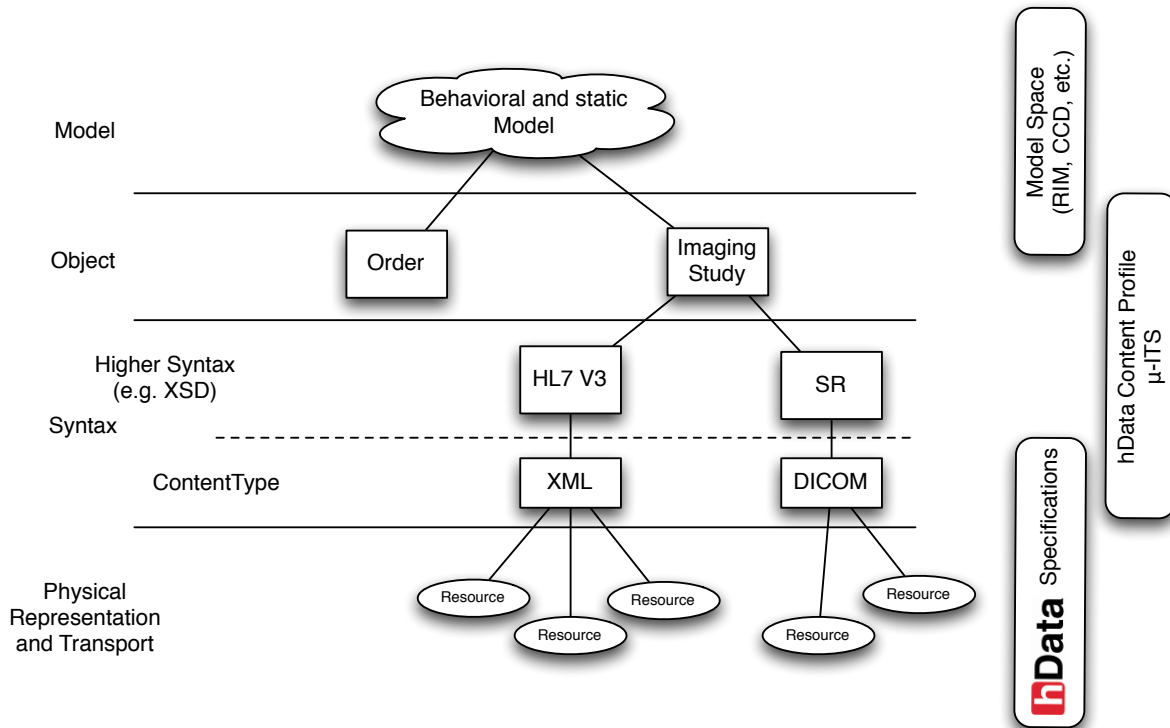


hData Record Format v0.15

2 Gerald Beuchelt, Robert Dingwell, Andrew Gregorowicz, Harry Sleeper
 3 The MITRE Corporation
 4 202 Burlington Rd.
 5 Bedford, MA 01730
 6 U.S.A.
 7 © 2010 The MITRE Corporation. All rights reserved.

1 Introduction

9 The hData Record Format (HRF) describes the logical organization of the information in an electronic health
 10 record (EHR). The HRF is implemented through component-specific documents which are linked and organized
 11 through a “master document” called the “root.xml” document. For better organization, the individual
 12 documents are put into a hierarchy, with the master document at the root of this hierarchy. While the HRF
 13 defines a core set of components, it is fully extensible and can easily be adopted for more complex situations.
 14 This specification only describes the organization of data within an hData Record (HDR). Another specification
 15 describes how a HDR is serialized [1]. The relationship of this and other hData specifications to other
 16 specifications in the HL7 standards stack is illustrated in the following diagram.



17

1.1 Namespaces

This document uses the following namespaces. This specification uses a number of namespace prefixes throughout; they are listed in Table 1. Note that the choice of any namespace prefix is arbitrary and not semantically significant.

Namespace Prefix	Namespace URI	Description
hrf	http://projecthdata.org/hdata/schemas/2009/06/core	Namespace for elements in this document
hcp	http://projecthdata.org/hdata/schemas/2010/04/hcp	Namespace for hData Content Profile Description language
hrf-md	http://projecthdata.org/hdata/schemas/2009/11/metadata	Namespace for meta data
xs	http://www.w3.org/2001/XMLSchema	XML Schema namespace
ds	http://www.w3.org/2000/09/xmldsig#	Namespace for XML Digital Signature
atom	http://www.w3.org/2005/Atom	Namespace for the Atom syndication format
rddl	http://www.rddl.org/	Namespace for RDDL

1.2 Glossary (Non-Normative)

hData Record Format (HRF) - This specification, which is a part of the hData specification that defines the abstract hierarchy, meta-data schema, and document organization of the hData record.

hData Record (HDR) - an single instantiation of the HRF.

hData Restful API (HRA) - the part of the hData specification that defines the basic HTTP-based API for accessing or modifying an HDR.

hData Specification - a set of normative specifications that defines the HRF, the HRA, and a file-based serialization format.

hData Content Profile (HCP) - a profile of the medical content of an HDR. An HCP is specified separately from the HRF.

Electronic Medical Record (EMR) - the medical record or records of a single patient in the IT system of an actor (health provider, government entity, payer, etc.). In this definition, an HDR is a type of EMR.

