Using IHE XDS with the Canadian Client Registry Services

Version: 3.1
Date: 2010-04-17

Prepared by:

Document Details

Document Author(s)

Name	Organization	E-mail	Phone
José Mussi	JRS Partners	jmussi@jrspartners.com	(416) 802-3949

Document Owner(s)

For a current version of this document, please contact:

Name	Organization	E-mail	Phone

Document Version History

Version Date	No.	Changes	Source / Author
2010.03.30	1.0	First Version	José Mussi
2010.04.14	2.0	Updates with comments from stakeholders	José Mussi
2010.04.16	3.0	Updates with comments from conference call	José Mussi
2010.04.17	3.1	Changes suggested by Alberta pHIE Netcare	José Mussi

Table of Contents

Introduction:

This whitepaper discusses the technical considerations of using the IHE Cross-Enterprise Document Sharing (XDS) within a jurisdictional EHR infostructure, in particular in respect to its integration with the Client Registry (CR) services. This material here presented is based on the most recent XDS and XDS-I specifications. It has also been created in response to a recent analysis document prepared by the pHIE Netcare initiative about the introduction of Link/Unlink support in XDS and XDS-I solutions.

XDS Background

The XDS suite of profiles has been created by IHE to support the exchange of a patient's longitudinal health record across multiple enterprises. Its basic premise i s that clinical information will be shared using a "clinical documents model", where a collection of clinical information pertaining to the same patient is grouped into one or more "documents" and published to a shared XDS infrastructure composed of one or more document repositories and a common document registry containing metadata and pointers to all shared documents across these repositories. Systems that publish documents are known as "document sources" while those who retrieve them are called "document consumers". The same system can be both a source and consumer of XDS documents. All these "XDS actors" are contained within a single "affinity domain", which establishes a set of conventions on what type of documents, security constraints and other applicable policies must be used by all organizations (i.e. enterprises) that have come together to publish and share documents.

Crucial to the ability to share documents reliably is the need to uniquely and correctly identify the person (i.e. patient or client) to whom the information belongs to. This is a non-trivial problem, as each clinical system that participates in the affinity domain may (or more likely will) use different identification means for its patients. The challenge is to find a common identifier that can be used across the entire affinity domain. The XDS specifications do not attempt to resolve the identification problem, rather it assumes that the affinity domain will have some common means to create a unique identifier for the domain and allow document sources to find the appropriate patient ID prior to publishing documents to the XDS infrastructure. IHE recommends the use of the PIX profile where a PIX manager provides each document source (and later document consumers as well) a match between the patient's local identifier (i.e. that which is known to the Point of Service (POS) application) and the common, affinity domain identifier. With this information in hand, the document source system can publish documents, create submission sets and organize information into XDS folders, all assigned to the same patient identifier.

The Document Repository, which receives the information from the document sources, is mostly agnostic to the patient identifier contained in each submission set. Its job is basically to receive the documents from the source, store it within its persistence layer, calculate the hash and size of the

document assign a unique id and assign and URI (only if it supports ITI-17)to each object and pass all this information to the document registry. It does not validate or change the patient identifier contained within the "provide and register" transactions.

On the other hand, the patient identifier is a key piece of information for the document registry. After all, its role is to organize and group documents that belong to the same person. It must ensure that a proper identifier is provided with each submission set. To accomplish this, the document registry receives patient identity feeds from the affinity domain patient identity source containing information about valid identities for the affinity domain. With this information in hand, the document registry can ensure that:

1. patient identifier contained in the submission set is valid and active for the affinity domain
2. all documents within the submission set belong to the same patient (i.e. contains the same patient identifier)
3. all documents within the same XDS folder also belong to the same patient.

The key point about this approach is that the affinity domain patient identifier is the authoritative means for identifying patients and grouping documents. Although the local patient identifier can also be provided with each submission set, it is considered a mere attribute of the transaction.

Once a document has been published to the XDS services, there are only two ways where the assigned patient identifier may be changed:

1. By the original document source – in case where documents were published with the wrong patient identifier, the document source can deprecate the original posting and re-submit the same document with the correct identifier. The document registry will change the status (using availability status attribute) of the incorrect entry and add the new information as provided by the document source. This process is integral to the document lifecycle management defined by the XDS specifications and the XDS infrastructure is not concerned about the reasons for the correction.
2. By the Patient Identity Source – a recent addition to the XDS specifications allows support for patient identity merges. In this scenario, two affinity domain patient identifiers are discovered to belong to the same person. Through some process external to the XDS services, one of the two identifiers is chosen to be subsumed (i.e. no longer in use) by the other (i.e. the survivor). A merge notification is sent to the XDS registry and from that moment on:
    - All documents that were published with the subsumed patient identifier are now joined with documents belonging to the surviving ID.
    - Any further submission sets referencing a subsumed ID will be rejected by the document registry with an "*XDSUnknownPatientId* "error.
    - All queries referencing a subsumed identifier return no content.

- All queries referencing a surviving identifier return the entire recorded merge tree and return appropriate metadata.

