
39   <components:figcaption>The overlap of the europium emission and
40     the DAPI stain is minimal and with time-gating is
41     insignificant. However, if a third conventional fluorochrome
42     were introduced, compensation would be necessary.
43   </components:figcaption>
44 </relations:figure>
45 </relations:Image_Data_File_W_Relation_Ref>

```

Code Fragment 4 above provides a limited description of a second binary data containing image instance file (File2). Element 3 is a hyperlink to this file. The reason for the inclusion of this Image\_Data\_File\_W\_Relation\_Ref (Element 1) is that its Role (Element 5) is being the Compensated\_Binary\_Data that was obtained by the transformation described in the original data produced by the cytometer (not shown). The number that ends the Compensated\_Data Instance file name, File2, is lower than that of its child (Element 11), the Image of the segmented data (Most\_Relevant\_Data\_File), File3, because of the Compensated\_Data file parent relation (Elements 6-13) and child relation (Elements 14-20). Thus, this description is of a Compensated\_Data file that is an element of a doubly linked list (Elements 6-20). This data structure permits a RDF type relation between the Compensated\_Data file and the Most\_Relevant\_Data\_File as well as a second relation between the Compensated\_Data file and the original data. It differs from RDF by being based on an element, which is derived from a complexType instead of being an attribute, which is based on a simple Type. The Relation\_Type complexType is based on the unambiguous structure of a simple sentence. This complex-Type consists of three elements: Subject, Predicate, and Object. As is the case for the Role element there is a choice of the enumeration in Predicate\_Std\_Type or user defined values in Predicate\_Other\_Type. In the case of Element 8, the Subject is "Self", which is equivalent to File2. The Predicate (Element 10) is "is parent of" and the Object (11) is "File3". This unambiguously states that Self (File2) is the parent of File3. The second relation (Elements 14-20) has a similar structure. In this case the Subject (Element 16) is Self ("file2"); the Predicate (Element 17) is "is child of" and the Object (Element 18) is "File1". The use of both the parent of and child of relations permits the Instance(s) to be in the form of a linked list. Since the subject of a relation is often the material in the relation, a Self element was introduced into the design because a hyperlink is not required to remain in the same place. The use of the Self element also decreases the size of the file and increases readability. A more flexible and more complex alternative exists where the data structures for the Subject and Object of a Relation element both have the data structure shown for the Object. In this case the Subject element has a hyperlink that is not local and the Object has the local link. Thus, relations can be in either direction.

The doubly linked-list data structure described in Code Fragment 4 can serve as the conceptual basis for other schemas that describe staining or specimen preparation.

### Code Fragment 5

```

1 <relations:Metadata_File_W_Relation_Ref mimeType="text/xml"
  UID_Value="1.111.211.121" id="compMay12.xml">
2 <relations:Curie_Link>
3   <links:a href="comp:Comp6May12.xml">Compensation Description
   </links:a>
   </relations:Curie_Link>
4 <relations:Role>
5   <relations:Data_Role_Std>Compensation description including

```

