

1 A New XML Representation for HDF5

2 (First Draft)

3 Date:

4 11 November 2011

5 Editors:

6 Mike Folk, mfolk@hdfgroup.org

7 Gerd Heber, gheber@hdfgroup.org

8 Quincey Koziol, koziol@hdfgroup.org

9

10 The HDF Group, 1800 South Oak Street, Suite 203, Champaign, IL 61820, USA

11 Abstract:

12 In this document, a brief overview of a new XML representation of HDF5 is given. Its purpose is
13 to rekindle the discussion about different XML representations of HDF5, and to arrive at a solution
14 that will serve our users well for the next decade.

15 Motto:

16 *Ein neues Lied, ein besseres Lied,*

17 *O Freunde, will ich euch dichten!*¹

18 Heinrich Heine

19 1 Introduction

20 The idea of an XML representation of HDF5 is probably almost as old as HDF5 itself. HDF5 wraps
21 something that is structurally rich into an opaque container and creates a demand for *introspection* – for
22 looking under the surface. However, with the possible exception of data archival [7], this curiosity doesn't
23 extend all the way down to the byte level.

24 Technically, the HDF5 library provides all the facilities for introspection and the tool `h5dump` [8] can
25 produce textual renderings (in Backus-Naur Form [1]). Despite these capabilities, the library leaves the
26 definition of a description largely to the user and BNF is probably too esoteric of a choice when it comes
27 to tool support and interoperability. Creating an XML representation of HDF5 opens data stored in HDF5

¹ "A novel song, a better song, my friends, I'll have you try!" Germany – A Winter's Tale (translation by Walter W. Arndt)

28 up to the formidable machinery built around W3C standards such as XML schema [9], XSLT [10], and
29 XQuery [11].

30 The genesis of the existing XML representation for HDF5 [16] can be summarized as follows: In the
31 beginning there was the HDF5 BNF grammar [1] that was paraphrased to become an HDF5 DTD that
32 was later “converted” and enhanced to become the current HDF5 XML schema. A conscious decision
33 was made to adopt the “hybrid approach” already present in the BNF representation: *For HDF5 objects*
34 *that may be shared (Groups, Datasets, Named Datatypes) the XML element is defined to be either a*
35 *description of the object or a “pointer” to an element that describes the object. A shared object should be*
36 *described in exactly one element, and all other instances should point to that element.* [19]

37 While the choices made appear to result in an XML representation that provides improved readability and
38 a natural representation of the HDF5 structure, they also result in a de-normalized representation, a
39 substantial increase in XML processing complexity, and make ad hoc querying rather difficult because of
40 a priori knowledge requirements. Seen in hindsight, the “hybrid approach” was a questionable decision.

41 The requirements and use case analysis given in [19] are still valid and illuminating today. In fact, the
42 “new” HDF5/XML schema was discussed as an alternative in [19]. In meeting notes from February 2000
43 [21], it was characterized as ‘elegant’ and deemed to represent *the actual way that HDF5 works*. Despite
44 this, it was ultimately rejected in favor of the hybrid solution.

45 We cannot ignore a decade of advances in XML technology that lets yesterday’s challenges appear in a
46 different light and that urges us to review earlier decisions. As the use of the indefinite article in the title of
47 this document suggests, there is no canonical XML representation of HDF5.

48 The choices and design decisions made in the development of the new HDF5 XML schema (HDF5/XML)
49 were driven by the aspects of HDF5 that are to be represented and by the intended consumers. With that
50 in mind, the representation should meet these requirements:

51 1) Reflect the simplicity of the HDF5 data model,

52 The HDF5 data model is of the utmost simplicity: a collection of HDF5 datasets, a *database*, is
53 logically structured via a collection of HDF5 groups, a *linkbase*. Together these define a global
54 structure that can be navigated starting from a *root*.

55 2) Offer a sufficient degree of fidelity to the “real thing” (HDF5).

56 The BNF representation (and the current XML representation) gives a distorted view of the original in
57 its attempt to create the fiction of object names. This obscures the simplicity of HDF5 and has
58 unintended side-effects, which we would like to avoid.

59 3) Be neutral with respect to application domains.

60 Not only should the representation be neutral with respect to application domains, it must not limit its
61 own applicability by constraining potential applications in any shape or form.

62 4) Be durable and capable of absorbing peripheral changes to HDF5.

63 The durability requirement is necessary in order for the representation to be a sound investment.
 64 Let HDF5/XML be an XML schema that satisfies these requirements. As a domain expert, you might ask
 65 of what, if any, value such a representation may be to your domain. The answer can range from “none at
 66 all” to “one less thing to think about”. In case you are contemplating the creation of an XML-based
 67 description for your domain, HDF5/XML provides a stable target that lets you “factor out” the HDF5 side of
 68 things. You are more than welcome to reuse the type definitions from HDF5/XML in your own XML
 69 schema. Alternatively, you may provide XSL or XQuery transformations that map parts of your domain
 70 XML into HDF5/XML. Finally, although this appears to be the least common scenario, you may represent
 71 your domain directly in HDF5/XML. No matter what approach you take, as HDF5 tools are extended to
 72 support HDF5/XML, you can utilize them for your domain.

73 This document is neither an HDF5/XML specification nor a primer. We assume that you have a working
 74 knowledge of HDF5 and XML schema, XPath, XLink [12], and preferably XSLT or XQuery. This is a
 75 request for comments. If you are interested in the synergy between HDF5 and XML, we would like to hear
 76 from you. Please contact us by email or, even better, share your comments, criticisms, suggestions, and
 77 thoughts with the community on the HDF forum hdf-forum@hdfgroup.org.

78 2 HDF5/XML High-Level Structure

79 All XML information items defined by the HDF5/XML schema are identified by the namespace URI

80 `http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11`

81 The schema uses the following prefixes:

Prefix	Namespace
h5	http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11
xs	http://www.w3.org/2001/XMLSchema
xlink	http://www.w3.org/1999/xlink

82 The schema can be found at the namespace URI which happens to be a URL.

83 Many elements in instance documents are identified by universally unique identifiers (UUID) [13]:

84 `<xs:simpleType name="uuidT">`
 85 `<xs:restriction base="xs:anyURI">`
 86 `<xs:pattern value="[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}"/>`
 87 `</xs:restriction>`
 88 `</xs:simpleType>`

89 The HDF5/XML schema defines a single global element of type `h5:domainT`. All other elements are
 90 local elements. The type `h5:domainT` is defined as follows:

```

91 <xs:complexType name="domainT">
92   <xs:sequence>
93     <xs:element name="root" type="xlink:simple" minOccurs="1" maxOccurs="1"/>
94
95     <xs:element name="linkbase" type="h5:linkbaseT" minOccurs="1" maxOccurs="1"/>
96
97     <xs:element name="database" type="h5:databaseT" minOccurs="1" maxOccurs="1"/>
98
99     <xs:element name="encodingbase" type="h5:encodingbaseT"
100       minOccurs="1" maxOccurs="1"/>
101   </xs:sequence>
102   <xs:attribute name="id" type="h5:uuidT" use="required"/>
103   <xs:attribute name="created" type="xs:dateTime" use="required"/>
104   <xs:attribute name="last-modified" type="xs:dateTime" use="required"/>
105 </xs:complexType>