Electronic Health Record (EHR) - the collection of all EMRs of a single patient, across organizational and national boundaries.

37 **EHR System** - An IT system that creates, stores, and manages EHRs.

38 **Clinical Document Architecture (CDA)** - an XML specification by Health Layer 7 International (HL7) that is
39 intended to be used to convey health related information.

40 **Continuity of Care Record (CCR)** - a specification by ASTM that is intended to be used for summary/continuity
41 of care documentation. A CCM is a type of EMR.

42 **Continuity of Care Document (CCD)** - a profile of the CDA that accommodates the medical information of the
43 CCR.

44 **1.3 Notational Conventions**

45 The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT",
46 "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#).

47 When describing concrete XML schemas, this specification uses the following notation: each member of an
48 element's [children] or [attributes] property is described using an XPath notation (e.g.,
49 /x:MyHeader/x:SomeProperty/@value1).

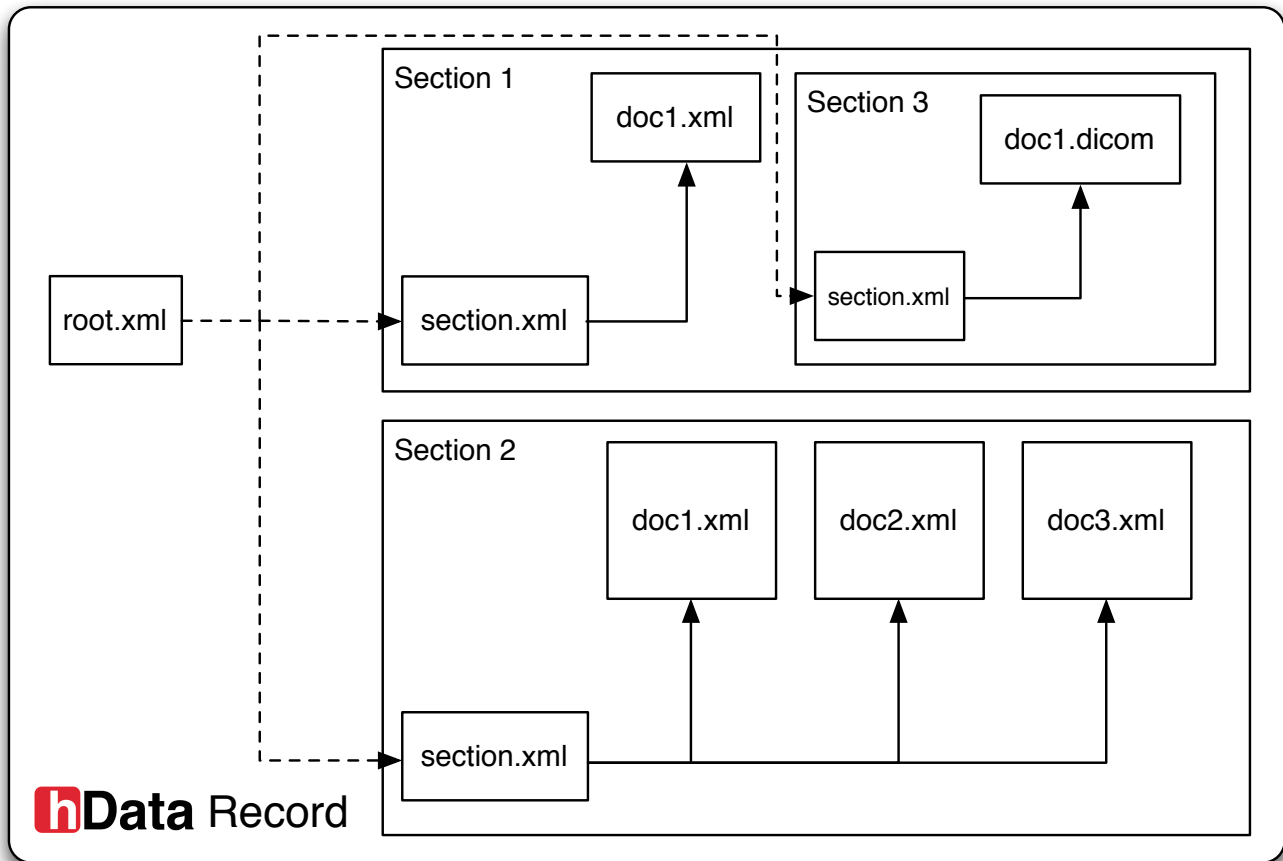
50

51 Note also that only the W3C XML schemas linked in section 3 at the end of this document are normative – any
52 schema fragment or other schema description within the main body of this document are informational only.

53

53 **2 Hierarchical Organization**

54 The basic approach of the hData Record Format is to represent the medical data through linked documents,
 55 which are organized through an abstract hierarchy. The hData RESTful API specification and the hData
 56 Serialization specification map this abstract hierarchy to a concrete implementation, such as a directory folder
 57 or web resource hierarchy.



58

59 In order to be able to accommodate more complex situations, HRF was designed with a number of extension
 60 points that allow the definition and insertion of new components.

61 **2.1 Overall Structure**

62 At the root of the hierarchy is the root document (RD). The content is contained in component specific
 63 sections, some of which are REQUIRED. The component specific sections are the primary extension points
 64 within the hierarchy. Implementers can either extend existing component sections or define new sections.
 65 Such newly created sections MUST be registered in the RD to be accessible.

