

hData TB Progress Note Implementation Guide

Gerald Beuchelt, Andrew Gregorowicz

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

© 2010 The MITRE Corporation. All rights reserved.

1 Introduction

This document provides an introduction to using hData to produce and consume CDA R2 Based TB Progress Notes. While the HL7 Clinical Document Architecture (CDA) provides a standards-based framework for expressing the information contained in a progress note, hData can provide an alternate interface to this data. System implementers may choose to use hData due to its design characteristics, which differ from the CDA. hData focuses on small XML documents, where element names match business names for XML tags. An emphasis is also placed on having tightly defined XML Schemas. This contrasts with the CDA, which focuses on communicating precise semantic meaning through a set of reusable XML elements and attributes.

This guide walks through the creation of an hData Record that contains the information needed for a TB Progress Note. It will then describe how to use a translator tool to convert the record into a CDA document. Conversely, the guide will illustrate how to convert a CDA-based TB Progress Note into an hData Record. This allows implementers who prefer the hData approach to data encoding to easily interoperate with others who prefer CDA.

1.1 Intended Audience

This document is intended for system implementers who need to consume or produce CDA R2 TB Progress Notes. It assumes a familiarity with XML and XML Schema. It also assumes some rudimentary knowledge of the HL7 CDA.

2 hData Overview

hData is a simple XML framework for the creation, storage, and exchange of health data. hData contains two main components:

1. hData Record Format (HRF): The HRF describes an abstract architecture of how data is stored in multiple XML documents and organized in a hierarchy. It also contains a concrete schema for the HRF metadata. Records conforming to the HRF are called hData Records (HDRs).

2. hData Content Profiles (HCP): Concrete schemata that define the data to be stored in an hData record as well as the hierarchy the data should conform to.

Separating the hData Content Profiles from the technical container and communication specification serves the interests of both the health care and technical communities: the HCPs should be defined and managed by medical domain experts, while the data organization issues can be better addressed by data management experts. By separating the problem areas and creating two distinct specification sets, the respective communities and their standardization organizations can employ their own expertise and follow their own time lines.

The hData Record Format (HRF) follows the approach taken by the Open Document Format (ODF) and other modern XML file formats: at the core of the document is a “root” document containing metadata describing the actual medical data documents, which are located within a hierarchy of sections. These individual XML documents are referred to as “section documents” and are located within a section. Any given section can only contain section-documents of one type or other sections. These sections can easily be represented as a file folder hierarchy on disk or within a ZIP file, or as web resources.

The hData Record Format was created with extensibility in mind. Since we do not expect to be able to address all potential use cases with a single HCP, hData can be extended by defining new sections for additional XML documents to deliver additional functionality with almost no limitations to the format of the extensions. While highly desirable, we do not expect that all consumers of hData will be required or, indeed, will be capable of parsing all documents, so extensions must be marked as mandatory or optional.

For the case of the TB Progress Note, we can construct an HCP that encapsulates all of the information stored in the CDA based version. This HCP will be the basis of translation between hData Records and CDA documents.

3 Translation Process

The conversion between hData Records and CDA documents is achieved through the use of XProc. This technology allows for the definition of an XML Pipeline, which is a list of operations to be performed on a set of input XML documents. These operations include inserting into or replacing the contents of an XML element, transforming a document using XSLT and examining the contents of a directory. Through use of these procedures, we can construct pipelines that will convert documents between the two formats.

