HMIS XML
Version 3.0
Cumulative Package Overview

Version 3.0: April 2010
Version 2.8: April 2008
Version 2.7: May 2006
First Public Version (2.6): January 2005

Revisions for Version 3.0
Prepared for Abt Associates Inc. by
Eric Jahn, Alexandria Consulting LLC
in consultation with:
Matthew Simmonds, Simtech Solutions Inc. and
Brian Sokol, Abt Associates Inc.

Original Title:
"HMIS Data Integration Cumulative Package Overview"
Original Authors:
Oscar Gutierrez, Ph.D., Change and Strategy Solutions
Brian Sokol, Abt Associates Inc.

U.S. Department of Housing and Urban Development
Office of Community Planning and Development

HMIS XML CUMULATIVE PACKAGE OVERVIEW

I. Preface

This document is intended for HMIS project managers and implementing technicians seeking guidance on the purpose and intended use of the HUD XML Schema Definition (XSD) for HMIS data, version 3.0. This is not a policy document; much of its language is technical and assumes a basic understanding of the W3C XSD specification[1]. For HMIS implementers seeking an alternative format for HMIS client data transmission, see the HUD Comma-Separated Values (CSV) for HMIS data, version 3.0. A separate document entitled "Housing Inventory Chart and Point-In-Time Count Reporting with *HMIS XML v.3.0*" covers reporting using this format.[2]

II. Introduction

Continua of Care across the nation are struggling to gather more comprehensive data on homeless persons within and across Continuum boundaries. Within a Continuum of Care, homeless services data are often maintained outside the HMIS in separate databases maintained by individual partner agencies. Also, multiple HMIS software solutions may be used within a single reporting area or Continuum. Both scenarios require data integration to overcome the administrative and technical implementation barriers to comprehensive Continuum-wide homeless client services reporting.

The goal of the HMIS XML Schema is to provide a single, validated software platform neutral, client data format as the basic building block for standardized HMIS data integration between diverse systems.

A. Contents of Package

The current package consists of multiple parts, all available at :

http://www.hmis.info/schema/3_0/HUD_HMIS.xsd

- This document, which includes:

  - A rationale for the schema, including an overview of the process and a description and explanation of the model.

  - A description of the steps involved beyond creation of a data standard, including development of communication protocols and documentation of responsibilities.

  - A brief discussion of the future path of HMIS XSD development.

- A set of two XML Schema Definition (XSD) documents. The main document is the HMIS XSD v. 3.0. The other is the referenced AIRS XSD v. 3.0, modified with the addition of a target namespace and the removal of <xs:any/> tags, so the schema may be imported. The tService type is also added.

- A sample, valid XML document with fictitious data.

- An example extension schema of the HMIS 3.0 XSD, adding an additional element.

---

[1]   see http://en.wikipedia.org/wiki/XML_Schema_%28W3C%29

[2]   see http://www.hmis.info/schema/3_0/HUD_HIC_PIT_XML_Overview.pdf

- A sample, valid XML instance document for the extended schema.

- Online documentation for version 3.0, with graphical representation, is available at: http://www.hmis.info/schema/3_0/docs/HUD_HMIS_3_0.xsd.html

Inline documentation in the HMIS XSD correlates each schema element to an item in the *HMIS Revised Draft Notice*. Searching the HMIS Schema for the corresponding *HMIS Revised Draft Notice* element numeral provides a cross-walk between technical schema and Federal Register notice.

III. Section 2: Data Integration Rationale
A. Process
The XML format is better suited than flat files, e.g. Comma-separated Values[3] (CSV) files, for conveying hierarchical data. The schema can enforce a specific structure to XML data, while there is no accepted standard for enforcing a similar structure for flat files. Thus, the primary product of this effort is an XML Schema Definition (XSD) document.

Three concerns guided the original XSD development:
1) Modeling the data precisely as expressed by the *HMIS Revised Notice*. The scope was largely limited to those data required by the *HMIS Revised Notice*, leaving out other data that might be collected on a community level.
2) Accounting for the mechanics of data integration itself within the model. For example, data elements were required to track the originating database of particular records and the date when the data were collected, in order to properly synchronize the data.
3) Creating a clear and simple schema that could be used and understood by all HMIS developers and local database engineers and consultants who may be employed to convert data to the HMIS XSD standard.

Viewing the *HMIS Revised Notice* as the underlying requirements document, the following needs added layers of complexity:

- Certain elements are collected generally once for a person, other elements are collected once per program (SiteService) enrollment (*HMIS Revised Draft Notice*, Sec. 2). A third set of elements are historical data collected multiple times within a single program (SiteService) enrollment (see *HMIS Revised Draft Notice*, Exhibits 1-2 and 1-3), for example, during SiteService entry, and at exit.

- In several instances, the *HMIS Revised Draft Notice* supplies a list of response codes and mandates that more than one answer must be allowed.

  - Examples of this are race, educational degrees, income, and school barriers. These elements are relationally modeled as child tables rather than as yes/no fields for each element, which would not have allowed the inclusion of the

---

[3]   see http://en.wikipedia.org/wiki/Comma-separated_values

standard codes HUD requires.[4] In the HMIS XSD, they are unbounded child elements, either simple or complex as appropriate.

- Since HUD has mandated particular numeric codes for all standard values, the model must enable the transmission of numeric codes, rather than the descriptive values.

Version 3.0 of the HMIS XSD incorporates the Revised HMIS Data Standards issued by HUD in June 2009 as well as the draft data standards issued in July 2009 [5] and subsequent changes prior to the OMB approval process, completed in March 2010. These include a numerous changes to the data elements. Most significantly, the 2009 Standards include data elements required for "Homelessness Prevention and Rapid Rehousing Program" (HPRP) reporting as well as required "Program Descriptors," which convey extensive information about homeless programs themselves.

This version also attempts to incorporate suggestions raised by those who have implemented previous versions of the schema. It also adds or changes data elements and response values mirroring changes to the *HMIS Revised Draft Notice*. A listing of the specific changes made in Version 3.0 are described in Appendix 1.

Note: This document does not attempt to describe the meaning of data elements that are already included in the *HMIS Revised Draft Notice* or to justify, refine, or clarify any of the data definitions given in the Notice. Instead, this document describes the decisions made in modeling the data and highlights data elements that were not in the *HMIS Revised Draft Notice* but were included in the Schema for technical reasons or out of anticipation of need.

B. Overview of Schema

For readers who are already familiar with XML, the following section describes some aspects of the HMIS XSD. Otherwise, there are many good introductions[6] to XML and XML Schema on the web, which should be read before continuing this document.

