Plataforma de Integración de Servicios en el Área de la Salud

Proyecto de Grado 2013

Instituto de Computación – FING – UDELAR

Tutores:

MSc. Ing. González, Laura

MSc. Ing. Llambías, Guzmán

Estudiantes:

Ardanaz Cabrera, Paula

Arocena San Martin, Rodolfo

Delgado Alcain, Rafael

Resumen

La integración e interoperabilidad de los sistemas que manejan información de salud es un tema de gran interés, ya que resulta clave a la hora de brindar una atención de calidad a los pacientes. En la actualidad existen una variedad de estándares y buenas prácticas definidas por distintas organizaciones para la industria de la salud, siendo dos de las más importantes Integrating the Healthcare Enterprise (IHE) y Health Level Seven (HL7). Sin embargo, la integración e interoperabilidad de sistemas en esta área todavía se dificulta, debido a que no todas las organizaciones cuentan con sistemas de apoyo de atención en salud adheridos a estándares o buenas prácticas. Por otro lado, el hecho de que la información de salud está catalogada como información sensible, es confidencial y su acceso se encuentra regulado por leyes, introduce problemáticas adicionales que hacen más difícil aún la situación.

En Uruguay, la Agencia de Gobierno Electrónico y Sociedad de la Información (AGESIC) en el marco del proyecto denominado Salud.uy, se planteó como objetivo el desarrollo de una plataforma de interoperabilidad para la integración de sistemas de información de Salud, a nivel de Estado y en general a nivel de todo el ámbito de la salud. Si bien esto ha sido un avance importante, quedan aún varios problemas por resolver como por ejemplo, lograr la integración y compatibilidad de la Plataforma de Gobierno Electrónico (PGE) con los estándares y buenas prácticas más utilizados en el área de la salud a nivel mundial.

Este proyecto propone lineamientos de diseño para la creación de una Plataforma de Interoperabilidad en Salud compatible con la PGE. Para esto, en primer lugar se examinó todo lo referente a estándares y buenas prácticas definidos hoy en día para dicha industria y luego se analizó cómo esos estándares se relacionaban con los requerimientos que tenía el proyecto de Salud.uy. Por último se planteó una propuesta de solución en este sentido, es decir, para integrar e interoperar los sistemas de las organizaciones de salud, a través de la Plataforma de Gobierno Electrónico (PGE) con el fin de compartir documentos.

La solución que se diseñó es de gran porte e involucra muchos sistemas, por lo que se desarrolló un prototipo para un subconjunto de la misma como forma de prueba de concepto.

Palabras Clave: Integración, Interoperabilidad, Gobierno Electrónico, Sistemas de Información Sanitarios, Seguridad, HL7, IHE.

Contenido

Índice de Figuras

Índice de Tablas

1 Introducción

En el presente documento se detallan los distintos aspectos involucrados, así como la solución propuesta, durante la realización del Proyecto de Grado denominado “Plataforma de Integración de Servicios en el Área de la Salud” de la carrera Ingeniería en Computación de la Facultad de Ingeniería - UDELAR (Universidad de la República Oriental del Uruguay).

1.1 Contexto, Motivación y Definición del Problema

Hoy en día es muy importante en los distintos ámbitos de atención sanitaria a la población, contar con el soporte de Sistemas de Información que faciliten la multiplicidad de tareas y procedimientos que implica la correcta atención clínica de los pacientes. Muchos de esos procedimientos implican la realización de variadas tareas, que en muchos casos son realizadas por distintas áreas dentro de una misma Institución o directamente por varias Instituciones que trabajan en conjunto para poder completarlos. Por lo tanto, tan importante como contar con dichos sistemas de apoyo, es el hecho de que los mismos se encuentren bien diseñados para así poder interoperar y compartir la información que sea necesaria para completar la atención, tanto dentro como fuera de la organización.

Por otro lado, a lo largo de su vida, una persona por variadas razones puede llegar a cambiar su residencia ya sea permanentemente o porque realiza un viaje, por ejemplo de vacaciones. En algunos de estos casos eso implica también cambiar la Organización de Salud, o recibir asistencia de otra distinta a la que pertenece. En este tipo de situaciones sería de mucha utilidad poder contar con la información clínica de la persona en formato electrónico, así como también tener la posibilidad, ya sea de transferirla a la nueva Organización o simplemente accederla en casos de necesidad.

A su vez, la mayor parte de la información personal y clínica de los pacientes, es confidencial y su acceso se encuentra regulado y penalizado por leyes en los distintos países. Esto implica que tanto la construcción como la utilización de dichos sistemas de apoyo, se encuentren obligados a asegurar el debido cumplimiento de las mismas.