3.1 XProc Environment

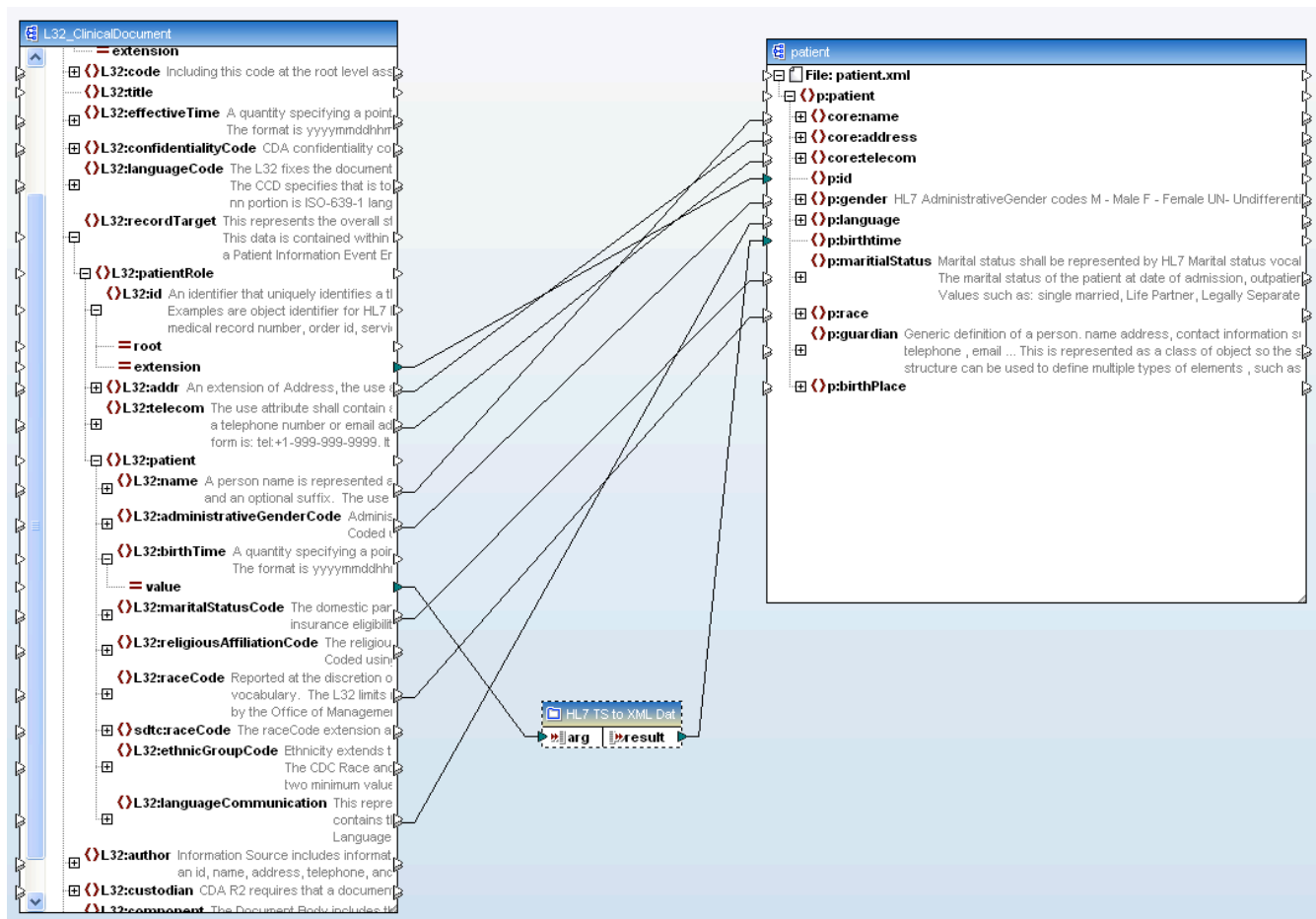
This implementation guide was created using Calabash, a free implementation of XProc. Other implementations of XProc are available. The pipelines created to facilitate the hData/CDA conversion process should work in any XProc implementation, but they have only been tested in Calabash.

Calabash is a java-based tool. To assist in its usage, an Apache Ant build file is provided that will ensure that Calabash has the necessary environment available. This Ant file is provided as a convenience and is not necessary for proper execution of the XProc pipelines.