66 Each section corresponds to a single set of documents. Documents are grouped into sections by type. For
 67 example, a section may contain laboratory result documents.

2.2 Root Document

The root document is at the root of the hierarchy. It contains the following elements (REQUIRED if not marked otherwise):

- /hrf:id - This element uniquely identifies the document, e.g. through a textual representation of a UUID. It is RECOMMENDED to not use absolute URIs, but only fragments that may be used within a URI.
- /hrf:version - The version of the hData Record Format used within this document. It is an integer that corresponds to the version number of the hData Record Specification that is implemented. The version number for records complying with this version of the specification is 1.
- /hrf:created - Creation date of the document, using the W3C XML Schema Date data type. This data SHOULD be significant to at least the second.
- /hrf:lastModified - Last modification of the document, using the W3C XML Schema Date data type. This data SHOULD be significant to at least the second.
- /hrf:extensions - Node containing a list of extensions (list of hrf:extension elements). Any extension to this specification MUST register itself in this section.
- /hrf:extensions/hrf:extension (OPTIONAL) - This text element contains a unique identifier for the extension. It is RECOMMENDED to use an URL. For elements of content type “application/xml”, it is RECOMMENDED that the text element contains an URL that provides a RDDDL document [3] that describes the format of instances of XML document of this extension type by including a <rddl:resource> element with the xlink:role attribute set to the schema definition. For other content types, it is RECOMMENDED that the RDDDL document resolves to documentation of the Section Document format, such as a PDF or HTML description.
- /hrf:extensions/hrf:extension/@contentType (OPTIONAL) - This attribute contains the content type of all documents in a section that registers with this extension. If the attribute is not present, the documents in the section MUST be of content type “application/xml”.
- /hrf:extensions/hrf:extension/@extensionId – This attribute contains an identifier for the extension. It MUST be unique within the root document.
- /hrf:sections - This node contains references to all component-specific sections (hrf:section)
- /hrf:sections/hrf:section (OPTIONAL) - A hrf:section describes an abstract collection of data elements within an hData record.
- /hrf:sections/hrf:section/@path - This text attribute is a path segment, used to construct the full path to the section from the root of the HDR document. Valid characters are [a-z][A-Z][0-9]. The full path to a section is obtained by starting with a forward slash (“/”), and concatenating the path segments, separated by forward slashes.
- /hrf:sections/hrf:section/@extensionId - This identifier MUST be equal to the identifier of any of the registered extension elements, as identified by the id attribute of the <extension> element. It describes the default contentType for documents contained in this section. Note that the metadata for each individual document MAY override the default contentType.
- /hrf:sections/hrf:section/@name (OPTIONAL) - Used for a human-friendly name to this section.
- /hrf:sections/hrf:section/@requirement (OPTIONAL) – this attribute indicates if a given section is required or optional. Valid values are “required” or “optional”. If this attribute is not present, the section is “required”. NOTE: This attribute is ignored in the root document for HDRs. It is only used for the hData Content Profile Description Language (see section 2.6).

110 The root document schema MAY be extended to support additional features such as a mechanism to record
111 versions of the data contained in the document.

112 2.3 Extensions

113 Extensions define the default type of Section Documents that MAY appear in a Section. Extensions MUST be
114 identified by a globally unique identifier. It is RECOMMENDED that this unique identifier be a URL pointing to a
115 RDDL document. Section Documents MAY override the default type in their metadata, but only with Extensions
116 that are registered in the RD.

117 The RECOMMENDED RDDL document that this URL resolve into will assist in the creation, consumption or
118 validation of Section Documents. It is RECOMMENDED that Extensions using XML-based Section Documents
119 include a <rddl:resource> element with the xlink:role attribute set to “http://www.w3.org/2001/XMLSchema”.
120 For Extensions using other content types, it is RECOMMENDED that the RDDL document includes a description
121 of the acceptable content in Section Documents.

122 2.4 Sections

123 Sections within an hData record form an abstract hierarchy, similar to the file folder structure commonly used
124 in hierarchical file systems. Sections can contain either Section Documents or other Sections. Sections are
125 identified by their path. The path to a Section is constructed by starting with a forward slash (“/”) and
126 appending all section path names from the root of the HDR to the Section. Section Documents contained in
127 Sections comply with the contentType of an Extension registered in the RD. An Extension MUST be listed in
128 /hrf:extensions for it to be used by a Section. Sections MAY use the same semantics for confidentiality, access
129 control, and consent as described in the meta data for Section Document in 2.5.1.

130 2.5 Section Documents

131 At each section a collection of documents can be obtained. Sections MAY be empty. Within each Section, the
132 documents MUST conform to the type defined by the Extension unless declared otherwise by the Section
133 Document’s meta data. Section documents can be of any media type, including binary media types.

134 2.5.1 Section Document Meta Data

135 Each section contains a collection of meta data artifacts that are associated with each Section Document. The
136 container format for this meta data is Atom, described in [RFC 5023](#). Each Section Document MUST have a
137 corresponding </atom:feed/atom:entry> element. If the Section Document type is different from the type
138 defined in the Section’s Extension, it MUST indicate its type in the /atom:feed/atom:entry/atom:link/@type
139 attribute. Each </atom:feed/atom:entry> must contain an <atom:link> element where the href attribute refers
140 to the Section Document. Additional meta data is contained in the </atom:feed/atom:entry/atom:content>
141 element, in an XML fragment starting with <hrf-md:DocumentMetaData>

- 142 • /hrf-md:DocumentMetaData - DocumentMetaData is the top-level element for the hData meta data
143 specification.
- 144 • /hrf-md:DocumentMetaData/hrf-md:PedigreeInfo (OPTIONAL) - This optional node holds the pedigree
145 information for the Section Document. It is of type <hrf-md:PedigreeInfo>
- 146 • /hrf-md:DocumentMetaData/hrf-md:DocumentId - This required text element holds an identifier for
147 the Section Document. It MUST be unique over any given Section.

