

HL7 hData Record Format v0.19

Gerald Beuchelt, Robert Dingwell, Andrew Gregorowicz, Mark Kramer, Samuel Sayer, Harry Sleeper

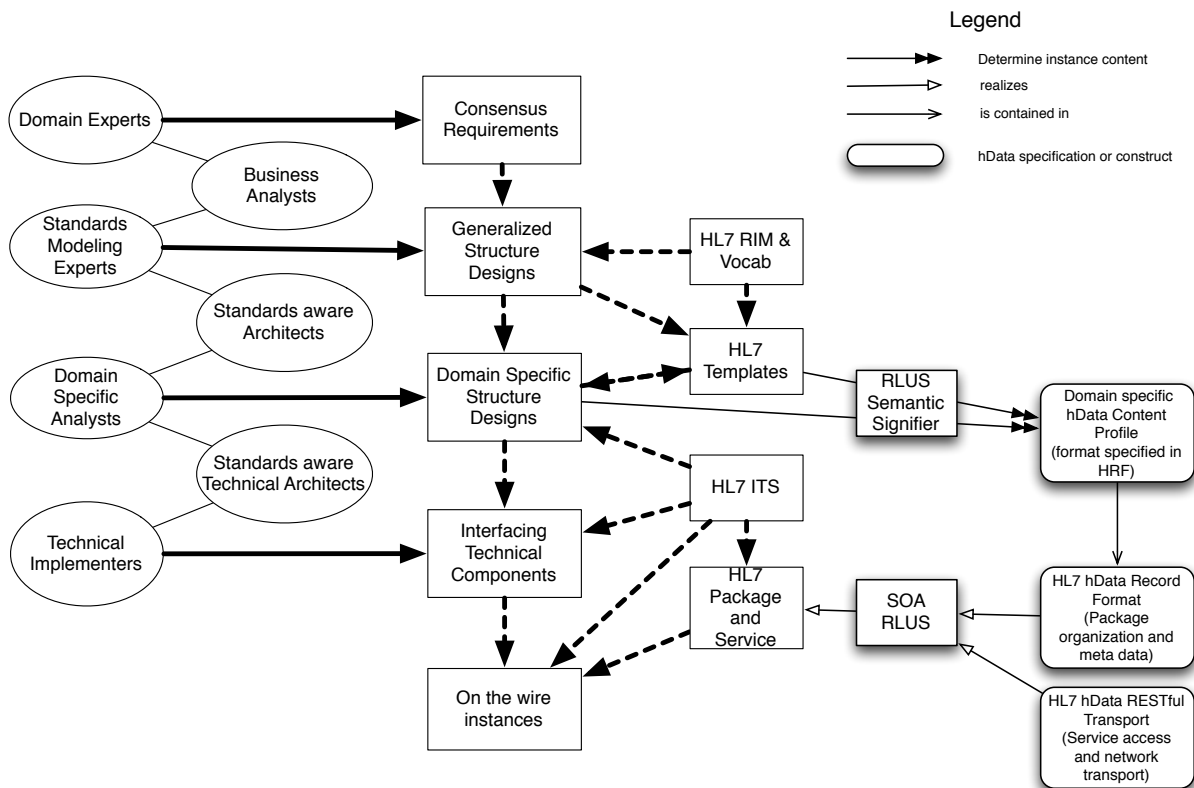
The MITRE Corporation
 202 Burlington Rd.
 Bedford, MA 01730
 U.S.A.

© 2009-2011 The MITRE Corporation. All rights reserved.

1 Introduction

The hData Record Format (HRF) describes the logical organization of the information in an electronic health record (EHR). The HRF is implemented through component-specific documents, which are linked and organized through a “master document” called the “root.xml” document. For better organization, the individual documents are put into a hierarchy, with the master document at the root of this hierarchy. While the HRF defines a core set of components, it is fully extensible and can easily be adopted for more complex situations.

The HRF can be accessed by any appropriate transport, including the HL7 hData RESTful Transport [1]. This transport provides means to locate, access, and update portions or the entire hData record.



Given this characterization, the hData specifications can be seen as a realization of the SOA Retrieve, Locate, Update, Service (RLUS) model. Domain Analysts should determine what content is appropriate to be included in a given hData Content Profile, which is determined by a Localized Information Model (LIM). The HCP also requires business justifications and behavioral modeling (where appropriate).

The information identified by the domain analysts must be rendered into “exchangeable goods” – this process is governed by the applicable ITS (expand acronyms). This is fairly straightforward when using the standard XML ITS, but future developments within the ITS WG may also allow simplified XML structures or JSON encoded content for wire-level exchanges.

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 metadata
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 RDDDL

1.2 Glossary (Non-Normative)

HL7 hData Record Format (HRF) - this specification specifies an abstract hierarchical organization, packaging, and metadata for individual documents (referred to as “Section Documents” within the HRF specification). Section Documents can be of any type, either XML documents (such as CDA documents, H7v3 messages, or simplified XML wire formats, etc.) or of other media types (such as e.g. MS Word documents or DICOM files). Also contained in this specification is the format for specifying the content that goes into an hData record, which is called the hData Content Profile (HCP) format.

hData Record (HDR) - a single instance of the HRF.

HL7 hData Restful Transport - this specification defines how the abstract hierarchical organization defined within the HRF specification is access and modified through a RESTful approach, using HTTP as the access protocol. It creates a unique mapping to an URL structure, and defines how HTTP verbs such as GET, PUT, DELETE, etc. affect the underlying information.

hData Content Profile (HCP) - a profile of the content of an HDR. The HRF specification contains the definition of the HCP format.