3.2 XSLT Creation

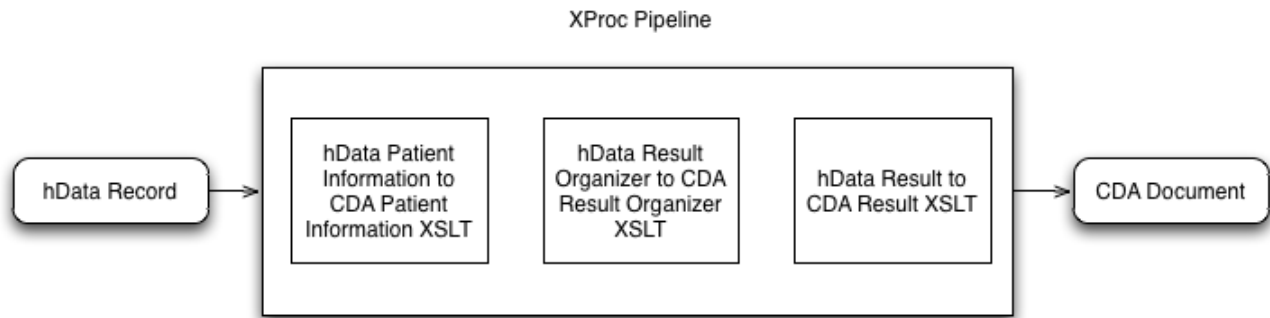
XSLT Stylesheets used in the pipelines were created using Altova MapForce. This is a tool that allows for the creation of mappings between two XML Schemas. XML Schemas from the hData TB Progress Note Content Profile were used to represent the information in hData Records. For CDA, we used the XML Schemas in the L32 project (<http://www.projecthdata.org/l32.html>), a set of strongly typed XML Schemas for the HITSP C32. These schemas were chosen because the templates used in the TB Progress Note are derived from the HL7 CCD. Since L32 is also a derivative of the CCD, the schemas could easily be repurposed for usage in the mapping.

Below is an illustration of the mapping from CDA Person Information to hData Patient Information:



MapForce generates XSLT from these visual mappings. In most cases these transforms were used without modification. Stylesheets for translating results required some manual editing since MapForce does not have strong support for anyTypes in XML Schema. Handwritten XSLT was used to perform the appropriate conversions.

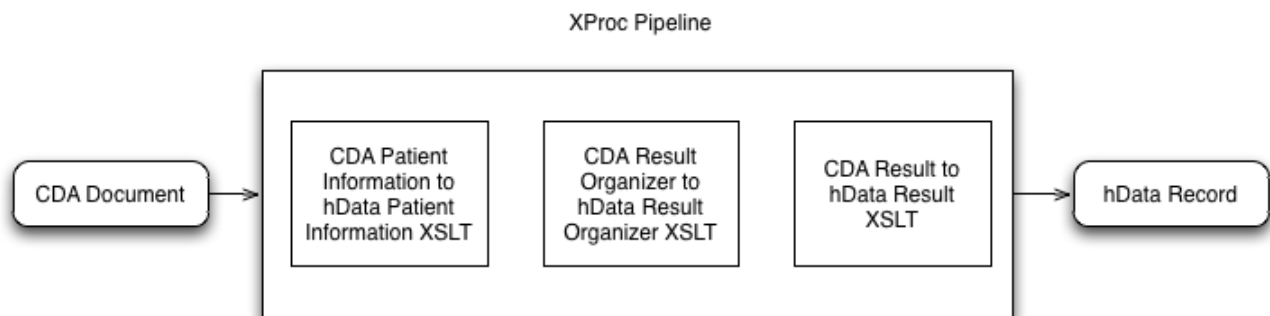
3.3 hData Record to CDA Translation



In the pipeline to convert hData Records into a CDA Document, there are three main XSLT processes involved. The first takes the hData Person Information and converts it into an XML fragment representing the CDA record target. XProc then takes that XML fragment and inserts it into a stub CDA document. This document contains the proper template ids for a TB Progress Notes and has place holders for the record target and results sections.

Next, result organizers are converted with XSLT, followed by individual results. These XML fragments are then joined and inserted into the stub CDA document. The result is a full CDA representation of the hData Record.

3.4 CDA to hData Record Translation



This process is extremely similar to the hData to CDA conversion. Three main stylesheets are involved in transforming the CDA to an hData Record. The main difference is that the output of the stylesheet execution is used to create multiple XML documents that are stored in folders within the hData Record.