```

106 Conceptually, an HDF5 domain consists of a collection of HDF5 datasets (a database), a collection of
107 HDF5 groups and links (a linkbase), and a designated HDF5 root group (the root). The linkbase tracks
108 how HDF5 datasets and groups are organized and the root tells us “where to start”:

```

109 <xs:complexType name="linkbaseT">
110   <xs:sequence>
111     <xs:element name="group" type="h5:groupT" minOccurs="1" maxOccurs="unbounded"/>
112   </xs:sequence>
113 </xs:complexType>
114
115 <xs:complexType name="groupT">
116   <xs:sequence>
117     <xs:element name="attribute" type="h5:attributeT" minOccurs="0"
118       maxOccurs="unbounded"/>
119     <xs:element name="participant" type="h5:participantT" minOccurs="0"
120       maxOccurs="unbounded"/>
121   </xs:sequence>
122   <xs:attribute name="id" type="h5:uuidT" use="required"/>
123   <xs:attribute ref="xlink:type" fixed="extended" use="required"/>
124 </xs:complexType>
125
126 <xs:complexType name="participantT">
127   <xs:complexContent>
128     <xs:extension base="xlink:locatorType">
129       <xs:attribute name="corder" type="xs:long" use="optional"/>
130       <xs:attribute name="cencoding" type="h5:cencodingT" use="optional"/>
131     </xs:extension>
132   </xs:complexContent>
133 </xs:complexType>

```

134 HDF5 groups can be viewed as collections of named single source/single destination, unidirectional links.
135 This is a special case of XLink extended links [12] which we use in HDF5/XML to model HDF5 groups.
136 What XLink refers to as ‘link participants’ is equivalent to what HDF5 refers to as ‘members of an HDF5
137 group’. A group member is represented by an element of type `xlink:locatorType` with XML attributes
138 `xlink:title` for the link name and `xlink:href` for the link target.

139 The database is just a collection of HDF5 datasets that is modeled as follows:

```

140 <xs:complexType name="databaseT">
141   <xs:sequence>
142     <xs:element name="dataset" type="h5:datasetT" minOccurs="0"
143       maxOccurs="unbounded"/>
144   </xs:sequence>
145 </xs:complexType>
146
147 <xs:complexType name="datasetT">
148   <xs:sequence>
149     <xs:element name="attribute" type="h5:attributeT" minOccurs="0"

```

```

150         maxOccurs="unbounded"/>
151
152     <xs:group ref="h5:functionGraphG"/>
153
154     <xs:element name="layout" type="h5:layoutT" minOccurs="1" maxOccurs="1"/>
155     <xs:element name="fill-value" type="xs:string" minOccurs="0" maxOccurs="1"/>
156     <xs:element name="fill-time" type="h5:fillTimeT" minOccurs="0" maxOccurs="1"/>
157     <xs:element name="allocation-time" type="h5:allocationTimeT" minOccurs="0"
158         maxOccurs="1"/>
159     <xs:element name="filters" type="h5:filterPipelineT" minOccurs="0"
160         maxOccurs="1"/>
161     <xs:element name="storage" type="h5:storageT" minOccurs="0" maxOccurs="1"/>
162 </xs:sequence>
163     <xs:attribute name="id" type="h5:uidT" use="required"/>
164 </xs:complexType>
165
166 <xs:group name="functionGraphG">
167     <xs:sequence>
168         <xs:element name="type" type="xlink:simple" minOccurs="1" maxOccurs="1"/>
169         <xs:element name="shape" type="h5:dataspaceT" minOccurs="1" maxOccurs="1"/>
170         <xs:element name="value" type="h5:valueT" minOccurs="0" maxOccurs="1"/>
171     </xs:sequence>
172 </xs:group>

```

173 HDF5 datasets (and attributes) can be viewed as array variables. Their defining characteristics are:

- 174 1) A *shape* - a multidimensional lattice (HDF5 dataspace)
- 175 2) An array element *type* (HDF5 datatype)
- 176 3) The *value* of the array variable (an actual array of data elements)

177 This “trinity” is modeled with an XML `group` element called `functionGraphG`. (The value of an HDF5
178 dataset or attribute is the graph of an HDF5 datatype-valued function defined on an HDF5 dataspace.)
179 The `type` child element is a simple link that refers to an HDF5 datatype encoding in the `encodingbase`
180 which is the one-stop shop for the various datatype encodings in an HDF5 domain.

181 That’s all. There are of course many more XML schema datatypes and local elements dealing with the
182 details that fully describe dataset layout, filter pipelines, HDF5 attributes etc., but in terms of the high-level
183 structure this is all there is. Remember, an HDF5 domain is nothing but a database held together by a
184 linkbase with a designated entry point (root).

185 See Appendix B for a sample XML instance.

186 3 Examples

187 Throughout the discussion we will use an imaginary HDF5 file whose (multi-graph) structure is shown in
188 Figure 1. Circles symbolize HDF5 groups (root = blue circle), rectangles symbolize HDF5 datasets, and
189 triangles symbolize HDF5 datatypes. An HDF5/XML representation, `sample.xml`, can be found in
190 Appendix B.

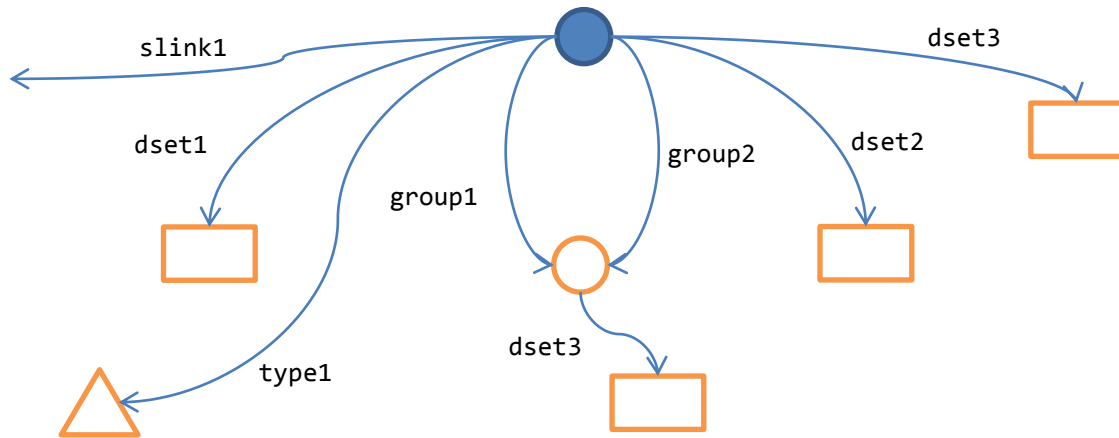


Figure 1. A schematic of the HDF5 domain from Appendix B.

191 XPath, XSLT, and XQuery are the main workhorses when dealing with HDF5/XML representations.
 192 HDF5/XML-style introspection doesn't require any C programming. Below are a few simple examples that
 193 may give you an idea of what's involved. Use your favorite XSLT/XQuery processor (e.g., SAXON [14] or
 194 Zorba [15]) to play along!

