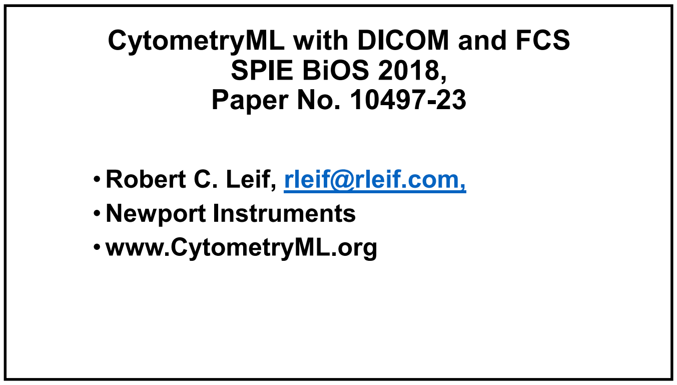
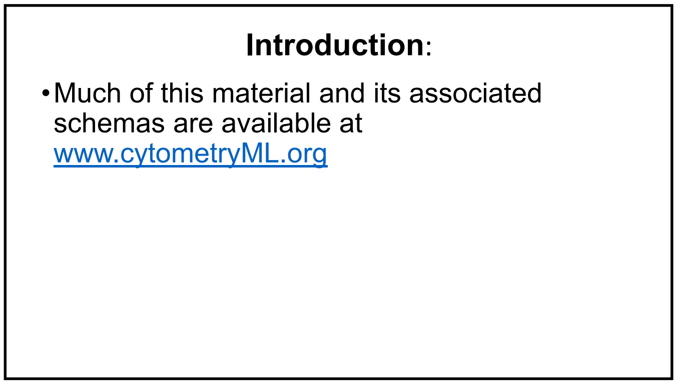


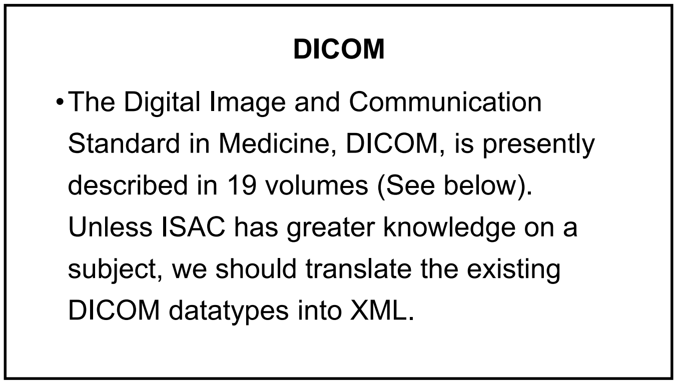
CytometryML with DICOM and FCS
SPIE BiOS 2018,
Paper No. 10497-23

- **Robert C. Leif, rleif@rleif.com,**
- **Newport Instruments**
- **www.CytometryML.org**



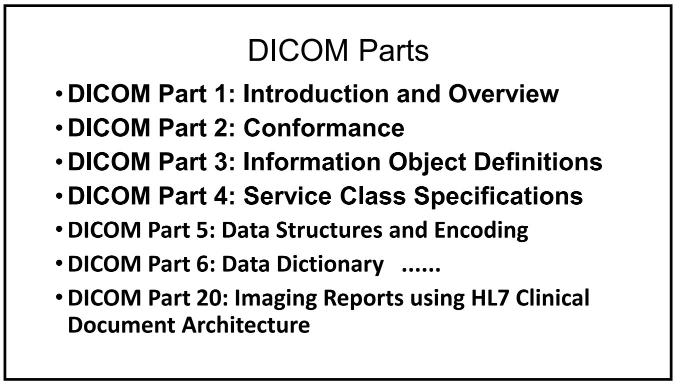
Introduction:

- Much of this material and its associated schemas are available at www.cytometryML.org