3.5 Translation Scope

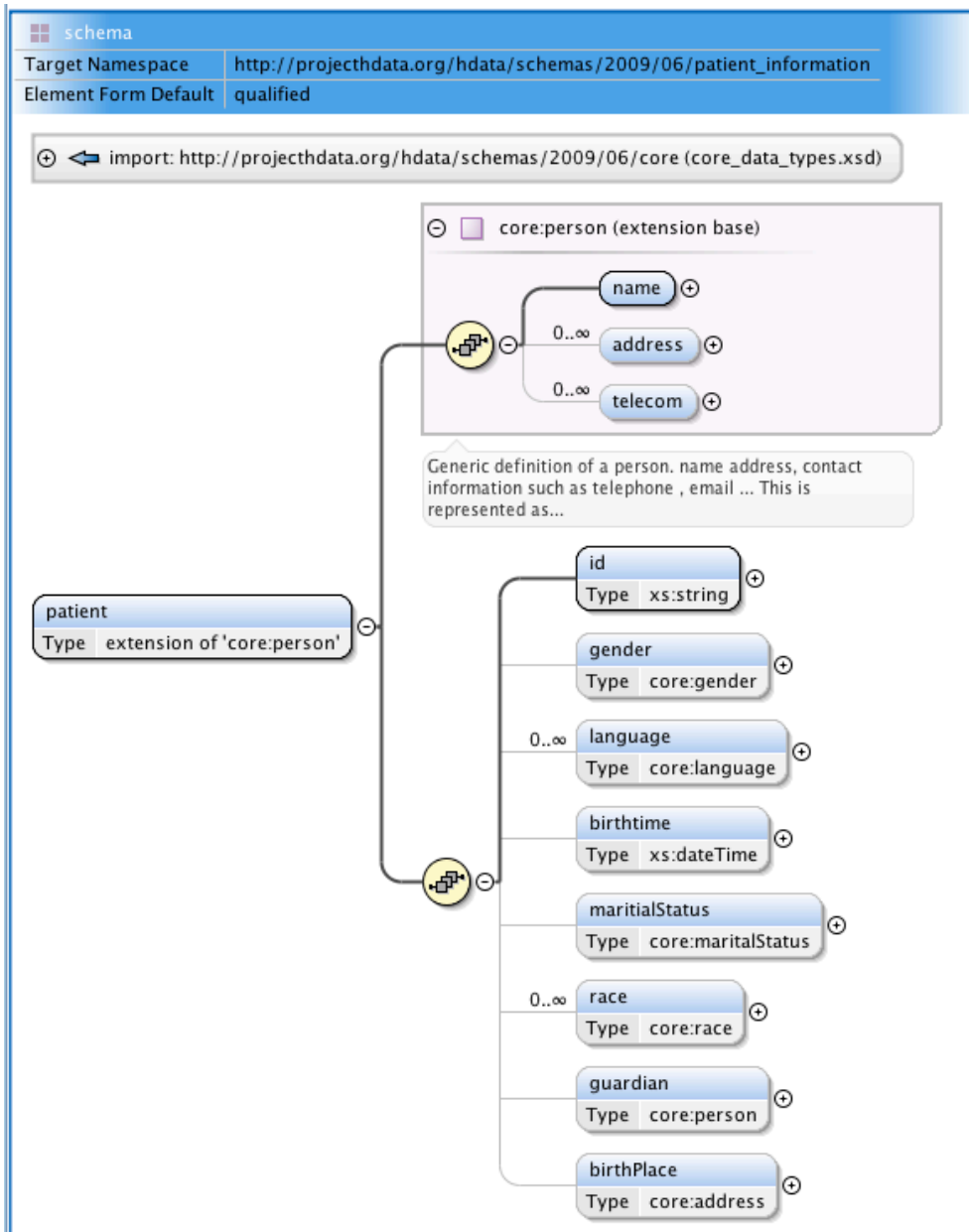
The XProc pipelines described above are only valid for converting between CDA documents conforming to the TB Progress Note template and hData Records conforming to the TB Progress Note hData Content Profile. Supporting different CDA templates or hData Content Profiles will require additional mapping to create appropriate XSLT and XProc pipeline modification. However, much of the existing pipeline should be reusable, and for CCD based CDA templates, the L32 Project can again be used as a mapping target.