---

[4] In relational modeling, a "child table" refers to a table that can have multiple records compared to a "parent" table, which would have only one record. Thus, if one person can have multiple races, then the person table is the parent table, and the table of the person's races is the child. In XML, when one element appears inside another element, the container element is the parent.

[5] In June 2009, in order to accommodate the Congressionally mandated deadlines of American Recovery and Reinvestment Act (ARRA), HUD received "emergency clearance" from the Office of Management and Budget (OMB) for a revision to the data standards after an abbreviated Notice and Comment period. This clearance is only valid for six months. In July of 2009, HUD released a revised draft version of the Standards for a full 60-day Notice and Comment period. The final data elements released subsequent to this period will be valid for three years or until superseded by further Notice or Regulation. The only distinction between the June "final emergency" Standards and the July draft standards is that Section 4.15 "Client Outcome Measures" appears in the July version, but was not in the June version. This insertion caused a renumbering of June Section 4.15 "Optional Data Elements" to become Section 416 in the July version.

[6]   http://en.wikipedia.org/wiki/XML_Schema and http://www.w3.org/XML/Schema are two useful sites to begin learning XML Schema.
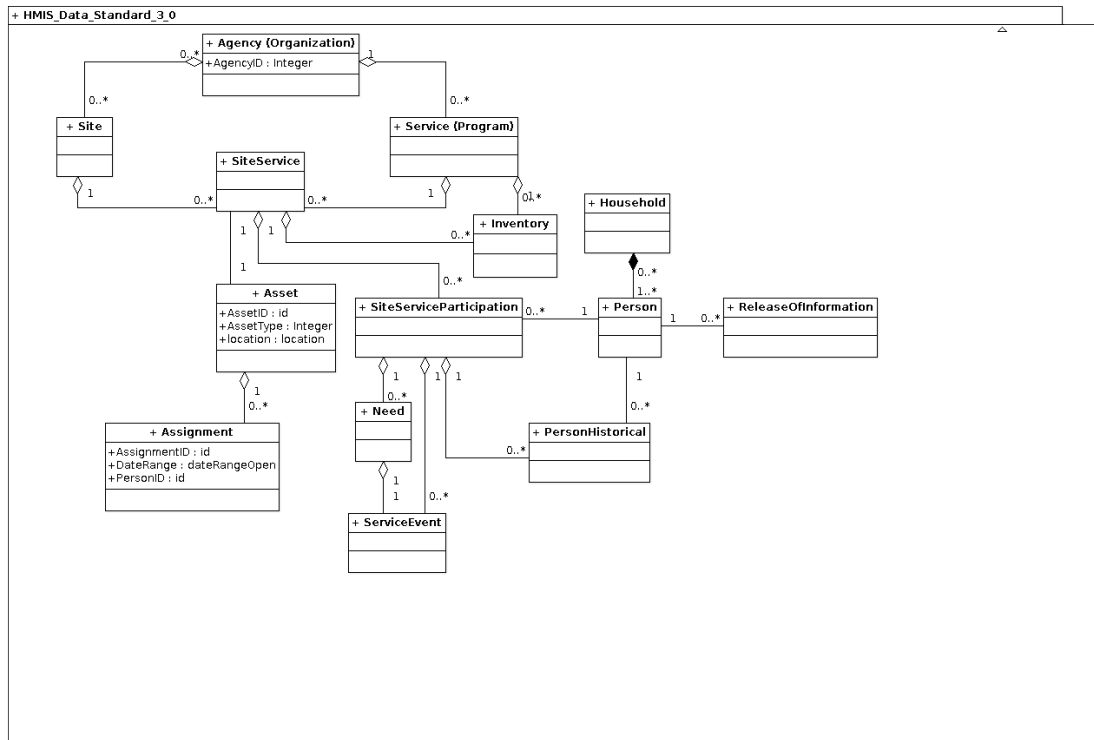
UML Class Diagram for HMIS XSD, version 3.0

Very few data elements are mandatory. The mandatory elements are limited to those data that are logically necessary in order to produce meaningful information. The mandatory elements should not be confused with the Universal data elements mandated by HUD. Programs are mandated to collect Universal data elements, but the entire data set is not invalid if one person's record is missing one data element.

All elements are based on top level types. These top level types are often reused multiple times within this schema, and can be imported into a new schema for extension. Some types are imported from the AIRS Schema[7], so the namespaces "hmis" and "airs" and "xsd" are used to keep elements' origins clearly denoted.

A simplified view of the hierarchy of major complex elements in an XML instance document is shown in Figure 1, followed by a description and rationale for these elements. In the description, each Roman numeral (I, II, III) section represents a cluster of elements grouped for logical reasons, and Arabic numbered (1, 2, 3) sections represent distinct complex element sequences.

---

[7]   AIRS is the Alliance of Information and Referral Services, "AIRS is the professional association for over 1,200 community Information and Referral (I&R) providers, primarily in the United States and Canada" http://airs.org/

IV. XML Instance Element Documentation.

A. Integration Structural Elements

These elements collect information specific to the data integration environment, enabling synchronization and the tracking of data to its source, which may be needed for resolving errors or other data validation or quality issues.

1. Sources

Parent Element: None, it is the "root" element

Attribute: version

The 'Sources' element holds a sequence of Source subelements. Each Source subelement represents a unique data source for XML data contained within. The version attribute simply tracks the version number of the HMIS Schema used by the XML instance document (*i.e.* if this particular schema is used, the version would be 3.0). Sources contains one or more Source elements. Within each Source, the SourceID should be initially assigned by the target database, *i.e.*, the database integrating the data, and it should be unique across the implementation. SourceName is simply a string intended to give the SourceID a familiar name. SoftwareVendor describes the provider of the XML generating software (including community maintained systems with no vendor) and SoftwareVersion details the version number of the software system being used. The remaining elements before Export hold contact information for the individual directly responsible for the sending database, *i.e.* a database administrator or the IT department manager.

2. Export

Parent Element: Source

Over time, each data source will produce multiple exports. This element records information about the beginning and end of the period for which data were extracted and the actual date of the export. In an XML instance file, a Source element can have many exports. All the following payload elements are part of an export.

B. Payload Elements

1. Agency

Parent Element: Export

This element describes an organization participating in a Continuum of Care and is just a reference to the Alliance of Information and Referral Systems (AIRS) XSD version 3.0 tAgency[8] type. It indexes the agencies using the tAgency.Key element. This data could also be transmitted separately using just AIRS XML version 3.0[9], and was included in the

---

[8]   See the AIRS 3.0 style guide at: http://www.airs.org/StyleGuide

[9]   AIRS Schema, version 3.0 available at http://cse.unl.edu/~bkutsch/3_0_Final.xsd

HMIS Schema as a convenience, so only one XML file need be transmitted in certain data integration scenarios.