195 3.1 XPath

196 Example: Dataset Count

197 Return the total number of datasets.

```
198 => count(/domain/database/dataset)
199 4
```

200 Example: Group Member Count

201 For each group return its member count.

```
202 => for $g in /domain/linkbase/group return count($g/participant)
203 7
204 1
205 1
```

206 Example: IDs of Leaf Elements

207 Display the UUIDs of all leaf elements, i.e., all elements that represent HDF5 objects that have no
 208 members (groups without members, datasets and committed datatypes).

```
209 =>/domain/(linkbase|database|encodingbase)/(dataset|group[empty(/participant)]|datatype
210 [./@id=/domain/linkbase/group/participant/@xlink:href])/@id
211
212 30292613-8d2a-4dc4-a277-b9d59d5b0d20
213 0a68caca-629a-44aa-9f37-311e7ffb8417
214 4b43748e-817f-44c6-a9f1-16e242fd374b
215 42f5e3a2-5e70-4faf-9893-fd216257a0d9
```

```
216 a93ff089-d466-44e7-b3f0-09db34ec2ef5
```

217 Example: Reference Count

218 Return the reference counts of all linked objects.

```
219 => for $n in
220 /domain/(linkbase|database|encodingbase)/(group|dataset|datatype[./@id=/domain/linkbase/
221 group/participant/@xlink:href])
222 return concat(string($n/@id), ' : ',
223 string(count(/domain/linkbase/group/participant[@xlink:href=$n/@id])))
224
225 be8dcb22-b411-4439-85e9-ea384a685ae0 : 2
226 903d1d75-e617-4767-a3bf-0cb3ee509027 : 0
227 30292613-8d2a-4dc4-a277-b9d59d5b0d20 : 1
228 0a68caca-629a-44aa-9f37-311e7ffb8417 : 1
229 4b43748e-817f-44c6-a9f1-16e242fd374b : 1
230 42f5e3a2-5e70-4faf-9893-fd216257a0d9 : 1
231 a93ff089-d466-44e7-b3f0-09db34ec2ef5 : 1
```

232 Example: Attribute Search

233 Return the identifiers of all datasets and groups that have an attribute starting with an 'a'.

```
234 => //(dataset|group)[./attribute[matches(@name,"a*")]]/@id
235
236 903d1d75-e617-4767-a3bf-0cb3ee509027
```

237 3.2 XSLT

238 Converting XML documents from the older XML schema for HDF5 into HDF5/XML is a standard XML-to-
239 XML translation task and can be easily accomplished using an XSLT stylesheet. Shown below is an
240 excerpt from an XSLT stylesheet that generates the HDF5/XML <domain> element.

```
241 <xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
242 xmlns:xs="http://www.w3.org/2001/XMLSchema"
243 xmlns:xlink="http://www.w3.org/1999/xlink"
244 xmlns:hdf5="http://hdfgroup.org/HDF5/XML/schema/HDF5-File" version="2.0">
245
246 <xsl:output method="xml" version="1.0" encoding="UTF-8" indent="yes"/>
247
248 <!-- datetimes need to be supplied via external parameters -->
249 <xsl:param name="created"/>
250 <xsl:param name="last-modified"/>
251
252 <xsl:template match="/hdf5:HDF5-File">
253
254 <domain xmlns="http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11"
255 xmlns:xlink="http://www.w3.org/1999/xlink"
256 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
257 xsi:schemaLocation="http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11
258 http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11">
259
260 <xsl:attribute name="id">
261 <xsl:value-of select="generate-id(.)"/>
262 </xsl:attribute>
263 <xsl:attribute name="created">
264 <xsl:value-of select="$created"/>
265 </xsl:attribute>
266 <xsl:attribute name="last-modified">
267 <xsl:value-of select="$last-modified"/>
```

```

268     </xsl:attribute>
269
270     <root xmlns="http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11">
271       <xsl:attribute name="href" namespace="http://www.w3.org/1999/xlink">
272         <xsl:value-of select="//hdf5:HDF5-File/hdf5:RootGroup[@OBJ-XID]"/>
273       </xsl:attribute>
274     </root>
275
276     <linkbase xmlns="http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11">
277       <xsl:apply-templates select="//hdf5:Group|hdf5:RootGroup[@OBJ-XID]"/>
278     </linkbase>
279
280     <database xmlns="http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11">
281       <xsl:apply-templates select="//hdf5:Dataset"/>
282     </database>
283
284     <encodingbase xmlns="http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11">
285       <xsl:apply-templates
286         select="//hdf5:DataType[not(ancestor::hdf5:DataType)]"/>
287     </encodingbase>
288
289     </domain>
290
291   </xsl:template>
292
293   ...
294 </xsl:stylesheet>

```

295 This way, for example, the XML generated by `h5dump` can be easily converted into HDF5/XML.

296 3.3 XQuery

297 Where did the HDF5 path names go? The linkbase contains all the information that's needed to generate
298 them. The path names can be obtained by traversing the linkbase from the HDF5 root group and
299 concatenating link names (the `xlink:title` XML attributes of `member` elements) interspersed with
300 slashes. This is what the XQuery function `h5xq:pathnames` does. Given an HDF5 group and an initial
301 path it iterates over all group members and recursively descends into member groups that aren't empty.
302 (The function is guarded against infinite recursion through a maximum recursion depth parameter.)

```

303 xquery version "1.0" encoding "UTF-8";
304
305 declare namespace h5xq = "http://www.hdfgroup.org/HDF5/XML/xquery";
306 declare namespace xlink = "http://www.w3.org/1999/xlink";
307
308 import schema namespace h5 =
309   "http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11"
310   at http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11";
311
312 declare function h5xq:pathnames(
313   $domain      as element(h5:domain,h5:domainT),
314   $group       as element(h5:group,h5:groupT),
315   $path        as xs:string,
316   $maxRecDepth as xs:integer
317 ) as xs:string*
318 {
319   let $depth := $maxRecDepth - 1
320   return
321     if ($depth ge 0) (: maximum recursion depth ? :)
322     then (
323       $path,
324       for $m in $group/h5:participant[@xlink:type="locator"]
325       let $newPath := concat($path, concat('/', $m/@xlink:title))
326       let $x := $domain/h5:linkbase/h5:group[@id = $m/@xlink:href]

```



```

327     let $isGroupWithParts := not(empty($x)) and
328                             not(empty($x/h5:participant[@xlink:type = "locator"]))
329     return
330     if ($isGroupWithParts)
331     then h5xq:pathnames($domain,
332         $domain/h5:linkbase/h5:group[@id = $m/@xlink:href], $newPath, $depth)
333     else ($newPath)
334 )
335 else ($path)
336 };

```

337 The function is then invoked on the root group of `sample.xml` with initial path `'.'`.

```

338 declare variable $doc := validate { doc("sample.xml") };
339
340 let $domain := $doc/h5:domain
341 let $rootGroup := $domain/h5:linkbase/h5:group[@id = $domain/h5:root/@xlink:href]
342 let $maxRecDepth := 8
343
344 let $names := h5xq:pathnames($domain, $rootGroup, '.', $maxRecDepth)
345 for $n in $names
346 order by $n
347 return $n