RLUS --- fill in!!

1.3 Notational Conventions

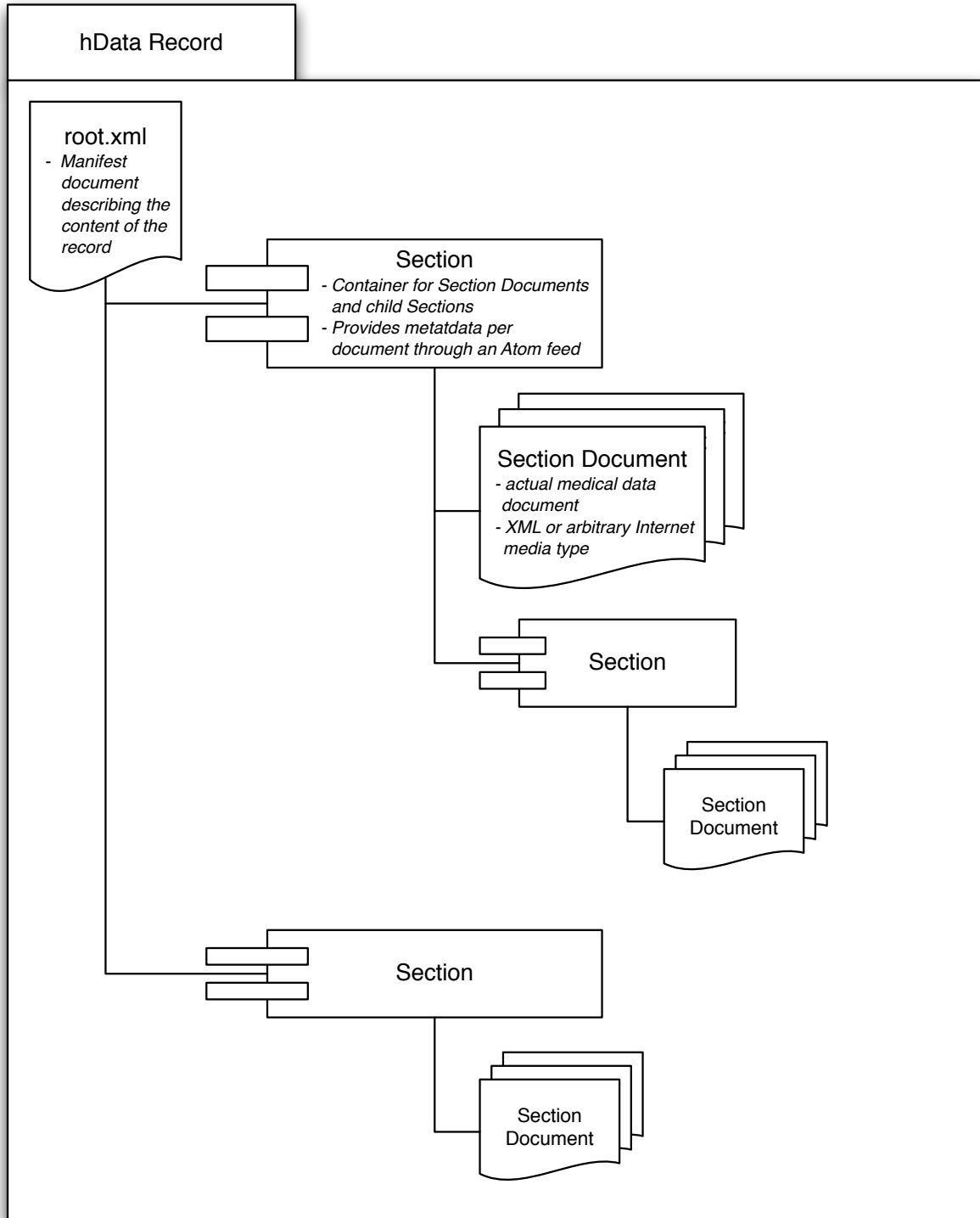
The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#).

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

Note also that only the W3C XML schemas linked in section 4 at the end of this document are normative – any schema fragment or other schema description within the main body of this document are informational only.

2 Hierarchical Organization

The basic approach of the hData Record Format is to represent the medical data through linked documents, which are organized through an abstract hierarchy. The hData RESTful API specification maps this abstract hierarchy to a concrete implementation, such as a web resource hierarchy.



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

2.1 Overall Structure

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

Each section corresponds to a single set of documents. Documents are grouped into sections by type. For 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. The list of children of this element represents the list of HL7 RLU semantic signifiers, as required by [5], section 5.2.1.
- /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. To allow sections that store no Section Documents, a root.xml MUST define an extension node of value “urn:empty”.
- /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 a local 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] and [.]. The full path to a section is obtained by starting with a forward slash ("/"), and concatenating the path segments, separated by forward slashes. It is RECOMMENDED that organizations creating hData Content Profile use their domain name in the first path segment (such as e.g. org.hl7.sample) to avoid namespace collisions in the full path name.
- /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 3).

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

2.3 Extensions

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

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

From an HL7 RLU perspective, each <Extension> is a semantic signifier, and the root.xml document defines a default semantic signifier for each Section.