2. Household

Parent Element: Export

This element allows household groupings of persons to be conveyed. Household can contain multiple Member elements, each describing their ID and relationship to the head of household. In the household element, one or more of the members can have their ID designated as HeadOfHouseholdID. The *HMIS Revised Draft Notice* (Section 3.15) states that "a household is a group of persons who together apply for homeless assistance services." Thus, the household is not static for a particular person, but is based on a site service enrollment. A person can be a member of multiple households. Multiple persons in the same household should have the same household identification number. Household identifiers should be generated by the source database. A combination of the SiteService identifier and the locally generated HouseholdID will be unique across the implementation.

3. Person

Parent Element: Export

This element includes identifying information about an individual and any other personal data that will very rarely change over time or between collecting agencies. As such, only minimal data is associated directly with the Person entity.

All of the identifying elements directly under Person may occur only once, except Race and OtherNames. The Race element can occur multiple times allowing for multiple races for each person. OtherNames is a complex type containing the various parts of the person's other names. Multiple sets of OtherNames can also occur.

The PersonID is populated from the data source and uniquely identifies a person in the system. The SourceID concatenated with the PersonID should avoid PersonID collisions when merging multiple databases. The merged database should implement algorithms to unduplicate across databases using the personal information within the Person element.

To make the schema more flexible for purposes of implementing local policies for de-identification and unduplication, the acceptable formats for the Social Security Number and the child elements of Person allow for a choice of either plain text or hashed values.

```xml
<xsd:element name="LegalFirstName" type="hmis:hashingChoice" minOccurs="0"></xsd:element>
...
<xsd:complexType name="hashingChoice">
        <xsd:choice>
```

```
                            <xsd:element name="Unhashed" type="hmis:string50"/>
                            <xsd:element name="Hashed" type="hmis:string"/>
                </xsd:choice>
</xsd:complexType>
```

In the XML instance documents, the various elements contributing to the PersonID hash result may or may not actually have data in them. The reason for this is to provide implementers flexibility in how they want to implement (or not implement) hashing security and unduplication algorithms. One has the choice of:

- Not transmitting personal identifying information for client confidentiality and simply transmitting the hashed PersonID for unduplication

Example:

```
<hmis:PersonID>
        <hmis:Hashed hmis:dateCollected="2006-05-04T18:13:51.0Z" hmis:dateEffective="2006-05-04T18:13:51.0Z" hmis:dataCollectionStage="2">g8u7Kyg6</hmis:Hashed>
</hmis:PersonID>
...
<hmis:LegalFirstName hmis:dateCollected="2009-08-01T00:00:00"/>
```
Note: omitting the empty LegalFirstName element entirely is a more efficient transmission format for the above example.

- Transmitting hashed personal identifying information and also transmitting the hashed PersonIDStr for unduplication

Example:

```
<hmis:PersonID>
        <hmis:Hashed hmis:dateCollected="2006-08-01T00:00:00" hmis:dateEffective="2006-05-04T18:13:51.0Z" hmis:dataCollectionStage="2">DCF017Y0</hmis:Hashed>
</hmis:PersonID>
...
<hmis:LegalFirstName>
        <hmis:Hashed hmis:dateCollected="2006-08-01T00:00:00">A2TD87YU</hmis:Hashed>
</hmis:LegalFirstName>
```

- Transmitting clear text personal information and also transmitting the hashed PersonID for unduplication

Example:

```
<hmis:PersonID>
        <hmis:Hashed hmis:dateCollected="2004-08-01T00:00:00" hmis:dateEffective="2006-05-04T18:13:51.0Z" hmis:dataCollectionStage="2">DCF017Y0</hmis:Hashed>
</hmis:PersonID>
...
<hmis:LegalFirstName>
        <hmis:Unhashed hmis:dateCollected="2004-0801T00:00:00">Washington-</hmis:Unhashed>
```

```
</hmis:LegalFirstName>
```

One could also present clear text PersonIDs as integers or strings with the combinations above. Regardless of the method, the XML instance documents clearly indicate which is occurring.

a. OtherNames

Parent Element: Person

This element is not required in the *HMIS Revised Draft Notice*, however, it is useful for shared case management. This element is not required for every person, and can occur multiple times to hold many additional names.

b. PersonHistorical

Parent Element: Person or SiteServiceParticipation (depending on whether wrapped within an entry/exit or not)

The bulk of the "Program-specific" data elements (versus "Universal", per *HMIS Revised Standards*) collected on each person are captured within the PersonHistorical element and its subelements. Each person can have zero or many PersonHistorical records. PersonHistorical can be transmitted independently of SiteServiceParticipation, by placing it directly under a Person element.

The PersonHistorical element includes many simple subelements that are collected for all persons, as well as many that are collected only for adults or only for children.

All subelements of PersonHistorical are unbounded elements, allowing for multiple or zero responses. PersonHistoricalID establishes a unique index for the transmitted set of PersonHistorical data. SiteServiceID attributes the PersonHistorical data set to a particular SiteService, and corresponds to airs:tSiteService.Key. Many elements have their own indexes, since they contain clusters of subelements.

HUDChronicHomeless and HUDHomelessEpisodes are not specifically listed as data elements in the *HMIS Revised Draft Notice*, however, they are frequently tracked in HMIS systems and reported upon by CoCs.

PersonAddress (except ZIP Code), PersonEmail, and PersonPhoneNumber are not located in the *HMIS Revised Draft Notice*, but are commonly tracked in HMIS systems and useful for shared case management purposes.

c. ReleaseOfInformation

Parent Element: Person

The Release of Information complex element contains a record of a person's consent to share their personal information for a particular SiteService during a defined time frame. ReleaseOfInformation includes subelements ReleaseOfInformationID to index the releases, SiteServiceID to relate the release to a specific site service, and EffectivePeriod to limit the release to a specific time frame.

d. ServiceEvent

Parent Element: Person or SiteServiceParticipation (depending on whether wrapped within an entry/exit or not)

The ServiceEvent element describes particular services actually rendered for a particular person or household, via the subelement HouseholdID. This entity stores data on the date and type of service, as described by the *HMIS Revised Draft Notice.* The ServiceEvent can be classified as one of:

- HMISServiceType: An additional field is grouped with the element to allow for free text description of the service if the code for "other" is used.
- FinancialAssistance: Required for HPRP services only. Subindexed with numerous subelements
- RelocationStabilizationServiceType: Required for HPRP services only.