```

348 The output reads as follows:

```

349 . ./dset1 ./dset2 ./dset3 ./group1 ./group1/dset3 ./group2 ./group2/dset3 ./slink1
350 ./type1

```

351 A similar utility functions `h5xq:whois` returns the value of the `id` XML attribute, if an HDF5 path name
352 can be resolved to an element through the linkbase.

```

353 xquery version "1.0" encoding "UTF-8";
354
355 declare namespace h5xq = "http://www.hdfgroup.org/HDF5/XML/xquery";
356 declare namespace xlink = "http://www.w3.org/1999/xlink";
357
358 import schema namespace h5 =
359     "http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11"
360     at http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11";
361
362 declare function h5xq:whois(
363     $domain as element(h5:domain,h5:domainT),
364     $group as element(h5:group,h5:groupT),
365     $relpath as xs:string
366 ) as xs:string*
367 {
368     let $names := tokenize($relpath, "/")
369     return
370     if (count($names) > 1)
371     then
372         let $newRelPath := string-join(subsequence($names, 2, count($names)), "/")
373         let $id := $group/h5:participant[@xlink:title=$names[1]]/@xlink:href
374         let $newGroup := $domain/h5:linkbase/h5:group[@id=$id]
375         return if ($newGroup)
376             then h5xq:whois($domain, $newGroup, $newRelPath)
377             else ()
378     else
379         $group/h5:participant[@xlink:title=$names[1]]/@xlink:href
380 };

```

381 The function is then invoked on the root group of `sample.xml` with a sequence of path names.

```

382 declare variable $doc := validate { doc("../sample.xml") };
383
384 let $domain := $doc/h5:domain
385 let $root := $domain/h5:linkbase/h5:group[@id=$domain/h5:root/@xlink:href]

```

386
387
388

```
for $name in ("group1", "group2", "dset3", "dset4", "group1/dset3")  
return h5xq:whois($domain, $root, $name)
```

389 The output reads:

390
391

```
be8dcb22-b411-4439-85e9-ea384a685ae0 be8dcb22-b411-4439-85e9-ea384a685ae0 4b43748e-817f-  
44c6-a9f1-16e242fd374b 42f5e3a2-5e70-4faf-9893-fd216257a0d9
```

392 (There are 5 path names but only 4 UUIDs returned since the path `dset4` cannot be resolved.)

393 4 Comments

394 Most HDF5/XML use cases probably involve (1) going from HDF5 to XML or (2) going from XML to HDF5,
395 or (3) going both directions. Anyone who intends to use both representations faces a fundamental
396 coherence problem unless precautions are taken against independent modification. Maintaining some
397 form of checksum or fingerprint might be sufficient to mitigate most of the risk.

398 4.1 From HDF5 to XML

399 The `h5dump` tool already supports creating old-style XML representations **[16]** and will support
400 HDF5/XML at some point in the future. Initially, we'll provide an XSLT stylesheet that can be used to
401 transform the old-style XML output from `h5dump` into HDF5/XML. Of course, users are free to create
402 HDF5/XML instances that are only partial representations (e.g., drop dataset values) of the underlying
403 HDF5 file. This might be sufficient, for example, for certain structural validation tasks, further XML
404 processing, or application configuration. Through mechanisms like XLink and XPointer, HDF5/XML can
405 also be used to link HDF5 to external, non-HDF5 resources including ontologies.

406 4.2 From XML to HDF5

407 Under certain circumstances it might be desirable to create HDF5 files from HDF5/XML instances or
408 blueprints. Loading an HDF5/XML file into HDFView **[17]** should largely look and feel the same as loading
409 and manipulating an actual HDF5 file. Determined users are free to load HDF5/XML into their favorite
410 XML editor and manipulate it directly. At some point in the future, the magic "instantiate" button or a
411 command line tool like `h5gen` will create an HDF5 file according to spec.

412 In most practical applications, there will probably be an intermediate step in the form of a domain XML to
413 HDF5/XML transformation. HDF5/XML is good at representing HDF5 accurately, but it is pretty oblivious
414 about, and hence unsuited for, modeling domain specific concepts. A domain expert should create an
415 XML representation that is convenient for her users. She should also create an XML transformation
416 (XSLT or XQuery) that converts domain XML into HDF5/XML. (If necessary, she must also provide the
417 inverse transformation.)

418 As an example, consider the MXADataModel [18],[20]. In Appendix C, an MXA domain XML sample is
419 shown. It should give a domain expert a fair understanding of the nature of the underlying experimental
420 data, but, taken at face value, there's nothing about the actual MXA/HDF5 layout in this representation
421 (and there shouldn't be!). The heavy lifting is done by HDF5 reader and writer objects that hardcode the
422 HDF5 details. HDF5/XML is intended to reduce, if not eliminate, this kind of hardcoding. XML
423 transformations (XSLT, XQuery) are much easier to maintain and less brittle than the equivalent Python,
424 C/C++, or JAVA code for the HDF5 readers and writers.

425 A. References

- 426 [1] DDL in BNF for HDF5, HDF Group, May 2011. Available at
427 <http://www.hdfgroup.org/HDF5/doc/ddl.html>.
- 428 [2] HDF5 File Format Specification Version 2.0, HDF Group, May 2011. Available at
429 <http://www.hdfgroup.org/HDF5/doc/H5.format.html>.
- 430 [3] HDF5: API Specification Reference Manual, HDF Group, May 2011. Available at
431 http://www.hdfgroup.org/HDF5/doc/RM/RM_H5Front.html.
- 432 [4] HDF5 User's Guide, HDF Group, May 2011. Available at
433 <http://www.hdfgroup.org/HDF5/doc/UG/index.html>.
- 434 [5] XML Linking Language (XLink) Version 1.1, W3C, ed. S. DeRose, E. Maler, D. Orchard and
435 N. Walsh, May 2010. Available at <http://www.w3.org/TR/xlink11/>.
- 436 [6] HDF5 Information Set, M. Folk, G. Heber and Quincey Koziol, August 2011
- 437 [7] HDF4 Mapping Project, HDF Group, May 2011. Available at
438 <http://www.hdfgroup.org/projects/h4map/>
- 439 [8] h5dump, HDF Group, 2011. Available at
440 <http://www.hdfgroup.org/HDF5/doc/RM/Tools.html#Tools-Dump>
- 441 [9] W3C XML Schema, W3C, 2001. Available at <http://www.w3.org/XML/Schema>
- 442 [10] XSL Transformations (XSLT), W3C, 2007. Available at <http://www.w3.org/TR/xslt20/>
- 443 [11] XQuery 1.0: An XML Query Language, W3C, 2010. Available at
444 <http://www.w3.org/TR/xquery/>
- 445 [12] XML Linking Language (XLink), W3C, 2010. Available at <http://www.w3.org/TR/xlink11/>
- 446 [13] Universally Unique Identifier, Wikipedia, 2011. Available at
447 http://en.wikipedia.org/wiki/Universally_unique_identifier
- 448 [14] SAXON – The XSLT and XQuery Processor, Michael H. Kay, 2011. Available at
449 <http://saxon.sourceforge.net/>