2.4 Sections

Sections within an hData record form an abstract hierarchy, similar to the file folder structure commonly used in hierarchical file systems. Sections can contain either Section Documents or other Sections. Sections are identified by their path. The path to a Section is constructed by starting with a forward slash ("/") and appending all section path names from the root of the HDR to the Section. Section Documents contained in Sections comply with the contentType of an Extension registered in the RD. An Extension MUST be listed in /hrf:extensions for it to be used by a Section. Sections MAY use the same semantics for confidentiality, access control, and consent as described in the metadata for Section Document in 2.6. Sections MAY be empty.

2.5 Section Documents

At each section a collection of documents can be obtained. Within each Section, the documents MUST conform to the type defined by the Extension unless declared otherwise by the Section Document's metadata. Section documents can be of any media type, including binary media types.

2.6 Section Document Metadata

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

2.6.1 Section Feed Element Requirements

The following Atom feed level elements are RECOMMENDED:

- `<atom:title>` - This element SHOULD provide the full path from the root of the hData record to the Section, beginning with a "/" character, and separating each Section path segment with "/" characters.
- `<atom:updated>` - This element SHOULD provide the time when the Section or any of its child elements were last modified. Modifications could be new, updated, or deleted Section Documents or Sections, or changes to the metadata.
- `<atom:link rel="self" type="application/atom+xml">` This OPTIONAL element applies to transport that identify Sections by web resource identifiers (see [Atom 1.0], section 4.2.7). It has an href attribute with a globally unique URI that identifies the Section.

2.6.2 `<atom:entry>` Element Requirements

For each child of a section (either Section Document or Section) the Atom feed of the parent section provides one `<atom:entry>` node. The following list of child nodes defines how they MUST be populated:

- `<atom:id>` - This element contains a name for the document that is unique over the parent Section. For child Sections this name is the path segment for the child Section, as defined in the root.xml document. This element MUST be identical to the DocumentId element in the document metadata (see section 2.6.3).
- `<atom:link>` This OPTIONAL element applies to transport that identify Section Documents and Sections by web resource identifiers (see [Atom 1.0], section 4.2.7). It has an href attribute with a globally unique URI that identifies the Section or Section Document. For Sections, it MUST contain a type attribute with value "application/atom+xml". For Section Documents it SHOULD contain a type attribute that is identical to the media type of the referenced Section Document. If the media type is different from the Section default media type (as identified by the root.xml extension node), the type attribute is REQUIRED.
- `<atom:updated>` - For Section Documents, this element contains a W3C Date that is identical to the Section Document's metadata CreatedDateTime or the newest ModifiedDateTime (see section 2.6.3) time.

2.6.3 Section Document Metadata Definition

- /hrf-md:DocumentMetaData - DocumentMetaData is the top-level element for the hData metadata specification.
- /hrf-md:DocumentMetaData/hrf-md:PedigreeInfo (OPTIONAL) - This optional node holds the pedigree information for the Section Document. It is of type <hrf-md:PedigreeInfo>
- /hrf-md:DocumentMetaData/hrf-md:DocumentId - This required text element holds an identifier for the Section Document. It MUST be unique over any given Section.
- /hrf-md:DocumentMetaData/hrf-md:LinkedDocuments (OPTIONAL) - This optional node holds a list of URI links to documents that are related to this Section Document. Use depends on the semantics of the Section Document Type. It can have <hrf-md:LinkInfo> typed child elements.
- /hrf-md:DocumentMetaData/hrf-md:RecordDate - This required node holds the information about Document creation and modification.
- /hrf-md:DocumentMetaData/hrf-md:RecordDate/hrf-md:CreatedDateTime - This required element of type <xs:dateTime> contains the dateTime of creation of this document. If this document is not derived (see PedigreeInfo), this is the time of the creation of the original. If this document is derived from another origin, this element contains the date of derivation.
- /hrf-md:DocumentMetaData/hrf-md:RecordDate/hrf-md:Modified (OPTIONAL) - This optional node is first created when the document is changed for the first time. It contains a collection of modification dates with optional pedigree information of the modifier.
- /hrf-md:DocumentMetaData/hrf-md:RecordDate/hrf-md:Modified/hrf-md:ModifiedDateTime - This required element of type <xs:dateTime> records a dateTime when the document was modified.
- /hrf-md:DocumentMetaData/hrf-md:RecordDate/hrf-md:Modified/hrf-md:PedigreeInfo (OPTIONAL) – This optional node of type <hrf-md:PedigreeInfo> contains the pedigree information of the modifier.
- /hrf-md:DocumentMetaData/hrf-md:Confidentiality (OPTIONAL) – This element contains controls for confidentiality - details are out of scope for this specification and MAY be specified by an hData Access Control specification.
- /hrf-md:DocumentMetaData/hrf-md:AccessControl (OPTIONAL) - This element contains controls for access control - details are out of scope for this specification and MAY be specified by an hData Access Control specification.
- /hrf-md:DocumentMetaData/hrf-md:Consent (OPTIONAL) - This element contains controls for consent - details are out of scope for this specification and MAY be specified by an hData Access Control specification.

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

- /hrf-md:PedigreeInfo - This node contains the document pedigree information.
- /hrf-md:PedigreeInfo/hrf-md:XmlSignature (OPTIONAL) - This optional node contains the signature information on the document or this metadata. This signature MUST conform to the W3C XML Signature Syntax and Processing (Second Edition) [2] specification.
- /hrf-md:PedigreeInfo/hrf-md:XmlSignature/@documentMethod - This optional attribute indicates what method was used to transform binary Section Document media types into XML files for signature. Currently the only permitted methods are xml, sha256 and base64. xml is the default XML signature

over XML documents. base64 encodes a data stream into an XML document. The root contains the BASE64 encoded data. sha256 calculates a hash over the binary stream and signs this hash.