To increase the flexibility and usefulness of ServiceEvent data certain elements are included that are not described in the *HMIS Revised Draft Notice.* These include:

- <ServicePeriod>...<EndDate>, which captures the end date of the service.

- ServiceUnit, which indicates whether the ServiceEvent was delivered to an individual or the entire household. In the latter case, the ServiceEvent record need only be included under the head of household. A family apartment unit is an example of a service delivered to an entire household.

- QuantityOfService, which can be used to indicate, for example, dollar amounts if the service is rental disbursements.

- QuantityOfServiceMeasure describes the unit of measure used for the QuantityofService element.

- AIRSCode, this string can be used to track the standard taxonomic code from the Alliance or Information and Referral Services (AIRS), *e.g.*, "BH-180" if the service is Emergency Shelter.

- IsReferral allows a ServiceEvent to be flagged as a referral as opposed to a directly provided service.

e. SiteServiceParticipation

Parent Element: Person

SiteServiceParticipation relates a person to a SiteService and tracks data that are specific to a person's enrollment in a SiteService. The SiteService is associated with the SiteServiceID subelement.

A Person record may have multiple SiteService enrollments for different SiteServices, or for the same SiteService with different dates. SiteServiceParticipation should not be confused with the ServiceEvent element. The former references overall enrollment in a SiteService (*i.e.* program site), the latter references particular actions/events performed on behalf of a client. Thus, if a SiteService entails multiple counseling sessions, say one per week over ten weeks, there would be one SiteServiceParticipation record and ten ServiceEvent records.

HouseholdID is included as an element within SiteServiceParticipation to flag an entire household, of which the enclosing Person record is a member, as having participated as well in the same SiteService. The *HMIS Revised Draft Notice* (Section 3.15) states that " A household is a single individual or a group of persons who together apply to a CoC program for services. ". Thus, the household membership is not static for a particular person, but is based on a SiteServiceParticipation. Multiple persons in the same household should have the same household identification number. Household identifiers should be generated by the source database. A combination of the SiteService identifier and the locally generated HouseholdID will be unique across the implementation.

A SiteServiceParticipation can optionally encapsulate Need, PersonHistorical, and ServiceEvent (by itself without an associated Need) subelements.

i. Need
Parent Element: SiteServiceParticipation

Needs are not part of the *HMIS Revised Draft Notice*, however fulfillment of identified client needs are central to most HMIS software architectures, as well as CoC board reporting. Need elements track AIRS services/service categories a person requires, as opposed to any actual services received. A Need element contains many subelements, including NeedID as an index, ServiceEventID to tag specific service events as pursuant to need fulfillment, Taxonomy to hold the AIRS Taxonomy Code of the need, NeedStatus to indicate fulfillment state of the need, and SiteServiceID to credit a specific site service with registering the need.

4. Service
Parent Element: Export

Service is equivalent to a "Program" in the *HMIS Revised Draft Notice*. Its terminology originates from the AIRS Agency/Site/Service model, which separates legal entity from locus and program, respectively. Service represents a multi-site delivery of coordinated assistance to clientèle, perhaps funded by multiple funding sources. Subelements are

COCCode, ConfigurationType, DirectServiceCode, GranteeIdentifier, IndividualFamilyCode, Inventory, ResidentialTrackingMethod, ServiceType, TargetPopulationA, and TargetPopulationB. The HMIS XSD inline documentation cross-references each subelement to its associated *HMIS Revised Draft Notice* location. IndividualFamilyCode is an exception. It tracks whether the SiteService serves individuals, families, or both. This information is useful for data analysis and reporting. For example, the Annual Homeless Assessment Report (AHAR) asks for data on families and individuals separately.

5. Site

Parent Element: Export

Site is simply the AIRS XSD version 3.0 tSite type extended with a deleteStampGroup attribute group for data integration purposes (see deleteStampGroup in Section IV. C.). It is included with the HMIS Schema so that AIRS agency/site/service information can be transmitted in a single HMIS XML file along with HMIS client information.

6. SiteService

Parent Element: Export

This element captures information about services occurring at a particular physical site or location. The SiteService is defined within the Alliance of Information and Referral Services' (AIRS) XML Schema,[10] and is analogous to a specific "program site" as described in the *HMIS Revised Draft Notice*. Also, AIRS and HMIS XML data are used in common scenarios, and this naming harmonization reduces confusion. The HUD HMIS site service simply extends the AIRS Schema v. 3.0's version of site service so that there is congruence between HUD HMIS and AIRS XML data elements. Included within the HMIS XSD's SiteService are subelements SiteID, GeographicCode, HMISAsset, HousingType, Inventory, Principal, SiteType. All these are referenced directly by the *HMIS Revised Draft Notice* and cross-referenced by the inline annotations in the HMIS XSD, except SiteID and HMISAsset. SiteID links the SiteService to a Site.Key, specifying the location of the SiteService. HMISAsset tracks individual beds or housing units, as opposed to tracking them in aggregate using SiteService.Inventory or Service.Inventory (when aggregate Inventory will not be tracked at the location level, and rather at the "Program" level as optionally allowed by the *HMIS Revised Draft Notice*).

The SiteService element also includes a SiteID element, which can be used to designate the physical location at which a SiteService exists. In AIRS terminology, an agency may possess many Sites which in turn may possess many Services, and the SiteService represents the intersection of the Site and Service. Use the AIRS Schema in isolation to represent complex multi-layered parent agencies, sites and site service relationships as AIRS XML.

---

[10]   See http://tinyurl.com/ywsypz for the AIRS Wiki

Site service data will not necessarily need to be sent in every periodic upload. Rather, they can be sent during an initial transfer and only updated if changes occur.

C. Attribute Groups

Two attribute groups are used throughout the schema: dateStampGroup and deleteStampGroup.   They allow for synchronization between sending and receiving systems.

dateStampGroup encapsulates dateCollected (when data collected), dateEffective (when data is effective, so back-dating possible), and dataCollectionStage (1 = Entry, 2 = During Program Enrollment, 3 = Exit, 4 = Followup).

deleteStampGroup encapsulates delete, deleteOccurred, and deleteEffective and exists to provide instructions for removing stored data by referring to its index and requesting a deletion. delete ="1" overrides the default add/update "0" for an index, which need not be specified.   deleteOccurred conveys when the deletion was executed, and deleteEffective allows backdating of the deletion effective date.

D. Types

Since the release of HMIS XSD version 2.8, top level types, both simple and complex, exist for every element declared in the schema. This allows any type, including the root element, SourceDatabase, to be imported and referenced or extended within another schema. This enables flexible, validated customization of the HUD HMIS 3.0 Schema, unlike the "Custom" tags present in version 2.7 of the Schema. An example schema demonstrating the syntax for extending the HUD HMIS 3.0 is included in the data integration package. An example XML instance document which validates against this extended schema is also included in the package.

