Integrating the Healthcare Enterprise

IHE Quality, Research and Public Health Technical Framework Supplement

Aggregate Data Exchange (ADX)

Rev. 2.1 – Trial Implementation

Date: August 18, 2017

Author: QRPH Technical Committee

Email: qrph@ihe.net

Please verify you have the most recent version of this document. See here for Trial Implementation and Final Text versions and here for Public Comment versions.

Foreword

This is a supplement to the IHE Quality, Research and Public Health (QRPH) Technical Framework. Each supplement undergoes a process of public comment and trial implementation before being incorporated into the volumes of the Technical Frameworks.

This supplement is published on August 18, 2017 for trial implementation and may be available for testing at subsequent IHE Connectathons. The supplement may be amended based on the results of testing. Following successful testing it will be incorporated into the QRPH Technical Framework. Comments are invited and may be submitted at http://www.ihe.net/QRPH_Public_Comments.

"Boxed" instructions like the sample below indicate to the Volume Editor how to integrate the relevant section(s) into the relevant Technical Framework volume.

Amend Section X.X by the following:

Where the amendment adds text, make the added text **bold underline**. Where the amendment removes text, make the removed text **~~bold strikethrough~~**. When entire new sections are added, introduce with editor's instructions to "add new text" or similar, which for readability are not bolded or underlined.

General information about IHE can be found at www.ihe.net.

Information about the IHE QRPH domain can be found at http://www.ihe.net/IHE_Domains.

Information about the organization of IHE Technical Frameworks and Supplements and the process used to create them can be found at http://www.ihe.net/IHE_Process and http://www.ihe.net/Profiles.

The current version of the IHE QRPH Technical Framework can be found at http://www.ihe.net/Technical_Frameworks.

CONTENTS

Introduction to this Supplement

The Aggregate Data Exchange (ADX) Profile supports interoperable public health reporting of aggregate health data. These most typically take the form of routine reports (weekly, monthly, quarterly etc.) from a health facility to some administrative jurisdiction such as a health district, though there are numerous other use cases such as international reporting and community health worker reporting.

The motivating context for this profile originates in health systems management in developing countries though its potential use is not restricted to these environments. What are collectively called developing countries are very diverse environments. Information systems need to be able to scale and adapt across diverse and changing conditions. Electronic medical record systems (EMR) penetration is often limited (but also often growing) leading to varied and mixed modes of data collection and transmission.

The health information system which enjoys national coverage in such diverse settings is often the routine reporting system. This is the national health management information system (HMIS) which gathers aggregate data from all health facilities so that indicators related to population health and facility service utilization can be generated, compared and analyzed to produce and assess relevant action interventions within the health system.

Data is currently being exchanged using a variety of ad-hoc formats which are not formally described. Increasing use of health facility and laboratory based electronic systems as well as administrative systems related to logistics and human resources starts to expose the value of interoperable systems in emerging health information exchange settings.

There is a need for a standard means of representing such routine reports and for profiles which govern their electronic exchange.

The nature of routine reports

Data is reported from facilities for the construction of indicators. In many countries this data is aggregated from paper registries and tally sheets and entered on a paper form provided to all health facilities by the Ministry of Health. In a non-computerized environment these forms are then physically taken to the health district office where they may be entered into a computer system. Forms vary greatly in their complexity as they will have evolved over time to meet a complex mix of (often competing) requirements. They are historical, cultural and political products.

The following examples are of sections taken from the monthly reporting form used by health facilities in Rwanda (currently published at http://moh.gov.rw/index.php?id=135).

Simple example without disaggregation

The example in Figure 1 shows a section of the Antenatal reporting form. The structure is simple, requiring an integer value in the right column for each data element label in the left. The

data values will be uniquely keyed by the data element, the reporting month and the reporting facility.

XII. Antenatal Consultations		
1	ANC New Registrations	
2	ANC First standard visit 1st trimester	
3	ANC 4th standard visit	
4	ANC high risk pregnancy detected (including pregnancy under 15 years)	
5	ANC pregnancy under 15 years	
6	ANC high risk referred	
7	ANC TT 1given	
8	ANC TT 2 to 5 given	
9	ANC TT new registrations fully vaccinated	
10	ANC new registrations who received full course of Iron and Folic Acid supplements (90 tablets)	
11	ANC Insecticide Treated Bed nets distributed	
12	ANC deworming performed	
13	ANC new registrations screened for malnutrition (MUAC)	
14	ANC new registrations screened who were malnourished (MUAC < 21 cm)	

Figure 1: Sample monthly reporting form without disaggregation

A more complex example using disaggregation

The example in Figure 2 below shows a section of the monthly reporting form related to gender based violence. In this case the data elements are in rows, but the "answers" are divided into columns to reflect that the data has to be further disaggregated across age and sex.

Further, some cells in the resulting grid are greyed out to indicate that those particular disaggregate combinations are not valid for data entry.

A single reporting form may contain sections which are disaggregated differently.

HMIS Health Center/Dispensary HMIS Monthly Report 03/03/2015

XI. Gender Based Violence

		Under 5 years		5-9 years		10-18 years		19 -24 years		25years and above	
		M	F	M	F	M	F	M	F	M	F
1	GBV victims with symptoms of sexual violence (new cases).										
2	GBV victims with symptoms of physical violence (new cases).										
3	GBV victims referred for care to higher level health facility										
4	GBV victims referred to this facility by police										
5	GBV victims referred to this facility by community health workers										
6	GBV victims HIV+ sero-conversion 3 months after exposure										
7	GBV victims with irreversible disabilities due to GBV										
8	GBV victim deaths										
9	GBV victims pregnant 4 weeks after exposure										
10	GBV victimsreceived emergency contraception within 72 hours										

Figure 2: Sample monthly reporting form with disaggregation

Approach

This specification profiles an SDMX v2.1 Data Structure Definition (DSD) to normatively describe the structure of routine aggregate data reports as XML. It introduces two new actors (Content Data Structure Creator and Content Data Structure Consumer) which produce and consume the DSD respectively.

The DSD is used by a Content Creator and Content Consumer to define the structural metadata of ADX/XML data messages exchanged.

The Content Data Structure Creator can also render the DSD as an XML schema to be used by Content Creator and Content Consumer for message validation.

Open Issues and Questions

Should we register application/adx+xml as per IETF RFC6838 Media Type Specifications and Registration Procedures? See https://www.iana.org/form/media-types

There is some overlap in the aims with QRDA. We need to justify and explain.

Closed Issues

Should we register application/adx+xml as per IETF RFC6838 Media Type Specifications and Registration Procedures? (See https://www.iana.org/form/media-types)

RESOLUTION: Submitted provisional registration for application/adx-xml to internet assigned numbers authority (IANA)

Maybe we have to create new actors if Content Creator/Content Consumer do not fit.
RESOLUTION: A new actor has been defined: the Content Data Structure Creator.

Placing dimensions at the dataset or data row level requires further discussion.

RESOLUTION: For ADX mandatory dimensions, orgUnits and period dimensions shall be placed at the dataset level and data element dimension shall be placed at the data row level.

CSV format may be considered too low a bar for profiling. It should not be at the same conformance level as the other two.

RESOLUTION: CSV format has been dropped from the profile.

SDMX v2.1 describes and xml and CSV rendition of the data. It doesn't currently describe a json rendition. There is considerable pressure from implementers to support json. Should we incorporate a mapping in an informative appendix?

RESOLUTION: Only XML formats are normatively defined.

SDMX 2.0 vs 2.1: version 2.0 is the most widely adopted version of SDMX, but it is 2.1 which is published by ISO. V2.0 is published by the SDMX consortium.

RESOLUTION: Because it is the version balloted by ISO, ADX profiles SDMX v2.1.

General Introduction

Update the following appendices to the General Introduction as indicated below. Note that these are not appendices to Volume 1.

Appendix A – Actor Summary Definitions

Add the following actors to the IHE Technical Frameworks General Introduction list of actors:

Actor	Definition
Content Data Structure Creator	The Content Data Structure Creator creates a message structure definition that may be employed by a Content Creator to develop profile-conformant messages for exchange with a Content Consumer.
Content Data Structure Consumer	The Content Data Structure Consumer consumes a message structure definition that may be employed by a Content Creator to develop profile-conformant messages for exchange with a Content Consumer.

Appendix B – Transaction Summary Definitions

Add the following transactions to the IHE Technical Frameworks General Introduction list of Transactions:

Transaction	Code	Definition
ADX POST Content	QRPH-53	The POST Content transaction is used by the Content Creator to perform an HTTP POST request on the Content Consumer.

Glossary

Add the following glossary terms to the IHE Technical Frameworks General Introduction Glossary:

No new glossary terms.

Volume 1 – Profiles

Copyright Licenses

NA

Add the following to the IHE Technical Frameworks General Introduction Copyright section:

Domain-specific additions

NA

Add Section X …

X Aggregate Data Exchange (ADX) Profile

The Aggregate Data Exchange (ADX) Profile enables interoperable public health reporting of aggregate health data. ADX will typically be used to represent routinely reported aggregate data such as the numerators and denominators which can be used in the construction of public health indicators.

The central concern of an ADX is the reporting of data tuples. These tuples are sets of values which are keyed according to a data element subject, a temporal dimension, and a spatial dimension. An example data tuple is the number of live births recorded in January 2015 at Nyamandhlovu Clinic.

Aggregate health data elements are defined in an implementing jurisdiction, e.g., a single health facility or a geographic/administrative area such as district, province or country, and are currently being exchanged using a variety of ad-hoc and application-specific formats which are not formally described.

ADX defines a Content Data Structure Creator that creates two message structures that enable an implementing jurisdiction to formally define the aggregate health data to be exchanged:

1. ADX profiles the SDMX v2.1 Data Structure Definition (DSD) specification
2. ADX normatively describes how a DSD file is transformed to develop a W3C XML schema definition (XSD) file and an ISO Schematron schema to validate ADX data messages.

ADX defines a Content Data Structure Consumer that consumes a DSD.

The DSD and schema describe lightweight, formal XML messages containing aggregate health data that meet the requirements of the use cases in the implementing jurisdiction.

Content Creator and Consumer Actors use the ADX DSD and schema to construct and validate ADX/XML messages containing aggregate health data in their jurisdiction.

The ADX Profile contains few constraints regarding the nature and source of coding systems, and there are liberal extension points intended to allow ADX content to be embedded within different envelopes and its message attributes extended in locally-meaningful ways.

In summary:

- The ADX Profile defines the process for a Content Data Structure Creator to develop an SDMX-conformant DSD which describes the base constraints for a valid ADX XML data message. Individual jurisdictions will extend the DSD by specifying relevant codelists and additional dimensions of data to satisfy their message exchange use cases.
- The Content Data Structure Creator transforms the resulting ADX DSD into a W3C XML schema definition (XSD) and an ISO Schematron schema.
- A Content Creator then transmits ADX-conformant XML messages that are conformant with the structure described by the ADX-conformant XSD and DSD files to the Content Consumer. This may be using the ADX POST Content transaction defined in this profile.

X.1 ADX Actors, Transactions, and Content Modules

This section defines the actors, transactions, and/or content modules in this profile. General definitions of actors are given in the Technical Frameworks General Introduction Appendix A at http://ihe.net/Technical_Frameworks.

Figure X.1-1 shows the actors directly involved in the ADX Profile and the direction that the content is transmitted. Although the Content Creator employs the message structure definition files that result from a Content Data Structure Creator, there is not a message exchange transaction, per se, between these actors.

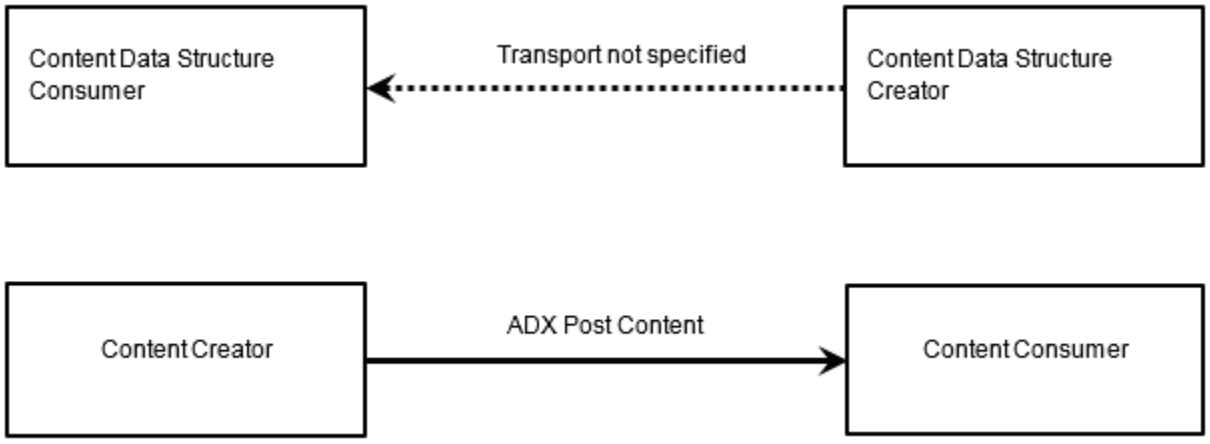

Figure X.1-1: ADX Actor Diagram

Table X.1-1 lists the content module(s) defined in the ADX Profile. To claim support for this profile, an actor shall support all required content modules (labeled "R") and may support optional content modules (labeled "O").