- /hrf-md:PedigreeInfo/hrf-md:XmlSignature/ds:Signature (0..unbounded) - A collection of XML Signatures. This Signature MUST contain:
 1. A valid Reference to either the metadata or the Section Document
 2. The ds:KeyInfo for the signer (optional with DSig - required here)
- /hrf-md:PedigreeInfo/hrf-md:Source (OPTIONAL) - This node indicates the source of this data.
- /hrf-md:PedigreeInfo/hrf-md:Source/@derived - If the data is derived (i.e. copied or compiled from other sources) this attribute of type <xs:boolean> MUST be set to true.
- /hrf-md:PedigreeInfo/hrf-md:Source/hrf-md:PedigreeInfo (0..unbounded) – This element contains the <hrf-md:PedigreeInfo> of the all source from which this document was derived.
- /hrf-md:PedigreeInfo/hrf-md:Source/hrf-md:Document (0..unbounded) – This element of type <hrf-md:LinkInfo> contains links to all documents from which this document was derived.
- /hrf-md:PedigreeInfo/hrf-md:Author (0..unbounded) – This element contains the names or identifiers of all author(s).
- /hrf-md:PedigreeInfo/hrf-md:Organization (0..unbound) - This element identified the organization(s) at which this document was created.

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

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

2.6.4 Metadata Processing Instructions

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

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

3 hData Content Profiles

This specification does not specify which sections are required for an hData Record. This is done in separate hData Content Profiles (HCP) which are specified through a HCP documentation package. An hData Content Profile prescribes the required and optional content a record must provide, and allocates the place for the

Section Documents within the hierarchical structure. Note that a single hData record can be compliant with multiple HCPs.

3.1 Relationship to HL7 RLUS

Similar to RLUS [4], the definition of the payloads contained in an HDR is beyond the scope of the HL7 hData specification. The HCP definition document (see section 3.3) or the metadata for each Document contains the RLUS semantic signifiers ([4], section 9.1) for the SectionDocument resource in the form of a URI. As such, any HL7 Version 3 compliant HCP will need to be accompanied by a Localized Information Model (LIM) that formally describes the semantic signifiers. As such, an HCP constitutes Semantic Profile in the sense of [5], Section 6.1.

3.2 HCP Documentation Package

The HCP documentation package is composed of the following documents:

1. HCP definition document (see section 3.3) – This document is REQUIRED
2. Complete set of all XML schemas referenced in the HCP definition document – The schemas are REQUIRED.
3. Sample instance – A sample instance of an hData record that complies with the HCP is RECOMMENDED. The sample instance SHOULD be provided through serialized XML Section Documents that are stored in a hierarchical file system corresponding to the Section layout described in the HCP definition document. To simplify transmission these documents can be stored in a ZIP file supporting hierarchical file storage.
4. Transforms to XML ITS/CDA – If the HCP uses a simplified wire format, it SHOULD provide XML transforms for converting the simplified wire format into HL7 v3 XML ITS CDA if the exchanged data can be mapped to HL7 data types. The transforms are OPTIONAL.
5. Master documentation – This REQUIRED text document MUST including the purpose, applicable business requirements, and a basic justification for the HCP need. It is REQUIRED, but its scope depends on the original intent of the HCP.
6. Behavioral model and business rules – If the domain for the HCP must adhere to a behavioral model or a set of business rules, these MUST be documented. This documentation SHOULD include applicable UML diagrams or similar documentation. This document is REQUIRED if a behavioral model or business rules exist.
7. Use cases – The applicable use cases SHOULD be captured in a suitable document.
8. Testing and conformance documents – If there are any additional testing or conformance requirements, these SHOULD be documented here.
9. Change log – This REQUIRED document must contain a table with the following columns:
 - a. Date of change
 - b. Name of editors and/or organization
 - c. Version-aware identification URL (see 10.a)

- d. Major changes
 - e. Other comments
10. Other Metadata – This REQUIRED metadata can be documented in a single HTML document.
- a. URL for identification – This URL MUST be version aware, i.e. it MUST be different for differing versions of the HCP. This can be achieved by including the change year and month in the URL. The URL SHOULD resolve to a publicly accessible HTML resource that contains links to all documents needed for the HCP.
 - b. Name, summary, creation date

3.3 hData Content Profile Definition Document

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

- /hcp:hcp – the root element for a HCP definition file.
- /hcp:hcp/@name – a simple display name
- /hrf:hcp/@id – a URI identifying the hData Content Profile. It is RECOMMENDED to use a URL that can be resolved into the HCP definition document.
- /hrf:hcp/hrf:extensions – this element describes the extensions used in this HCP. It uses the same syntax as in the root document as described in section 2.2.
- /hrf:hcp/hrf:sections – this element describes the sections that are to be included in a hData record that claims conformance to the HCP. It uses the same syntax as in the root document as described in section 2.2. NOTE: the requirements attribute is being used in the HCP, as described above.