Surge entonces la necesidad de contar con Sistemas de Apoyo a la atención médica de los pacientes que brinden la posibilidad de compartir la información que manejan, de forma segura, controlada y confidencial, con el objetivo de garantizar la privacidad de la misma y facilitar situaciones cotidianas comunes en este contexto, como las anteriormente expuestas.

Este proyecto, propuesto por el cliente AGESIC (Agencia de Gobierno Electrónico y Sociedad de la Información), se enmarca principalmente en dar solución a esta necesidad planteada dentro el contexto del Uruguay, es decir resolver la integración entre los sistemas de atención médica utilizando la plataforma de Gobierno Electrónico (PGE) del Estado Uruguayo. En lo referente a seguridad, confidencialidad y privacidad de la información, se da soporte para hacer cumplir las leyes definidas e impuestas por el gobierno Uruguayo.

Con el fin de comprender el problema planteado, guiar el análisis realizado en el proyecto y concretar los objetivos definidos (descriptos en la siguiente sección de este capítulo), se definió un Caso de Estudio, compuesto por dos escenarios, que ha servido de referencia a lo largo de la definición de la solución propuesta y se resume a continuación.

En la Figura 1 se muestran los distintos actores que participan en los escenarios definidos.

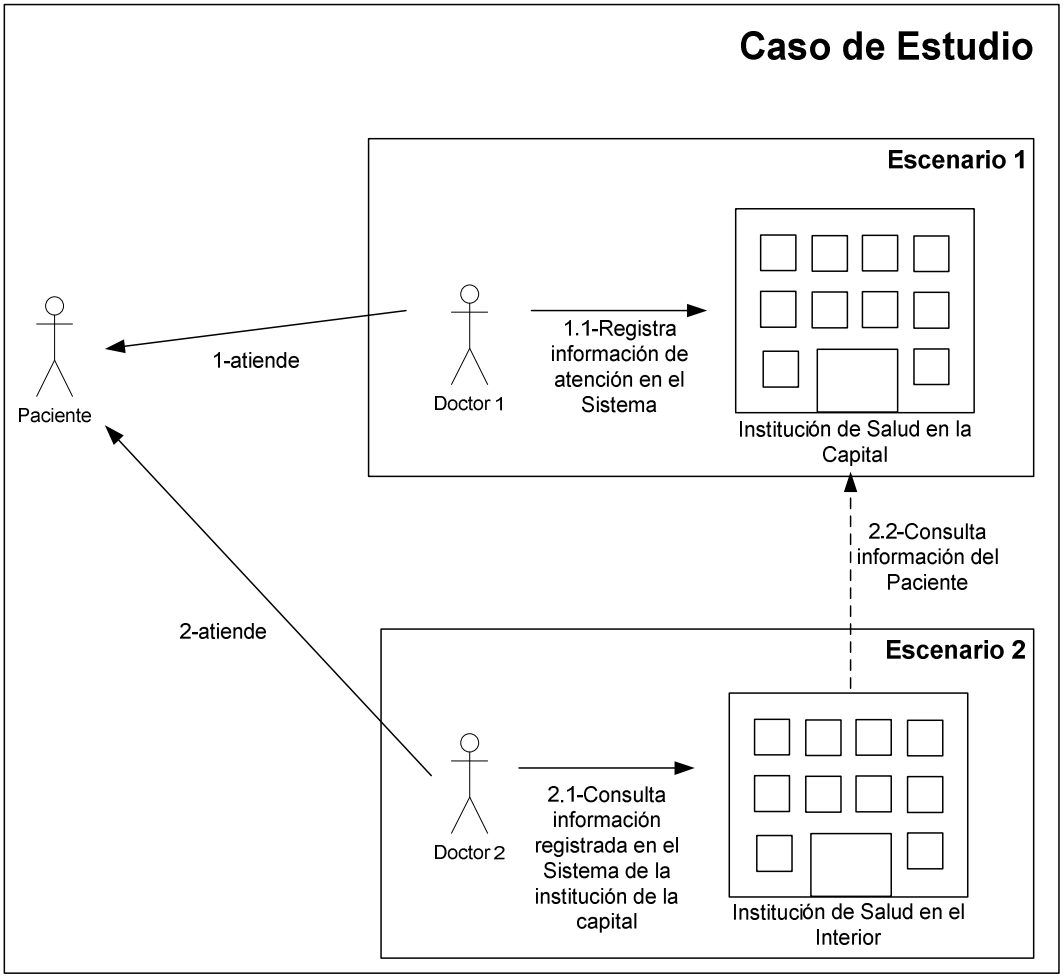


Figura 1: Caso de Estudio

En el primer escenario el paciente se atiende en emergencia de la institución a la que pertenece ubicada en la capital del país. Como resultado de esta atención se le brinda un diagnóstico y una prescripción de medicamentos. Tanto el diagnóstico como la prescripción, son registradas en el sistema de atención médica de la institución.