Table X.1-1: ADX Profile - Actors and Content Modules

Actors	Content Modules	Optionality	Reference
Content Data Structure Creator	ADX/DSD	R	QRPH TF-3: Sec 8.2-8.3, App. 8A, 8E
	ADX/XSD	R	QRPH TF-3: Sec 8.2-8-3, App. 8B, 8F
Content Data Structure Consumer	ADX/DSD	R	QRPH TF-3: Sec 8.2-8.3, App. 8A, 8E
Content Creator	ADX/XML	R	QRPH TF-3: App 8G
Content Consumer	ADX/XML	R	QRPH TF-3: App 8G

X.1.1 Actor Descriptions and Actor Profile Requirements

X.1.1.1 Content Data Structure Creator

A Content Data Structure Creator defines the structure of XML data to be exchanged between a Content Creator and Content Consumer. Typically, the Content Data Structure Creator will be an implementing jurisdiction such as a ministry of health, a global non-government organization (NGO) or a donor.

A Content Data Structure Creator shall create two normative message structure definition files.

1. an SDMX v2.1-conformant Data Structure Definition (DSD) file conformant to the normative ADX schematron specification (see QRPH TF-3: 8.2 and Appendix 8A)
2. a W3C XML Schema Definition (XSD) and an ISO Schematron schema matching the result generated by the normative XSLT transform from DSD to XSD and from DSD to Schematron (see QRPH TF-3: 8.3 and Appendix 8B and Appendix 8D).

Informative examples of a DSD and generated schemas are included in QRPH TF-3: Appendices 8F, 8G and 8H.

X.1.1.2 Content Data Structure Consumer

A Content Data Structure Consumer consumes an ADX DSD file produced by a Content Data Structure Creator. Typically the system implementing this actor role will also be a Content Creator. The DSD can be used to configure the Content Creator to produce valid content.

X.1.1.3 Content Creator

A Content Creator shall be able to generate an XML file that is conformant to the schema defined by the DSD and schemas produced by the Content Data Structure Creator.

X.1.1.4 Content Consumer

A Content Consumer shall be able to process an XML stream that is conformant to the schema defined by the DSD and schemas produced by the Content Data Structure Creator. What it means to process the xml stream depends on the nature of the processor. For example it might persist the individual data tuples, or it might format them for display or perform further aggregation on the data.

X.2 ADX Actor Options

Table X.2-1: ADX - Actors and Options

Actor	Option Name	Reference
Content Data Structure Creator	No options defined	--
Content Data Structure Consumer	No options defined	--

Actor	Option Name	Reference
Content Creator	POST Content	X 2.1
Content Consumer	No options defined	--

X.2.1 ADX POST Content

A Content Creator may support the action to post content to a Content Consumer using the ADX POST Content transaction. This is an option rather than a mandatory conformance requirement to support use cases where a Content Creator produces a conformant ADX data message but may have to transport it by other means (for example USB memory stick or email).

X.3 ADX Required Actor Groupings

There are no Required Actor Groupings defined.

X.4 ADX Overview

ADX defines the normative process used to develop a simple message structure for representing aggregate health data.

The ADX message structure is defined using a Data Structure Definition (DSD) file conformant to the Statistical Data and Metadata Exchange (SDMX v2.1) specification. SDMX v2.1 is defined in ISO 17639:2013(E). ADX profiles this ISO standard; it constrains the SDMX specification and articulates how an ADX-conformant DSD is developed. The schemas for exchanged messages are generated from the DSD via a normative transformation.

This profile sets constraints on the mandatory dimensions which shall be in a DSD. Additional data element dimensions may be defined as necessary within the context of use - for example, within a particular country or implementing jurisdiction. Similarly, whereas ADX assumes that codelists and other structural metadata will be exchanged between Content Creators and Content Consumers, the content of that structural metadata is out of scope for ADX and shall be defined by the jurisdictional bodies governing the data exchange. These inputs to the ADX message schema definition are conceptually illustrated by Figure X.4-1.

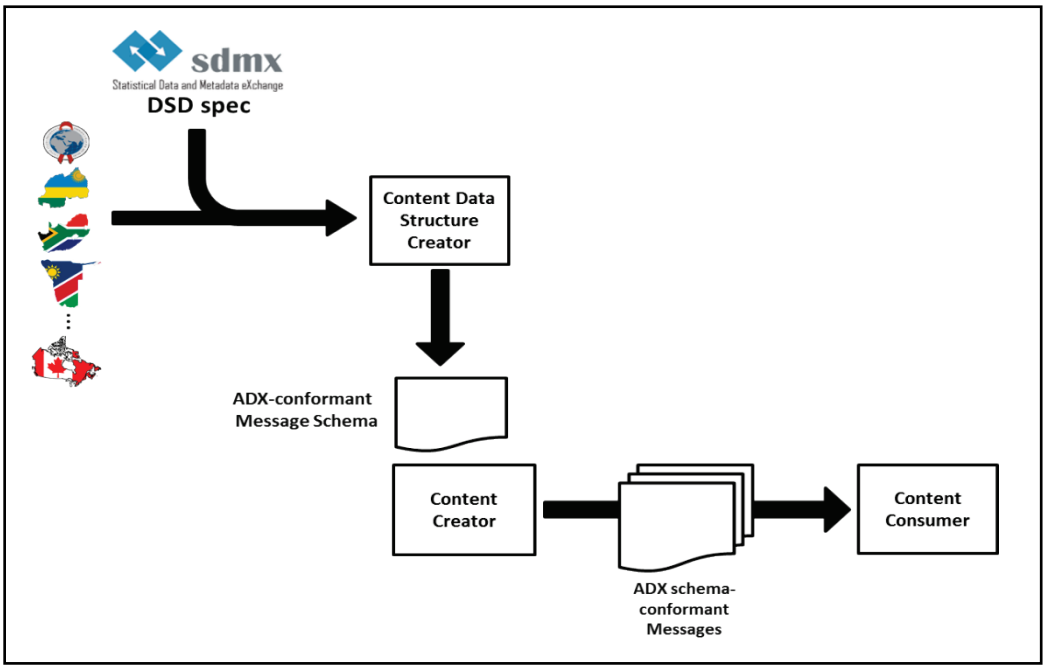

Figure X.4-1: An illustration of ADX Profile inputs and outputs

The shape of aggregate data reports can be different, as illustrated in the use cases in Section X.4.2. In some cases the data is "rectangular" (all data rows share the same dimensionality). In other cases, where the electronic data must model a complex paper form (for example), some data elements are disaggregated differently from others, leading to a "ragged-right" shape. Rectangular datasets are shaped "long and skinny"; for each combination of organization unit (typically a "facility") and time period, there will be a list of data element and value pairs. Ragged-right messages, on the other hand, may have tuples of differing widths depending on whether disaggregation attributes have been specified for the data element (e.g., number of malaria cases, disaggregated by gender and by age range). These two characteristic message shapes are illustrated in Figure X.4-2.

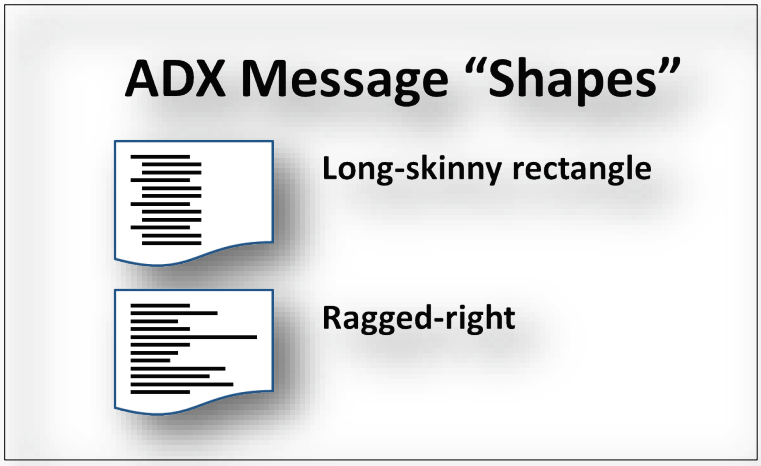

Figure X.4-2: ADX Message Shapes

In summary:

- The ADX Profile defines the process for a Content Data Structure Creator to develop an SDMX-conformant DSD which describes the base constraints for a valid ADX XML data message. See QRPH TF-3: 8.2.
- Individual jurisdictions will extend the DSD by specifying relevant codelists and additional dimensions of data to satisfy their message exchange use cases.
- The Content Data Structure Creator transforms the resulting ADX DSD into an XML schema definition (XSD) file. See QRPH TF-3: 8.3.
- A Content Creator then transmits ADX-conformant XML messages that are conformant with the structure described by the ADX XSD and DSD to the Content Consumer.

X.4.1 Concepts

The following concepts are used in this profile:

Data Structure Definition (DSD): a formal SDMX v2.1 definition of the structural metadata of an ADX message.

Data value: a reported value with its associated keys

Data Value Set: a collection of data values conforming to the ADX DSD.

Data element: the atomic subject of data collection such as "number of bed nets distributed", "number of confirmed malaria cases", "number of midwives". This concept is required for the construction of public health indicators, but is not in itself an indicator.

Organization Unit: the spatial dimension of the data value tuple. It can identify a single health facility or a geographic/administrative area such as district, province or country. Within a

national reporting structure, an organization unit represents a single node in a reporting hierarchy.

Time: the temporal dimension of the data value tuple. For routine data this would represent a time period such as the month of January 2015, the ISO8601 week number 3 of 2015, or the year 2015.

Value: the recorded value which is keyed in the data value tuple.

X.4.2 Use Cases

X.4.2.1 Use Case #1: Community health workers filing routine reports

X.4.2.1.1 Community health workers filing routine reports: Use Case Description

An Auxiliary Nurse Midwife (ANM) in an Indian state uses a simple SMS message to report 40 simple aggregate data elements. These are reported monthly and some of the more critical ones also daily. For example

- Number of ANY new cases registered
- Number of children given full immunization

These are all related to mother and child health services provided by the sub center and either in her clinic or outreach. She would take these numbers from her register. The data elements are all simple without further disaggregation.

The SMS message is sent to a gateway system, which (today) pushes the aggregate data report into the state level Health Management Information System (HMIS) system using a proprietary format. In the future state, the Content Data Structure Creator defines the structure of the mother and child data to be exchanged, and the gateway system acts as a Content Creator to produce these reports in ADX format which can be readily consumed by a Content Consumer.

For ADX messages sent from the gateway that relate to a particular ANM for a particular time period, this message exchange would be a potential example of the “long skinny” rectangular shape.

X.4.2.2 Use Case #2: Reporting health worker data

X.4.2.2.1Reporting health worker data Use Case Description

In Sierra Leone the two systems central to the management of the health system are the national HMIS and the human resource system, or health worker registry (HWR). A routine human resource report is reported by the health facility every six months. The reports consist of counts of health worker cadres working at the facility (number of doctors, nurses, community health workers etc.). There is an additional disaggregation on all report rows which indicates the salaried status of the employee.

This data could instead be extracted directly from the health worker registry and represented as an ADX summary report to be consumed by the national HMIS. The benefits of doing this can

be either to reduce the reporting burden at the facility by replacing the manual data entry or to supplement it by acting as a control to detect staffing anomalies. The report extracted from the health worker registry would contain data for a single time period but for multiple health facilities.

In this use case, the Content Data Structure Creator defines the structure of the human resource report, the health worker registry acts as a Content Creator, and the national HMIS is a Content Consumer.

X.4.2.3 Use Case #3: Routine reporting from EMR

This use of ADX enables generation and reporting of aggregate data from EMRs to HMIS without the need for double data entry by facility staff in both systems.

X.4.2.3.1 Routine reporting from EMR: Use Case Description

In the majority of countries, routine health facility reports are compiled manually at the facility from paper registers and tally sheets. To deal with the challenge of continuity of care, the antiretroviral therapy (ART) treatment center of a health facility has acquired an EMR system to manage patients. The routine ART reports demanded of the facility are complex, so there are considerable benefits in terms of reducing reporting burden as well as increasing accuracy to extracting reports from the EMR.

Current state

In the current state, aggregate data from a facility EMR is generated as a printed report and sent to a district office. Here, this aggregate data is hand-entered into the district health information system (DHIS), which feeds the national HMIS. This workflow is illustrated by Figure X.4.2.3.1-1.

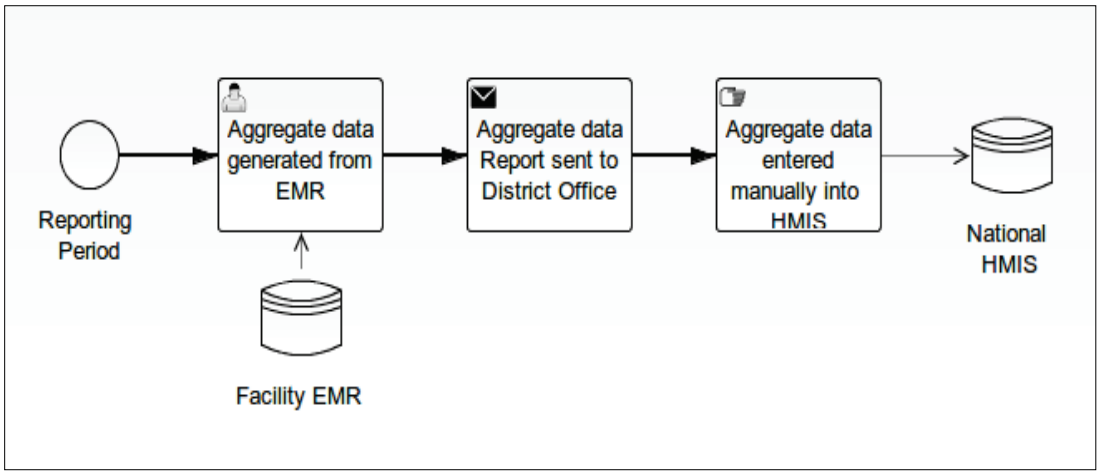

Figure X.4.2.3.1-1: Current state of EMR to HMIS data exchange

Desired state

The desired state is illustrated by Figure X.4.2.3.1-2. In the desired state, data from the facility EMR is used to generate an ADX-conformant message. The ADX-conformant message is sent by the facility EMR (acting as a Content Creator) to the DHIS (acting as a Content Consumer). Such a message exchange improves the timeliness and efficiency of the HMIS data reporting workflow.