4 Schemas

4.1 Root Document

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

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Copyright 2009 The MITRE Corporation
```

```
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0
```

```
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied.
```

```
©2009-11 The MITRE Corporation. Unlimited distribution. Public Release Approval 09-4511
```

See the License for the specific language governing permissions and limitations under the License. -->

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:core="http://projecthdata.org/hdata/schemas/2009/06/core"
elementFormDefault="qualified"
targetNamespace="http://projecthdata.org/hdata/schemas/2009/06/core">
  <xs:element name="root">
    <xs:complexType>
      <xs:all>
        <xs:element ref="core:id"/>
        <xs:element ref="core:version"/>
        <xs:element ref="core:created"/>
        <xs:element ref="core:lastModified"/>
        <xs:element ref="core:extensions"/>
        <xs:element ref="core:sections"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
  <xs:element name="id" type="xs:string"/>
  <xs:element name="version" type="xs:string"/>
  <xs:element name="created" type="xs:date"/>
  <xs:element name="lastModified" type="xs:date"/>
  <xs:element name="extensions">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded"
ref="core:extension"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="extension">
    <xs:complexType mixed="true">
      <xs:attributeGroup ref="core:extension"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="sections">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded"
ref="core:section"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:attributeGroup name="extension">
    <xs:attribute name="contentType" type="xs:string"
use="optional"/>
    <xs:attribute name="extensionId" type="xs:string"

```

```

use="required"/>
</xs:attributeGroup>
<xs:element name="section">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded"
ref="core:section"/>
    </xs:sequence>
    <xs:attribute name="path" use="required"/>
    <xs:attribute name="name" use="optional"/>
    <xs:attribute name="extensionId" use="required"/>
    <xs:attribute name="requirement" use="optional">
      <xs:simpleType>
        <xs:restriction base="xs:token">
          <xs:enumeration value="mandatory"/>
          <xs:enumeration value="optional"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
</xs:element>
</xs:schema>

```

4.2 hData Content Profile Definition

This section contains the schema for the hData Content Profile definition (see section 3). All instances of HCP definition documents MUST validate against this schema definition.

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Copyright 2010 The MITRE Corporation
Licensed under the Apache License, Version 2.0 (the "License"); you
may not use this file except in compliance with the License. You may
obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied.
See the License for the specific language governing permissions and
limitations under the License. -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:hcp="http://projecthdata.org/hdata/schemas/2010/04/hcp"
xmlns:core="http://projecthdata.org/hdata/schemas/2009/06/core"
elementFormDefault="qualified"
targetNamespace="http://projecthdata.org/hdata/schemas/2010/04/hcp">

```

```

<xs:import
namespace="http://projecthdata.org/hdata/schemas/2009/06/core"
schemaLocation="http://github.com/projecthdata/hData/raw/master/schem
as/2009/06/root.xsd"/>
  <xs:element name="hcp">
    <xs:complexType>
      <xs:all>
        <xs:element ref="core:extensions"/>
        <xs:element ref="core:sections"/>
      </xs:all>
      <xs:attribute name="name" use="required" type="xs:string"/>
      <xs:attribute name="id" use="required" type="xs:anyURI"/>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

4.3 Section Document Metadata

This section contains the schema for the Section Document metadata (see Section 2.6). All instances of metadata documents MUST validate against this schema definition.

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Copyright 2009 The MITRE Corporation

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing,
software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied.
See the License for the specific language governing permissions
and
limitations under the License. -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:hd-
md="http://projecthdata.org/hdata/schemas/2009/11/metadata"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
elementFormDefault="qualified"
targetNamespace="http://projecthdata.org/hdata/schemas/2009/11/metada
ta">
  <xs:import namespace="http://www.w3.org/2000/09/xmldsig#"
schemaLocation="http://www.w3.org/TR/2008/REC-xmldsig-core-
20080610/xmldsig-core-schema.xsd"/>

```

```

<xs:element name="DocumentMetaData">
  <xs:annotation>
    <xs:documentation>
      DocumentMetaData is the top-level element for the
hData meta data specification. It is
      embedded with every Atom 1.0 Content node.
    </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="PedigreeInfo" type="hd-
md:PedigreeInfo">
        <xs:annotation>
          <xs:documentation>
            This optional node holds the pedigree
information for the Section Document.
          </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="DocumentId" type="xs:string">
        <xs:annotation>
          <xs:documentation>
            This required element holds an identifier
for the Section Document. It MUST be unique over any given
            Section feed.
          </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="Title" type="xs:string">
        <xs:annotation>
          <xs:documentation>
            This required element holds the title of
the Section Document.
          </xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element minOccurs="0" name="LinkedDocuments">
        <xs:annotation>
          <xs:documentation>
            This optional node holds a list of URI
links to documents that are related to this
            Section Document. Use depends on the
semantics of the Section Document Type.
          </xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

```

```

        <xs:element maxOccurs="unbounded" name="Link" type="hd-
md:LinkInfo" />
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="RecordDate">
    <xs:annotation>
        <xs:documentation>
            This required node holds the information
about Document creation and modification.
        </xs:documentation>
    </xs:annotation>
<xs:complexType>
    <xs:sequence>
        <xs:element name="CreatedDateTime" type="xs:dateTime">
            <xs:annotation>
                <xs:documentation>
                    This required element
contains the dateTime of creation of this document. If this document
is not derived (see PedigreeInfo), this is the time of the creation
of the original. If this document is derived from another origin,
this element contains the date of derivation.
                </xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element minOccurs="0" name="Modified">
            <xs:annotation>
                <xs:documentation>
                    This optional node is first
created when the document is changed for the first time. It contains
a collection of modification dates with optional pedigree information
of the modifier.
                </xs:documentation>
            </xs:annotation>
<xs:complexType>
            <xs:sequence minOccurs="1" maxOccurs="unbounded">
                <xs:element name="ModifiedDateTime"
type="xs:dateTime">
                    <xs:annotation>
                        <xs:documentation>
                            This required
element record a dateTime when the document was modified.
                        </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element minOccurs="0" name="PedigreeInfo"
type="hd-md:PedigreeInfo">

```