- 450 [15] Zorba: The XQuery Processor, 2011. Available at <http://www.zorba->
451 [xquery.com/site2/html/index.html](http://www.zorba-xquery.com/site2/html/index.html)
- 452 [16] The HDF5 XML Information Page, HDF Group, 2011. Available at
453 <http://www.hdfgroup.org/HDF5/XML/>
- 454 [17] HDFView, The HDF Group, 2011. Available at <http://www.hdfgroup.org/hdf-java-html/hdfview/>
- 455 [18] MXADataModel Documentation, 2009, Available at <http://mxa.web.cmu.edu/index.html>
- 456 [19] The XML DTD for HDF5: Design Notes, HDF Group, 2000, Available at
457 <http://ftp.hdfgroup.org/HDF5/XML/design-notes.html>
- 458 [20] MXA: a customizable HDF5-based data format for multi-dimensional data sets, M Jackson,
459 JP Simmons, M De Graef, Modelling Simul. Mater. Sci. Eng. 18 065008 doi:10.1088/0965-
460 0393/18/6/065008.
- 461 [21] Some key points from the Data Modelling meeting, 2/16/00, Robert McGrath, HDF Group,
462 2000, Available at <http://www.hdfgroup.org/HDF5/XML/Version-0/mtg-2-16-00.html>

463 B. An HDF5/XML Sample

```

464 <?xml version="1.0" encoding="UTF-8"?>
465 <domain id="e203fee7-89b4-4216-894d-7aef0e3a199d" created="1985-04-12T23:20:50.52"
466 last-modified="1996-12-19T16:39:57" xmlns:xlink="http://www.w3.org/1999/xlink"
467 xmlns="http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11"
468 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
469 xsi:schemaLocation="http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11
470 http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11">
471
472 <root xlink:href="903d1d75-e617-4767-a3bf-0cb3ee509027"/>
473
474 <linkbase>
475 <group id="be8dcb22-b411-4439-85e9-ea384a685ae0" xlink:type="extended">
476 <participant xlink:title="dset3" xlink:type="locator"
477 xlink:href="42f5e3a2-5e70-4faf-9893-fd216257a0d9" />
478 </group>
479 <group id="903d1d75-e617-4767-a3bf-0cb3ee509027" xlink:type="extended">
480 <attribute name="attr1">
481 <type xlink:href="74f1d847-27cb-414c-ac5f-ed72a88fd1c9"/>
482 <shape>
483 <scalar/>
484 </shape>
485 <value>
486 string attribute
487 </value>
488 </attribute>
489 <participant xlink:title="dset1" xlink:type="locator"
490 xlink:href="30292613-8d2a-4dc4-a277-b9d59d5b0d20" />
491 <participant xlink:title="group1" xlink:type="locator"
492 xlink:href="be8dcb22-b411-4439-85e9-ea384a685ae0" />
493 <participant xlink:title="group2" xlink:type="locator"
494 xlink:href="be8dcb22-b411-4439-85e9-ea384a685ae0" />
495 <participant xlink:title="dset2" xlink:type="locator"
496 xlink:href="0a68caca-629a-44aa-9f37-311e7ffb8417" />
497 <participant xlink:title="dset3" xlink:type="locator"
498 xlink:href="4b43748e-817f-44c6-a9f1-16e242fd374b"/>
499 <participant xlink:title="slink1" xlink:type="locator"
500 xlink:href="somevalue"/>
501 <participant xlink:title="type1" xlink:type="locator"

```

```

502                                     xlink:href="a93ff089-d466-44e7-b3f0-09db34ec2ef5"/>
503     </group>
504 </linkbase>
505
506 <database>
507     <dataset id="30292613-8d2a-4dc4-a277-b9d59d5b0d20">
508         <type xlink:href="774a0564-a47e-4e69-aa86-051682aef065"/>
509         <shape>
510             <simple cur="10 10" max="10 10"/>
511         </shape>
512         <value media-type="application/json"
513             serializer="http://www.hdfgroup.org/HDF5/serialization/JSON">
514             [ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
515               0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
516               0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
517               0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
518               0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
519               0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
520               0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
521               0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
522               0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
523               0, 1, 2, 3, 4, 5, 6, 7, 8, 9 ]
524         </value>
525         <layout>
526             <contiguous/>
527         </layout>
528     </dataset>
529     <dataset id="0a68caca-629a-44aa-9f37-311e7ffb8417">
530         <type xlink:href="1b7f6991-17a4-478e-9e05-9c38f68a0288"/>
531         <shape>
532             <simple cur="5" max="5"/>
533         </shape>
534         <value media-type="application/json"
535             serializer="http://www.hdfgroup.org/HDF5/serialization/JSON">
536             [[1,0.1,0.01], [2,0.2,0.02], [3,0.3,0.03], [4,0.4,0.04], [5,0.5,0.05]]
537         </value>
538         <layout>
539             <contiguous/>
540         </layout>
541     </dataset>
542     <dataset id="4b43748e-817f-44c6-a9f1-16e242fd374b">
543         <type xlink:href="709f3060-fa3a-4e38-846b-b6bd5f85bc87"/>
544         <shape>
545             <simple cur="4" max="4"/>
546         </shape>
547         <value media-type="application/json"
548             serializer="http://www.hdfgroup.org/HDF5/serialization/JSON">
549             [[0], [10, 11], [20, 21, 22], [30, 31, 32, 33]]
550         </value>
551         <layout>
552             <contiguous/>
553         </layout>
554     </dataset>
555     <dataset id="42f5e3a2-5e70-4faf-9893-fd216257a0d9">
556         <type xlink:title="/type1"
557             xlink:href="a93ff089-d466-44e7-b3f0-09db34ec2ef5"/>
558         <shape>
559             <simple cur="5" max="5"/>
560         </shape>
561         <value media-type="application/json"
562             serializer="http://www.hdfgroup.org/HDF5/serialization/JSON">
563             [
564             [
565                 [ 0, 1, 2, 3 ],
566                 [ 0.1, 0.1, 0.1, 0.1, 0.1, 0.1,
567                   0.2, 0.2, 0.2, 0.2, 0.2, 0.2,
568                   0.3, 0.3, 0.3, 0.3, 0.3, 0.3,
569                   0.4, 0.4, 0.4, 0.4, 0.4, 0.4,
570                   0.5, 0.5, 0.5, 0.5, 0.5, 0.5 ]
571             ],
572             [

