



# The AMIE Project

## Application Architecture Overview

Author	Anthony Christianson
Title/Role	Architect
Version: 4.0	01/29/2009



## Revision & Sign-off Sheet

### Change Record

Date	Author	Version	Change Reference
4/29/08	M. Kelley	3	Added content from Solution Documents
1/29/09	S. Koka	4	Updated the references.

### Reviewers

Name	Version Approved	Position	Date

### Distribution

Name	Position

### Document Properties

Item	Details
Document Title	Physical Design
Author	
Creation Date	
Last Updated	



# The AMIE Application Architecture Overview

## Table of Contents

Revision & Sign-off Sheet .....	2
Table of Contents.....	3
1 Introduction .....	4
2 AMIE High Level Functional Architecture/Data Flow .....	4
3 Solution Overview .....	5
3.1 Solution Description.....	6
3.2 Description of the Components.....	9
4 AHCCCS Interfaces .....	10
4.1 RLS.....	10
4.1.1 RLS Publish .....	10
4.1.1.1 RLS Receive Publish.....	10
4.1.1.2 RLS Acknowledge Publish .....	11
4.1.2 RLS Lookup.....	11
4.1.2.1 RLS Receive Lookup.....	11
4.1.2.2 RLS Lookup Response .....	12
4.2 CDX.....	12
4.2.1 CDX Publish .....	12
4.2.1.1 CDX Receive Publish.....	13
4.2.1.2 CDX Acknowledge Publish.....	13
4.2.2 CDX Retrieve .....	13
4.2.2.1 CDX Retrieve Request .....	13
4.2.2.2 CDX Retrieve Response .....	14

## 1 Introduction

The AMIE Utility uses a combination of open source software applications, internally developed software and off the shelf products. The AMIE Utility is designed to meet standards established by the Office of the National Coordinator for Health Information Technology (ONCHIT), the Medicaid Information Technology Architecture (MITA), the Arizona statewide technology standards, AHCCCS standards, as well as industry standards for data exchange (such as HL7, SNOMED, NCPDP, LOINC), and technical standards common to web technologies and internet connectivity. AHCCCS intends the Utility to be able to meet the certification by the Certification Commission for Healthcare Information Technology (CCHIT). The base technology for the Utility is Microsoft Windows, .NET and C#.

Construction of the overall Utility has been broken out into separate releases or projects for each of the major parts of the system. Phase I of the AMIE Utility provides an electronic health information exchange (AMIE), and a simple web-based record viewer. This document describes the overall architecture of this effort. In particular, the Service Oriented Architecture (SOA) deployed and the interfaces built for the AMIE Utility Project are explained.