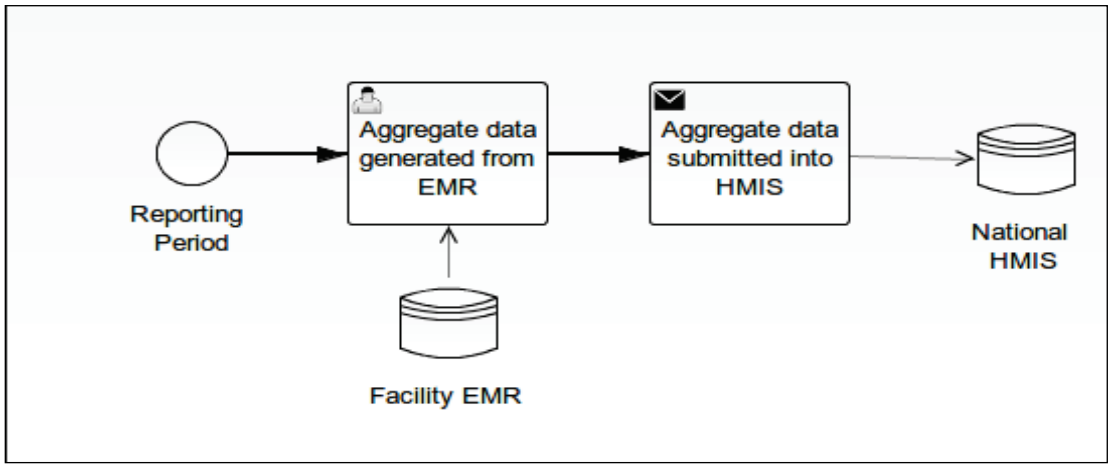

Figure X.4.2.3.1-2: Desired state of EMR to HMIS data exchange

The ADX Content Data Structure Creator models the facility monthly reports off the paper reporting form and consists of a set of data elements which form a logical set but are disaggregated differently. For example the number of ART stage 1 new enrollment is disaggregated by three age groups and by gender, whereas the number of patients on particular first line regimens is collected as simple counts without further disaggregation. The ART

monthly summary report from the facility will then consist of data which is not neatly rectangular in terms of dimensions, but will instead be an example of the "ragged-right" ADX message shape.

X.4.2.4 Use Case #4: Reporting data from HMIS to Other Aggregate Data Repositories

This use case describes ADX enabling the reporting of aggregate data from a national HMIS and consumed by other aggregate data repositories used for global reporting.

X.4.2.4.1 Reporting national data from HMIS: Use Case Description

To facilitate routine reporting from countries, country offices may generate aggregate data from national HMIS for submission to funding agencies. In this case the national HMIS is the Content Data Structure Creator and Content Creator. This data can be consumed by other aggregate data repositories, such as UNAIDS Global AIDS Response Progress Reporting (GARPR) online tool and PEPFAR information system, Data for Accountability, Transparency and Impact (DATIM), for monitoring health goals. As in the previous use case, the reported data will have a ragged disaggregation dimensionality.

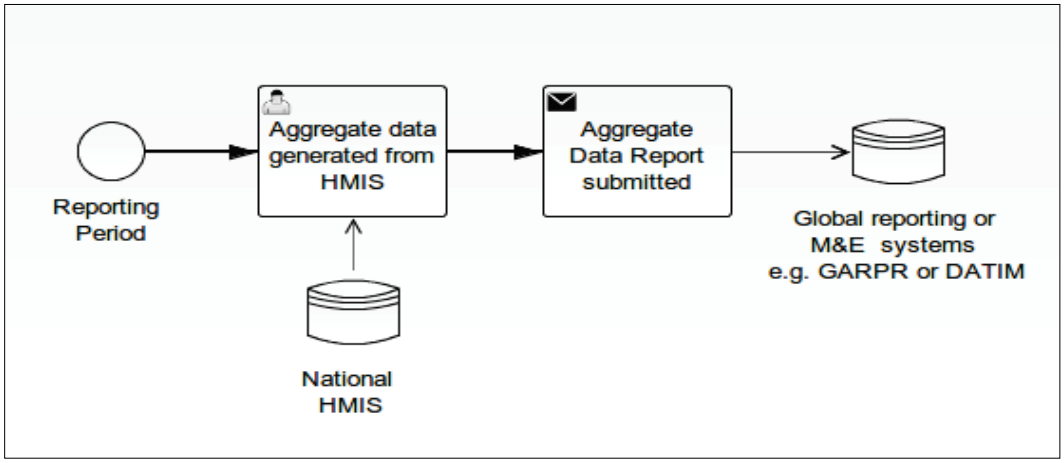

Figure X.4.2.4.1-1: Global reporting to M&E Systems (e.g., DATIM)

X.5 ADX Security Considerations

The ADX Profile does not support the exchange of person-centric health information. Therefore, this profile does not specify security mechanisms, such as the ITI Audit Trail and Node Authentication (ATNA) Profile, that would be required were that the case. Implementers should nevertheless be sensitive to the possibility of approximate personal identification arising from aggregate data derived from small population sets. Transport of such data should be safeguarded according to jurisdictional guidelines.

X.6 ADX Cross Profile Considerations

There are no Cross Profile considerations.

Appendices

None

Volume 2 – Transactions

Add Section 3.53

3.53 ADX POST Content [QRPH-53]

This section corresponds to Transaction QRPH-53 of the IHE QRPH Technical Framework. Transaction QRPH-53 is used by the Content Creator and Content Consumer Actors to share aggregate health data within a jurisdiction.

3.53.1 Scope

This transaction is used to communicate aggregate health data from the Content Creator to the Content Consumer at the end of each reporting cycle.

3.53.2 Actor Roles

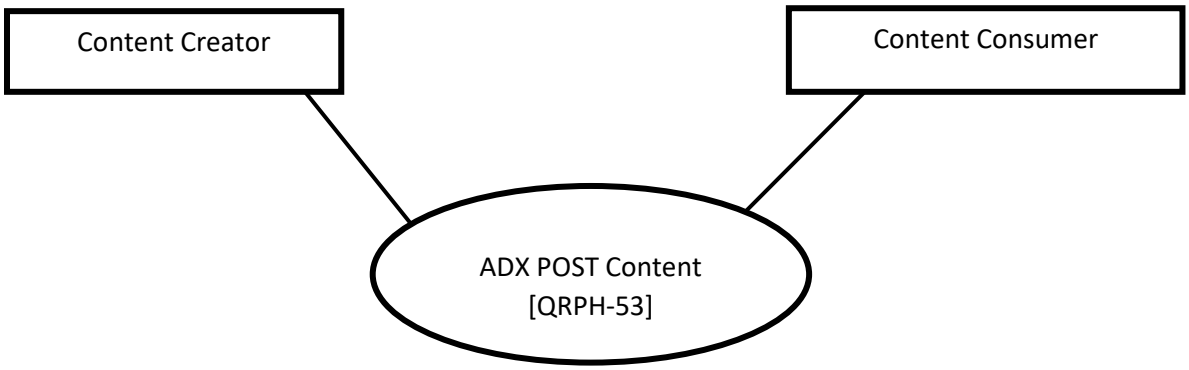

Figure 3.53.2-1: Use Case Diagram

The roles in this transaction are defined in the following table and may be played by the actors shown here:

Table 3.53.2-1: Actor Roles

Actor:	Content Creator
Role:	The Content Creator is responsible for the creation of an ADX message containing aggregate health data conformant to the jurisdiction defined ADX DSD and transmitting this message to a Content Consumer.
Actor:	Content Consumer

Role:	A Content Consumer is responsible for receiving the ADX message containing aggregate health data conformant to the jurisdiction defined ADX DSD from the Content Creator and processing it.

3.53.3 Referenced Standards

- IETF RFC2616 HyperText Transfer Protocol HTTP/1.1

3.53.4 Interaction Diagram

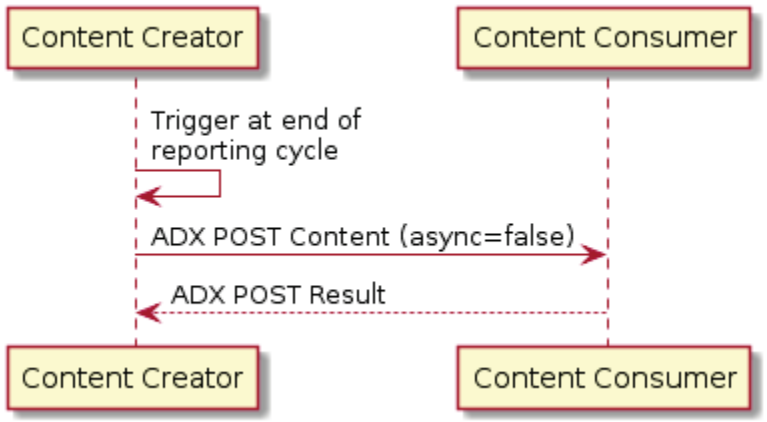

Figure 3.53.4-1: ADX POST Content Diagram

3.53.4.1 ADX POST Content

This transaction transmits ADX-conformant messages containing aggregate health data from the Content Creator to the Content Consumer. A Content Consumer implemented at a jurisdiction may receive this transaction from multiple Content Creators.

The ADX POST Content is implemented as an HTTP request using the POST method as specified in RFC2616.

3.53.4.1.1 Trigger Events

There are a wide variety of implementation and jurisdiction specific events which might trigger an ADX POST Content transaction. This might be automated, for example a timeout indicating the end of a routine reporting period, or manually triggered in response to prevailing business logic. The trigger event is implementation specific.

3.53.4.1.2 Message Semantics

The Content Creator creates an ADX conformant message containing aggregate health data that meets the requirements of the ADX DSD in their jurisdiction. The Content Creator **MAY** send the message using ADX POST Content. The Content Consumer **SHALL** consume the message that meets the requirements of the ADX DSD in their jurisdiction.

The table below describes the request.

	Description	
URL	The ADX Profile does not prescribe the form of the URL to be advertised by a Content Consumer except that the scheme of the URL **SHALL** be "https". The following is a non-exhaustive list of valid examples: https://hmis.gov.rw/datasets/adx https://hmis.gov.rw/routinereports/adx https://hmis.gov.rw/routinereports	
Headers	The http POST request **SHALL** contain a Content-type header identifying the payload Type:Content-type: application/adx+xml The request **MAY** contain any additional headers. For example, a Content Consumer may require an Authorization header.	
HTTP Parameters	async=true|false (optional)	When this parameter is absent, "async=false" is assumed. When "async=false", the Content Consumer **SHALL** process the request body and return an ADX POST result in the Response of the http request. When "async=true", and assuming there is no error condition detected, the Content Consumer **SHALL** return an HTTP 202 code and produce a URL in the Location header of the http response which the Content Producer may use to poll for ongoing status. *Processing of the async parameter is not optional for a Content Consumer. Where the parameter is present the Content Consumer **SHALL** parse and interpret it according to the semantics above or return an error code.*
	atomic=true|false (optional)	When this parameter is absent, "atomic=false" is assumed. When atomic=true, and if Content Consumer supports atomic transactions, the processing of the POST is guaranteed to either succeed entirely, or fail entirely. *Processing of the atomic parameter is not optional for a Content Consumer. Where the parameter is present the Content Consumer **SHALL** parse and interpret it according to the semantics above or return an error code.*
	A Content Consumer **MAY** support additional parameters.	
BODY	The body of an ADX POST Content request **SHALL** contain a valid ADX data xml payload as described in Section 8.2	

3.53.4.1.3 Expected Actions

The Content Consumer **SHALL** processes the ADX message received and return the status of the transaction as an ADX POST Result.

3.53.4.2 ADX POST Result

This transaction is an acknowledgement of ADX POST Content transaction from the Content Consumer to the Content Creator.

The ADX POST Result is implemented as an HTTP response. It can be emitted synchronously in response to the initial HTTP POST request, or maybe made available at a later time. This behavior is determined by the **async** parameter in the request.

When **async=false** in the ADX POST Content Request, then ADX POST Result will be returned as the HTTP response of the originating ADX POST request.

When **async=true**, the ADX POST Result may not be immediately available from the Content Consumer. For example, it may be queued by the Content Consumer for later processing or the processing may be a long-running process. In this case, the Content Consumer **SHALL** implement the mechanism described below for the Content Creator to poll for the status and completion of the transaction.