```

```

573         [ 0, 1, 2, 3 ],
574         [ 0.1, 0.1, 0.1, 0.1, 0.1, 0.1,
575           0.2, 0.2, 0.2, 0.2, 0.2, 0.2,
576           0.3, 0.3, 0.3, 0.3, 0.3, 0.3,
577           0.4, 0.4, 0.4, 0.4, 0.4, 0.4,
578           0.5, 0.5, 0.5, 0.5, 0.5, 0.5 ]
579     ],
580     [
581     [ 0, 1, 2, 3 ],
582     [ 0.1, 0.1, 0.1, 0.1, 0.1, 0.1,
583       0.2, 0.2, 0.2, 0.2, 0.2, 0.2,
584       0.3, 0.3, 0.3, 0.3, 0.3, 0.3,
585       0.4, 0.4, 0.4, 0.4, 0.4, 0.4,
586       0.5, 0.5, 0.5, 0.5, 0.5, 0.5 ]
587     ],
588     [
589     [ 0, 1, 2, 3 ],
590     [ 0.1, 0.1, 0.1, 0.1, 0.1, 0.1,
591       0.2, 0.2, 0.2, 0.2, 0.2, 0.2,
592       0.3, 0.3, 0.3, 0.3, 0.3, 0.3,
593       0.4, 0.4, 0.4, 0.4, 0.4, 0.4,
594       0.5, 0.5, 0.5, 0.5, 0.5, 0.5 ]
595     ],
596     [
597     [ 0, 1, 2, 3 ],
598     [ 0.1, 0.1, 0.1, 0.1, 0.1, 0.1,
599       0.2, 0.2, 0.2, 0.2, 0.2, 0.2,
600       0.3, 0.3, 0.3, 0.3, 0.3, 0.3,
601       0.4, 0.4, 0.4, 0.4, 0.4, 0.4,
602       0.5, 0.5, 0.5, 0.5, 0.5, 0.5 ]
603     ]
604   ]
605 </value>
606 <layout>
607   <contiguous/>
608 </layout>
609 </dataset>
610 </database>
611
612 <encodingbase>
613   <datatype id="a93ff089-d466-44e7-b3f0-09db34ec2ef5">
614     <compound>
615       <member name="a">
616         <array dims="4">
617           <predefined>H5T_STD_I32BE</predefined>
618         </array>
619       </member>
620       <member name="b">
621         <array dims="5 6">
622           <predefined>H5T_IEEE_F32BE</predefined>
623         </array>
624       </member>
625     </compound>
626   </datatype>
627   <datatype id="774a0564-a47e-4e69-aa86-051682aef065">
628     <predefined>H5T_STD_I32BE</predefined>
629   </datatype>
630   <datatype id="74f1d847-27cb-414c-ac5f-ed72a88fd1c9">
631     <stringN length="17" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
632   </datatype>
633   <datatype id="1b7f6991-17a4-478e-9e05-9c38f68a0288">
634     <compound>
635       <member name="a">
636         <predefined>H5T_STD_I32BE</predefined>
637       </member>
638       <member name="b">
639         <predefined>H5T_IEEE_F32BE</predefined>
640       </member>
641       <member name="c">
642         <predefined>H5T_IEEE_F64BE</predefined>
643       </member>

```

```

644     </compound>
645   </datatype>
646   <datatype id="709f3060-fa3a-4e38-846b-b6bd5f85bc87">
647     <vlen>
648       <predefined>H5T_STD_I32LE</predefined>
649     </vlen>
650   </datatype>
651 </encodingbase>
652
653 </domain>

```

654 C. An MXA Sample File

655 The XML sample below is can be found at http://mxa.web.cmu.edu/XML_Usage.html.

```

656 <?xml version="1.0" encoding="UTF-8"?>
657 <!DOCTYPE File_Root SYSTEM "http://mxa.web.cmu.edu/mxa_0.4.dtd">
658 <File_Root>
659   <Data_Model Model_Type="MXA" Model_Version="0.4">
660     <Data_Root Name="DataModelTest/Data/Root/Is/Here/" />
661     <Data_Dimensions>
662       <Dimension Alt_Name="Vol Frac"
663         Count="15" End_Value="50" Increment="2" Index="0"
664         Name="Volume Fraction" Start_Value="20" Uniform="1" />
665       <Dimension Alt_Name="Rnd Seed" Count="10"
666         End_Value="5000" Increment="500" Index="1"
667         Name="Random Seed" Start_Value="1000" Uniform="1" />
668       <Dimension Alt_Name="TS" Count="100" End_Value="99"
669         Increment="1" Index="2"
670         Name="Timestep" Start_Value="0" Uniform="1" />
671       <Dimension Alt_Name="slice" Count="256" End_Value="255"
672         Increment="1" Index="3"
673         Name="Slice" Start_Value="0" Uniform="1" />
674     </Data_Dimensions>
675     <Data_Records>
676       <Signal Alt_Name="AltComp" Name="Composition" />
677       <Signal_Group Alt_Name="OP" Name="Order Parameters">
678         <Signal Alt_Name="Alt Eta1" Name="Eta1" />
679         <Signal Alt_Name="Alt Eta2" Name="Eta2" />
680         <Signal Alt_Name="Alt Eta3" Name="Eta3" />
681       </Signal_Group>
682       <Signal_Group Alt_Name="OP 2" Name="Order Parameters 2">
683         <Signal Alt_Name="Alt Eta1" Name="Eta1" />
684         <Signal Alt_Name="Alt Eta2" Name="Eta2" />
685         <Signal Alt_Name="Alt Eta3" Name="Eta3" />
686       </Signal_Group>
687     </Data_Records>
688   </Data_Model>
689   <Meta_Data>
690     <Required_MD Creator="Mike Jackson"
691       Date="2006:12:24 15:34.51"
692       Description="Just a test case showing how to organize data"
693       Distribution_Rights="Unlimited"
694       Name="Testing Example Data Model"
695       Original_Source_File="Original Data Files"
696       Pedigree="Original"
697       Release_Number="AFRL/WS07-0476" />
698     <UserDefined_MD>
699       <UserMetaData key="Vector Int8" dims="5 2" type="H5T_NATIVE_INT8">
700         0 1 3 4 6
701         7 9 10 12 13
702       </UserMetaData>
703     </UserDefined_MD>
704   </Meta_Data>
705 </File_Root>