## 2 AMIE High Level Functional Architecture/Data Flow

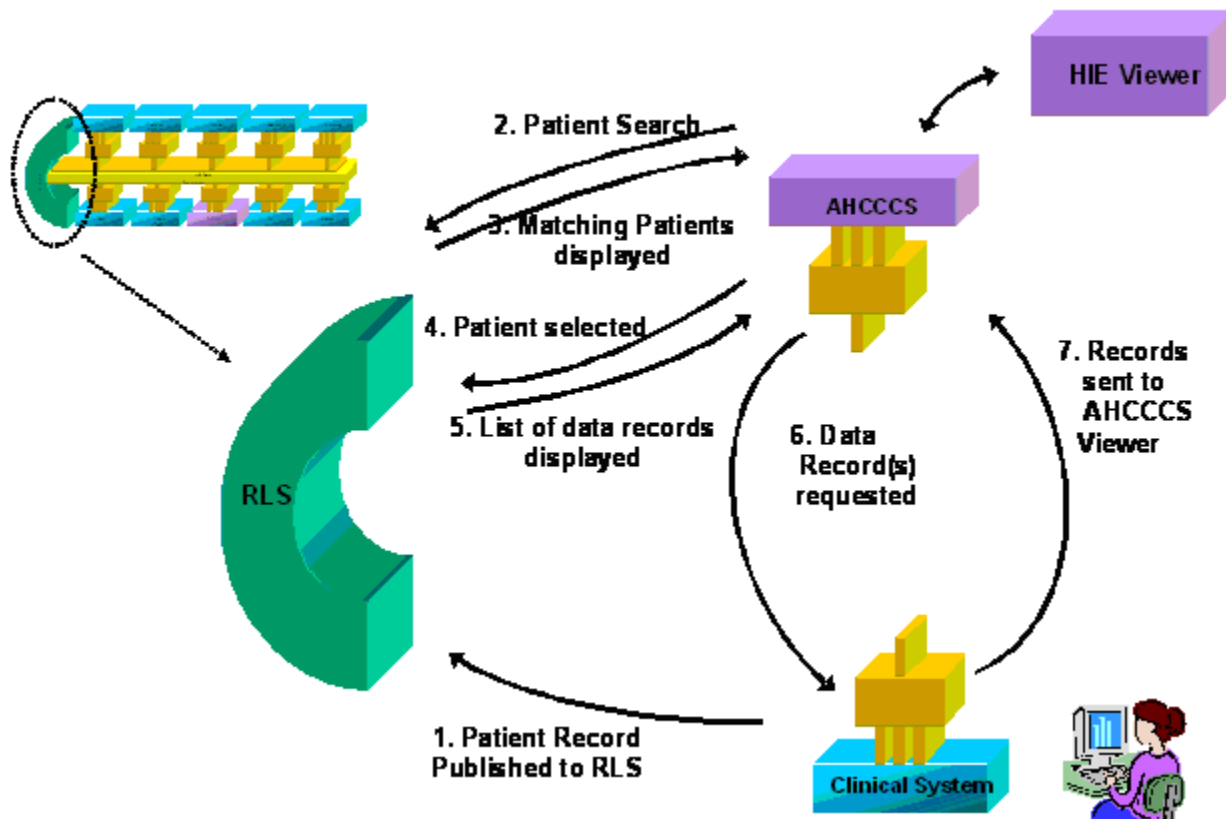


Figure 1



## The AMIE Application Architecture Overview

The diagram above (Figure 1) depicts the data flow observed with the use of the AMIE Utility:

1. All Data Providers publish notification of the existence of available patient records from the local data source to the central registry (Record Locator Service or RLS) following service delivery. Patient demographics, a Record ID and the Source System ID are logged, specifying the location of each record. At the point of care, these records are made available through the web-based Viewer application.
2. The Provider accesses the Viewer and searches for a Patient using valid criteria.
3. The Viewer queries the RLS and returns Patient records matching entered criteria to the screen for review by the Provider.
4. The Provider selects the appropriate Patient.
5. The Viewer then queries the RLS and displays a list of available Records for the selected Patient (In Phase I, this may include Medication History, Lab Results, and/or Discharge Summaries).
6. The Provider selects one or more records for Viewing.
7. The Viewer acquires the address (record location) from the central registry (RLS – see step 1).
8. The Record data is retrieved directly from the Data Provider on peer-to-peer basis.

• *Conforms to Web services interoperability pattern: Publish/Find/Bind*

Adapted from: Linking Healthcare Information: Proposed Methods for Improving

As described above, each Data Provider as a component of the AMIE Utility integrates with the Exchange and actively publishes indices of available records (Step 1 of Fig 1). Data Providers enable a mechanism for Users to locate and view available records for selected Patients (Steps 2-7 of Fig 1). In this manner, the key integration points for each Data Provider are: a) Using web services, publish to the RLS when medication history exists for a Patient (Step 1 of Fig 1), and b) Respond to web service requests for a patient's records by returning all data using industry-standard formats (Step 8 of Fig 1).

### 3 Solution Overview

SOA is the architectural style for creating business processes as loosely-coupled units that communicate via a common protocol such as SOAP. The units can be distributed over a variety of platforms on a network and communicate with each other by passing data from one web service to another. AHCCCS has defined a technology interface based upon SOA for the exchange of health information. The interface is secured by encryption and other approved security technology and techniques to maintain privacy and integrity of the information. Standard messaging formats are used for interacting with Data Providers. This messaging is based upon the Clinical Document Architecture (CDA)..

Figure 2 below highlights these interactions between the Data Provider source system and the AMIE.