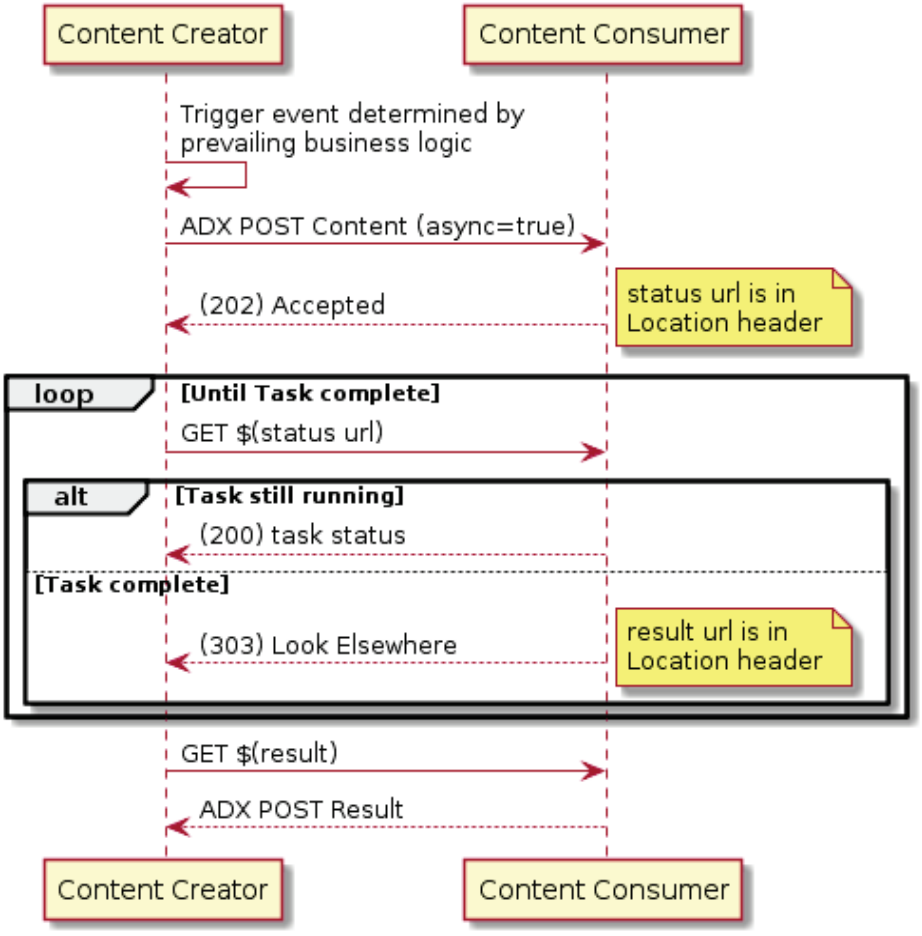

Figure 3.53.4.2-1: Async ADX POST Content Diagram

If it is prepared to process the request the Content Consumer **SHALL** respond with an HTTP code 202 and a URL in the Location header to poll for task status. Otherwise it may respond with an error code terminating the transaction.

Content Consumer **MAY** perform a series of GET requests on the status url, polling for completion. If there is no error, the response code to these requests **SHALL** be either 200, signifying task is still running, or 303 signifying the task is complete. The 303 response **SHALL** include a URL in the Location header indicating the URL where the ADX POST Result can be retrieved.

The Content Consumer makes no guarantee that either the status url or the result url will be available permanently.

3.53.4.2.1 Trigger Events

A Content Consumer sends an ADX POST Result after receiving and processing an ADX POST Content message from the Content Creator. For a synchronous request this will be the HTTP Response of the originating Request. For an asynchronous Request this will be in the HTTP Response of a later request that the Content Creator may make after polling for completion.

3.53.4.2.2 Message Semantics

The ADX POST Result is implemented as an HTTP Response. The response may include content in the body to provide an implementation and jurisdiction specific informative message on the completed status of the transaction. The response shall contain an HTTP status code. The table below describes the codes which may be produced by the Content Consumer which have a specific meaning related to the transaction.

Note that a Content Creator should be prepared to handle additional status codes not particular to the transaction, such as authorization, server or network error codes.

Table 3.53.4.2.2-1: ADX POST Result status codes

HTTP status code	Interpretation
200	ADX POST Request was successfully processed
202	ADX POST request has been accepted for processing, but the processing has not been completed. The request might or might not be eventually acted upon, and may be disallowed when processing occurs.
303	The response to the ADX POST when the task is complete can be retrieved from another URL. When received in response to an ADX POST, the client should presume that the server has received the data and should issue a redirect with a separate GET message.
400	Bad Request - XML content is badly formed or invalid
409	Conflict - invalid identifier in the XML content. If the atomic parameter was set to true, then no data will have been processed. If atomic was false, then partial data may have been processed.
415	Unsupported content-type or media
501	The request method is not implemented.

3.53.4.2.3 Expected Actions

There are no prescribed actions for what a Content Creator should do with an ADX POST Result.

3.53.5 Security Considerations

The envisaged use cases of the ADX POST Content [QRPH-53] transaction do not include the exchange of person-centric health information (PHI). Therefore, this transaction would not typically require security mechanisms that protects PHI, such as the ITI Audit Trail and Node Authentication (ATNA) Profile. If implementers *do* use ADX to carry PHI then it is anticipated that the transfers of Personal Health Information (PHI) will be protected. In that case, the IHE ITI Audit Trail and Node Authentication (ATNA) Profile **SHOULD** be implemented by both of the actors involved in the IHE transaction specified in this profile to protect node-to-node communication and to produce an audit trail of the PHI related actions when they exchange messages, though other appropriate security mechanisms **may** be used to secure content within enterprise managed systems.

Implementers **SHOULD** nevertheless be sensitive to the possibility of approximate personal identification arising from aggregate data derived from small population sets.

Transport of ADX data **SHOULD** be safeguarded according to jurisdictional guidelines. To protect data integrity these **SHOULD** include encryption of the transport layer and the use of an appropriate mutual authentication mechanism which meets these guidelines.

Content Consumers should also take adequate account of security considerations related to the generic processing of XML documents (RFC7303).

3.53.5.1 Security Audit Considerations

There is no specific ATNA security audit event that is associated with this transaction.

Appendices

None

Volume 2 Namespace Additions

Add the following terms to the IHE General Introduction Appendix G:

NA

Volume 3 – Content Modules

5 Namespaces and Vocabularies

The following are namespaces referred to in ADX Profile together with the prefix which is used to refer to that namespace within this document.

620

Namespace	Prefix	Description
http://www.sdmx.org/resources/sdmxml/schemas/v2_1/message	mes	SDMX 2.1 message
http://www.sdmx.org/resources/sdmxml/schemas/v2_1/structure	str	SDMX 2.1 structure definitions
http://www.sdmx.org/resources/sdmxml/schemas/v2_1/common	com	SDMX 2.1 common elements
http://www.w3.org/2001/XMLSchema	xs	W3C Schema
urn:ihe:qrph:adx:2015		ADX data payload

6 Content Modules

CDA®[1] Content Modules - Not applicable

[1] CDA is the registered trademark of Health Level Seven International.

7 Content Modules

625 DICOM®[2] Content Modules -Not applicable

[2] DICOM is the registered trademark of the National Electrical Manufacturers Association for its standards publications relating to digital communications of medical information.

8 ADX Content Modules

This section defines Content Modules for the Aggregate Data Exchange (ADX) Profile.

8.1 Overview of ADX process

A Content Data Structure Creator creates ADX-conformant DSD and schema streams. Typically, the ADX Content Data Structure Creator will be an implementing jurisdiction such as a ministry of health, a global non-government organization (NGO) or a donor.

- An ADX compliant DSD is a profile of the SDMX 2.1 DSD, as described in Section 8.2 and formally expressed as a Schematron rule set in Appendix 8A.
- • ADX Content data messages can be validated using schemas derived from the ADX compliant DSD. Normative XSLT 1.0 stylesheets are provided in Appendices 8B and 8D to generate a W3C XML schema definition and an ISO Schematron schema. Sample ADX compliant schema is provided in the informative Appendix 8G and 8H.

Content Creators and Content Consumers exchange a data payload that conforms to the ADX compliant schemas. A sample ADX compliant data payload is provided in the informative Appendix 8I.

Additional (informative) message constraints for interoperability of ADX data exchange which are outside of the scope of this profile are described in Section 8.4.

8.2 The Data Structure Definition (DSD)

An ADX compliant DSD shall be all of:

i. a well-formed XML 1.0 document;

ii. a valid SDMX 2.1 `mes:Structure` message;

iii. compliant with the additional constraints defined below.

The purpose of profiling the base DSD is to provide guidance regarding which parts of an ADX DSD are fixed by the profile and which parts may be extended to support implementation-specific requirements.

8.2.1 ADX DSD constraints

A Schematron (ISO/IEC 19757-3:2006) schema is provided in Appendix 8A which provides a formal means of validating ADX DSD documents. This Section 8.2.1 provides a description of the constraints within the Schematron schema. In the event of disagreement or ambiguity between the textual and the schema, the Schematron shall be considered normative.

8.2.1.1 Top level structure

The root element of an ADX DSD document shall be a `mes:Structure` element.

The root element shall have a single `mes:Structures` child element.

The `mes:Structures` element shall have the following three child elements:

Element	Description
str:Codelists	Defines sets of data values that are allowed when a data dimension or attribute is constrained to an element within a set.
str:Concepts	Used to produce the dimension attribute names in an ADX data message.
str:DataStructures	Specifies the dimensions and attributes of the data including their syntax, tying them to Codelists and Concepts.

Note: ADX does not preclude the Content Data Structure Creator from including other SDMX elements in the structure message. Elements that are not prescribed in this specification may be ignored by the XSLT transform when generating an ADX schema.

Central to the governance of SDMX structure definitions is the notion of Agency. Each structure element (e.g., Codelist, ConceptScheme, DataStructure) has a maintenance agency identifier associated with it. ADX defines one agency identifier, `'IHE_QRPH'`, which is the maintenance agency for the mandatory concept scheme listed in Appendix 8E. A Content Data Structure Creator which is defining an ADX conformant DSD shall make use of an agency identifier which indicates that it is the maintenance authority for those structures which extend the mandatory ADX-defined structures.

The illustrative sample DSD in Appendix 8F assumes a hypothetical use case where the West African Health Organisation uses the identifier `'WAHO'` to define a DSD.

8.2.1.2 Codelists

The `str:Codelists` element may have any number of `str:Codelist` child elements.

The ADX Profile does not specify the content of code lists; these are specified by the implementing jurisdiction and used by the Content Data Structure Creator. The code lists used for controlling vocabulary of dimensions shall be provided within the DSD in SDMX v2.1 compatible format. The following example, Figure 8.2.1.2-1, shows a hypothetical code list for the West Africa Health Organisation to represent data elements:

```
<str:Codelists>
  <str:Codelist id="CL_DataElements" agencyID="WAHO" version="1.0">
    <com:Name xml:lang="en">West Africa Health Organisation Data elements</com:Name>
    <str:Code id="MAL01">
      <str:Name xml:lang="en">Confirmed Malaria cases</str:Name>
    </str:Code>
    <str:Code id="MAL02">
      <str:Name xml:lang="en">Unconfirmed Malaria cases</str:Name>
```



```
    </str:Code>
    <str:Code id="MAL03">
      <str:Name xml:lang="en">Malaria deaths</str:Name>
    </str:Code>
  </str:Codelist>
    ….
</str:Codelists>
```

Figure 8.2.1.2-1: DSD Codelist example

The full example in Appendix 8C shows additional Codelists for orgUnits, Age Group and Sex representation.

8.2.1.3 Concepts

The `str:Concepts` element shall have one or more `str:ConceptScheme` child elements.

One of these `str:ConceptScheme` elements shall have an `@id` attribute of `ADX_MANDATORY_CONCEPTS` and `@agencyID` attribute of `IHE_QRPH`.

The mandatory `str:ConceptScheme` is shown in Figure 8.2.1.3-1. A Content Data Structure Creator shall not make any changes to the content of the mandatory `str:ConceptScheme` elements.

```
<str:ConceptSchemeid="ADX_MANDATORY_CONCEPTS" agencyID="IHE_QRPH"version="1.0"
isFinal="true">
  <com:Name xml:lang="en">Mandatory concepts defined by IHE ADX profile</com:Name>
  <str:Concept id="dataElement" >
    <com:Name xml:lang="en">Subject of data measure</com:Name>
  </str:Concept>
  <str:Concept id="orgUnit" >
    <com:Name xml:lang="en">Spatial dimension of measure</com:Name>
  </str:Concept>
  <str:Concept id="period" >
    <com:Name xml:lang="en">The temporal dimension of the measure</com:Name>
  </str:Concept>
  <str:Concept id="value" >
    <com:Name xml:lang="en">Observed measure value</com:Name>
```



```
    <str:CoreRepresentation>
      <str:TextFormat textType="Decimal" />
    </str:CoreRepresentation>
  </str:Concept>
</str:ConceptScheme>
```

Figure 8.2.1.3-1: DSD mandatory ConceptScheme

Note that the value concept is the only concept which provides a core representation in the mandatory concept scheme. The enumerated code list representation for orgUnit and data element shall be provided as a `str:LocalRepresentation` of the `str:Dimension` as described below.