E. Lookup Values/Enumerations

These elements declare the acceptable values for the data elements and can be based on patterns or enumerations.

In cases where the *HMIS Revised Draft Notice* defines a specific list of acceptable codes with their description, the schema uses the codes as the acceptable values. The interpretations of the values are included in the XSD file within documentation tags. This strategy has disadvantages in that it reduces the human readability of a given XML file. That is, an individual file would require the XSD or additional information to decode the XML element values. However, it is clearly in line with HUD's intention in the *HMIS Revised Draft Notice* and it also relieves programmers of the burden of having to convert codes to values and back again to codes.

Multiple data elements might reference the same lookup enumeration if they use the same code breakdown. For example, multiple fields accept the values 1, 2, 8 or 9 even if the meanings of these values differ. (The use of the non-sequential 8 and 9 values to indicate "Don't Know " and "Refused" is mandated by the *HMIS Revised Draft Notice* and has

the advantage of consistency across elements, even though it is somewhat non-intuitive within each element.)

Where enumerations are linked to multiple descriptions, they are named after the number of meaningful values. For example, "twoVal" refers to an enumeration with only two acceptable values.

V. Beyond the Current Package

This data integration package is only the first step in achieving full data integration. The current specification assumes that participating stakeholders agree to use the standard XML description and will have software convert their stored data to the standard XML format. Many more steps are needed in order to actually complete the integration process. Technical steps include implementing a process for validating the data, transferring the data via a standard communication protocol (commonly SOAP or REST messaging), creating a database to act as a central repository, and devising a synchronization method. In addition to the technical steps, a number of decisions need to be made and responsibilities divided between the central agency and the participating parties.

A. Validation

Before sending the XML, the contributing data source should validate the files. The following list presents the most commonly used XML parsers to date. Most of the tools listed here can test XML documents for being not only well formed (i.e. conform to the basic syntax of XML) but also valid (they conform to a particular schema or DTD).

Validating Parser	URL
CFX XML Parser (Coldfusion)	http://www.zrinity.com/xml/xmlparser
Crimson	http://xml.apache.org/crimson
JAXP	http://java.sun.com/webservices/jaxp
libxml	http://xmlsoft.org/
lxml	http://codespeak.net/lxml
MSXML	http://msdn2.microsoft.com/en-us/xml
Oracle	http://www.oracle.com/technology/tech/xml
Xerces	http://xerces.apache.org/xerces2-j
XSV (W3.org official reference implementation for validating the XML Schema)	http://www.ltg.ed.ac.uk/~ht/xsv-status.html convenient web form: http://www.w3.org/2001/03/webdata/xsv

In addition to validating the XML against the HMIS XML Schema, the data should also be scrubbed to ensure that the content of the data is reasonable. Additional checks might be appropriate to ensure, for example, that birth dates are earlier than the current date, or

that only women are marked as pregnant. This is beyond the level of validation handled by the parsers and schema.

B. Processes for Data Transfer

There are varying levels of technological sophistication that can be implemented for transferring the data.

1. Simple: Unidirectional Batched Data Uploads

The least technically sophisticated integration approach would involve minimal automation and rely heavily on people doing the work. Consequently, data would be sent infrequently, such as quarterly or yearly. The process would be one-way. Participating agencies will not receive data back from the central repository.

Database developers of each aggregate database will develop a process for exporting the data to the XML file and provide an interface for users of the system to easily export the data for a certain date range. Many database tools have functionality to export data as XML. In addition, various tools can assist developers with this conversion process. The exported data file is uploaded via secure FTP or SSH to an Internet site managed by the central repository. In very-low level implementations the file can even be emailed, as long as the email or the attached file is encrypted during delivery transport, with at least 128-bit encryption.

2. Complex: Bidirectional Asynchronous[11] Messaging

A fully integrated environment will be one that satisfies the three following characteristics. First, it allows for data to be transmitted to and from source and destination systems. Second, it is an environment where multiple human services domains such as mental health, substance abuse, and health care - in addition to homelessness - contribute data to an aggregate database. Third, it is a technically flexible environment where heterogeneous data management applications that operate under diverse messaging protocols can asynchronously push and pull data sets according to standard security and authorization mechanisms.

For this environment to be put in effect, developers at the local level must assess the most appropriate tools to implement asynchronous data transmission; develop both push and pull data extraction utilities; and integrate the array of necessary data messaging services that are more applicable to the locality.

In this environment, data transmission is bidirectional; in other words, a participating system contributes with data to an aggregate database, but also has the ability to extract up-to-date data sets from the aggregate database for analysis or consolidation purposes. Bi-directional does not necessarily mean that communication between participating and

---

[11]   Asynchronous means the sending server need not block/wait for the receiving server to respond before continuing on with the next task.

integrating systems is synchronous (i.e. the ability to handle two-way communication on a real time basis). However, the ultimate data integration environment is a flexible, asynchronous, bidirectional data transmission system.

Messages should implement the basic ACID[12] record functions of "add, change, inquire, and delete", and only necessary portions of valid XML, such as a single ServiceEvent, may be present in a single message.

Commonly used messaging protocols include SOAP and REST. These protocols are often referred to as web services, as they overlay the Internet as the underlaying physical transport. SOAP and REST attempt to accomplish the same means and they are competing messaging methods. Freely available SOAP/REST HMIS APIs and code are still needed by the HMIS data integration community.

C. Converting from XML to Databases

The administrator of the central repository creates the data warehouse to store the data sent by multiple data sources. In general, there are two approaches to storing XML in databases: data-centric and document-centric. In the document-centric model, the database stores each XML file as a complete document or in a manner that is easily suited to retrieving the data again as a separate XML file. The data-centric model stores the data within the XML document rather than the XML document itself. The data-centric model is often most appropriate for HMIS purposes since the primary data source and destination is typically a relational database.

The process of transferring data to a relational database is called "shredding" XML into tables. Most major relational database systems have built-in functionality to handle this process, especially if the database schema maps closely to the XML schema. A closely mapped database would consist of separate tables for every complex element as well as every element that may occur multiple times. The columns in each table will consist of all of the simple elements contained with the complex elements. Usually though, software must be written to handle the complex logic of shredding XML files into a relational database, since there is typically significant dissimilarity between the HMIS format and the destination relational model. In addition, a wide array of middleware products is available to assist with the process of mapping between XML schemas and databases. A good overview of the relationship between XML and databases as well as lists of databases and middleware that support conversion between XML and databases can be found at http://www.rpbourret.com/xml/XMLAndDatabases.htm .