4 hData XML Schemas

hData uses a number of XML Schemas to represent the information contained in a CDA R2 Based TB Progress Note. Together these schemas form an hData Content Profile for TB Progress Notes. They are described in detail in the following sections.

4.1 Patient Information

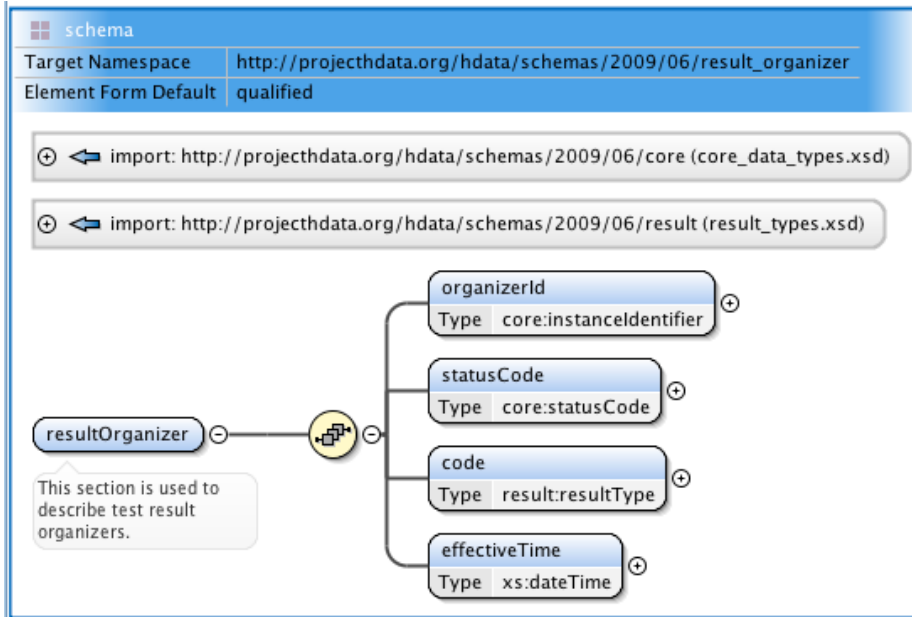
This is used to represent the information in a CDA recordTarget/patientRole.



Coded values reuse HL7 vocabularies where possible. For example, gender is recorded with HL7 Administrative Gender Codes and maritalStatus uses HL7 Martial Status.