- 148 • /hrf-md:DocumentMetaData/hrf-md:LinkedDocuments (OPTIONAL) - This optional node holds a list of
- 149 URI links to documents that are related to this Section Document. Use depends on the semantics of the
- 150 Section Document Type. It can have <hrf-md:LinkInfo> typed child elements.
- 151 • /hrf-md:DocumentMetaData/hrf-md:RecordDate - This required node holds the information about
- 152 Document creation and modification.
- 153 • /hrf-md:DocumentMetaData/hrf-md:RecordDate/hrf-md:CreatedDateTime - This required element of
- 154 type <xs:dateTime> contains the dateTime of creation of this document. If this document is not
- 155 derived (see PedigreeInfo), this is the time of the creation of the original. If this document is derived
- 156 from another origin, this element contains the date of derivation.
- 157 • /hrf-md:DocumentMetaData/hrf-md:RecordDate/hrf-md:Modified (OPTIONAL) - This optional node is
- 158 first created when the document is changed for the first time. It contains a collection of modification
- 159 dates with optional pedigree information of the modifier.
- 160 • /hrf-md:DocumentMetaData/hrf-md:RecordDate/hrf-md:Modified/hrf-md:ModifiedDateTime - This
- 161 required element of type <xs:dateTime> records a dateTime when the document was modified.
- 162 • /hrf-md:DocumentMetaData/hrf-md:RecordDate/hrf-md:Modified/hrf-md:PedigreeInfo (OPTIONAL) –
- 163 This optional node of type <hrf-md:PedigreeInfo> contains the pedigree information of the modifier.
- 164 • /hrf-md:DocumentMetaData/hrf-md:Confidentiality (OPTIONAL) – This element contains controls for
- 165 confidentiality - details are out of scope for this specification and MAY be specified by an hData Access
- 166 Control specification.
- 167 • /hrf-md:DocumentMetaData/hrf-md:AccessControl (OPTIONAL) - This element contains controls for
- 168 access control - details are out of scope for this specification and MAY be specified by an hData Access
- 169 Control specification.
- 170 • /hrf-md:DocumentMetaData/hrf-md:Consent (OPTIONAL) - This element contains controls for consent
- 171 - details are out of scope for this specification and MAY be specified by an hData Access Control
- 172 specification.

173 There are two more types that are being used in <hrf-md:DocumentMetaData>: <hrf-md:PedigreeInfo> and
 174 <hrf-md:LinkInfo>. This is the schema for <hrf-md:PedigreeInfo>

- 175 • /hrf-md:PedigreeInfo - This node contains the document pedigree information.
- 176 • /hrf-md:PedigreeInfo/hrf-md:XmlSignature (OPTIONAL) - This optional node contains the signature
- 177 information on the document or this meta data. This signature MUST conform to the W3C XML
- 178 Signature Syntax and Processing (Second Edition) [2] specification.
- 179 • /hrf-md:PedigreeInfo/hrf-md:XmlSignature/@documentMethod - This optional attribute indicates
- 180 what method was used to transform binary Section Document media types into XML files for signature.
- 181 Currently the only permitted methods are xml, sha256 and base64. xml is the default XML signature
- 182 over XML documents. base64 encodes a data stream into an XML document. The root contains the
- 183 BASE64 encoded data. sha256 calculates a hash over the binary stream and signs this hash.
- 184 • /hrf-md:PedigreeInfo/hrf-md:XmlSignature/ds:Signature (0..unbounded) - A collection of XML
- 185 Signatures. This Signature MUST contain:
 - 186 1. A valid Reference to either the metadata or the Section Document
 - 187 2. The ds:KeyInfo for the signer (optional with DSig - required here)
- 188 • /hrf-md:PedigreeInfo/hrf-md:Source (OPTIONAL) - This node indicates the source of this data.

- 189 • /hrf-md:PedigreeInfo/hrf-md:Source/@derived - If the data is derived (i.e. copied or compiled from
190 other sources) this attribute of type <xs:boolean> MUST be set to true.
- 191 • /hrf-md:PedigreeInfo/hrf-md:Source/hrf-md:PedigreeInfo (0..unbounded) – This element contains the
192 <hrf-md:PedigreeInfo> of the all source from which this document was derived.
- 193 • /hrf-md:PedigreeInfo/hrf-md:Source/hrf-md:Document (0..unbounded) – This element of type <hrf-
194 md:LinkInfo> contains links to all documents from which this document was derived.
- 195 • /hrf-md:PedigreeInfo/hrf-md:Author (0..unbounded) – This element contains the names or identifiers
196 of all author(s).
- 197 • /hrf-md:PedigreeInfo/hrf-md:Organization (0..unbound) - This element identified the organization(s)
198 at which this document was created.