The process of importing data may also incorporate the matching of client records immediately. In that case the import script would include the process of searching the current database to determine whether the person is already present. If so, the person record would be updated rather than inserted, and the database would be unduplicated at all times. Alternatively, it is possible to simply insert all records into the aggregate

---

12   http://en.wikipedia.org/wiki/CRUD_(acronym)

database when they are initially received, and at a later point, match the records. This decision will impact how the data warehouse is designed.

D. Documentation of Responsibilities and Decisions

Project documentation should be created to help guide administrators of participating data sources. The following is an outline of the responsibilities that need to be clarified and decisions that to be made:

1. Converting the data

- Whose job will it be to convert the data in source databases to the data standard?

- How much support will be provided by the staff of the central database?

2. Ensuring cross-participant consistency

- How will questions and answers about the meaning of the fields be managed?

- Will the implementation make efforts to ensure that all participating source databases understand the data standard in the same way?

- What messaging protocol will be used (SOAP, REST, other), or will the integration be file-based?

- Hashing the data. The standard includes the option for certain values to be hashed within the file. Which site services will be allowed to send hashed values?

3. Cleaning and validating the data

- Who is responsible for cleaning the data and removing invalid or incorrect information?

- How clean must the data be before being sent to the central agency?

- Should the data be converted to a certain format (e.g., upper case) before being sent?

- Beyond the technical requirements of the standard, what, if any validation rules should be applied (e.g., acceptable dates for date of birth, incomes within reasonable ranges, pregnancy only for women)? Should this validation occur prior to sending the data?

4. Unduplicating the original data.

- Should the data sent to the central database be unduplicated before being sent, so that the central database can treat all distinct person records sent from a particular agency as unique. Or should the data source extract all of the records and leave unduplication to the central repository.

5. Frequency

- How often should the data be sent?

6. Filtering the appropriate data for a period

- What records should be sent?

- If batch data are sent every month, should data be sent on every person that was served that month? Or only those who entered, exited, or had their records updated that month?

- Should a complete data dump be sent every time, leaving the central repository to filter based on dates in the system?

7. Sending the data

- What are the protocols for sending the data?

- What methods are available

8. Handling data conflicts and synchronization

- How should data conflicts be resolved (for data that are not longitudinal)?

9. Securing the database

- What information security protocols will be in place at the central database?

- Under what circumstances and in what manner will the data be released?

These decisions can be made by the individuals overseeing the integration effort, though some might require community input. Answers to the questions can be written in a Standards Manual or Frequently Asked Question (FAQ) format. The outline above can be used as the starting point for this documentation. An overview of the data integration process responsibilities can be found at:

- http://hmis.info/ClassicAsp/documents/8 Simple Rules - Data Integration Institute.pdf

- http://hmis.info/ClassicAsp/resources.asp?resource_id=860

E. Recommendations for Continued Standardization
The current project was limited to development of an extensible schema for file-based data integration. Additional standardization can help to move integration projects further down the road to success. In addition, the development of both proprietary and non-proprietary software and the sharing of code to support this schema on various platforms are highly encouraged. A section of the HMIS Technical Assistance portal has been made available for these purposes.

VI. New Features and Modifications in HMIS XSD 3.0

The HMIS XSD 3.0 introduces a number of significant changes and several minor adjustments. These changes are described below.

A. Significant Changes

1. Addition of Service element

In the HMIS Schema, hmis:service closely models 'Program' in the *HMIS Revised Draft Notice*. It extends the tService type added into modified AIRS XML 3.0. tService existed in the AIRS XSD 2.07 and was removed in 3.0.

2. Addition of Inventory element to hmis:service and hmis:siteService

Inventory enables aggregate bed/unit reporting at either the program or program site level, depending on each Continuum's tracking preference. If, alternatively, inventory is to tracked by the individual bed/unit (and then aggregated for reporting) use the SiteService.Asset.Bed and SiteService.Asset.Unit elements.

3. Addition of an Asset to hmis:siteService

Assets can individual beds or housing units, and are assignable to individuals. This is an alternate, more detailed way of transmitting bed or unit inventory than using the aggregate Inventory element. Assets could also be extended by a Continuum to describe any other type of asset tracked by a Continuum.

4. IncomeAndSources changed to reflect changes in *HMIS Revised Draft Notice,* Section 4.1

Income and sources has a new structure, with choices and an index.

5. Grouped element sets whose groupings need to be preserved are indexed.

Most of these indexes exist within hmis:personHistorical.

6. SourceDatabase renamed to Sources

Within Sources, there can be many hmis:source elements. Within each source, there can be many hmis:export elements, which hold the payload elements (*e.g.* hmis:person, hmis:agency, etc.). Some elements formerly within hmis:export were moved to hmis:source, such as SoftwareVendor.

B. Minor Adjustments

1. Addition of HousingStatus element within hmis:personHistorical

Added per *HMIS Revised Draft Notice* Section 3.11.

2. Addition of ServiceEvent directly within hmis:person

To account for scenarios where services are rendered without a SiteServiceParticipation.

3. Addition of SiteServiceID within hmis:serviceEvent

Since ServiceEvent isn't necessarily within a SiteServiceParticipation, SiteServiceID now directly gives the ServiceEvent program context without requiring nesting.

4. Removal of hmis:siteService.SiteServiceID

SiteService.SiteServiceID was redundant with SiteService.Key, causing ambiguity in the HMIS XSD.

5. hmis:warZoneServed.ReceivedFire changed into an hmis:twoval

ReceivedFire was formerly an hmis:unsignedInt.

6. OrganizationIdentifier is mapped to airs:tAgency.Key

Both have the same functionality of providing an Agency index. For that reason, OrganizationIdentifier referenced in *HMIS Revised Standards*, Section 2.2 is not present in the HMIS XSD.

7. OrganizationName is mapped to airs:tAgency.Name

8. ProgramIdentifier is mapped to airs:tService.Key

9. ProgramName is mapped to airs:tService.Name

10. DirectServiceCode is added to hmis:service

Added per *HMIS Revised Draft Notice,* Section 2.5.

11. ConfigurationType added to hmis:siteService

Added per *HMIS Revised Draft Notice,* Section 2.6A.

12. Added Principal (to designate the principle program service site) to hmis:SiteService

Added per *HMIS Revised Draft Notice,* Section 2.6B.

13. Added GeographicCode and SiteType to hmis:siteService

Added per *HMIS Revised Draft Notice,* Section 2.6C.

14. SiteType added to hmis:siteService

Added per *HMIS Revised Draft Notice,* Section 2.6D.