```

        <xs:annotation>
            <xs:documentation>
                This optional
node contains the pedigree information of the modifier.
                </xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element minOccurs="0" name="Confidentiality"
type="xs:string">
    <xs:annotation>
        <xs:documentation>
            This element contains controls for
confidentiality - details are TBD.
        </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element minOccurs="0" name="AccessControl">
    <xs:annotation>
        <xs:documentation>
            This element contains controls for access
control - details are TBD.
        </xs:documentation>
    </xs:annotation>
</xs:element>
<xs:element minOccurs="0" name="Consent">
    <xs:annotation>
        <xs:documentation>
            This element contains controls for
consent - details are TBD.
        </xs:documentation>
    </xs:annotation>
</xs:element>
</xs:sequence>
<xs:attribute name="MediaType" type="xs:string">
    <xs:annotation>
        <xs:documentation>
            This attribute contains the media type of the
document itself. If it is not present, the
            default media type of the content type is
assumed.
        </xs:documentation>
    </xs:annotation>
</xs:documentation>

```

```

        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="ContentType" type="xs:anyURI"
use="optional">
        <xs:annotation>
            <xs:documentation> This attribute contains the URI for the
content type of this document. If it is not present, the content type
for the Section is implied. Note that the current hData Content
Profiles assume that the content type for all Section Documents
within a given Section is uniform.
            </xs:documentation>
        </xs:annotation>
    </xs:attribute>
</xs:complexType>
</xs:element>
<xs:complexType name="PedigreeInfo">
    <xs:annotation>
        <xs:documentation>
            This node contains the pedigree information.
        </xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element minOccurs="0" name="XmlSignature"
maxOccurs="unbounded">
            <xs:annotation>
                <xs:documentation> This optional node contains the
signature information on
                    the document or this meta data.
            </xs:documentation>
        </xs:annotation>
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="ds:Signature">
                    <xs:annotation>
                        <xs:documentation> This Signature MUST contain:
1. a valid Reference to either the metadata or the Section Document
2. the ds:KeyInfo for the signer (optional with DSig - required here)
                        </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        <xs:attribute name="documentMethod">
            <xs:annotation>
                <xs:documentation>This optional attribute indicates
what method was used to transform binary Section Document mediatypes
into XML files for signature. Currently the only permitted methods
are xml and base64. xml is the default XML signature over XML

```

documents. base64 encodes a data stream into an XML document. The root node is root and contains the BASE64 encoded data.