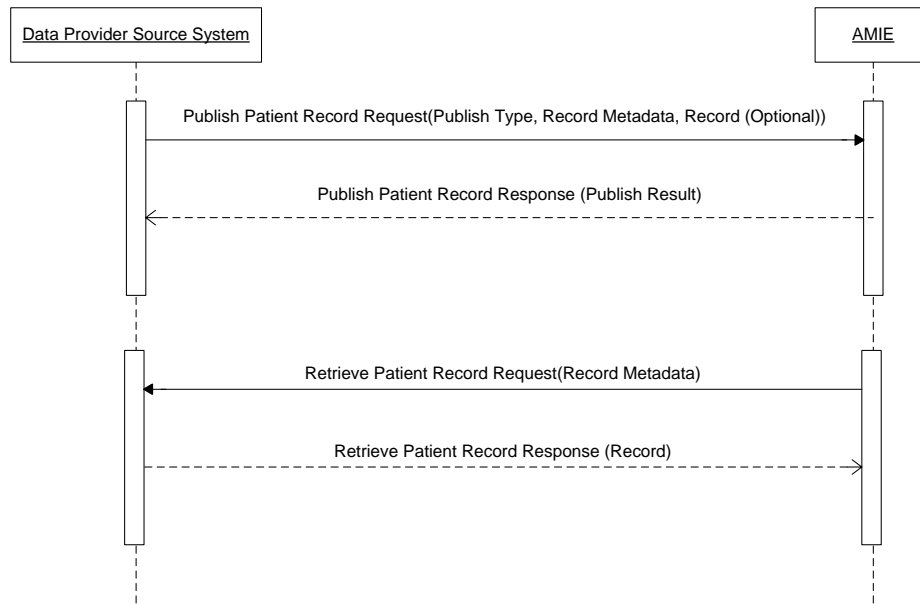


Figure 2

### 3.1 Solution Description

The Data Partner may interact with the AMIE directly from their source system or interface engine (Cloverleaf, eGate, etc.) based on the current environment. Use of the AHCCCS Emulator, described later in this document, is optional. This software is used when the Data Partner desires assistance in the production of the required CDA.

**Publish:** Data is submitted via the use of web services. Based on triggers in their organizational workflow, the Data Providers will send their patient/encounter metadata, calling the Receive Publish web service of the RLS. For example, hospitals may publish a discharge summary when the Discharge Summary record is signed. A mutually determined unique ID will be used to identify Data Providers in the network.

**Request/Response:** Whenever a request is generated from the AMIE Viewer to the RLS for the patient/record look up, the RLS uses this information to locate the patient/record metadata and respond to the Viewer with a list of matching records. When the User selects one of the matching records for retrieval, the Viewer then submits this record request to the AMIE. The request is sent from the AMIE gateway, via the internet, to the gateway located at the source Data Provider. The Data Provider gateway forwards the request to either the source system or the Emulator, which then responds to the request. The response is then routed back from the Data Provider gateway to the AMIE gateway and ultimately rendered for display in the Viewer.

A more detailed drawing, including the web services described in this document, is pictured in Figure 3 below. The Emulator is optionally available to each Data Provider to assist with a number of functions necessary for effective interaction with the AMIE. The Emulator can manage the Publish of patient demographics and record metadata to the AMIE RLS. The Emulator can also support the transformation of all record types received into the required CDA format. The Emulator also provides storage functionality in terms



## **The AMIE Application Architecture Overview**

of saving both original and transformed data in the Emulator database to support record requests from the Viewer. This solution eases the load on the production systems in order to respond to such requests from the AMIE in a timely manner.







## The AMIE Application Architecture Overview

### 3.2 Description of the Components

This solution as depicted in Figure 3 consists of two (2) domains with the following components and data stores:

#### *Data Provider*

- Data Provider Source System
  - The source system contains all records to be published to AMIE. Any system providing data to the AMIE is considered a source, whether the record originated in or has migrated to the system as part of Data Provider's workflow.
- Interface Engine (optional)
  - An interface engine may serve as a centralized interface to the data provider source systems. In this manner, the interface engine supports industry-standard messaging for delivering information to other systems.
- Emulator (optional)
  - The Emulator is the application component that will process all data received from the data provider source system or interface engine. It will transform the data delivered into a CDA format and persist it in the emulator database. Furthermore, it will publish all transformed and persisted records to the AMIE. Also, it will provide the CDA records to the AHCCCS AMIE upon request.
- Emulator Database (optional)
  - The Emulator database contains all original records received, as well as any transformed records to facilitate retrieval upon request. The Patient Master stores patient demographic information and original messages. The Encounter Master stores records in CDA format and the original data used to generate the CDA record (i.e. HL7 or NCPDP messages).