```

706 There's not much about HDF5 in MXA/XML (and there shouldn't be!). Contrast that with an HDF5/XML
707 representation (outline) of the underlying HDF5 file.

```
708 <?xml version="1.0" encoding="UTF-8"?>
709 <domain id="63aa9ba0-ae7a-464f-a9e0-494644de5e34" created="2006-12-24T15:34:51"
710   last-modified="2006-12-24T15:34:51" xmlns:xlink="http://www.w3.org/1999/xlink"
711   xmlns="http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11"
712   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
713   xsi:schemaLocation="http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11
714     http://www.hdfgroup.org/HDF5/XML/schema/2011/11/11">
715
716   <root xlink:href="ae6c5a7d-d595-47d7-ad24-f40fe9f716d3"/>
717
718   <linkbase>
719     <group id="ae6c5a7d-d595-47d7-ad24-f40fe9f716d3" xlink:type="extended">
720       <!-- HDF5 root group participants -->
721       <participant xlink:title="DataModel" xlink:type="locator"
722         xlink:href="953dcb90-5b7b-4b97-a261-8913293a32c7"/>
723       <participant xlink:title="Meta Data" xlink:type="locator"
724         xlink:href="f262ba12-ab59-4fcc-8d9c-8e78b82b4dcc"/>
725       <participant xlink:title="RoboMet Data" xlink:type="locator"
726         xlink:href="ae5f8dd2-cb8a-4550-a9f7-2a6b95054b95"/>
727       <participant xlink:title="SupportFiles" xlink:type="locator"
728         xlink:href="162d687d-0fd2-43f4-af68-10560262a852"/>
729     </group>
730     <group id="953dcb90-5b7b-4b97-a261-8913293a32c7" xlink:type="extended">
731       <!-- Data Model attributes and participants -->
732       <attribute name="Model Type">
733         <type xlink:href="ed5c780c-0150-4648-86c3-6a93865a9edd"/>
734         <shape>
735           <scalar/>
736         </shape>
737         <value>MXA</value>
738       </attribute>
739       <attribute name="Model Version">
740         <type xlink:href="50e3046c-df0a-4e54-9184-cbe3dcb31401"/>
741         <shape><scalar/></shape>
742         <value>0.4</value>
743       </attribute>
744       <participant xlink:title="Data Dimensions" xlink:type="locator"
745         xlink:href="12f5325a-17bf-426f-9c9c-d435f68a52dd"/>
746       <participant xlink:title="Data Records" xlink:type="locator"
747         xlink:href="cfalac99-342f-49ed-a5a1-ebac1a796c4f"/>
748       <participant xlink:title="Data Root" xlink:type="locator"
749         xlink:href="509eae8d-fd08-4047-9959-e957526a1f6a"/>
750     </group>
751     <group id="f262ba12-ab59-4fcc-8d9c-8e78b82b4dcc" xlink:type="extended">
752       <!-- Meta Data participants -->
753       <participant xlink:title="Required" xlink:type="locator"
754         xlink:href="a28c0255-ae6d-4588-abea-cf4dca95e151"/>
755       <participant xlink:title="User Defined" xlink:type="locator"
756         xlink:href="5c6eb012-1f32-480f-b1d8-f7b57e63ae66"/>
757     </group>
758     <group id="ae5f8dd2-cb8a-4550-a9f7-2a6b95054b95" xlink:type="extended">
759       <!-- RoboMet Data participants omitted -->
760     </group>
761     <group id="162d687d-0fd2-43f4-af68-10560262a852" xlink:type="extended">
762       <!-- SupportFile participants -->
763       <participant xlink:title="0" xlink:type="locator"
764         xlink:href="db7f3af3-5039-40df-ba89-0c430bf57888"/>
765     </group>
766     <group id="12f5325a-17bf-426f-9c9c-d435f68a52dd" xlink:type="extended">
767       <!-- Data Dimensions participants -->
768       <participant xlink:title="0" xlink:type="locator"
769         xlink:href="59f21d1a-bd9b-4513-ac45-e435d21881a7"/>
770       <participant xlink:title="1" xlink:type="locator"
771         xlink:href="1a6db2d8-7c77-4590-81ea-5c6786a85f58"/>
772     </group>
773     <group id="cfalac99-342f-49ed-a5a1-ebac1a796c4f" xlink:type="extended">
```



```

774      <!-- Data Records participants -->
775      <participant xlink:title="0" xlink:type="locator"
776                xlink:href="2120637c-34ce-4668-a2bb-46f86bfad397"/>
777    </group>
778  </linkbase>
779
780  <database>
781    <dataset id="509eae8d-fd08-4047-9959-e957526a1f6a">
782      <type xlink:href="6fb36117-f97f-4b0f-87bc-83dac598ec9e"/>
783      <shape>
784        <simple cur="1" max="1"/>
785      </shape>
786      <value>RoboMet Data</value>
787      <layout>
788        <contiguous/>
789      </layout>
790      <allocation-time>H5D_ALLOC_TIME_LATE</allocation-time>
791    </dataset>
792    <dataset id="a28c0255-ae6d-4588-abea-cf4dca95e151">
793      <attribute name="Creator">
794        <type xlink:href="c483a4c8-fb53-4795-8ff1-86cc6376ea0a"/>
795        <shape><scalar/></shape>
796        <value>RoboMet Operator</value>
797      </attribute>
798      <attribute name="Date">
799        <type xlink:href="dc7f0ee9-f8a5-4d7e-9614-7d59c9d163fd"/>
800        <shape><scalar/></shape>
801        <value>2006:12:24 15:34.51</value>
802      </attribute>
803      <attribute name="Description">
804        <type xlink:href="cd59397c-107d-43cf-af6c-d0d38095b820"/>
805        <shape><scalar/></shape>
806        <value>A beta-processed titanium alloy, courtesy of Dr. H. Fraser at
807          The Ohio State University. 100 Sections collected total.
808          Every 5th section, starting with section 0.
809          (0,5,10,15....95)</value>
810      </attribute>
811      <attribute name="Distribution Rights">
812        <type xlink:href="e4163515-0f4c-440b-91a9-ab17b477c661"/>
813        <shape><scalar/></shape>
814        <value>Limited</value>
815      </attribute>
816      <attribute name="Name">
817        <type xlink:href="2c4c01c3-2165-41cf-82a2-5ae43fc3539c"/>
818        <shape><scalar/></shape>
819        <value>mosaic test3 0816_10</value>
820      </attribute>
821      <attribute name="Original Source File">
822        <type xlink:href="dc7f0ee9-f8a5-4d7e-9614-7d59c9d163fd"/>
823        <shape><scalar/></shape>
824        <value>Original Data Files</value>
825      </attribute>
826      <attribute name="Pedigree">
827        <type xlink:href="dd913c15-6bbb-4369-8232-1bd5441ad985"/>
828        <shape><scalar/></shape>
829        <value>Original</value>
830      </attribute>
831      <attribute name="Release Number">
832        <type xlink:href="068d1607-36c0-4cc4-b421-f3a167162679"/>
833        <shape><scalar/></shape>
834        <value>Not Applicable</value>
835      </attribute>
836      <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
837      <shape>
838        <simple cur="1" max="1"/>
839      </shape>
840      <value>[0]</value>
841      <layout>
842        <contiguous/>
843      </layout>
844    </dataset>