15. HousingType added to hmis:siteService

Added per *HMIS Revised Draft Notice,* Section 2.6E.

16. FIPSCode removed from hmis:siteService

FIPSCode was removed from the *HMIS Revised Draft Notice.*

17. COCCode moved from hmis:siteService to hmis:service

Continuum of Care Number moved to hmis:service, since service is a Continuum's program and applies to all its subordinate SiteService elements.

18. hmis:serviceType (Program Type Code) moved from hmis:siteService to hmis:service

Picklist values in hmis:serviceTypeBase (formerly hmis:siteServiceTypeBase) also changed to conform to *HMIS Revised Draft Notice,* Section 2.8.

19. Added hmis:version type for hmis:source.SoftwareVersion to replace use of hmis:string

20. Added hmis:sources.HMISXMLVersion

21. Added TargetPopulationA/B to hmis:service

Added per *HMIS Revised Draft Notice,* Section 2.10.

22. Added ResidentialTrackingMethod to hmis:service

Added per *HMIS Revised Draft Notice,* Section 2.12.

23. Added GranteeIdentifier to hmis:service

Added per *HMIS Revised Draft Notice,* Section 2.13.

24. Ethnicity now an hmis:fourValHashingChoice

Due to picklist changes in the *HMIS Revised Draft Notice, Section 3.5*.

25. hmis:person.Race changed from an hmis:fiveValHashingChoice to an hmis:sevenValHashingChoice

Due to picklist changes in the *HMIS Revised Draft Notice , Section 3.4*.

26. hmis:dOBHashingChoice.DateOfBirthType added

Added per *HMIS Revised Draft Notice,* Section 3.3.

27. hmis:sevenValHashingChoice added

28. Hmis:sevenValDKUnknown2HashingChoice added

29. sevenValDKUnknownHashingChoice added

30. hmis:residenceBase picklist values changed, made into an enumeration

Values changed per *HMIS Revised Draft Notice,* Section 3.3.

31. hmis:personHistorical.LengthOfStayAtPriorResidence now an hmis:sevenValDKRefused2

Due to picklist changes in the *HMIS Revised Draft Notice, Section 3.9*.

32. hmis:zIPQualityCodeBase picklist value #1 now reads "Full or Partial Zip Code Reported"

Due to picklist changes in the *HMIS Revised Draft Notice, Section 3.10*.

33. Moved hmis:twoValPlus and hmis:twoValYesNo to hmis:fourVal

34. Removed of hmis:twoValHashingChoice

35. hmis:sixValDKRefused added for hmis:personHistorical.HousingStatus

36. <u>hmis:incomeAndSources.IncomeAndSourceId added</u>

37. hmis:personHistorical.IncomeLast30Days added

Added per *HMIS Revised Draft Notice,* Section 4.1.

38. hmis:incomeAndSources.IncomeSourceAmount put into an xsd:choice along with a new yes/no option, hmis:incomeAndSources.ReceivingIncomeSource

Changed per *HMIS Revised Draft Notice,* Section 4.1.

39. hmis:incomeSourceCodeBase picklist values changed, and made into an enumeration

#8 and #9 removed, #10 changed to TANF, #11 changed to General Assistance, #12 changed to Retirement income from Social Security, #13 changed to Veteran's pension, #14 changed to Pension from a Former Job, #15 changed to Child Support, #16 changed to Alimony or other spousal support, #17 Other Source added.

40. hmis:personHistorical.TotalIncome Changed

hmis:personHistorical.TotalIncome changed to hmis:personHistorical.IncomeTotalMonthly, and no longer an hmis:decimal but an hmis:unsignedInt

41. hmis:incomeAndSources.Amount changed to hmis:incomeAndSources.IncomeSourceAmount hmis:unsignedInt from hmis:decimal

42. Added hmis:nonCashBenefits complex type

hmis:nonCashBenefits has an index, captures the non-cash source code, and a yes/no response, per *HMIS Revised Draft Notice,* Section 4.2.

43. Added complex type hmis:nonCashSourceCode

Formerly, hmis:nonCashSourceCode was an hmis:elevenval, many changes to picklist values per *HMIS Revised Draft Notice,* Section 4.2.

44. hmis:physicalDisability type replaces hmis:twoval for hmis:personHistorical.PhysicalDisability

45. Added hmis:physicalDisability.HasPhysicalDisability and hmis:physicalDisability.ReceivePhysicalDisabilityServicesTreatment

Changed per *HMIS Revised Draft Notice,* Section 4.2.

46. hmis:developmentalDisability type replaces hmis:twoval

47. Added hmis:developmentalDisability.HasDevelopmentalDisability and hmis:developmentalDisability.ReceiveDevelopmentalDisabilityServicesTreatment

Changed per *HMIS Revised Draft Notice,* Section 4.4.

48. Added ChronicHealthCondition

Per *HMIS Revised Draft Notice,* Section 4.5.

49. Added hmis:personHistorical.HasChronicHealthCondition and hmis:personHistorical.ReceiveChronicHealthConditionServicesTreatment

Per *HMIS Revised Draft Notice,* Section 4.5.

50. Changed HIVAIDSStatus to hmis:hIVAIDSStatus from hmis:twoVal

51. Added hmis:hIVAIDSStatus.HasHIVAIDS and hmis:hIVAIDSStatus.ReceiveHIVAIDSServicesTreatment

Changed per *HMIS Revised Draft Notice,* Section 4.6.

52. Added hmis:personHistorical.MentalHealthProblem

hmis:personHistorical.MentalHealthProblem contains one new picklist value and consolidates hmis:personHistorical.MentalHealthIndefinite and formerly existing hmis:personHistorical.MentalHealth per *HMIS Revised Draft Notice,* Section 4.7.

53. Added hmis:personHistorical.DomesticViolence

hmis:dVHowLong now an hmis:sixValDKRefused2. hmis:personHistorical.DVHowLong now hmis:domesticViolence.DVOccured per *HMIS Revised Draft Notice,* Section 4.9.

54. hmis:destination now its own type, instead of reusing hmis:residence

Changed per *HMIS Revised Draft Notice,* Section 4.10. Also, Destination moved out of hmis:siteServiceParticipation into hmis:personHistorical.Destinations.

55. Former hmis:siteServiceParticipation.DestinationTenure removed and folded into hmis:destination enumeration

Per *HMIS Revised Draft Notice,* Section 4.10.

56. Added hmis:personHistorical.ContactMade

Also added hmis:contact complex type, per *HMIS Revised Draft Notice,* Section 4.11.