The XDS specification does not specify changes to the internal state of the document registry. Instead it specifies required future behaviours on the part of the three transactions listed above. Also, the patient identity merge notification is not propagated to the document repositories, which means that their record for the document may (it is not a mandatory behaviour) still contain the original patient identifier used when the submission set was first published.

Document sources must also use the PIX manager to determine the corresponding affinity domain patient identifier before submitting any queries to the document registry. As mentioned previously, the affinity domain patient identifier is the only identifier that is recognized by the XDS services.

EHR Client Registry Services and Client Identification

In the Canadian interoperable EHR architecture, the responsibility for managing and cross referencing client (i.e. patient) identities falls with the Client Registry (CR) services. The CR collects registration information from various patient identity domains in its jurisdiction and groups these records together through a combination of automatic algorithms and manual linking events. These sets of linked Ids are given a unique identifier, known as the Enterprise Client Identifier (ECID). The ECID can be a new ID created by the CR (i.e. shadow ECID) or can be assigned (through jurisdictional policy) by one of the patient identity domain (i.e. provincial health card number). Regardless of the model, the CR is the only authoritative source of ECIDs for the jurisdictional EHR infostructure.

For the most part, the CR role is very close to a combined PIX/PDQ manager. The most significant difference is that the relationship local ID → ECID is not guaranteed to be permanent. In fact, as described in the Alberta Link/Unlink whitepaper, it can very often change because a new local patient ID will be given an initial ECID when ii is first published to the CR. It may or may not remain the same, depending on the result of the matching algorithm and possibly manual linking events. If a match is found, the local ID is moved from the original ECID to another, pre-existing ECID. In this case, the original ECID is deprecated and no longer in use[1]. There are also other scenarios where, for different reasons, the original match of a local ID may be wrong and a new ECID assignment is required.

In all these cases, the implication to the XDS services occurs if there is one or more published documents that belong to the local patient ID in question. Since this identifier is now linked (i.e. assigned) to a different ECID, the XDS services needs to be informed and its information corrected.

These examples highlight the key nature of the issue: in the CR model, the authoritative patient identifier for clinical data published to the EHR (and this would include XDS documents) is the local

[1] This is similar to a patient identity merge described previously since the original ECID was a singleton, that is, it only contained one linked local ID.

patient ID from the POS source system; the ECID can be seen as just an attribute, externally assigned by the Client Registry that can change!

Note: There are other different EHR client identification models applied in Canada:

1. In some cases (i.e. Quebec), the ECID is a public identifier that must be resolved by each EHR source, much like the XDS model. This model does not require the support of link/unlink and can easily adopt XDS solutions as specified.
2. In other places (i.e. Manitoba), the CR manages the linked list as described but the ECID is not used by EHR repositories. Records are created in the EHR just with the local identifiers and linked together, via the CR linked set, dynamically at query time. Link/unlink events are handled exclusively by the CR and repositories need not be notified. However, this model does not fit well with the XDS specifications which require a single patient identification domain. It is likely that the ECID would be used for XDS documents, bringing us back to the model in discussion.

Integrating EHR and XDS Services

The essence of the issue in discussion is: can XDS be adapted to support the CR services and client identification services as described?

The Alberta Link/Unlink whitepaper brings to the surface a main consequence of this integration, which is, supporting changes to the ECID linkage sets. If the ECID assigned to a local ID never changed, than there would be no impact to the XDS specifications, but since this is not the case, how can the XDS infrastructure deal with the dynamic nature of the ECID linkage sets?

The whitepaper provides a very good description of the use cases involved in EHR patient identification as implemented in Alberta. It also provides three suggestions that resolve the problem of how to notify link/unlink events to the document registry. In all cases, the solution requires that the document source system provide with the submission set the local identifier that was used to execute the client resolution (i.e. the PIX query).

The proposed solutions try to address the link/unlink problem using existing ITI transactions, one using ITI-8 (currently used for patient identity feed between Patient Identity Source and the XDS registry) and the other two using ITI-30, defined as part of the Patient Administration Management (PAM) profile.

In the first solution, ADT^A31 – Update Person Information message would be used by the document registry to determine if there has been any change to the linked Ids for a particular ECID. This approach requires that a first message be sent to unlink the local ID from the current ECID and a second message to re-link the same ID to a new ECID. The advantage of this solution is that it uses the existing ITI-8 transaction and message set. On the other hand, using two asynchronous messages to correct a single ECID assignment is a concern since it could leave the registry in an incorrect state if one fails.

The two other solutions propose using the PAM profile (ITI-30) to manage the synchronization of patient identity information between the PIX manager and the document registry. These recommendations would require a larger change to the XDS profile since it changes the set of messages between the two actors.

Variants for this approach are described based on the two PAM options:

1. Option 2a: Using merge – in this option, the link/unlink information is done through ADT^A47 – Change Patient Identifier List, very similar to how the CR uses ADT^A43 – Move Patient Information Event , to notify the EHR of link/unlink occurrences. This approach only requires a single message to complete the change.