```

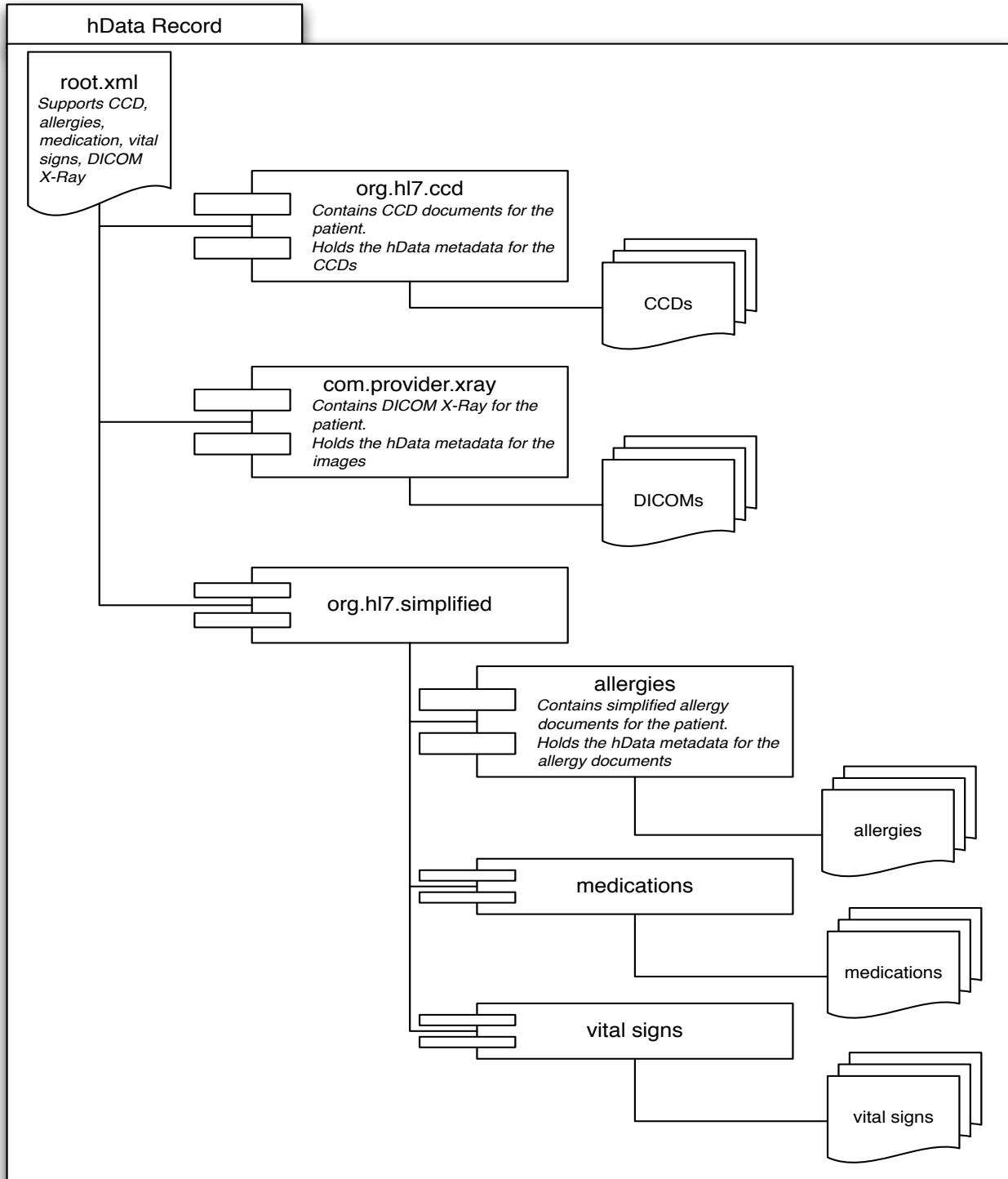
</xs:documentation>
  </xs:annotation>
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="base64"/>
      <xs:enumeration value="xml"/>
      <xs:enumeration value="sha256"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
</xs:complexType>
</xs:element>
<xs:element minOccurs="0" maxOccurs="1" name="Source">
  <xs:annotation>
    <xs:documentation>This node indicates the source of this
data. </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="PedigreeInfo" type="hd-md:PedigreeInfo"
minOccurs="0"/>
      <xs:element maxOccurs="unbounded" minOccurs="0"
name="Document" type="hd-md:LinkInfo"/>
    </xs:sequence>
    <xs:attribute name="derived" type="xs:boolean">
      <xs:annotation>
        <xs:documentation>If the data is derived (i.e. copied
or compiled from other sources) this attribute MUST be set to true.
</xs:documentation>
      </xs:annotation>
    </xs:attribute>
  </xs:complexType>
</xs:element>
<xs:element minOccurs="0" name="Author" type="xs:string">
  <xs:annotation>
    <xs:documentation>The identifier of the creators of this
document. For derived documents, this is the author. Note that this
identifier can identify machines as well as humans.
</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element minOccurs="0" name="Organization" type="xs:string">
  <xs:annotation>
    <xs:documentation>This element identifies the organization.
</xs:documentation>
  </xs:annotation>
</xs:element>
</xs:sequence>

```

```
</xs:complexType>
<xs:complexType name="LinkInfo">
  <xs:sequence>
    <xs:element name="Target" type="xs:anyURI"/>
    <xs:any maxOccurs="unbounded" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
</xs:schema>
```

5 hData Record Example (Non-Normative)

This section outlines a simple hData Record that contains HL7 Continuity of Care Documents (CCD), simplified documents containing information about allergies, medications, and vital signs, and radiology imagery. The following figure illustrates the high-level structure of the example record.



Note that the HRF specification does not dictate how the information making up the Section Documents is stored. For example, the CCD documents in this example can be populated from the same underlying data source as the simplified documents.

The following subsections describe each of the hData Record components.

5.1 root.xml Document

The contents of the hData Record are described within the root.xml document (see section 2.2). For the hData record outlined above, the root.xml document looks like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<root xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://projecthdata.org/hdata/schemas/2009/06/core">
  <id>125123124312</id>
  <version>1</version>
  <created>2010-02-09T15:01:35-5:00</created>
  <lastModified>2010-02-09T19:35:44-5:00</lastModified>
  <extensions>
    <extension extensionId="1"
      contentType="application/hl7-sda+xml">
      http://hl7.org/v3/cda-r2/ccd
    </extension>
    <extension extensionId="2"
      contentType="application/dicom">
      http://picturealliance.example.com/x-ray/2011/05
    </extension>
    <extension extensionId="3"
      contentType="application/xml">
      http://hl7.org/hdata/2012/01/allergies
    </extension>
    <extension extensionId="4"
      contentType="application/xml">
      http://hl7.org/hdata/2012/01/medications
    </extension>
    <extension extensionId="5"
      contentType="application/xml">
      http://hl7.org/hdata/2012/01/vitalsigns
    </extension>
    <extension extensionId="6">
      urn:empty
    </extension>
  </extensions>
  <sections>
    <section path="org.hl7.ccd" extensionId="1" />

    <section path="com.provider.xray" extensionId="2" />
    <section path="org.hl7.simplified" extensionId="6">
      <section path="allergies" extensionId="3" />
      <section path="medications" extensionId="4" />
      <section path="vital signs" extensionId="5" />
    </section>
  </sections>
</root>
```

```
</root>
```

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

The root document also contains a list of extensions. Extensions identify the format of section documents. As such, they are the HL7 RLUS semantic signifiers for the Section Documents. The record format only requires that a globally unique string is used to identify an extension. It is recommended, but not necessary, that the string be a URL where information can be found about the extension. Note that the use of URLs with the domain part of the author guarantees global uniqueness. If the extension is describing XML documents, it is again recommended that the URL resolve to a RDDL document that provide a machine-resolvable link to the XML Schema for the documents that this extension is describing. For non-XML content, a description of the content type (such as e.g. DICOM documentation) should be provided at the URL used for identifying the extension.