199 This is the schema for <hrf-md:LinkInfo>:

- 200 • /hrf-md:LinkInfo – This node contains the link information
- 201 • /hrf-md:LinkInfo/hrf-md:Target –This required element of type <xs:anyURI> contains the absolute link
202 to the referenced Section Document.
- 203 • /hrf-md:LinkInfo/##any (OPTIONAL) – extension point.

204 2.5.2 Meta Data Processing Instructions

205 The meta data for a Section Document is only valid for the system that currently hosts the Section Document. If
206 an HDR is copied in portions or in its entirety, the system to which it is copied (referred to below as “new
207 system”) MUST recompute the meta data according to the following rules:

- 208 1. The DocumentId MUST be kept unchanged.
- 209 2. The RecordData MUST be updated by adding a new RecordDate/Modified element. This element MUST
210 contain the DateTime of the operation. The RecordDate/Modified does not need to contain a
211 PedigreeInfo field for the new system, including a KeyInfo, if the document was not modified. The
212 Source/@derived attribute MUST be set to true, and a LinkInfo to the original Section Document
213 location SHOULD be provided.
- 214 3. Confidentiality, AccessControl, and Consent SHOULD be copied verbatim.

215 2.6 hData Content Profiles

216 This specification does not specify which sections are required for an hData Record. This is done in separate
217 hData Content Profiles (HCP).

218 To describe hData Content Profiles, the following schema is used for the HCP definition file:

- 219 • /hcp:hcp – the root element for a HCP definition file.
- 220 • /hcp:hcp/@name – a simple display name
- 221 • /hrf:hcp/@id – a URI identifying the hData Content Profile. It is RECOMMENDED to use a URL that can
222 be resolved into the HCP definition document.
- 223 • /hrf:hcp/hrf:extensions – this element describes the extensions used in this HCP. It uses the same
224 syntax as in the root document as described in section 2.2.

- 225 • /hrf:hcp/hrf:sections – this element describes the sections that are to be included in a hData record
 226 that claims conformance to the HCP. It uses the same syntax as in the root document as described in
 227 section 2.2. NOTE: the requirements attribute is being used in the HCP, as described above.

228 3 Schemas

229 3.1 Root Document

230 This section contains the schema for the root document (see Section 2.2). All instances of root documents
 231 MUST validate against this schema definition.

```

232 <?xml version="1.0" encoding="UTF-8"?>
233 <!-- Copyright 2009 The MITRE Corporation
234
235 Licensed under the Apache License, Version 2.0 (the "License");
236 you may not use this file except in compliance with the License.
237 You may obtain a copy of the License at
238
239 http://www.apache.org/licenses/LICENSE-2.0
240
241 Unless required by applicable law or agreed to in writing, software
242 distributed under the License is distributed on an "AS IS" BASIS,
243 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
244 implied.
245 See the License for the specific language governing permissions and
246 limitations under the License. -->
247
248 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
249 elementFormDefault="qualified"
250 targetNamespace="http://projecthdata.org/hdata/schemas/2009/06/core"
251 xmlns:core="http://projecthdata.org/hdata/schemas/2009/06/core">
252   <xs:element name="root">
253     <xs:complexType>
254       <xs:all>
255         <xs:element ref="core:id"/>
256         <xs:element ref="core:version"/>
257         <xs:element ref="core:created"/>
258         <xs:element ref="core:lastModified"/>
259         <xs:element ref="core:extensions"/>
260         <xs:element ref="core:sections"/>
261       </xs:all>
262     </xs:complexType>
263   </xs:element>
264   <xs:element name="id" type="xs:string"/>
265   <xs:element name="version" type="xs:string"/>
266   <xs:element name="created" type="xs:date"/>
267   <xs:element name="lastModified" type="xs:date"/>
268   <xs:element name="extensions">
269     <xs:complexType>

```

```
270     <xs:sequence>
271         <xs:element minOccurs="0" maxOccurs="unbounded"
272 ref="core:extension"/>
273     </xs:sequence>
274 </xs:complexType>
275 </xs:element>
276 <xs:element name="extension">
277     <xs:complexType mixed="true">
278         <xs:attributeGroup ref="core:extension"/>
279     </xs:complexType>
280 </xs:element>
281 <xs:element name="sections">
282     <xs:complexType>
283         <xs:sequence>
284             <xs:element minOccurs="0" maxOccurs="unbounded"
285 ref="core:section"/>
286         </xs:sequence>
287     </xs:complexType>
288 </xs:element>
289 <xs:attributeGroup name="extension">
290     <xs:attribute name="contentType" type="xs:string"
291 use="optional"/>
292     <xs:attribute name="extensionId" type="xs:string"
293 use="required"/>
294 </xs:attributeGroup>
295 <xs:element name="section">
296     <xs:complexType>
297         <xs:sequence>
298             <xs:element minOccurs="0" maxOccurs="unbounded"
299 ref="core:section"/>
300         </xs:sequence>
301         <xs:attribute name="path" use="required"/>
302         <xs:attribute name="name" use="optional"/>
303         <xs:attribute name="extensionId" use="required"/>
304         <xs:attribute name="requirement" use="optional">
305             <xs:simpleType>
306                 <xs:restriction base="xs:token">
307                     <xs:enumeration value="mandatory"/>
308                     <xs:enumeration value="optional"/>
309                 </xs:restriction>
310             </xs:simpleType>
311         </xs:attribute>
312     </xs:complexType>
313 </xs:element>
314 </xs:schema>
315
```