A Content Data Structure Creator may provide this concept scheme inline in the DSD content, though it is recommended that it refer to it with an external reference such as in Figure 8.2.1.3-2:

```
<str:ConceptScheme id="ADX_MANDATORY_CONCEPTS" agencyID="IHE_QRPH" version="1.0"
       isExternalReference="true" structureURL="qrph_structures.xml">
  <com:Name xml:lang="en">Mandatory concepts defined by IHE ADX profile</com:Name>
</str:ConceptScheme>
```

Figure 8.2.1.3-2: External reference to ConceptScheme

Content Data Structure Creators may provide additional `str:ConceptScheme` elements to meet requirements of additional disaggregation. Figure 8.2.1.3-3 shows an example of one concept within an implementation-defined concept scheme:

```
<str:ConceptScheme id="ADX_WAHO_CONCEPTS" agencyID="WAHO" version="1.0">
  <com:Name xml:lang="en">Disaggregation concepts used in West Africa Health
     Organisation data warehouse</com:Name>
  <str:Concept id="AGE_GROUP" >
    <com:Name xml:lang="en">Age group</com:Name>
    <str:CoreRepresentation>
      <str:Enumeration>
        <Ref agencyID="WAHO" id="CL_AgeGroup" version="1.0"/>
      </str:Enumeration>
    </str:CoreRepresentation>
  </str:Concept>
```



```
    ...
</str:ConceptScheme>
```

Figure 8.2.1.3-3: DSD Implementation ConceptScheme example

Note that concept schemes other than the mandatory concept scheme may, and should where possible, provide a `str:CoreRepresentation` which refers to a `str:Codelist` within the DSD. Where this is not possible (for example if the Content Data Structure Creator is not the maintenance Agency for the `str:ConceptScheme`), then the representation shall be provided as a `str:LocalRepresentation` of the `str:Dimension` as described in Section 8.2.1.4.1.

8.2.1.4 DataStructures

The `str:DataStructures` element defines the format of the data dimensions and ties them to concepts and codelists. SDMX v2.1 DSD allows the `str:DataStructures` element to have any number of child `str:DataStructure` elements, however the ADX Profile specifies that there will be exactly one.

```
<str:DataStructures>
  <str:DataStructure id="MALARIA" agencyID="WAHO" version="1.0">
    <com:Name xml:lang="en">ADX data structure</com:Name>
    <str:DataStructureComponents>
      <str:DimensionList> ...  </str:DimensionList>
      <str:Group id="OUTER_DIMENSIONS"> ... </str:Group>
      <str:MeasureList> ... </str:MeasureList>
    </str:DataStructureComponents>
  </str:DataStructure>
</str:DataStructures>
```

Figure 8.2.1.4-1: DataStructure example

SDMX allows for the definition of `str:Attributes` in addition to `str:Dimensions`. The xslt transformation which creates a data payload schema ignores any such `str:Attribute` definitions.

8.2.1.4.1 Dimensions

The `str:DataStructure` element shall have exactly one `str:DimensionList` child element. The `str:DimensionList` element shall have all of the following child elements:

- a str:Dimension with a str:ConceptIdentity/Ref child which refers to the concept in the mandatory concept scheme with @id='dataElement'.

- a str:Dimension with a str:ConceptIdentity/Ref child which refers to the concept in the mandatory concept scheme with @id='orgUnit'.
- a str:TimeDimension with a str:ConceptIdentity/Ref child which refers to the concept in the mandatory concept scheme with @id='period'.

The `str:DimensionList` may contain additional `str:Dimension` elements which refer to concepts which are provided by one or more `str:ConceptSchemes` outside of the ADX mandatory `ConceptScheme`. These may be used by a Content Data Structure Creator to provide additional optional disaggregation dimensions in addition to `dataElement, orgUnit` and `period`. The `dataElement` dimension in the sample is shown in the Figure 8.2.1.4.1-1 below:

```
<str:Dimension id="dataElement">
  <str:ConceptIdentity>
    <Ref id="dataElement" maintainableParentID="ADX_MANDATORY_CONCEPTS"
        maintainableParentVersion="1.0" agencyID="IHE_QRPH"/>
  </str:ConceptIdentity>
  <str:LocalRepresentation>
    <str:Enumeration>
      <Ref agencyID="WAHO" id="CL_DataElements" version="1.0"/>
    </str:Enumeration>
  </str:LocalRepresentation>
</str:Dimension>
```

Figure 8.2.1.4.1-1: The data element str:Dimension element

Note that because the data element dimension is linked to a mandatory concept which has no `str:CoreRepresentation`, the Content Data Structure Creator shall provide a `str:LocalRepresentation` which constrains the vocabulary to that provided by a `str:Codelist` in the DSD.

An additional disaggregation dimension is shown in Figure 8.2.1.4.1-2 below:


```
<str:Dimension>
  <str:ConceptIdentity>
    <Ref id="ageGroup" maintainableParentID="ADX_WAHO_CONCEPTS"
       maintainableParentVersion="1.0" agencyID="WAHO"/>
  </str:ConceptIdentity>
</str:Dimension>
```

Figure 8.2.1.4.1-2: An optional disaggregation str:Dimension element

Note that this `str:Dimension` refers to a `str:Concept` which has a `str:CoreRepresentation` so there is no requirement to provide a `str:LocalRepresentation`.

The @id attribute is mandatory:

i. If it is one of the mandatory dimensions, it SHALL have an @id attribute of 'dataElement', 'orgUnit' or 'TIME_PERIOD' respectively;

8.2.1.4.2 The TimeDimension

The `str:TimeDimension` shall have a `str:LocalRepresentation` child element which encloses a `str:TextFormat` element. The `str:TextFormat` shall have a `@textType` attribute with a value of either '`DateTime`' or '`TimeRange`', depending on whether the DSD is to be used for event based or routinely reported data. The choice of value for this attribute determines the xml data type for the `period` attribute which will be used in an ADX data payload.

TextType	Data type of period
dateTime	xs:dateTime
TimeRange	com:TimeRange

The SDMX `com:TimeRange` is defined in the SDMX common namespace and is modelled after (xs:dateTime OR xs:date)/xs:duration. An informative description of the use of this data type is provided in Appendix 8G.

8.2.1.4.3 Dimension grouping

SDMX 2.1 allows dimensions to be grouped so that each dimension does not have to be repeated for each measure value in the payload. ADX mandates that the orgUnit and period dimensions SHALL be attached at the group level and that the data element dimension shall be attached at the data value level. ADX also mandates that there SHALL be a dimension at group level called "dataSet" which has the value of the datastructure identifier (`DataStructure/@id`). Optional

disaggregation dimensions can be attached at either the group or datavalue level. The informative sample ADX data payload in Appendix 8F illustrates the implication of this grouping.

- A `str:Group` child element with an ID of `TIME_PERIOD` that refers as shown in the example below to the `Concept` elements with ID `period`.
- Under `str:TimeDimension` will be a `str:localRepresentation` element, and under this will be a `str:TextFormat` element with attribute `textType` having the value "`ObservationalTimePeriod`". SDMX v2.1 DSD allows many different `ObservationalTimePeriod` formats in data messages, but the ADX Profile further constrains these to the time periods as described in Appendix 8J.
- The `str:DataStructure` element shall have exactly one `str:Group` child element with ID `OUTER_DIMENSIONS`, as shown in the following example (this specifies which data attributes must be present in a `group` element within a data payload conforming to this DSD).

Figure 8.2.1.4.3-1 below shows an example of the `str:Group` element showing the mandatory orgUnit and period dimensions as well as an additional disaggregation dimension with `@id='mechanism'`.


```
<str:Group id="OUTER_DIMENSIONS">
  <str:GroupDimension>
    <str:DimensionReference>
      <Ref id="orgUnit"/>
    </str:DimensionReference>
  </str:GroupDimension>
  <str:GroupDimension>
    <str:DimensionReference>
      <Ref id="TIME_PERIOD"/>
    </str:DimensionReference>
  </str:GroupDimension>
  <str:GroupDimension>
    <str:DimensionReference>
      <Ref id="mechanism"/>
    </str:DimensionReference>
  </str:GroupDimension>
</str:Group>
```

Figure 8.2.1.4.3-1: Example of str:Group element

8.2.1.4.4 The measure dimension

ADX requires that there be a primary measure which is linked to the mandatory value concept as shown in Figure 8.2.1.4.4-1 below.

```
<str:MeasureList>
  <str:PrimaryMeasure>
    <str:ConceptIdentity>
      <Ref id="value" maintainableParentID="ADX_MANDATORY_CONCEPTS"
      maintainableParentVersion="1.0" agencyID="IHE_QRPH"/>
    </str:ConceptIdentity>
  </str:PrimaryMeasure>
</str:MeasureList>
```

Figure 8.2.1.4.4-1: Example of str:Group element

The Representation for the value concept is provided as a CoreRepresentation in the ConceptScheme. Implementers may not override this with a LocalRepresentation.

8.2.2 Constraining optional disaggregation

Optional disaggregation dimensions refer to additional `str:Dimension` elements in the `str:DimensionList` which refer to concepts which are provided by one or more `str:ConceptSchemes` outside of the ADX mandatory `ConceptScheme`. An example is provided above in Figure 8.2.1.4.1-2 for an ageGroup dimension.

ADX allows for data rows within the same dataset to be disaggregated differently depending on the data element. Content Data Structure Creators **may** indicate the required disaggregations for individual data elements by making use of Annotations in the dataElement code list.

```
<str:Code id="MAL04">
  <com:Annotations>
    <com:Annotation id="Disaggregation">
      <com:AnnotationText>ageGroup</com:AnnotationText>
    </com:Annotation>
    <com:Annotation id="Disaggregation">
      <com:AnnotationText>sex</com:AnnotationText>
    </com:Annotation>
  </com:Annotations>
  <com:Name xml:lang="en">Unconfirmed Malaria deaths</com:Name>
</str:Code>
```

Figure 8.2.2: Example of Annotation of Data element codes

Note in the example above that

(i) the `Annotation/@id` value of 'Disaggregation' is mandatory to distinguish the interpretation of this `Annotation`. The SDMX DSD allows for arbitrary additional `Annotation` elements.

(ii) the content of the `AnnotationText` element is a reference to a `Dimension/@id` as described in Section 8.2.1.4.1.

8.3 Generating the schema for an ADX data payload

ADX data within a particular jurisdiction shall be constrained by schemas generated from the ADX DSD. That schema can be generated by applying the XSLT transformations given in Appendices 8B and 8D to the ADX compliant DSD stream.

This transformation assumes that all structure elements appear inline in the DSD XML document. Where external references are used, the DSD may be pre-processed, for example using the transformation provided in Appendix 8C.

8.4 ADX Message Exchange Constraints (Informative)

There are implicit assumptions regarding the sharing of additional metadata between Content Creators and Content Consumers which are not covered within the scope of the ADX Profile. In order to ensure semantic validity of the exchanged data, the constraints described below shall be specified using processes not defined by this profile:

- The data elements to be reported for particular orgUnits. For example, reporting 'Number of lab Tests performed' from a health facility which did not perform laboratory tests is not constrained by ADX.
- The Value type (real or integer) expected for individual data elements.
- Which period types are appropriate for which data elements.

The preceding is not an exhaustive list. Other business rules that may affect interoperability may need to be established such as bounds on data values, bounds on date ranges, relationships between different data elements (e.g., the sum of these data elements cannot be greater than the sum of those data elements, etc.).