57. Added hmis:personHistorical.EngagedDate

Per *HMIS Revised Draft Notice,* Section 4.12.

58. hmis:barrier.BarrierOther and hmis:barrier.Barrier Code grouped and indexed with hmis:barrier.BarrierID

59. hmis:degree.DegreeCodeOther and hmis:degree.DegreeCode grouped and indexed with hmis:degree.DegreeID

60. hmis:priorResidence.PriorResidenceCode and hmis:priorResidence.PriorResidenceOther grouped and indexed with hmis:priorResidence.PriorResidenceID

61. hmis:reasonForLeaving.ReasonForLeaving and hmis:reasonForLeaving.ReasonForLeavingOther grouped and indexed with hmis:reasonForLeaving.ReasonForLeavingID

62. hmis:personHistorical.SubsidyType removed

63. Cardinality on all hmis:personHistorical subelements changed to minOccurs="0" maxOccurs="unbounded"

64. Added hmis:serviceEvent.FinancialAssistance element

Per *HMIS Revised Draft Notice,* Section 4.13. Added as a choice next to preexisting hmis:serviceEvent.ServiceType (renamed HMISServiceType)

65. Added hmis:serviceEvent.RelocationStabilizationServiceType

Per *HMIS Revised Draft Notice,* Section 4.14. Added as a choice next to preexisting hmis:serviceEvent.ServiceType (renamed HMISServiceType)

66. hmis:employment.EmploymentTenure type changed from an hmis:threeVal to an hmis:fiveValDKRefused

Per *HMIS Revised Draft Notice,* Section 4.16A.

67. hmis:employment.LookingForWork changed from an hmis:twoVal to an hmis:fourValDKRefused

Per *HMIS Revised Draft Notice,* Section 4.16A.

68. hmis:personHistorical.CurrentlyInSchool changed from an hmis:twoVal to an hmis:fourValDKRefused

Per *HMIS Revised Draft Notice,* Section 4.16B.

69. hmis:personHistorical.VocationalTraining changed from an hmis:twoVal to an hmis:fourValDKRefused

Per *HMIS Revised Draft Notice,* Section 4.16B.

70. hmis:highestSchoolLevelBase enumerations changed

Per *HMIS Revised Draft Notice,* Section 4.16B.

71. hmis:degreeCode changed to enumeration with different picklist values

Per *HMIS Revised Draft Notice,* Section 4.16B.

72. hmis:healthStatusBase enumeration added 'Refused' enumeration

Per *HMIS Revised Draft Notice,* Section 4.16C.

73. hmis:personHistorical.Pregnancy element consolidates and indexes hmis:pregnancy.PregnancyStatus and hmis:pregnancy.DueDate

Per *HMIS Revised Draft Notice,* Section 4.16D.

74. hmis:pregnancy.PregnancyStatus changed from hmis:twoVal to hmis:fourValDKRefused

75. Cardinalities in hmis:veteran.ServiceEra changed

76. hmis:serviceEraBase picklist values changed, made into an enumeration

Per *HMIS Revised Draft Notice,* Section 4.16E.

77. hmis:veteran.ServedInWarZone changed from hmis:twoVal to hmis:fourValDKRefused

Per *HMIS Revised Draft Notice,* Section 4.16E.

78. hmis:veteran.WarZonesServed is a new sequence of hmis:warZoneServed.WarZone and related elements with an index.

Per *HMIS Revised Draft Notice,* Section 4.16E.

79. hmis:veteran.WarZone changed from hmis:tenVal to hmis:warZone type which is an enumeration with a change picklist

Per *HMIS Revised Draft Notice,* Section 4.16E.

80. hmis:veteran.ReceivedFire changed from hmis:twoVal to hmis:fourValDKRefused

Per *HMIS Revised Draft Notice,* Section 4.16E.

81. hmis:veteran.MilitaryBranch moved into the hmis:militaryBranches sequence with MilitaryBranchOther with an index

82. hmis:dischargeBase picklist values changed and made into an enumeration

Per *HMIS Revised Draft Notice,* Section 4.16E.

83. hmis:personHistorical.ChildEnrollmentStatus sequence created with an index

84. hmis:personHistorical.ChildCurrentlyEnrolledInSchool changed from hmis:twoVal to hmis:fourValDKRefused

Per *HMIS Revised Draft Notice,* Section 4.15F.

85. hmis:schoolType (formerly hmis:schoolBase) changed to hmis:fourValDKRefused2

Per *HMIS Revised Draft Notice,* Section 4.15F.

86. Barrier use of hmis:tenValBase removed, replaced by hmis:barrierCode type

Per *HMIS Revised Draft Notice,* Section 4.16F.

87. Picklist values changed in hmis:elevenValBase for hmis:reasonForLeaving

Per *HMIS Revised Draft Notice,* Section 4.16G.

88. Removed hmis:tenVal

89. hmis:typeOfServiceBase values changed

Per *HMIS Revised Draft Notice,* Section 4.16H.

90. VeteranStatus moved from hmis:siteServiceParticipation to hmis:veteran

91. Destination now an element with a complex type in hmis:personHistorical, named hmis:destinations

92. DisablingCondition moved to hmis:personHistorical from hmis:siteServiceParticipation

93. SiteServiceID added to hmis:personHistorical so an agency structure could be attributed with the historical client info

94. SiteServiceID in SiteServiceParticipation changed to type hmis:unsignedInt for better fit with AIRS Schema

95. Added hmis:employment type

To hold *HMIS Revised Draft Notice,* Section 4.16A elements

96. All sequences alphabetized with exception of indexes, which come always appear first

97. hmis:serviceEvent.QuantityOfService made maxOccurs = 1

98. hmis:hMISServiceType groups TypeOfService with free text field TypeOfServiceOther for use when 'Other' response is selected

99. All airs: key types brought in alignment with hmis: keys (unsignedInt instead of hmis:id)

100. HUDHomeless deprecated in favor of HousingStatus

Per *HMIS Revised Draft Notice,* Section 3.11.

101. ClientOutcomesMeasure added to PersonHistorical

Per *HMIS Revised Draft Notice,* Section 4.15.

102. Regions added to schema to group SiteServices arbitrarily for reporting.

103. hmis:hashingChoice.Hashed/<u>Unhashed</u> changed to <u>minoccurs</u>=1 instead of <u>zero</u>

104. <u>TargetPopulationA/B</u> moved from hmis:<u>siteService</u> to hmis:service

105. Removed FacilityCode from hmis:<u>siteService</u>, since Facility Code was removed from *HMIS Revised Draft Notice*

106. IndividualFamilyCode moved to hmis:service from hmis:siteService

107. Removed of SiteServiceType/Base