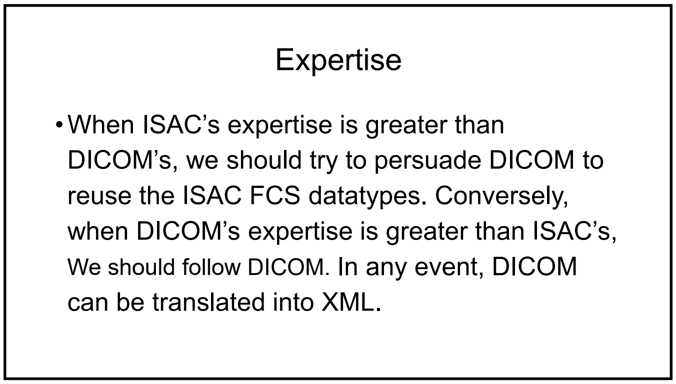
DICOM

- The Digital Image and Communication Standard in Medicine, DICOM, is presently described in 19 volumes (See below). Unless ISAC has greater knowledge on a subject, we should translate the existing DICOM datatypes into XML.



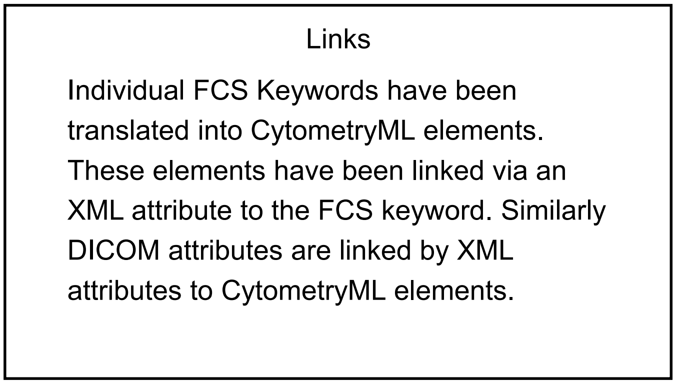
DICOM Parts

- **DICOM Part 1: Introduction and Overview**
- **DICOM Part 2: Conformance**
- **DICOM Part 3: Information Object Definitions**
- **DICOM Part 4: Service Class Specifications**
- **DICOM Part 5: Data Structures and Encoding**
- **DICOM Part 6: Data Dictionary ......**
- **DICOM Part 20: Imaging Reports using HL7 Clinical Document Architecture**



Expertise

- When ISAC's expertise is greater than DICOM's, we should try to persuade DICOM to reuse the ISAC FCS datatypes. Conversely, when DICOM's expertise is greater than ISAC's, We should follow DICOM. In any event, DICOM can be translated into XML.



Links

Individual FCS Keywords have been translated into CytometryML elements. These elements have been linked via an XML attribute to the FCS keyword. Similarly DICOM attributes are linked by XML attributes to CytometryML elements.

Flow and Image Overlap

- It should be noted that some flow systems can produce images and digital microscopes can produce FCS files, The use of one system to store flow and image data solves this problem.
- It also would simplify the operation of flow and image cytometry instruments if their data could be stored on a common instrument and network.

Draft of FCS 3.2 to 4.0 FCS Keyword

Number of bits reserved for FCS dimension number n

$P1B/32/ $P1B is the keyword and 32 is its value;

$ is the start of all FCS 4.0 Keywords;

/ is the delimiter for FCS 4.0, which is placed before the start of and the end of the value;

The values of the $PnB keywords specify the number of bits allocated for storage of values of FCS dimension number n, which appears in the template of the datatype. $PnB

FCS & DICOM

Although FCS and DICOM are based on extensive, superb domain knowledge. Both have some common problems. they are isolated systems, which lack the capability to interoperate or work with other standards.

•FCS does not take advantage of data structures and both FCS and DICOM require special programs to read and write the data.

$P1B/32/ Example

- The letter P was originally for parameter and now refers to dimension
- The n in the dimension keywords shall be replaced with a number (Positive integer?) corresponding to the FCS dimension number. For example, $P1N is the keyword capturing the short name of FCS dimension 1. Each successive value of n results in a keyword including the value of that keyword.
- The letter B probably is the abbreviation for Bits. The value is the number of bits.