3.2 hData Content Profile Definition

This section contains the schema for the hData Content Profile definition (see section 2.6). All instances of HCP definition documents MUST validate against this schema definition.

```

319 <?xml version="1.0" encoding="UTF-8"?>
320 <!-- Copyright 2009 The MITRE Corporation
321
322 Licensed under the Apache License, Version 2.0 (the "License");
323 you may not use this file except in compliance with the License.
324 You may obtain a copy of the License at
325
326 http://www.apache.org/licenses/LICENSE-2.0
327
328 Unless required by applicable law or agreed to in writing, software
329 distributed under the License is distributed on an "AS IS" BASIS,
330 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
331 implied.
332 See the License for the specific language governing permissions and
333 limitations under the License. -->
334
335 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
336 elementFormDefault="qualified"
337 targetNamespace="http://projecthdata.org/hdata/schemas/2010/04/hcp"
338 xmlns:hcp="http://projecthdata.org/hdata/schemas/2010/04/hcp"
339 xmlns:core="http://projecthdata.org/hdata/schemas/2009/06/core">
340   <xs:element name="hcp">
341     <xs:complexType>
342       <xs:all>
343         <xs:element ref="core:extensions"/>
344         <xs:element ref="core:sections"/>
345       </xs:all>
346       <xs:attribute name="name" use="required" type="xs:string" />
347       <xs:attribute name="id" use="required" type="xs:anyURI" />
348     </xs:complexType>
349   </xs:element>
350 </xs:schema>

```

3.3 Section Document Meta Data

This section contains the schema for the Section Document meta data (see Section 2.5.1). All instances of meta data documents MUST validate against this schema definition.

```

355 <?xml version="1.0" encoding="UTF-8"?>
356 <!-- Copyright 2009 The MITRE Corporation
357
358 Licensed under the Apache License, Version 2.0 (the "License");
359 you may not use this file except in compliance with the License.
360 You may obtain a copy of the License at
361

```

```

362     http://www.apache.org/licenses/LICENSE-2.0
363
364     Unless required by applicable law or agreed to in writing,
365 software
366 distributed under the License is distributed on an "AS IS" BASIS,
367 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
368 implied.
369     See the License for the specific language governing permissions
370 and
371     limitations under the License. -->
372
373 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
374     xmlns:hd-
375 md="http://projecthdata.org/hdata/schemas/2009/11/metadata"
376     xmlns:ds="http://www.w3.org/2000/09/xmlsig#"
377
378 targetNamespace="http://projecthdata.org/hdata/schemas/2009/11/metadata"
379 ta">
380     <xs:import namespace="http://www.w3.org/2000/09/xmlsig#"
381         schemaLocation="http://www.w3.org/TR/2002/REC-xmlsig-
382 core-20020212/xmlsig-core-schema.xsd"/>
383     <xs:complexType name="DocumentMetaData">
384         <xs:annotation>
385             <xs:documentation>
386                 DocumentMetaData is the top-level element for the
387 hData meta data specification. It is
388                 embedded with every Atom 1.0 Content node.
389             </xs:documentation>
390         </xs:annotation>
391         <xs:sequence>
392             <xs:element minOccurs="0" name="PedigreeInfo" type="hd-
393 md:PedigreeInfo">
394                 <xs:annotation>
395                     <xs:documentation>
396                         This optional node holds the pedigree
397 information for the Section Document.
398                     </xs:documentation>
399                 </xs:annotation>
400             </xs:element>
401             <xs:element name="DocumentId" type="xs:string">
402                 <xs:annotation>
403                     <xs:documentation>
404                         This required element holds an identifier for
405 the Section Document. It MUST be unique over any given
406                         Section feed.
407                     </xs:documentation>
408                 </xs:annotation>
409             </xs:element>
410             <xs:element minOccurs="0" name="LinkedDocuments">
411                 <xs:annotation>

```

```

412         <xs:documentation>
413             This optional node holds a list of URI links
414 to documents that are related to this
415             Section Document. Use depends on the
416 semantics of the Section Document Type.
417         </xs:documentation>
418     </xs:annotation>
419     <xs:complexType>
420         <xs:sequence>
421             <xs:element maxOccurs="unbounded" name="Link"
422 type="hd-md:LinkInfo"/>
423         </xs:sequence>
424     </xs:complexType>
425 </xs:element>
426 <xs:element name="RecordDate">
427     <xs:annotation>
428         <xs:documentation>
429             This required node holds the information
430 about Document creation and modification.
431         </xs:documentation>
432     </xs:annotation>
433     <xs:complexType>
434         <xs:sequence>
435             <xs:element name="CreatedDateTime"
436 type="xs:dateTime">
437                 <xs:annotation>
438                     <xs:documentation>
439                         This required element contains
440 the dateTime of creation of this document.
441                         If this document is not derived
442 (see PedigreeInfo), this is the time of the
443                         creation of the original. If this
444 document is derived from another origin, this element
445                         contains the date of derivation.
446                     </xs:documentation>
447                 </xs:annotation>
448             </xs:element>
449             <xs:element minOccurs="0" name="Modified">
450                 <xs:annotation>
451                     <xs:documentation>
452                         This optional node is first
453 created when the document is changed for the first time.
454                         It contains a collection of
455 modification dates with optional pedigree information of the
456 modifier.
457                     </xs:documentation>
458                 </xs:annotation>
459             </xs:complexType>
460             <xs:sequence minOccurs="1"
461 maxOccurs="unbounded">