The sections identify their default content by referencing the extensionId attributes of the extension nodes. They also contain the path segment that is used to construct the full path to the section. It should be noted that sections can be nested. In this example, the results folder contains an empty folder (org.hl7.simplified) that cannot contain documents since its extension is "urn:empty". However, it will still have an Atom feed which will provide links to the nested sections.

5.2 Section Document Metadata

The metadata for Section Document is provided at the Section level through an Atom 1.0 feed (see section 2.6). Below is a sample feed for the /org.hl7.simplified/allergies Section. Other sections will have very similar feeds, that describe contained Section Documents and nested Sections.

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom"
      xmlns:hxf-md="http://projecthdata.org/hdata/schemas/2009/11/metadata">
  <title>/org.hl7.simplified/allergies</title>
  <link href="http://example.org/patient1234/org.hl7.simplified/allergies/"
        rel="self" />
  <updated>2011-12-13T18:30:02Z</updated>
  <entry>
    <id>allergy1.xml</id>
    <link
href="http://example.org/patient1234/org.hl7.simplified/allergies/allergy1.
xml" type="application/xml"/>
    <updated>2011-12-13T18:30:02Z</updated>
    <content>
      <hxf-md:DocumentMetaData>
        <hxf-md:DocumentId>allergy1.xml</hxf-md:DocumentId>
        <hxf-md:RecordDate>
          <hxf-md:CreateDateTime>
            2009-10-10T09:21:55Z
```

```

        </hrf-md:CreatedDateTime>
        <hrf-md:Modified>
          <hrf-md:ModifiedDateTime>
            2011-12-13T18:30:02Z
          </hrf-md:ModifiedDateTime>
        </hrf-md:Modified>
      </hrf-md:RecordDate>
      <hrf-md:LinkedDocuments>
        <hrf-md:LinkInfo>
          <hrf-md:Target>
            http://example.com/additionalPatientInfo/patient1234/allergyhistory
          </hrf-md:Target>
        </hrf-md:LinkInfo>
      </hrf-md:LinkedDocuments>
    </hrf-md:DocumentMetaData>
  </content>
</entry>
<entry>
  <id>allergy2.xml</id>
  <link
href="http://example.org/patient1234/org.hl7.simplified/allergies/allergy2.
xml" type="application/xml"/>
  <updated>2010-02-27T12:21:11Z</updated>
  <content>
    <hrf-md:DocumentMetaData>
      <hrf-md:DocumentId>allergy1.xml</hrf-md:DocumentId>
      <hrf-md:PedigreeInfo>
        <hrf-md:Author>Dr. John Doe</hrf-md:Author>
        <hrf-md:Organization>Sample Provider, Inc.</hrf-md:Organization>
      </hrf-md:PedigreeInfo>
      <hrf-md:RecordDate>
        <hrf-md:CreateDateTime>
          2010-02-27T12:21:11Z
        </hrf-md:CreateDateTime>
      </hrf-md:RecordDate>
    </hrf-md:DocumentMetaData>
  </content>
</entry>
</feed>

```

5.3 hData Content Profile Definition Document

The following hData Content Profile description document describes the hData Record in this example. Note that the simplified Sections have been marked as optional.

```

<?xml version="1.0" encoding="UTF-8"?>
<hcp xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://projecthdata.org/hdata/schemas/2010/04/hcp"

```



```

xmlns:hrf="http://projecthdata.org/hdata/schemas/2009/06/core"
id="http://example.com/hdata/hcp/2011/05/sample"
name="Example hData Content Profile" >
<hrf:extensions>
  <hrf:extension extensionId="1"
    contentType="application/hl7-sda+xml">
    http://hl7.org/v3/cda-r2/ccd
  </hrf:extension>
  <hrf:extension extensionId="2"
    contentType="application/dicom">
    http://picturealliance.example.com/x-ray/2011/05
  </hrf:extension>
  <hrf:extension extensionId="3"
    contentType="application/xml"
    http://hl7.org/hdata/2012/01/allergies
  </hrf:extension>
  <hrf:extension extensionId="4"
    contentType="application/xml"
    http://hl7.org/hdata/2012/01/medications
  </hrf:extension>
  <hrf:extension extensionId="5"
    contentType="application/xml"
    http://hl7.org/hdata/2012/01/vitalsigns
  </hrf:extension>
  <hrf:extension extensionId="6">
    urn:empty
  </hrf:extension>
</hrf:extensions>
<hrf:sections>
  <hrf:section path="org.hl7.ccd" extensionId="1" />
  <hrf:section path="com.provider.xray" extensionId="2" />
  <hrf:section path="org.hl7.simplified" extensionId="6"
    requirement="optional">
    <hrf:section path="allergies" extensionId="3"
      requirement="optional" />
    <hrf:section path="medications" extensionId="4"
      requirement="optional" />
    <hrf:section path="vital signs" extensionId="5"
      requirement="optional" />
  </hrf:section>
</hrf:sections>
</hcp>

```

6 Bibliography

[1] G. Beuchelt, R. Dingwell, A. Gregorowicz, and H. Sleeper, "HL7 hData RESTful 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

©2009-11 The MITRE Corporation. Unlimited distribution. Public Release Approval 09-4511

[4] HL7 Resource Location and Updating Service (RLUS), DSTU Release 1, Health Level Seven, Inc., December 2006