Problem

- **FCS 3.2 & 4.0 Problem:** One major change in nomenclature of FCS is that the term **parameter** has been changed to dimension. However, to maintain backward compatibility the letter “P” has been retained as the second character in FCS Keywords.

FCS Keyword Type example, $P1B

```
<simpleType
    name="FCS_Keyword_Type">
    <restriction
        base="string">
        <minLength value="2"/>
        <maxLength value="32"/>
        <pattern
            value="[$]{1}[A-z|0-9]{1,31}"/>
    </restriction>
</simpleType>
```

FCS Keyword Type Comments

- A string of 2 to 32 characters is first created. It is then fixed to follow a pattern.
- The first character is fixed to be a dollar sign '$'.
- The rest of the string can be from 1 to 31 letters (characters). This provides the range of 2 to 32 characters, The capital letter 'P' or any other upper or lower case letter is also acceptable as the second character to a total of 31 letters. A third part of the string can also include numbers. This design to have a letter as its second character and permits the use of a number for the third character.

Num_Bits_Allocated Cont.

```
 <attribute
   name="Keyword_Name“
   type="keyword:Keyword_Name_Type"
         fixed="$PnB"/>
 </extension>
 </simpleContent>
</complexType>
```

Byte_Steps_*8_Type

```
•<simpleType
      name="Byte_Steps_x8_Type">
      <restriction base="integer">
         <enumeration value="8"/>
         <enumeration value="16"/>
         <enumeration value="32"/>
         <enumeration value="64"/>
         <enumeration value="128"/>
         <enumeration value="256"/>
      </restriction>
   </simpleType>
```

Assertions

Only works with attributes!

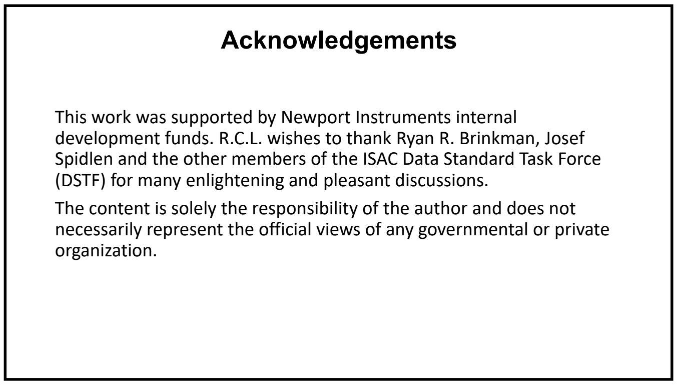
```
<assert
  test="@Num_Bits_Stored &lt;=
    @Num_Bits_Allocated"/>
```

Num_Bits_Allocated

```
<complexType
    name="Num_Bits_Allocated_Type"
      <simpleContent>
         <extension
            base="keyword:Byte_Steps_x8_Type">
            <attribute
               name="Tag"
               type="dicom:Tag_Type"
               fixed="00280100"/>
            <attribute
               name="VR"
                   type="dicom:VR_Type"
                       fixed="US"/>
```

Advantages of CytometryML

1. Based on ubiquitous commercially supported standards, XML, HTML, CSS & EPUB. No requirement for commercial software to read a file,
2. Includes links to its predecessors. Takes advantage of 3 knowledge bases: FCS, DICOM, and XML.
3. Strongly typed at object and relationship levels. Protect against data entry errors and perhaps unauthorized searches.
4. Maximizes readability (Requirement)



Acknowledgements

This work was supported by Newport Instruments internal development funds. R.C.L. wishes to thank Ryan R. Brinkman, Josef Spidlen and the other members of the ISAC Data Standard Task Force (DSTF) for many enlightening and pleasant discussions.

The content is solely the responsibility of the author and does not necessarily represent the official views of any governmental or private organization.