Appendices to Section 8

Appendix 8A – (Normative) Schematron constraining ADX/DSD

An electronic copy of the file is available at ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/ADX/schema/dsd_validation.sch. In case of any difference between the electronic copy and the appendix below, this appendix SHALL be considered normative.

```
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://purl.oclc.org/dsdl/schematron"
    queryBinding="xslt"
    xmlns:h="http://www.w3.org/1999/xhtml">

    <title>Validation of ADX Data Structure Definition</title>

    <p>An ADX profiled Data Structure Definition (DSD) is
        (i)   a well formed XML document and
        (ii)  a valid SDMX 2.1 Strucure message and
        (iii) is further subject to additional constraints expressed in
        this schematron schema.
    </p>

    <p>The following are namespaces defined in SDMX 2.1 which are used in
    an ADX profiled DSD</p>
    <ns prefix="mes" uri="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/message" />
    <ns prefix="str" uri="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/structure" />
    <ns prefix="com" uri="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/common" />

    <pattern >
        <title>Testing that Structures are all present</title>
        <rule context="mes:Structure">
            <assert test="count(mes:Structures)=1">
                There shall be a single mes:Structures element in the message.
            </assert>
        </rule>

        <rule context="mes:Structures" >
            <assert test="count(str:Codelists)=1">
                There shall be a single Codelists element.
            </assert>
```



```
            <assert test="count(str:Concepts)=1">
                There shall be a single Concepts element.
            </assert>

            <assert test="count(str:DataStructures/str:DataStructure)=1">
                There shall be a single DataStructure element.
            </assert>

            <let name="ADX_Concepts" value="str:Concepts/str:ConceptScheme[
                @id='ADX_MANDATORY_CONCEPTS' and @agencyID='IHE_QRPH']"/>

            <assert
                test="count($ADX_Concepts)=1">
                There shall be a ConceptScheme with @id='ADX_MANDATORY_CONCEPTS' and
                @agencyID='IHE_QRPH'.
            </assert>

        </rule>

    </pattern>

    <pattern>
        <title>Testing DataStructureComponents</title>

        <let name="components" value="str:DataStructureComponents" />
        <let name="dimensions" value="$components/str:DimensionList"/>

        <rule context="str:DataStructure/str:DataStructureComponents">

            <assert test="count(str:Group[@id='OUTER_DIMENSIONS'])=1">
                There shall be a Group with @id='OUTER_DIMENSIONS'
            </assert>

        </rule>

        <p>The dimension list shall include the mandatory dimensions</p>
        <rule context="str:DimensionList">
```



```
            <let name="dataElementDimension" value="str:Dimension[@id='dataElement']" />
            <let name="orgUnitDimension"     value="str:Dimension[@id='orgUnit']" />
            <let name="periodDimension"      value="str:TimeDimension[@id='TIME_PERIOD']" />

            <assert test="count($dataElementDimension)=1">
                There shall be a dimension with @id='dataElement'.
            </assert>

            <assert test="count($orgUnitDimension)=1">
                There shall be a dimension with @id='orgUnit'.
            </assert>

            <assert test="count($periodDimension)=1">
                There shall be a TimeDimension with @id='TIME_PERIOD'.
            </assert>

        </rule>

        <p>
            The dataElement dimension must be linked to the mandatory 'dataElement' concept and
            provide a LocalRepresentation.
        </p>
        <rule context="str:Dimension[@id='dataElement']">

            <assert
                test="str:ConceptIdentity/Ref/@id='dataElement'">
                @id of dataElement concept reference must be 'dataElement'.
            </assert>

            <assert
                test="str:ConceptIdentity/Ref/@maintainableParentID='ADX_MANDATORY_CONCEPTS'">
                @maintainableParentID of dataElement dimension concept reference
                must be 'ADX_MANDATORY_CONCEPTS'.
            </assert>

            <assert test="count(str:LocalRepresentation)=1">
                dataElement dimension must provide LocalRepresentation.
            </assert>
```



```
        </rule>

        <p>
            The orgUnit dimension must be linked to the mandatory 'orgUnit' concept and
            provide a LocalRepresentation.
        </p>
        <rule context="str:Dimension[@id='orgUnit']">

            <assert
                test="str:ConceptIdentity/Ref/@id='orgUnit'">
                @id of orgUnit concept reference must be 'orgUnit'.
            </assert>

            <assert
                test="str:ConceptIdentity/Ref/@maintainableParentID='ADX_MANDATORY_CONCEPTS'">
                @maintainableParentID of orgUnit dimensions concept reference
                must be 'ADX_MANDATORY_CONCEPTS'.
            </assert>

            <assert test="count(str:LocalRepresentation)=1">
                orgUnit dimension must provide LocalRepresentation.
            </assert>

        </rule>

        <p>
            The TimeDimension must be linked to the mandatory 'period' concept and
            provide a LocalRepresentation.
        </p>
        <rule context="str:TimeDimension">
            <assert
                test="str:ConceptIdentity/Ref/@id='period'">
                @id of orgUnit concept reference must be 'period'.
            </assert>

            <assert
                test="str:ConceptIdentity/Ref/@maintainableParentID='ADX_MANDATORY_CONCEPTS'">
```



```
                @maintainableParentID of period dimension concept reference
                must be 'ADX_MANDATORY_CONCEPTS'.
            </assert>

            <let name="periodFormat" value="str:LocalRepresentation/str:TextFormat/@textType"/>

            <assert test="$periodFormat='TimeRange' or $periodFormat='DateTime'">
                The time period format must be either 'TimeRange' or 'DateTime'.
            </assert>

        </rule>

        <p>
            The orgUnit and period dimensions must be attached to the outer group.  The dataElement
            dimesnion may not be in the outer group.
        </p>
        <rule context="str:Group[@id='OUTER_DIMENSIONS']">

            <assert test="count(str:GroupDimension/str:DimensionReference/Ref[@id='orgUnit'])=1" >
                The orgUnit dimension must appear once in the 'OUTER_DIMENSIONS' group.
            </assert>

            <assert test="count(str:GroupDimension/str:DimensionReference/Ref[@id='TIME_PERIOD'])=1" >
                The period dimension must appear once in the 'OUTER_DIMENSIONS' group.
            </assert>

            <assert test="count(str:GroupDimension/str:DimensionReference/Ref[@id='dataElement'])=0" >
                The period dimension must appear once in the 'OUTER_DIMENSIONS' group.
            </assert>

        </rule>

        <p>
            The PrimaryMeasure must be linked to the mandatory 'value' concept.
        </p>
        <rule context="str:PrimaryMeasure">

            <assert
```



```
                test="str:ConceptIdentity/Ref/@id='value'">
                @id of PrimaryMeasure concept reference must be 'value'.
            </assert>

            <assert
                test="str:ConceptIdentity/Ref/@maintainableParentID='ADX_MANDATORY_CONCEPTS'">
                @maintainableParentID of PrimaryMeasure concept reference
                must be 'ADX_MANDATORY_CONCEPTS'.
            </assert>

        </rule>

    </pattern>

</schema>
```

Appendix 8B – (Normative) XSLT to generate ADX/XSD schema from DSD

An electronic copy of the file is available at ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/ADX/xslt/dsd2adx2.xsl. In case of any difference between the electronic copy and the appendix below, this appendix SHALL be considered normative.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:str="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/structure"
    exclude-result-prefixes="xs" version="1.0">

    <xsl:output encoding="UTF-8" xml:space="preserve" method="xml" indent="yes"/>

<!--
        This stylesheet is a normative part of the ADX profile (urn:ihe:qrph:adx:2015)

        When applied to an ADX conformant SDMX Data Structure Definition it emits
        a W3C XML Schema document sutiable for validation of ADX data payloads.
-->

<!-- ============================================================================
     Variable declarations
     ============================================================================ -->
    <!-- Dimension nodes -->
    <xsl:variable name="dimensions" select="//str:Dimension"/>

    <!-- Jurisdiction specific Dimensions at group level -->
    <xsl:variable name="outerDimensions"
        select="$dimensions[//str:Group[@id='OUTER_DIMENSIONS']/descendant::Ref/@id = @id
          and @id != 'orgUnit']"/>

    <!-- Jurisdiction specific Dimensions at dataValue level -->
    <xsl:variable name="innerDimensions"
        select="$dimensions[not (//str:Group[@id='OUTER_DIMENSIONS']/descendant::Ref/@id = @id )
          and str:ConceptIdentity/Ref/@id != 'dataElement']"/>

    <!-- Mandatory dimensions -->
```



```
    <xsl:variable name="orgUnitDimension" select="$dimensions[@id='orgUnit']" />
    <xsl:variable name="dataElementDimension" select="$dimensions[@id='dataElement']" />

    <!-- Reference to the orgUnit code list -->
    <xsl:variable
        name="orgUnitCLRef"
        select="$orgUnitDimension/str:LocalRepresentation/str:Enumeration/Ref" />

    <!-- Construction of the orgUnit type name -->
    <xsl:variable name="orgUnitType"
        select="concat($orgUnitCLRef/@id,'_',$orgUnitCLRef/@agencyID,'_',
          $orgUnitCLRef/@version,'_Type')" />

    <!-- Reference to the dataElement code list -->
    <xsl:variable
        name="dataElementCLRef"
        select="$dataElementDimension/str:LocalRepresentation/str:Enumeration/Ref" />

    <!-- Construction of the dataElement type name -->
    <xsl:variable name="dataElementType"
        select="concat($dataElementCLRef/@id,'_',
          $dataElementCLRef/@agencyID,'_',$dataElementCLRef/@version,'_Type')" />

    <!-- The dataSet identifier -->
    <xsl:variable name="dataSetId"
        select="//str:DataStructure/@id"/>


<!-- ============================================================================
     Root Template Match
     ============================================================================ -->

    <xsl:template match="/">

        <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
            xmlns="urn:ihe:qrph:adx:2015"
            xmlns:common="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/common"
            targetNamespace="urn:ihe:qrph:adx:2015" elementFormDefault="qualified">
```



```
            <!--  Copyright notice -->

            <xs:annotation>
                <xs:documentation> This is an example of a tightly constrained schema which should
                    validate an adx data document which has been formed in compliance with the
                    sample SDMX DSD. </xs:documentation>
            </xs:annotation>

            <xs:import namespace="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/common"
                schemaLocation="sdmx/SDMXCommon.xsd"/>

            <!-- generate enumerated types for dimensions -->
            <xsl:apply-templates select="//str:Codelist"/>

            <!-- generate dateTime type -->
            <xsl:apply-templates select="//str:TimeDimension" />

            <!-- generate complex types -->
            <xsl:call-template name="adx" />
            <xsl:call-template name="group" />
            <xsl:call-template name="dataValue" />

            <xs:element name="adx" type="adxType"/>

        </xs:schema>
    </xsl:template>

<!-- ============================================================================
     Complex Element types

     1. ADX Root element type
     ============================================================================ -->
    <xsl:template name="adx">
        <xs:complexType name="adxType">
            <xs:sequence maxOccurs="unbounded">
                <xs:element name="group" type="groupType"/>
            </xs:sequence>
```

```
            <xs:attribute name="exported" use="required" type="xs:dateTime"/>
            <xs:anyAttribute processContents="skip"/>
        </xs:complexType>
    </xsl:template>

<!-- ========================================================================
     2.  groupType
     ======================================================================== -->
     <xsl:template name="group">
        <xs:complexType name="groupType">
            <xs:sequence maxOccurs="unbounded">
                <xs:element name="dataValue" type="DataValueType"/>
            </xs:sequence>
            <xs:attribute name="dataSet" use="required" type="xs:string" fixed="{$dataSetId}"/>
            <xs:attribute name="orgUnit" use="required" type="{$orgUnitType}"/>
            <xs:attribute name="period" use="required" type="periodType"/>

            <xsl:apply-templates select="$outerDimensions" />

            <xs:anyAttribute processContents="skip"/>
        </xs:complexType>
     </xsl:template>

<!-- ========================================================================
     3.  dataValueType
     ======================================================================== -->
    <xsl:template name="dataValue">
        <xs:complexType name="DataValueType">
                <xs:sequence maxOccurs="1" minOccurs="0">
                    <xs:element name="annotation" />
                </xs:sequence>

                <xs:attribute name="dataElement" use="required" type="{$dataElementType}"/>
                <xs:attribute name="value" use="required" type="xs:decimal"/>

                <xsl:apply-templates select="$innerDimensions" />

                <xs:anyAttribute processContents="skip"/>
```



```
            </xs:complexType>
    </xsl:template>

<!-- ========================================================================
     Type restrictions derived from SDMX DSD Codelists
     ======================================================================== -->
    <xsl:template match="str:Codelist">
        <xsl:variable name="type" select="concat(@id,'_',@agencyID,'_',@version,'_Type')"/>
        <xs:simpleType name="{$type}">
            <xs:restriction base="xs:token">
                <xsl:for-each select="str:Code">
                    <xs:enumeration value="{@id}"/>
                </xsl:for-each>
            </xs:restriction>
        </xs:simpleType>
    </xsl:template>

<!-- ========================================================================
     Time dimension type
     ======================================================================== -->
    <xsl:template match="str:TimeDimension">
        <xsl:variable name="timeFormat" select="str:LocalRepresentation/str:TextFormat"/>
        <xs:simpleType name="periodType">
            <xsl:choose>
                <xsl:when test="$timeFormat/@textType='DateTime'">
                    <xs:restriction base="xs:dateTime"/>
                </xsl:when>
                <xsl:when test="$timeFormat/@textType='TimeRange'">
                    <xs:restriction base="common:TimeRangeType"/>
                </xsl:when>
                <xsl:otherwise>
                    <xsl:message>
                        Only SDMX DateTime and TimeRange are supported types
                    </xsl:message>
                </xsl:otherwise>
            </xsl:choose>
        </xs:simpleType>
```

```
    </xsl:template>

<!-- ========================================================================
     Produce attributes for dimension
     ======================================================================== -->
    <xsl:template match="str:Dimension">
        <xsl:variable name="conceptID" select="str:ConceptIdentity/Ref/@id"/>
        <xsl:variable name="conceptSchemeID" select="str:ConceptIdentity/Ref/@maintainableParentID"/>

        <xsl:choose>
            <!-- if there is a LocalRepresentation, use that -->
            <xsl:when test="str:LocalRepresentation">
                <xsl:variable name="codelist" select="str:LocalRepresentation/str:Enumeration/Ref"/>
                <xsl:variable name="type"
                    select="concat($codelist/@id,'_',$codelist/@agencyID,'_',$codelist/@version,'_Type')"/>
                <xs:attribute name="{$conceptID}" type="{$type}" use="optional"/>
            </xsl:when>
            <!-- otherwise lookup the CoreRepresentation for the Concept -->
            <xsl:otherwise>
                <xsl:variable name="concept"
                    select="//str:ConceptScheme[
                    @id=$conceptSchemeID]/str:Concept[@id=$conceptID]"/>
                <xsl:variable name="codelist"
                    select="$concept/str:CoreRepresentation/str:Enumeration/Ref"/>
                <xsl:variable name="type"
                    select="concat($codelist/@id,'_',$codelist/@agencyID,'_',$codelist/@version,'_Type')"/>
                <xs:attribute name="{$conceptID}" type="{$type}" use="optional"/>
            </xsl:otherwise>
        </xsl:choose>
    </xsl:template>

</xsl:stylesheet>
```


Appendix 8C – (Informative) DSD pre-processor to resolve external references

An electronic copy of the file is available at ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/ADX/xslt/dsd_preprocess.xsl. In case of any difference between the electronic copy and the appendix below, this appendix SHALL be considered normative.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:mes="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/message"

    exclude-result-prefixes="xs"
    version="1.0">

<!--
     Stylesheet to pre-process an SDMX 2.1 DSD to pull in external references
-->
   <xsl:strip-space elements="*"/>
   <xsl:output method="xml" indent="yes" />

   <!-- The default action is to just copy each node to output - identity template -->
   <xsl:template match="/ | @* | node()">
         <xsl:copy>
               <xsl:apply-templates select="@* | node()" />
         </xsl:copy>
   </xsl:template>

<!-- Copy in external references where indicated -->
    <xsl:template match="*[@isExternalReference = 'true' and @structureURL]">
        <xsl:message>
            External reference for <xsl:value-of select="name()"/> fetched from <xsl:value-of
select="@structureURL"/>
        </xsl:message>
        <xsl:variable name="elementName" select="name()"/>
        <xsl:variable name="id" select="@id"/>
        <xsl:variable name="agencyID" select="@agencyID"/>
        <xsl:variable name="version" select="@version"/>
        <xsl:copy-of select="document(@structureURL)//*[name()=$elementName
```



```
            and @id=$id and @agencyID=$agencyID and @version=$version]" />
    </xsl:template>
</xsl:stylesheet>
```