2. Option 2b: Using Link/Unlink – in this option, two separate messages, ADT^A24 – Link Patient Information and ADT^37 – Unlink Patient Information are required. Similar to option 1 described previously, this approach uses two asynchronous messages and could create a situation where the XDS registry is left in an incorrect state.

Analysis of the Proposed Solutions

The three solutions proposed in the document address the same problem: how the Patient Identity Source should notify the XDS document registry of changes to which ECID is assigned to a particular local ID. In this sense, all three solutions can work, although preference should be made to Option 2a which does not rely on asynchronous pairs of messages.

Another option not explored in the document, which in this author's view could be simpler to implement, would be to add ADT^A43 to ITI-8 as an optional message. It would be used strictly to inform the document registry that a new ECID has been assigned to an existing local ID, using a message structure as shown in Figure 4 of the whitepaper. It would have the same advantage as Option 2a (i.e. single message) and has a small impact since it does not require a change in the XDS integration profile from using ITI-8 to ITI-30 to synchronize patient identities. Since support for ADT^A43 would be optional, it would not require changes to any XDS implementation that does not involve link/unlink events as described.

However, the whitepaper does not address some key impacts to the XDS implementation that will occur regardless of which messaging approach is adopted. Although it does note the need for document sources to include the local identifier with each submission set, there are three other issues that need further clarification:

1) Link/Unlink events can cause inconsistencies in submission sets or folders, where documents from two (now) different patients are grouped together. The issue here is that when folders and submission sets are first created, they are validated to ensure that all documents they contain

belong to the same patient (i.e. they have the same affinity domain patient ID). After a link/unlink event, it is possible that this condition is no longer valid, as one or more documents that used to belong to a particular ECID is now changed to a different identifier.

One solution would be to include a new (optional) restriction where documents within submission sets and folders must have the same local patient ID. This would ensure that regardless of what happens in a link/unlink scenario, that integrity of the objects in respect to patient identity is preserved. However, this may be too restrictive in regards to folders, where some implementations are using them to combine documents from various sources.

A more pragmatic option would be to maintain the current rules (i.e. documents from the same patient, regardless of source) but add a new behaviour to the registry. If, in consequence of a link/unlink event, one or more documents within a given folder no longer have the same affinity domain patient ID (e.g. ECID), than these must be logically removed from the folder. In addition, the XDS document registry must generate an exception event (possibly through a report) to notify that such action has taken place and allow for subsequent corrective measures by the document source.

This change should not create an issue with submission sets, since documents from the same document source would likely have the same local ID. However, the same may not be true in regards to XDS folders. Since folders are not widely used, it is difficult to assess if this restriction would create implementation barriers.

2) Link/Unlink event notifications are not propagated to document repositories. The issue here is that document repositories may also contain a copy of the metadata published with each submission set. If this is the case, than existing record would no longer be correct after the link/unlink event occurs.

   This situation is similar to the patient demographic information kept in the metadata record. It is not guaranteed to be correct over the lifetime of a document, and only reflects the original state of the record. If this is acceptable, and document consumers are aware of possible discrepancies, than no further changes are required. Otherwise, the same message used by the document registry to handle link/unlink events will have to be passed on to all document repositories in an affinity domain.

   Note: Many of the current document repository implementations do not persist the document metadata, in which case this issue is not applicable.

3) Merge/Unmerge of local patient Identifiers need to be propagated to the document registry. Since the XDS services are now made aware and rely upon the correctness of the local patient identifiers, it should be informed of any merge/unmerge events that effect that information.

Here is the scenario: two local identifiers (Lid-A and Lid-B) have been in use for some time but have been determined to belong to the same patient (i.e. source system duplicates). A merge event occurs in the local system and is notified to the CR, where Lid-A is subsumed by Lid-B. Within the CR, Lid-B is the only surviving ID but the XDS infrastructure still has one or more documents attached to Lid-A. Since (presumably) Lid-A and Lid-B were linked to the same ECID (let's say ECID-123), this situation is not a problem as all queries to the registry would use that same ECID number. However, if in the (not very likely) situation where Lid-B is now determined to be linked to the wrong ECID (i.e. it should be linked to ECID-987), we would have a problem. The link/unlink solutions here described would see a notification that all documents belonging to Lid-B should be now assigned to ECID-987. Nothing would be said about Lid-A since it has been for all purposes subsumed. At this point, the document registry would be left in an incorrect state, where Lid-A documents remain attached to ECID-123 while Lid-B documents are moved to ECID-987.

There are different ways to address this problem; most require the propagation of local merge/unmerge events to the XDS infrastructure. This will add a new level of complexity to the XDS integration model and may only be required in very rare instances. Alternatively, one may define a solution where the client identity source needs to include all merged identifiers (i.e. all subsumed local Ids) when creating a notification of a link/unlink event.

Conclusion

Integrating XDS and XDS-I solutions into a pan-Canadian EHR infostructure requires support for use cases that were not considered when the integration profiles were first created. This document and the attached Albert pHIE Netcare whitepaper provide a description of the problem, typical use scenarios and considerations for possible solutions.