```

```
845 <dataset id="5c6eb012-1f32-480f-b1d8-f7b57e63ae66">
846   <attribute name="Mosaic Style">
847     <type xlink:href="e4163515-0f4c-440b-91a9-ab17b477c661"/>
848     <shape><scalar/></shape>
849     <value>Meander</value>
850   </attribute>
851   <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
852   <shape>
853     <simple cur="1" max="1"/>
854   </shape>
855   <value>[0]</value>
856   <layout>
857     <contiguous/>
858   </layout>
859 </dataset>
860 <dataset id="db7f3af3-5039-40df-ba89-0c430bf57888">
861   <attribute name="FileSystem_Path">
862     <type xlink:href="cc644faa-9214-4ff1-9194-ee053b40b252"/>
863     <shape><scalar/></shape>
864     <value>Robomet_100.pdf</value>
865   </attribute>
866   <attribute name="File_Type">
867     <type xlink:href="39708e3c-5aeb-4bfd-95de-85cc42e69147"/>
868     <shape><scalar/></shape>
869     <value>Binary</value>
870   </attribute>
871   <attribute name="Filename">
872     <type xlink:href="cc644faa-9214-4ff1-9194-ee053b40b252"/>
873     <shape><scalar/></shape>
874     <value>Robomet_100.pdf</value>
875   </attribute>
876   <type xlink:href="0f377d84-b85d-4717-b859-b173919af3c6"/>
877   <shape>
878     <simple cur="1028528" max="1028528"/>
879   </shape>
880   <!-- value omitted -->
881   <layout>
882     <contiguous/>
883   </layout>
884 </dataset>
885 <dataset id="59f21d1a-bd9b-4513-ac45-e435d21881a7">
886   <attribute name="Alt Name">
887     <type xlink:href="e11d7185-9b72-4c1a-811c-d65a65013a7c"/>
888     <shape><scalar/></shape>
889     <value>Slice</value>
890   </attribute>
891   <attribute name="Count">
892     <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
893     <shape><scalar/></shape>
894     <value>20</value>
895   </attribute>
896   <attribute name="End Value">
897     <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
898     <shape><scalar/></shape>
899     <value>95</value>
900   </attribute>
901   <attribute name="Increment">
902     <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
903     <shape><scalar/></shape>
904     <value>95</value>
905   </attribute>
906   <attribute name="Name">
907     <type xlink:href="e11d7185-9b72-4c1a-811c-d65a65013a7c"/>
908     <shape><scalar/></shape>
909     <value>Slice</value>
910   </attribute>
911   <attribute name="Start Value">
912     <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
913     <shape><scalar/></shape>
914     <value>0</value>
915   </attribute>
```

```
916 <attribute name="Uniform">
917 <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
918 <shape><scalar/></shape>
919 <value>1</value>
920 </attribute>
921 <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
922 <shape>
923 <simple cur="1" max="1"/>
924 </shape>
925 <value>[0]</value>
926 <layout>
927 <contiguous/>
928 </layout>
929 </dataset>
930 <dataset id="1a6db2d8-7c77-4590-81ea-5c6786a85f58">
931 <attribute name="Alt_Name">
932 <type xlink:href="f5a890ff-cf77-4da8-ac92-4df276bf9ce5"/>
933 <shape><scalar/></shape>
934 <value>Tile Index</value>
935 </attribute>
936 <attribute name="Count">
937 <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
938 <shape><scalar/></shape>
939 <value>6</value>
940 </attribute>
941 <attribute name="End_Value">
942 <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
943 <shape><scalar/></shape>
944 <value>5</value>
945 </attribute>
946 <attribute name="Increment">
947 <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
948 <shape><scalar/></shape>
949 <value>1</value>
950 </attribute>
951 <attribute name="Name">
952 <type xlink:href="f5a890ff-cf77-4da8-ac92-4df276bf9ce5"/>
953 <shape><scalar/></shape>
954 <value>Tile Index</value>
955 </attribute>
956 <attribute name="Start_Value">
957 <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
958 <shape><scalar/></shape>
959 <value>0</value>
960 </attribute>
961 <attribute name="Uniform">
962 <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
963 <shape><scalar/></shape>
964 <value>1</value>
965 </attribute>
966 <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
967 <shape>
968 <simple cur="1" max="1"/>
969 </shape>
970 <value>[1]</value>
971 <layout>
972 <contiguous/>
973 </layout>
974 </dataset>
975 <dataset id="2120637c-34ce-4668-a2bb-46f86bfad397">
976 <attribute name="Alt_Name">
977 <type xlink:href="dc7f0ee9-f8a5-4d7e-9614-7d59c9d163fd"/>
978 <shape><scalar/></shape>
979 <value>Zeiss Optical Image</value>
980 </attribute>
981 <attribute name="GUID">
982 <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
983 <shape><scalar/></shape>
984 <value>0</value>
985 </attribute>
986 <attribute name="LUID">
```

```
987         <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
988         <shape><scalar/></shape>
989         <value>0</value>
990     </attribute>
991     <attribute name="Name">
992         <type xlink:href="52ff0511-99a5-4ca6-9338-c17794fc45f2"/>
993         <shape><scalar/></shape>
994         <value>Zeiss Image</value>
995     </attribute>
996     <type xlink:href="038e6825-47f3-451a-a7c7-64d5d23d3b0e"/>
997     <shape>
998         <simple cur="1" max="1"/>
999     </shape>
1000     <value>[0]</value>
1001     <layout>
1002         <contiguous/>
1003     </layout>
1004 </dataset>
1005 </database>
1006
1007 <encodingbase>
1008     <datatype id="50e3046c-df0a-4e54-9184-cbe3dcb31401">
1009         <predefined>H5T_IEEE_F32LE</predefined>
1010     </datatype>
1011     <datatype id="038e6825-47f3-451a-a7c7-64d5d23d3b0e">
1012         <predefined>H5T_STD_I32LE</predefined>
1013     </datatype>
1014     <datatype id="0f377d84-b85d-4717-b859-b173919af3c6">
1015         <predefined>H5T_STD_U8LE</predefined>
1016     </datatype>
1017     <datatype id="ed5c780c-0150-4648-86c3-6a93865a9edd">
1018         <stringN length="4" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1019     </datatype>
1020     <datatype id="e11d7185-9b72-4c1a-811c-d65a65013a7c">
1021         <stringN length="6" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1022     </datatype>
1023     <datatype id="39708e3c-5aeb-4bfd-95de-85cc42e69147">
1024         <stringN length="7" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1025     </datatype>
1026     <datatype id="e4163515-0f4c-440b-91a9-ab17b477c661">
1027         <stringN length="8" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1028     </datatype>
1029     <datatype id="dd913c15-6bbb-4369-8232-1bd5441ad985">
1030         <stringN length="9" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1031     </datatype>
1032     <datatype id="f5a890ff-cf77-4da8-ac92-4df276bf9ce5">
1033         <stringN length="11" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1034     </datatype>
1035     <datatype id="52ff0511-99a5-4ca6-9338-c17794fc45f2">
1036         <stringN length="12" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1037     </datatype>
1038     <datatype id="6fb36117-f97f-4b0f-87bc-83dac598ec9e">
1039         <stringN length="14" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1040     </datatype>
1041     <datatype id="068d1607-36c0-4cc4-b421-f3a167162679">
1042         <stringN length="15" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1043     </datatype>
1044     <datatype id="cc644faa-9214-4ff1-9194-ee053b40b252">
1045         <stringN length="16" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1046     </datatype>
1047     <datatype id="c483a4c8-fb53-4795-8ff1-86cc6376ea0a">
1048         <stringN length="17" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1049     </datatype>
1050     <datatype id="dc7f0ee9-f8a5-4d7e-9614-7d59c9d163fd">
1051         <stringN length="20" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1052     </datatype>
1053     <datatype id="2c4c01c3-2165-41cf-82a2-5ae43fc3539c">
1054         <stringN length="21" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
1055     </datatype>
1056     <datatype id="cd59397c-107d-43cf-af6c-d0d38095b820">
1057         <stringN length="181" strpad="H5T_STR_NULLTERM" cset="H5T_CSET_ASCII"/>
```

1058
1059
1060
1061

```
</datatype>  
</encodingbase>  
</domain>
```