Appendix 8D – (Normative) XSLT for generating ADX validation Schema

An electronic copy of the file is available at ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/ADX/xslt/dsd2schematron.xsl. In case of any difference between the electronic copy and the appendix below, this appendix SHALL be considered normative.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:str="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/structure"
    xmlns:com="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/common"
    xmlns:sch="http://purl.oclc.org/dsdl/schematron"
    exclude-result-prefixes="xs" version="1.0">

    <xsl:output encoding="UTF-8" xml:space="preserve" method="xml" indent="yes"/>

<!--
        This stylesheet is a normative part of the ADX profile (urn:ihe:qrph:adx:2015)

        When applied to an ADX conformant SDMX Data Structure Definition it emits
        a W3C XML Schema document suitable for validation of ADX data payloads.
-->

<!-- ================================================================================
     Variable declarations
     ================================================================ -->

    <!-- The dataSet identifier -->
    <xsl:variable name="dataSetId"
        select="//str:DataStructure/@id"/>

    <xsl:variable name="customConcepts"
        select="//str:ConceptScheme[not(@id='ADX_MANDATORY_CONCEPTS')]/str:Concept/@id" />

<!-- ================================================================================
     Root Template Match
     ================================================================================ -->

```



```
    <xsl:template match="/">

        <sch:schema >
            <sch:ns uri="urn:ihe:qrph:adx:2015" prefix="adx"/>
            <xsl:apply-templates select="//str:Codelist[@id='CL_DataElements']"/>
        </sch:schema>
    </xsl:template>

    <xsl:template match="str:Codelist">
        <sch:pattern >
            <sch:title>Validating ADX aggregations</sch:title>
            <sch:p> The ADX xsd schema validates that correct codes are used in code lists. Applying
                this set of rules in addition ensures that datavalues are reported with the correct
                disaggregations. </sch:p>

            <xsl:apply-templates select="str:Code"/>

        </sch:pattern>
    </xsl:template>

    <xsl:template match="str:Code">
        <xsl:variable name="apos">'</xsl:variable>
        <xsl:variable name="code" select="@id"/>
        <xsl:variable name="context"
            select="concat('adx:dataValue[@dataElement=',$apos,$code,$apos,']')"/>
        <xsl:variable name="disaggs"
            select="com:Annotations/com:Annotation[@id='Disaggregation']/com:AnnotationText"/>

        <sch:rule context="{$context}">

            <xsl:for-each select="$customConcepts">
                <xsl:choose>
                    <xsl:when test=".=$disaggs">
                        <xsl:variable name="test" select="concat('@',.)"/>
                        <sch:assert test="{$test}">
                            <xsl:value-of select="concat('@',.,' must be present on element ',$code)"/>
                        </sch:assert>
                    </xsl:when>
```



```
                    <xsl:otherwise>
                        <xsl:variable name="test" select="concat('@',.)"/>
                        <sch:assert test="not({$test})">
                            <xsl:value-of select="concat('@',.,' is not permitted on element ',$code)"/>
                        </sch:assert>
                    </xsl:otherwise>
                </xsl:choose>
            </xsl:for-each>
        </sch:rule>

    </xsl:template>
</xsl:stylesheet>
```


Appendix 8E – (Normative) ADX Mandatory ConceptScheme

An electronic copy of the file is available at ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/ADX/sample/qrph_structures.xml.
In case of any difference between the electronic copy and the appendix below, this appendix SHALL be considered normative.

```
<?xml version="1.0" encoding="UTF-8"?>
<mes:Structure xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/message
../schema/sdmx/SDMXMessage.xsd"
    xmlns:com="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/common"
    xmlns:str="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/structure"
    xmlns:mes="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/message">
    <mes:Header>
        <mes:ID>ADX_STRUCTURES</mes:ID>
        <mes:Test>false</mes:Test>
        <mes:Prepared>2015-05-01T14:30:47.000Z</mes:Prepared>
        <mes:Sender id="IHE_QRPH">
            <com:Name xml:lang="en">QRPH Technical Committee</com:Name>
        </mes:Sender>
    </mes:Header>

    <mes:Structures>

        <str:OrganisationSchemes>
            <str:AgencyScheme id="AGENCIES" agencyID="IHE_QRPH">
                <com:Name>List of AGENCY identifiers maintained by IHE QRPH</com:Name>
                <str:Agency id="IHE_QRPH" >
                    <com:Name>IHE Quality Research and Public Health Technical Committee</com:Name>
                </str:Agency>
            </str:AgencyScheme>
        </str:OrganisationSchemes>

        <str:Concepts>
            <!-- ADX Mandatory concepts -->
            <str:ConceptScheme id="ADX_MANDATORY_CONCEPTS" agencyID="IHE_QRPH" version="1.0">
                <com:Name xml:lang="en">Mandatory concepts defined by IHE ADX profile</com:Name>
```



```
                <!--  There is no default representation for these concepts.  An Agency which makes
                      use of these to construct an ADX DataStructure has to provide a local representation
-->
                <str:Concept id="dataElement">
                    <com:Name xml:lang="en">Subject of the data measure</com:Name>
                </str:Concept>
                <str:Concept id="orgUnit">
                    <com:Name xml:lang="en">Spatial dimension of the measure</com:Name>
                </str:Concept>
                <str:Concept id="period">
                    <com:Name xml:lang="en">Temporal dimension of the measure</com:Name>
                </str:Concept>

                <!-- The value concept is represented by an integer or real number (xsd:decimal)-->
                <str:Concept id="value">
                    <com:Name xml:lang="en">Value of the data measure</com:Name>
                    <str:CoreRepresentation>
                        <str:TextFormat textType="Decimal"/>
                    </str:CoreRepresentation>
                </str:Concept>
            </str:ConceptScheme>
        </str:Concepts>
    </mes:Structures>
</mes:Structure>
```


Appendix 8F – (Informative) Sample ADX DSD

An electronic copy of the file is available at ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/ADX/sample/adx_sample_dsd2.xml. In case of any difference between the electronic copy and the appendix below, this appendix SHALL be considered normative.

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-model href="../schema/dsd_validation.sch" type="application/xml"
schematypens="http://purl.oclc.org/dsdl/schematron"?>
<mes:Structure xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/message
../schema/sdmx/SDMXMessage.xsd"
 xmlns:com="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/common"
 xmlns:str="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/structure"
 xmlns:mes="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/message">
  <mes:Header>
    <mes:ID>DSD_ADX</mes:ID>
    <mes:Test>true</mes:Test>
    <mes:Prepared>2015-01-13T14:30:47.000Z</mes:Prepared>
    <mes:Sender id="Bob">
      <com:Name xml:lang="en">Bob Jolliffe</com:Name>
    </mes:Sender>
  </mes:Header>

  <mes:Structures>

    <str:Codelists>
      <!-- These are used to constrain vocabulary of dimension values -->

      <str:Codelist id="CL_OrgUnits" agencyID="WAHO" version="1.0" >
        <com:Name xml:lang="en">Reporting health clinics</com:Name>
        <str:Code id="342">
          <com:Name xml:lang="en">Bob's clinic</com:Name>
        </str:Code>
        <str:Code id="343">
          <com:Name xml:lang="en">Jim's clinic</com:Name>
        </str:Code>
      </str:Codelist>
```



```
<str:Codelist id="CL_DataElements" agencyID="WAHO" version="1.0">
  <com:Name xml:lang="en">WAHO Data elements</com:Name>
  <str:Code id="MAL01">
    <com:Name xml:lang="en">Confirmed Malaria cases</com:Name>
  </str:Code>
  <str:Code id="MAL02">
    <com:Name xml:lang="en">Unconfirmed Malaria cases</com:Name>
  </str:Code>
  <str:Code id="MAL03">
    <com:Name xml:lang="en">Confirmed Malaria deaths</com:Name>
  </str:Code>
  <str:Code id="MAL04">
    <com:Annotations>
      <com:Annotation id="Disaggregation">
        <com:AnnotationText>ageGroup</com:AnnotationText>
      </com:Annotation>
      <com:Annotation id="Disaggregation">
        <com:AnnotationText>sex</com:AnnotationText>
      </com:Annotation>
    </com:Annotations>
    <com:Name xml:lang="en">Unconfirmed Malaria deaths</com:Name>
  </str:Code>
</str:Codelist>

<str:Codelist id="CL_AgeGroup" agencyID="WAHO" version="1.0">
  <com:Name xml:lang="en">WAHO Age Groups</com:Name>
  <str:Code id="under5">
    <com:Name xml:lang="en">Under 5 years of age</com:Name>
  </str:Code>
  <str:Code id="5andOver">
    <com:Name xml:lang="en">5 Years and over</com:Name>
  </str:Code>
</str:Codelist>

<str:Codelist id="CL_Sex" agencyID="WAHO" version="1.0">
  <com:Name xml:lang="en">WAHO Sex categories</com:Name>
  <str:Code id="M">
```



```
          <com:Name xml:lang="en">Male</com:Name>
        </str:Code>
        <str:Code id="F">
          <com:Name xml:lang="en">Female</com:Name>
        </str:Code>
      </str:Codelist>

      <str:Codelist id="CL_Mechanism" agencyID="WAHO" version="1.0">
        <com:Name xml:lang="en">Funding mechanisms</com:Name>
        <str:Code id="PEPFAR">
          <com:Name xml:lang="en">PEPFAR</com:Name>
        </str:Code>
        <str:Code id="GFATM">
          <com:Name xml:lang="en">Global Fund</com:Name>
        </str:Code>
        <str:Code id="OTHER">
          <com:Name xml:lang="en">Other</com:Name>
        </str:Code>
      </str:Codelist>

    </str:Codelists>

    <str:Concepts>
      <!-- These are named concepts which are referred to in the definition of
        data dimensions.  They can be constrained through the use of codelists
        or text format patterns using a CoreRepresentation
      -->

      <str:ConceptScheme id="ADX_MANDATORY_CONCEPTS" agencyID="IHE_QRPH" version="1.0"
        isExternalReference="true" structureURL="qrph_structures.xml" >
        <com:Name xml:lang="en">Mandatory concepts defined by IHE ADX profile</com:Name>
      </str:ConceptScheme>

      <str:ConceptScheme id="ADX_WAHO_CONCEPTS" agencyID="WAHO" version="1.0" >
        <!-- These concepts are user-defined by an Agency other than IHE_QRPH.
           A concept is declared together with it's core represenation reference
           to a codelist
        -->
```



```
        <com:Name xml:lang="en">Disaggregation concepts used in WAHO data warehouse</com:Name>
        <str:Concept id="ageGroup">
          <com:Name xml:lang="en">Age group</com:Name>
          <str:CoreRepresentation>
            <str:Enumeration>
              <Ref agencyID="WAHO" id="CL_AgeGroup" version="1.0"/>
            </str:Enumeration>
          </str:CoreRepresentation>
        </str:Concept>
        <str:Concept id="sex">
          <com:Name xml:lang="en">Sex</com:Name>
          <str:CoreRepresentation>
            <str:Enumeration>
              <Ref agencyID="WAHO" id="CL_Sex" version="1.0"/>
            </str:Enumeration>
          </str:CoreRepresentation>
        </str:Concept>
        <str:Concept id="mechanism">
          <com:Name xml:lang="en">Funding Mechanism</com:Name>
          <str:CoreRepresentation>
            <str:Enumeration>
              <Ref agencyID="WAHO" id="CL_Mechanism" version="1.0"/>
            </str:Enumeration>
          </str:CoreRepresentation>
        </str:Concept>
      </str:ConceptScheme>
    </str:Concepts>

    <str:DataStructures>
      <!-- An implementing Aency other than IHE_QRPH defines the DatStructure to
        use for ADX data payloads in that jurisdiction.  The dataStructure must
        have Dimensions which correspond to the core concepts and provide a
        local representation which links those dimensions to codelists.  -->

        <str:DataStructure id="ADX" agencyID="WAHO" version="1.0" >
        <com:Name xml:lang="en">ADX data structure</com:Name>

        <str:DataStructureComponents>
```



```
        <str:DimensionList>

          <str:Dimension id="dataElement" >
            <str:ConceptIdentity>
              <Ref id="dataElement" maintainableParentID="ADX_MANDATORY_CONCEPTS"
                maintainableParentVersion="1.0" agencyID="IHE_QRPH"/>
            </str:ConceptIdentity>
            <str:LocalRepresentation>
              <str:Enumeration>
                <Ref agencyID="WAHO" id="CL_DataElements" version="1.0"/>
              </str:Enumeration>
            </str:LocalRepresentation>
          </str:Dimension>

          <str:Dimension id="orgUnit">
            <str:ConceptIdentity>
              <Ref id="orgUnit" maintainableParentID="ADX_MANDATORY_CONCEPTS"
                maintainableParentVersion="1.0" agencyID="IHE_QRPH"/>
            </str:ConceptIdentity>
            <str:LocalRepresentation>
              <str:Enumeration>
                <Ref agencyID="WAHO" id="CL_OrgUnits" version="1.0"/>
              </str:Enumeration>
            </str:LocalRepresentation>
          </str:Dimension>

          <str:Dimension>
            <str:ConceptIdentity>
              <Ref id="sex" maintainableParentID="ADX_WAHO_CONCEPTS"
                maintainableParentVersion="1.0" agencyID="WAHO"/>
            </str:ConceptIdentity>
          </str:Dimension>

          <str:Dimension>
            <str:ConceptIdentity>
              <Ref id="ageGroup" maintainableParentID="ADX_WAHO_CONCEPTS"
                maintainableParentVersion="1.0" agencyID="WAHO"/>
```



```
            </str:ConceptIdentity>
          </str:Dimension>

          <str:Dimension id="mechanism">
            <str:ConceptIdentity>
              <Ref id="mechanism" maintainableParentID="ADX_WAHO_CONCEPTS"
                maintainableParentVersion="1.0" agencyID="WAHO"/>
            </str:ConceptIdentity>
          </str:Dimension>

          <str:TimeDimension id="TIME_PERIOD">
            <str:ConceptIdentity>
              <Ref id="period" maintainableParentID="ADX_MANDATORY_CONCEPTS"
                maintainableParentVersion="1.0" agencyID="IHE_QRPH"/>
            </str:ConceptIdentity>
            <str:LocalRepresentation>
              <!-- An ADX DSD allows either an SDMX 'TimeRange' or 'DateTime'
                representation of the time dimension -->
              <str:TextFormat textType="TimeRange"/>
            </str:LocalRepresentation>
          </str:TimeDimension>
        </str:DimensionList>

        <!-- These dimensions are identified as being used as outer dimensions
          of the data.  They must include orgUnit and period.  -->
        <str:Group id="OUTER_DIMENSIONS">
          <str:GroupDimension>
            <str:DimensionReference>
              <Ref id="orgUnit"/>
            </str:DimensionReference>
          </str:GroupDimension>
          <str:GroupDimension>
            <str:DimensionReference>
              <Ref id="TIME_PERIOD"/>
            </str:DimensionReference>
          </str:GroupDimension>
          <str:GroupDimension>
            <str:DimensionReference>
```



```
                <Ref id="mechanism"/>
              </str:DimensionReference>
            </str:GroupDimension>
          </str:Group>

          <!-- ADX requires that there be a primary measure which is linked to
            the value concept.  -->
          <str:MeasureList>
            <str:PrimaryMeasure>
              <str:ConceptIdentity>
                <Ref id="value" maintainableParentID="ADX_MANDATORY_CONCEPTS"
                  maintainableParentVersion="1.0" agencyID="IHE"/>
              </str:ConceptIdentity>
            </str:PrimaryMeasure>
          </str:MeasureList>

        </str:DataStructureComponents>

      </str:DataStructure>
    </str:DataStructures>
  </mes:Structures>
</mes:Structure>
```