#### *AHCCCS AMIE*

- Record Locator Service (RLS)
  - The RLS provides a web service-based interface for publishing the existence or availability of records to the AHCCCS AMIE. The RLS acts as a patient and record registry to enable the search of patient records among federated data provider source systems.
- AHCCCS AMIE Gateway
  - The AHCCCS AMIE Gateway provides the communication framework that supports the AHCCCS AMIE. A gateway is also deployed at each node of the AHCCCS AMIE to enable communication among all nodes (see Figure 3). The CDX web service resides on each Gateway device.
- RLS index/MPI
  - The RLS index and MPI is a database that stores all patient information and record information that has been published to the AHCCCS AMIE. The only option available to update this database is thru a publish update operation.
- Viewer



## The AMIE Application Architecture Overview

- The Viewer is an AHCCCS web application used to retrieve and display patient and encounter records such as discharge summary, lab results, and medication history to the provider. The RLS is used as an index to locate available patient records. Web services are then used to request, retrieve and display the record detail. Users may print records displayed on the Viewer, but will not be able save an electronic copy locally on their system.

### 4 AHCCCS Interfaces

The AMIE has two primary points of interface with each Data Provider, the RLS and the CDX. Additionally inter-gateway web services exist that request and forward records between the Gateway of the Data Provider and the AMIE Gateway within the AHCCCS domain. The focus of this document is the interfaces between the AMIE and Data Provider only.

#### 4.1 RLS

As noted earlier, the Record Locator Service (RLS) for this federated AMIE stores only the location of patient records. The RLS system receives patient record location information via a Publish interface from each Data Provider and supports searching this information using the Lookup interface from the AMIE Utility Viewer.

##### 4.1.1 RLS Publish

The Publish web services of the RLS create an indexing mechanism to store patient record locations. The Publish message from each Data Provider details the location of a patient record, including source gateway, organization, source system, patient identifier and encounter meta-data. This information can later be queried and used to retrieve the record from the source. In this manner, the RLS system does not store actual encounter clinical detail. Rather it stores the fact that the record exists and the location from which it can be retrieved.

##### 4.1.1.1 RLS Receive Publish

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Publish xmlns="http://www.ma-share.org/">
      <SenderID>string</SenderID>
      <CDA>string</CDA>
    </Publish>
  </soap:Body>
</soap:Envelope>
```

Currently Data Providers do not publish directly to the RLS. When a record is generated, notification is first published to the CDX web service on the Gateway device (A) in the Data Provider's DMZ environment. This CDX web service in turn forwards the Publish message to the RLS service on the Gateway device at AHCCCS (B). In fact, the Receive Publish web service of the RLS processes two parameters from the CDX service at the Data Provider:



## The AMIE Application Architecture Overview

- SenderID
  - The sender ID identifies the system making the publish request.
  - The sender ID is validated against the list of recognized senders to ensure that the system making the request is an authorized publisher.
- CDA
  - The CDA is an XML string.
  - The CDA represents the Continuity of Care Document as specified by the joint efforts of the HL7 standards and ASTM organizations.
  - The CDA should include an encounter history section.

### 4.1.1.2 RLS Acknowledge Publish

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <PublishResponse xmlns="http://www.ma-share.org/">
      <PublishResult>string</PublishResult>
    </PublishResponse>
  </soap:Body>
</soap:Envelope>
```

The Acknowledge Publish from the RLS returns a single XML string. This string will be in CDA format and will include the published demographics. Furthermore, the Response contains the CDA Purpose section with a success/fail flag and text node specifying any error messages that occurred during the Publish process.

### 4.1.2 RLS Lookup

The RLS Lookup service exposes the RLS for query of the location of a patient's records. The lookup receives a patient query based on patient demographics entered from the Viewer and then returns the location of all records that match the query data. The RLS lookup currently is based on an exact match.

#### 4.1.2.1 RLS Receive Lookup

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Lookup xmlns="http://www.ma-share.org/">
      <SenderID>string</SenderID>
      <SessionID>string</SessionID>
      <CDA>string</CDA>
    </Lookup>
  </soap:Body>
</soap:Envelope>
```



## The AMIE Application Architecture Overview

The Receive Lookup web service receives three parameters from a Sending entity:

- SenderID
  - The Sender ID identifies the system transmitting the lookup request. In Phase I, Lookups will only be initiated from the Viewer.
  - The Sender ID is validated to ensure that the system making the request is authorized to query the RLS data.
- SessionID
  - The Session ID is an identifier that can be used to monitor/audit a specific session of an entity (User, batch process, etc.)
- CDA
  - The CDA is an XML string.
  - The CDA will include the criteria for demographic data that should be used in the query of the RLS.

### 4.1.2.2 RLS Lookup Response

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <LookupResponse xmlns="http://www.ma-share.org/">
      <LookupResult>string</LookupResult>
    </LookupResponse>
  </soap:Body>
</soap:Envelope>
```

The Lookup Response returns a single XML string. This string will be in CDA format and include all records that match the query information.