```

    Compensation Matrix</relations:Data_Role_Std>
  </relations:Role>
6 <relations:Relation>
7   <relations:Subject>
8     <relations:Noun_Ref idREF="File1">
9       <relations:Curie_Link>
10        <links:a href="test:File1.html">Original Data</links:a>
11        </relations:Curie_Link>
12        </relations:Noun_Ref>
13      </relations:Subject>
14    <relations:Predicate_Std>was compensated by
15    </relations:Predicate_Std>
16    <relations:Object idREF="compMay12.xml">
17      <relations:Curie_Link>
18        <links:a href="comp:Comp6May12.xml">Compensation Description
19        </links:a>
20      </relations:Curie_Link>
21    </relations:Object>
22  </relations:Relation>
23 <relations:Additional_Info>This Relation has made the file that was
24 compensated the subject and the metadata file that describes the
25 compensation and contains the matrix the object.
26 </relations:Additional_Info>
27 </relations:Metadata_File_W_Relation_Ref>

```

In subsequent Relations, such as that shown (Element 1) in the Metadata\_File\_W\_Relation\_Ref, The value of the mimeType attribute (Element 1 is text/xml, which is reserved for xml documents. The URI for this metadata file is expressed as a CURIE (Element 3 ).The metadata file is identified by a link (Element 3) that is based on a Curie href attribute value. The structure of the relation (Elements 6 to 14) is the inverse of the one shown in Code Fragment 4 . The Subject (Element 7) is an external entity (Element 8), test:File1.html and the Object is the Metadata file (Element 14). The value of the Subject is an external file reference (Element 10) to “test:File1.html, which contains the original binary data. The value of the Predicate (Element 11) is was compensated by. The value of the Object (Element 12) is an idREF to Comp6May12.xml and Element 14 which includes a link to the same Comp6May12.xml. Comp6May11.xml is the metadata file itself. This relation is described by the sentence” File1.html was

compensated by Comp6May12.xml. Thus, it has been possible in CytometryML to describe a relationship as either something done by the subject to the object or by the object to the subject.

## CONCLUSIONS

The ACS design for a Table of Contents (ToC) has been extended and modified in CytometryML to an XML schema datatype, `Relation_List`, that is appropriate for the DICOM Instance data structure. The ToC (`Relation_List`) has also been extended to include a `Role` element and `Relation` elements based on an extended `Relation_Type` that includes the grammatical contents of a simple sentence, which disambiguates the RDF expression including its direction and also permits multiple relations with different directions between two files. This combination of `Role` and enhanced `Relation` types should provide a much richer vocabulary to describe objects and their relations than present RDF. The use of hypertext links will provide a simpler more user friendly means of traversing the data than the standard database query approach. The CytometryML design and that of Spidlen et al. have the advantage that since they have been created in XSD, they can be easily imported into other XSD schemas.

The CytometryML approach differs from potentially complex structure proposed Spidlen et al. by being a description that is limited to a single Instance or a small closely related group of instances. This permits the `Relation_List` element to be either a single or a small number of doubly linked lists. The order of these lists is based on starting with the element that contains the description of the binary data of greatest interest to the target user. In many cases, this element will be the only one of interest to the user.

The CytometryML design and that of Spidlen et al. have the advantage over both DICOM and FCS that since they have been created in XSD, they can be easily imported into other XSD schemas. The separation of the descriptions of the metadata and binary data from cytometry measurements into a `Series`, which primarily contains metadata that is applicable to multiple Instances, and Instances each of which contains a limited set of binary data, will decrease the amount of extraneous data transmitted. This decrease will, as experienced with DICOM, provide improved response times and lower costs.

In this work, the reuse of an ACS construct in a DICOM based data structure has been demonstrated, which is an extension of previous reuse of DICOM data structures in CytometryML. This leads to the conclusion that the ACS and CytometryML should both reuse DICOM; and DICOM should reuse ACS and CytometryML. These standards should be harmonized at the datatype semantics level. The harmonization of these standards and similar work on harmonizing other medical standards and translating them into XML will significantly assist in the goal of creation of the capability for universal data exchange.

The combination of the use of xhtml5, CSS, and EPUB for medical informatics including ISAC standard(s) produces universal, portable Web applications. It also has the added benefit that if Cytometry were published as an EPUB document, the transfer of information from data files to Cytometry articles and the checking of the MIFlowCyt information could be semi-automated.

## ACKNOWLEDGMENT

This work was sponsored by Newport Instruments internal research funds. One of us, R.C.L. wishes to thank Ryan R. Brinkman, Josef Spidlen and the other members of the ISAC DSTF for many enlightening and pleasant discussions.

## SELECTED REFERENCES:

- 1] Thrall M, Pantanowitz L, Khalbuss W. Telecytology: Clinical applications, current challenges, and future benefits, Review Article. *J Pathol Inform* 2011;2:51.
- 2] Spidlen, J. et al., "Archival Cytometry Standard ACS, International Society for Advancement of Cytometry, Draft Candidate Recommendation," Available at: <http://flowcyt.sf.net/acs/latest.pdf> (2010).
- 3] Spidlen, J., Leif, R.,C., Moore, W., Roederer, M., International Society for the Advancement of Cytometry Data Standards Task Force, Brinkman, R. R., "Gating-ML: XML-based gating descriptions in flow cytometry," *Cytometry Part A* 73A, 1151-1157 (2008).
- 4.] Pritchett, J., Gylling, M., International Digital Publishing Forum (IDPF) "EPUB Open Container Format (OCF) 3.0, Recommended Specification 11 October 2011," Available at: <http://idpf.org/epub/30/spec/epub30-ocf.html> (2011).
- 5] Leif, R.,C., "Toward the integration of cytomics and medicine," *J. Biophoton.* 2, 482-493 (2009).
- 6] Lee, K.,P., Hu, J., "XML Schema Representation of DICOM Structured Reporting," *Journal of the American Medical Informatics Association* Volume 10, 213-223, <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC150374/pdf/0100213.pdf> (2003).
- 7] DICOM Working Group 26, "DICOM Supplement 145: Whole Slide Imaging in Pathology," which applies to Parts 2,3,4,6,16,17 of the DICOM Standard Available at: (<http://www.dclunie.com/dicom-status/status.html#BaseStandard2001>) (2009)
- 8] Leif R.C., Leif S.B., and Leif S.H., "CytometryML, an XML Format Based on DICOM and FCS for Analytical Cytology Data," *Cytometry* 54A, 56-65 (2003).
- 9]. Leif R.C., Spidlen J., Brinkman R.R., "A Container for the Advanced Cytometry Standard (ACS)," *Proc. SPIE* 7182, 71821Q (2009).
- 10] Leif R.C., "CytometryML, Binary Data Standards," *Proc. SPIE* 5699, 325-333 (2005).
- 11] Spidlen J., Brinkman R., Leif R. C., and other members of the ISAC Data Standards Task Force., "Advanced Cytometry Standard (ACS) Requirements for a data file standard format to describe cytometry and related analytical cytology data, Version 0.070920," Available at: [http://cdnetworks-us-1.dl.sourceforge.net/project/flowcyt/Analytical Cytology Standard/Analytical Cytology Standard/Requirements-v070920.pdf](http://cdnetworks-us-1.dl.sourceforge.net/project/flowcyt/Analytical%20Cytology%20Standard/Analytical%20Cytology%20Standard/Requirements-v070920.pdf) (2007).
- 12] Birbeck, M., and McCarron, S. (Editors) "CURIE Syntax 1.0, A syntax for expressing Compact URIs", W3C Candidate Recommendation 16 January 2009," Available at: <http://www.w3.org/TR/2009/CR-curie-20090116> (2009).
- 13] McCarron, S., (Editor) "XHTML+RDFa 1.1, Support for RDFa via XHTML Modularization", W3C Candidate Recommendation 13 March (2012)