```

```

462         <xs:element
463 name="ModifiedDateTime" type="xs:dateTime">
464             <xs:annotation>
465                 <xs:documentation>
466                     This required element
467 record a dateTime when the document was modified.
468                 </xs:documentation>
469             </xs:annotation>
470         </xs:element>
471     <xs:element minOccurs="0"
472 name="PedigreeInfo"
473                 type="hd-md:PedigreeInfo">
474         <xs:annotation>
475             <xs:documentation>
476                 This optional node
477 contains the pedigree information of the modifier.
478             </xs:documentation>
479         </xs:annotation>
480     </xs:element>
481 </xs:sequence>
482 </xs:complexType>
483 </xs:element>
484 </xs:sequence>
485 </xs:complexType>
486 </xs:element>
487     <xs:element minOccurs="0" name="Confidentiality"
488 type="xs:string">
489         <xs:annotation>
490             <xs:documentation>
491                 This element contains controls for
492 confidentiality - details are TBD.
493             </xs:documentation>
494         </xs:annotation>
495     </xs:element>
496     <xs:element minOccurs="0" name="AccessControl">
497         <xs:annotation>
498             <xs:documentation>
499                 This element contains controls for access
500 control - details are TBD.
501             </xs:documentation>
502         </xs:annotation>
503     </xs:element>
504     <xs:element minOccurs="0" name="Consent">
505         <xs:annotation>
506             <xs:documentation>
507                 This element contains controls for consent -
508 details are TBD.
509             </xs:documentation>
510         </xs:annotation>
511     </xs:element>

```

```

512         </xs:element>
513     </xs:sequence>
514     <xs:attribute name="MediaType" type="xs:string">
515         <xs:annotation>
516             <xs:documentation>
517                 This attribute contains the media type of the
518 document itself. If it is not present, the
519                 default media type of the content type is
520 assumed.
521             </xs:documentation>
522         </xs:annotation>
523     </xs:attribute>
524     <xs:attribute name="ContentType" type="xs:anyURI"
525 use="optional">
526         <xs:annotation>
527             <xs:documentation>
528                 This attribute contains the URI for the content
529 type of this document. If it is not present,
530                 the content type for the Section is implied. Note
531 that the current hData Content Profiles assume
532                 that the content type for all Section Documents
533 within a given Section is uniform.
534             </xs:documentation>
535         </xs:annotation>
536     </xs:attribute>
537 </xs:complexType>
538 <xs:complexType name="PedigreeInfo">
539     <xs:annotation>
540         <xs:documentation>
541             This node contains the pedigree information.
542         </xs:documentation>
543     </xs:annotation>
544     <xs:sequence>
545         <xs:element minOccurs="0" name="XmlSignature"
546 maxOccurs="unbounded">
547             <xs:annotation>
548                 <xs:documentation> This optional node contains
549 the signature information on
550                 the document or this meta data.
551             </xs:documentation>
552         </xs:annotation>
553     </xs:complexType>
554     <xs:sequence>
555         <xs:element ref="ds:Signature">
556             <xs:annotation>
557                 <xs:documentation> This Signature
558 MUST contain: 1. a valid Reference
559                 to either the metadata or the
560 Section Document 2. the ds:KeyInfo
561

```

```

562                                     for the signer (optional with
563 DSig - required here)
564                                     </xs:documentation>
565                                     </xs:annotation>
566                                     </xs:element>
567                               </xs:sequence>
568
569
570                               <xs:attribute name="documentMethod">
571                                 <xs:annotation>
572                                   <xs:documentation>This optional attribute
573 indicates what method was used
574 to transform binary Section Document
575 mediatypes into XML files for
576 signature. Currently the only permitted
577 methods are xml and base64.
578 xml is the default XML signature over XML
579 documents. base64 encodes
580 a data stream into an XML document. The
581 root node is root and
582 contains the BASE64 encoded data.
583 </xs:documentation>
584                                   </xs:annotation>
585                                 <xs:simpleType>
586                                   <xs:restriction base="xs:string">
587                                     <xs:enumeration value="base64"/>
588                                     <xs:enumeration value="xml"/>
589                                     <xs:enumeration value="sha256"/>
590                                   </xs:restriction>
591                                 </xs:simpleType>
592                               </xs:attribute>
593                             </xs:complexType>
594                           </xs:element>
595                         <xs:element minOccurs="0" maxOccurs="1" name="Source">
596                           <xs:annotation>
597                             <xs:documentation>This node indicates the source
598 of this data. </xs:documentation>
599                           </xs:annotation>
600                         <xs:complexType>
601                           <xs:sequence>
602                             <xs:element name="PedigreeInfo" type="hd-
603 md:PedigreeInfo" minOccurs="0"/>
604                             <xs:element maxOccurs="unbounded"
605 minOccurs="0" name="Document"
606                               type="hd-md:LinkInfo"/>
607                           </xs:sequence>
608                         <xs:attribute name="derived" type="xs:boolean">
609                           <xs:annotation>
610                             <xs:documentation>If the data is derived
611 (i.e. copied or compiled from other sources) this attribute MUST be