Appendix 8G – (Informative) Generated sample ADX data schema

An electronic copy of the file is available at ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/ADX/sample/adx_sample_generated.xsd. In case of any difference between the electronic copy and the appendix below, this appendix SHALL be considered normative.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns="urn:ihe:qrph:adx:2015"
xmlns:common="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/common"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:str="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/structure"
targetNamespace="urn:ihe:qrph:adx:2015" elementFormDefault="qualified">
  <xs:annotation>
    <xs:documentation> This is an example of a tightly constrained schema which should
                  validate an adx data document which has been formed in compliance with the
                  sample SDMX DSD. </xs:documentation>
  </xs:annotation>
  <xs:import namespace="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/common"
schemaLocation="sdmx/SDMXCommon.xsd"/>
  <xs:simpleType name="CL_OrgUnits_WAHO_1.0_Type">
    <xs:restriction base="xs:token">
      <xs:enumeration value="342"/>
      <xs:enumeration value="343"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="CL_DataElements_WAHO_1.0_Type">
    <xs:restriction base="xs:token">
      <xs:enumeration value="MAL01"/>
      <xs:enumeration value="MAL02"/>
      <xs:enumeration value="MAL03"/>
      <xs:enumeration value="MAL04"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="CL_AgeGroup_WAHO_1.0_Type">
    <xs:restriction base="xs:token">
      <xs:enumeration value="under5"/>
```



```
      <xs:enumeration value="5andOver"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="CL_Sex_WAHO_1.0_Type">
    <xs:restriction base="xs:token">
      <xs:enumeration value="M"/>
      <xs:enumeration value="F"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="CL_Mechanism_WAHO_1.0_Type">
    <xs:restriction base="xs:token">
      <xs:enumeration value="PEPFAR"/>
      <xs:enumeration value="GFATM"/>
      <xs:enumeration value="OTHER"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="periodType">
    <xs:restriction base="common:TimeRangeType"/>
  </xs:simpleType>
  <xs:complexType name="adxType">
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="group" type="groupType"/>
    </xs:sequence>
    <xs:attribute name="exported" use="required" type="xs:dateTime"/>
    <xs:anyAttribute processContents="skip"/>
  </xs:complexType>
  <xs:complexType name="groupType">
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="dataValue" type="DataValueType"/>
    </xs:sequence>
    <xs:attribute name="dataSet" use="required" type="xs:string" fixed="MALARIA"/>
    <xs:attribute name="orgUnit" use="required" type="CL_OrgUnits_WAHO_1.0_Type"/>
    <xs:attribute name="period" use="required" type="periodType"/>
    <xs:attribute name="mechanism" type="CL_Mechanism_WAHO_1.0_Type" use="optional"/>
    <xs:anyAttribute processContents="skip"/>
  </xs:complexType>
  <xs:complexType name="DataValueType">
    <xs:sequence maxOccurs="1" minOccurs="0">
```



```
      <xs:element name="annotation"/>
    </xs:sequence>
    <xs:attribute name="dataElement" use="required" type="CL_DataElements_WAHO_1.0_Type"/>
    <xs:attribute name="value" use="required" type="xs:decimal"/>
    <xs:attribute name="sex" type="CL_Sex_WAHO_1.0_Type" use="optional"/>
    <xs:attribute name="ageGroup" type="CL_AgeGroup_WAHO_1.0_Type" use="optional"/>
    <xs:anyAttribute processContents="skip"/>
  </xs:complexType>
  <xs:element name="adx" type="adxType"/>
</xs:schema>
```


Appendix 8H – (Informative) Generated sample ADX schema for validating disaggregation

An electronic copy of the file is available at ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/ADX/sample/adx_disaggs_generated.sch. In case of any difference between the electronic copy and the appendix below, this appendix SHALL be considered normative.

```
<?xml version="1.0" encoding="UTF-8"?>
<sch:schema xmlns:sch="http://purl.oclc.org/dsdl/schematron"
  xmlns:str="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/structure"
  xmlns:com="http://www.sdmx.org/resources/sdmxml/schemas/v2_1/common" >

  <sch:ns uri="urn:ihe:qrph:adx:2015" prefix="adx"/>

  <sch:pattern>
    <sch:title>Validating ADX aggregations</sch:title>
    <sch:p> The ADX xsd schema validates that correct codes are used in code lists. Applying
            this set of rules in addition ensures that datavalues are reported with the correct
            disaggregations. </sch:p>
    <sch:rule context="adx:dataValue[@dataElement='MAL01']">
      <sch:assert test="not(@ageGroup)">@ageGroup is not permitted on element MAL01</sch:assert>
      <sch:assert test="not(@sex)">@sex is not permitted on element MAL01</sch:assert>
      <sch:assert test="not(@mechanism)">@mechanism is not permitted on element MAL01</sch:assert>
    </sch:rule>
    <sch:rule context="adx:dataValue[@dataElement='MAL02']">
      <sch:assert test="not(@ageGroup)">@ageGroup is not permitted on element MAL02</sch:assert>
      <sch:assert test="not(@sex)">@sex is not permitted on element MAL02</sch:assert>
      <sch:assert test="not(@mechanism)">@mechanism is not permitted on element MAL02</sch:assert>
    </sch:rule>
    <sch:rule context="adx:dataValue[@dataElement='MAL03']">
      <sch:assert test="not(@ageGroup)">@ageGroup is not permitted on element MAL03</sch:assert>
      <sch:assert test="not(@sex)">@sex is not permitted on element MAL03</sch:assert>
      <sch:assert test="not(@mechanism)">@mechanism is not permitted on element MAL03</sch:assert>
    </sch:rule>
    <sch:rule context="adx:dataValue[@dataElement='MAL04']">
      <sch:assert test="@ageGroup">@ageGroup must be present on element MAL04</sch:assert>
      <sch:assert test="@sex">@sex must be present on element MAL04</sch:assert>
```



```
      <sch:assert test="not(@mechanism)">@mechanism is not permitted on element MAL04</sch:assert>
    </sch:rule>
  </sch:pattern>
</sch:schema>
```


Appendix 8I – (Informative) Sample ADX/XML data

An electronic copy of the file is available at ftp://ftp.ihe.net/TF_Implementation_Material/QRPH/ADX/sample/adx_data_sample2.xml. In case of any difference between the electronic copy and the appendix below, this appendix SHALL be considered normative.

```
<?xml version="1.0" encoding="UTF-8"?>
<adx xmlns="urn:ihe:qrph:adx:2015"
     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     xsi:schemaLocation="urn:ihe:qrph:adx:2015 ../schema/adx_sample_generated.xsd"
     exported="2015-02-08T19:30:00Z">

    <group orgUnit="342" period="2015-01-01/P1M" dataSet="MALARIA" mechanism="PEPFAR">
        <dataValue dataElement="MAL01" value="32" />
        <dataValue dataElement="MAL02" value="20" />
        <dataValue dataElement="MAL04" value="10" ageGroup="under5" sex="M" />
        <dataValue dataElement="MAL04" value="10" ageGroup="under5" sex="F"/>
        <dataValue dataElement="MAL04" value="10" ageGroup="5andOver" sex="M"/>
        <dataValue dataElement="MAL04" value="10" ageGroup="5andOver" sex="F"/>
    </group>
    <group orgUnit="342" period="2015-01-01/P1M" dataSet="MALARIA" mechanism="OTHER" comment="Imported from
facility system">
        <dataValue dataElement="MAL01" value="32" />
        <dataValue dataElement="MAL02" value="20" />
        <dataValue dataElement="MAL03" value="0" >
            <annotation>Some qualifying text here on the datavalue</annotation>
        </dataValue>
        <dataValue dataElement="MAL04" value="10" ageGroup="under5" sex="M" />
        <dataValue dataElement="MAL04" value="10" ageGroup="under5" sex="F"/>
        <dataValue dataElement="MAL04" value="10" ageGroup="5andOver" sex="M"/>
        <dataValue dataElement="MAL04" value="10" ageGroup="5andOver" sex="F"/>
    </group>
</adx>
```


Appendix 8J – (Informative) Formatting of times and time intervals in ADX

The SDMX v2.1 DSD ObservationalTimePeriod data type allows a variety of formats for specifying time periods, with the result that there are several equivalent ways of expressing the same period. In the interest of greater and easier interoperability, the normative transform in Appendix 8B restricts the available options to an SDMX ObservationalTimePeriod. This can be either a `xs:dateTime`, or a `com:TimeRange`, as specified under the `str:TimeDimension` element of the DSD.

`xs:dateTime` is formatted as yyyy-mm-ddThh:mm:sszzzzzz, where:

yyyy is the year

mm is the month (01-12)

dd is the day of the month (01-31)

T indicates the time (required)

hh is the hours

mm is the minutes

ss is the seconds optionally followed by '.' and fractional seconds

zzzzzz is the optional time zone: a '+' or '-' followed by the hh:mm (hours and minutes) offsetfrom UTC, or Z to indicate UTC.

All parts are required except the fractional seconds and the time zone.

Examples:

2016-01-01T00:00:00	Midnight on January 1, 2016
2016-01-01T12:00:00	Noon on January 1, 2016
2016-01-01T12:00:12.34Z	Noon and 12.34 seconds on January 1, 2016, UTC
2016-01-01T12:00:00+03:00	Noon on January 1, 2016, three hours ahead of UTC

`com:TimeRange` is modelled after (`xs:dateTime` OR `xs:date`)/`xs:duration`.

- `xs:dateTime` is as described above.
- `xs:date` follows the same rules as `xs:dateTime` except that the elements Thh:mm:ss are not present.
- `xs:duration` is of the form P[n]Y[n]M[n]DT[n]H[n]M[n]S, where n is the number of units and the letters have the meaning:

P - Period indicator (always present for durations)

Y - Years (follows the number of years)

M - Months (follows the number of months)

D - Days (follows the number of days)

T - Time indicator (preceding any of the following):

H - Hours (follows the number of hours)

M - Minutes (follows the number of minutes)

S - Seconds (follows the number of seconds)

Examples:

2016-01-01/P1Y	One-year period starting on 2016-01-01
2016-04-01/P1Y	One-year period starting on 2016-04-01
2016-07-01/P6M	Six-month period starting on 2016-07-01
2016-03-01/P1M	One-month period starting on 2016-03-01
2016-01-03/P7D	One-week period starting on 2016-01-03
2016-01-01/P1Y6M	A year and six month period starting on 2016-01-01
2016-01-01T12:00:00/PT1H30M	A period of 1 hour and 30 minutes starting on 2016-01-01 at 12:00 noon

Volume 3 Namespace Additions

Add the following terms to the IHE Namespace:

None

Volume 4 – National Extensions

Add appropriate Country section

2010 Not applicable