## 4.2 CDX

As noted in Figure 3, the CDX (Clinical Document Exchange) is installed on the Gateway device that resides behind the firewall of each Data Provider. This web service provides a common mechanism for transmitting Publish, Retrieve, and Send messages to and from the Data Provider. Patient record information is published through the CDX service provided on the Gateway from the Source System. The CDX Retrieve allows retrieving an individual patient record from the source system. Finally, the CDX supports sending requested patient records to an end system for viewing and/or import. In Phase I, viewing of retrieved records is supported only through the AMIE Utility Viewer.

Note: As noted in section 2.2, the Emulator device is optionally installed between the source system and the Gateway within the Data Provider environment to assist with data transformation and caching.

### 4.2.1 CDX Publish

The CDX Publish allows a data provider to publish documents to the AMIE. Residing on the Gateway device, this service directly communicates with the RLS Receive Publish method



## The AMIE Application Architecture Overview

located within the AHCCCS domain. The CDX Publish method is able to pre-process and transform messages received from the Data Provider Source System or the Emulator to ensure that the correct routing information and document structure is attained.

### 4.2.1.1 CDX Receive Publish

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Publish xmlns="http://www.ma-share.org/">
      <documentString>string</documentString>
    </Publish>
  </soap:Body>
</soap:Envelope>
```

The Publish Request accepts a single XML string with patient demographics, encounter history, and routing information in it. This Request does not have to be in a CDA format, but must be in an approved format so that the necessary transformations can be done to ensure the RLS receives a properly formatted CDA message.

### 4.2.1.2 CDX Acknowledge Publish

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <PublishResponse xmlns="http://www.ma-share.org/">
      <PublishResult>string</PublishResult>
    </PublishResponse>
  </soap:Body>
</soap:Envelope>
```

The Acknowledge Publish is an XML string that specifies the publish details. A status code (success/fail flag) and text node specifying any error messages that occurred during the Publish process are included.

## 4.2.2 CDX Retrieve

The CDX Retrieve services are able to request and receive a clinical document from the source system. The CDX Retrieve routes a record request to the Source for a particular document and returns the requested document either directly or as a callback to the requestor.

### 4.2.2.1 CDX Retrieve Request

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
```



## The AMIE Application Architecture Overview

```
<soap:Body>
  <Retrieve xmlns="http://www.ma-share.org/">
    <senderID>string</senderID>
    <sessionID>string</sessionID>
    <requestDocumentString>string</requestDocumentString>
    <maxWaitSeconds>int</maxWaitSeconds>
  </Retrieve>
</soap:Body>
</soap:Envelope>
```

The Retrieve Request has four parameters:

- senderID
  - The sender ID identifies the system making the retrieve request.
  - The sender ID is validated to ensure that the system making the request is allowed.
- sessionID
  - The session ID is an identifier that can be used by the requestor to monitor a specific session.
  - The session ID is also used for any callbacks.
- requestDocumentString
  - the requestDocumentString is an XML string containing the routing information to the document being requested
- maxWaitSeconds
  - This parameter informs the service how long to wait synchronously for the message response.
  - If the wait seconds is exceeded, the method will return an acknowledgement of the request. Then when the record is retrieved, the senderId and sessionId will be used to callback the original requestor with the document.
  - The entire request will timeout after 60 seconds.

### 4.2.2.2 CDX Retrieve Response

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <RetrieveResponse xmlns="http://www.ma-share.org/">
      <RetrieveResult>string</RetrieveResult>
    </RetrieveResponse>
  </soap:Body>
</soap:Envelope>
```

The Retrieve Response will be one of two possible responses

1. If a record is retrieved within the maxWaitSeconds timeframe, then the response will return: a) the requested record, or b) a response describing why the requested record could not be returned.



## **The AMIE Application Architecture Overview**

2. If a record is not be retrieved in the maxWaitSeconds timeframe, an acknowledgement of the request will be returned. The requested record will be posted back to the sender when it is received.