4.2 Result Organizer

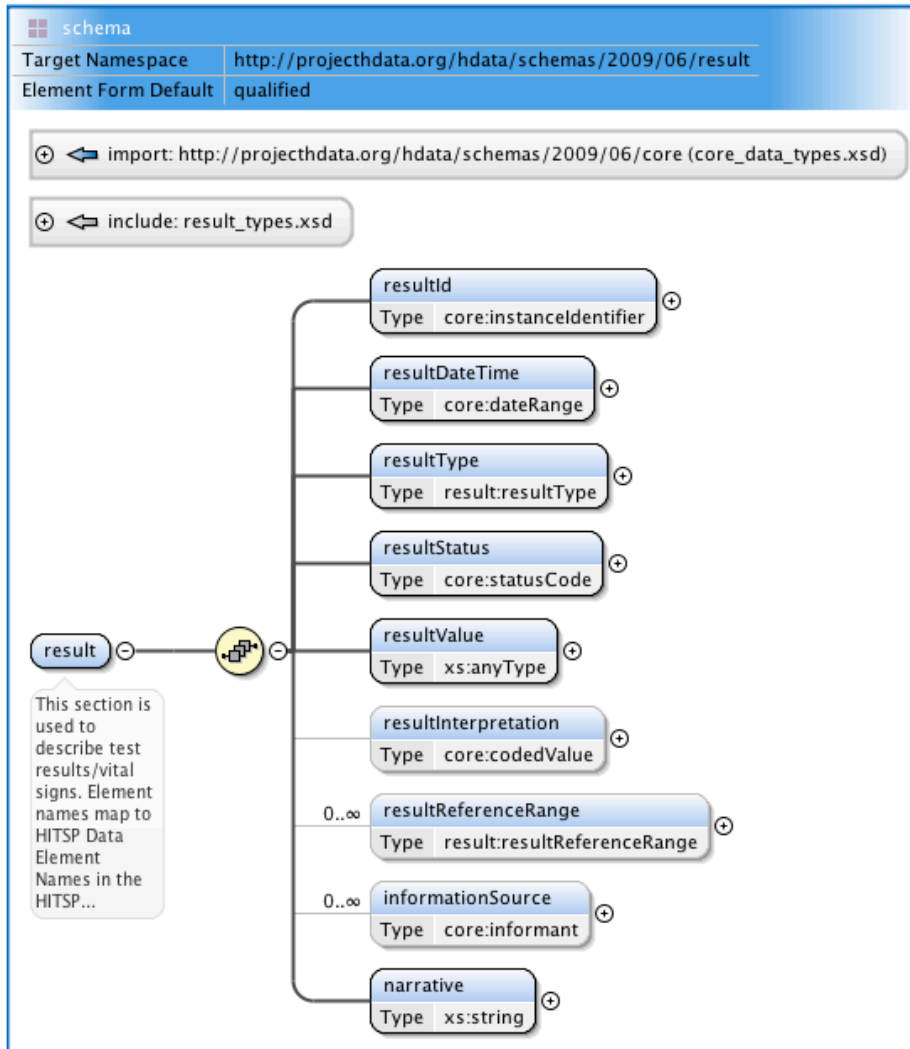
This schema maps to TB Progress Note template id 2.16.840.1.113883.10.20.15.3.21.



Unlike CDA Result Organizers, the Result Observations are not contained in the XML of the organizer. Instead, they are stored in their own XML documents and grouped within the folder structure of the hData Record. Again, HL7 vocabularies are reused as `statusCode` using the HL7 Act Status Normal code set.

4.3 Result

This schema maps to TB Progress Note template id 2.16.840.1.113883.10.20.15.3.13.



It should be noted that `resultValue` is an `anyType`. This corresponds to the value element in CDA based results, which also has an `anyType`. The current XSLT only performs conversions for CDA PQ values. Any other types will require manual adjustment of the appropriate stylesheets.

5 Implementation Considerations

Users of hData for consuming TB Progress Notes should be aware of some of the differences with HL7 CDA and how the hData Record Format is leveraged to encode the data.

5.1 Data Types

Where possible, hData Content Profiles prefer the use of XML data types to custom date types. This means that the native XML dateTime is used instead of HL7 Timestamps. Conversion between these types is handled in XSLT.

5.2 hData Record Structure

The hData Record Format allows for a nested folder structure. The TB Progress Note Content Profile expects the record to be structured in the following way:

- /patient_information – This folder will be present in the root of the record. It will contain XML conforming to the Patient Information schema.
- /result_organizers – This folder will be present in the root of the record. It will contain XML conforming to the Result Organizer schema
- /result_organizer/ORGANIZER_ID – Within the result_organizers folder, there will be nested folders that contain the results associated with the organizer. The name of the folder for the results will be id for the result organizer.

When creating a progress note in hData that you wish to convert into a CDA, it must conform to this structure. CDA based progress notes converted to hData will be stored in this same folder structure.

6 Examples and Resources

- Example hData Record meeting the TB Progress Note Content Profile: <http://github.com/projecthdata/hl7-tsc-project/tree/master/hdata-examples/>
- XProc Pipelines for CDA/hData Transformation: <http://github.com/projecthdata/hl7-tsc-project/tree/master/pipeline/>
- L32 Information: <http://www.projecthdata.org/l32.html>
- hData Specifications: <http://www.projecthdata.org/documents.html>
- Calabash: <http://xmlcalabash.com/>