```

```
612 set to true. </xs:documentation>
613         </xs:annotation>
614     </xs:attribute>
615     </xs:complexType>
616 </xs:element>
617 <xs:element minOccurs="0" name="Author" type="xs:string">
618     <xs:annotation>
619         <xs:documentation>The identifier of the creators
620 of this document. For derived documents, this is the author. Note
621 that this identifier can identify machines as well as humans.
622 </xs:documentation>
623     </xs:annotation>
624 </xs:element>
625 <xs:element minOccurs="0" name="Organization"
626 type="xs:string">
627     <xs:annotation>
628         <xs:documentation>This element identifies the
629 organization. </xs:documentation>
630     </xs:annotation>
631 </xs:element>
632 </xs:sequence>
633 </xs:complexType>
634 <xs:complexType name="LinkInfo">
635     <xs:sequence>
636         <xs:element name="Target" type="xs:anyURI"/>
637         <xs:any maxOccurs="unbounded" minOccurs="0"/>
638     </xs:sequence>
639 </xs:complexType>
640 </xs:schema>
```

641

641 4 Informative Example (Non-Normative)

642 In this example, we will show a hData record that contains patient information, results and dicom images. The
 643 definition of what those components are is out of scope for the record format. If we have serialized this record
 644 to disk, this is what the structure would look like:

```
645 example-hrd/
646 |-- images
647 |   |-- mri_of_knee.dicom
648 |   |-- section.xml
649 |-- patient_information
650 |   |-- 1.xml
651 |   |-- section.xml
652 |-- results
653 |   |-- vital_signs
654 |       |-- 1234abc3248.xml
655 |       |-- section.xml
656 |   |-- labs
657 |       |-- 4567asdf8976.xml
658 |       |-- section.xml
659 |   |-- section.xml
660 |-- root.xml
661
```

662 The contents of the hData Record are described within the root.xml document. Here is what an hData Root
 663 document for this example might look like:

```
664 <?xml version="1.0" encoding="UTF-8"?>
665 <root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
666     xmlns="http://projecthdata.org/hdata/schemas/2009/06/core">
667     <documentId>125123124312</documentId>
668     <version>0.10</version>
669     <created>2010-02-09</created>
670     <lastModified>2010-02-09</lastModified>
671     <extensions>
672
673     <extension>http://projecthdata.org/hdata/schemas/2009/06/patient_info
674     rmation</extension>
675
676     <extension>http://projecthdata.org/hdata/schemas/2010/01/result</exte
677     nsion>
678
679     <extension>http://projecthdata.org/hdata/profile/2010/06/dicom_image<
680     /extension>
681         <extension>urn:empty</extension>
682     </extensions>
683     <sections>
684         <section
```

```
685 | typeId="http://projecthdata.org/hdata/schemas/2009/06/patient_informa
686 | tion" path="patient_information" name="Patient Information"/>
687 |     <section typeId="urn:empty" path="results" name="Results">
688 |         <section
689 | typeId="http://projecthdata.org/hdata/schemas/2009/06/result"
690 | path="vital_signs" name="Vital Signs">
691 |     <section
692 | typeId="http://projecthdata.org/hdata/schemas/2009/06/result"
693 | path="labs" name="Labs">
694 |     <section>
695 |     <section
696 | typeId="http://projecthdata.org/hdata/schemas/2010/06/dicom_image"
697 | path="images" name="Images" />
698 |     </sections>
699 | </root>
700 |
```

701 This root document contains some basic metadata about itself: an identifier, the version of the hData Record
702 Format that it conforms to, as well as its created and modified date.

703 Also contained is a list of used extensions. Extensions identify the content of sections. The record format only
704 requires that a globally unique string is used to identify an extension. It is recommended, but not necessary,
705 that the string be a URL where information can be found about the extension. Note that the use of URLs with
706 the domain part of the author guarantees global uniqueness. If the extension is describing XML documents, it is
707 again recommended that the URL resolve to an XML Schema for the documents that this extension is
708 describing. For non-XML content, a description of the content type (such as e.g. DICOM documentation) should
709 be provided at the URL used for identifying the extension.

710 For example, an extension may be identified by a URL that points to a CDA Template Registry that lists
711 templates that should be applied to CDA documents that would be stored in a hData section. Alternatively, an
712 extension may be identified by a URL that resolves to an XML Schema that has been derived by a microITS or
713 greenCDA process.

714 Extension identifiers may also resolve to documentation on what may appear in sections. An extension may
715 describe the structure of JSON documents it expects to be used in a section.

716 Note in this example that sections can be nested. In this case we have a results folder. This folder is expected to
717 have no section documents. It will still have an Atom feed which will provide links to the nested sections. It
718 contains two nested sections: Labs and Vital Signs. Both conform to the same extension, but this is not
719 required. The rules for determining what section documents go into which section is out of scope of the
720 specification, and would be determined by the content profile.

721 The section feeds (named "section.xml") contain a feed of all child elements - either sections or section
722 documents, including the identifiers (typically URLs for the RESTful API to hData Records) of the child
723 resources.

5 Bibliography

- [1] G. Beuchelt, R. Dingwell, A. Gregorowicz, and H. Sleeper, "hData Packaging and Network Transport Specification," The MITRE Corporation, 2009.
- [2] W3C XML Signature Syntax and Processing (Second Edition), <http://www.w3.org/TR/xmlsig-core/>, W3C Consortium
- [3] Resource Directory Description Language (RDDL), <http://www.rddl.org/>, J. Bordon, T. Bray