En el segundo escenario el paciente se encuentra en el interior del país y vuelve a sentir malestar, lo que lo obliga a acudir a un centro asistencial en dicho departamento para ser asistido. En el marco de esta atención, el médico tratante consulta la información del paciente registrada en el primer escenario.

En resumen, el caso de estudio presenta las siguientes problemáticas a resolver:

- La necesidad de poder realizar consultas de documentos clínicos (informes, placas, etc.) por parte de un médico para uno de sus pacientes.
- El paciente para el cual se realizan las consultas, se realizó estudios en organizaciones médicas distintas a la que se está atendiendo actualmente.
- En cada uno de los lugares donde se realizó estudios, el paciente fue identificado de forma diferente de acuerdo a la identificación propia de cada organización.
- El paciente ha consentido políticas de privacidad que aplican a su información clínica, cuando es ingresado en cada una de las organizaciones.

1.2 Objetivos

El objetivo general de este proyecto es proponer mecanismos que permitan integrar e interoperar los sistemas de las distintas organizaciones de la salud, a través de la Plataforma de Gobierno Electrónico (PGE) con el fin de compartir información entre las mismas.

Como parte de la ejecución del proyecto, dicho objetivo ha sido desagregado en los siguientes objetivos específicos:

- Identificar problemáticas del contexto
- Definir lineamientos para la construcción de una Plataforma de integración e interoperabilidad de sistemas de Salud que permita el intercambio de documentos utilizando:
    - Los recursos de la PGE.
    - Los recursos existentes en las organizaciones.
    - Estándares y buenas prácticas (perfiles) de la industria.
- Detectar las problemáticas asociadas al objetivo anterior y proveer soluciones que la resuelvan.
- Realizar pruebas de herramientas para validar los distintos conceptos definidos en la solución planteada.

1.3 Aportes del Proyecto

Los principales aportes de este proyecto corresponden al análisis de las necesidades asociadas a la integración e interoperabilidad de los sistemas de salud.

Concretamente:

- Identificación de las principales problemáticas referentes a los requerimientos asociados al objetivo principal
- Relevamiento de estándares y perfiles de salud
- Relevamiento de estándares y perfiles asociados a la integración y seguridad con la PGE
- Análisis de correspondencia entre los diferentes estándares y perfiles con las problemáticas identificadas
- Definición de los lineamientos de diseño para la creación de una plataforma de interoperabilidad en salud compatible con la PGE, utilizando el resultado del análisis y relevamiento realizado en los puntos anteriores
- Realización de un prototipo que prueba herramientas y valida parte del modelo propuesto
- Generación de documentación asociada al prototipo donde se detallan aspectos puramente técnicos, como ser de configuración, uso, correspondencia y entendimiento del modelo teórico que implementan las herramientas utilizadas
- En lo personal nos aportó la posibilidad de adquirir y asimilar, conocimiento en la industria de las tecnologías de la información aplicadas a salud y también la experiencia de realizar un proyecto más orientado a la línea de investigación

1.4 Organización del Documento

El resto del documento se organiza de la siguiente manera:

En el capítulo 2 se describen los principales conceptos y antecedentes del proyecto, los cuales son utilizados en los capítulos subsiguientes.

El Capítulo 3 describe el análisis de requerimientos, las problemáticas identificadas, los estándares utilizados y también las distintas alternativas de solución.

En el Capítulo 4 se presenta una descripción del diseño de la solución propuesta como resultado de la ejecución del proyecto, así como también la explicación de los distintos artefactos que la componen.

En el Capítulo 5 se describen los aspectos tenidos en cuenta para definir el prototipo a implementar, su instanciación respecto a la solución propuesta en el Capítulo 4, los productos utilizados y la configuración necesaria para su ejecución, así como también los distintos desafíos presentados y encontrados al ejecutar el escenario de prueba definido. Por último se describe la experiencia con dichos productos.

En el Capítulo 6 se presentan las conclusiones y los posibles trabajos futuros.

2 Marco Conceptual

En este capítulo se describen los conceptos necesarios para comprender el resto del documento. Comienza introduciendo conceptos básicos de seguridad informática, para luego pasar a describir las tecnologías de web services y estándares de seguridad avanzados que se mencionan en este documento. El capítulo continúa con una descripción de las arquitecturas orientadas a servicios (SOAs), una revisión de estándares y perfiles relacionados a la interoperabilidad entre los sistemas de información sanitarios para finalizar describiendo el contexto nacional en que se desarrolla el proyecto y mencionando algunos trabajos relacionados.

2.1 Seguridad Informática

En esta sección se describen los conceptos de seguridad informática que se utilizarán en el resto del documento.

2.1.1 Identidad

Una identidad es la representación en un sistema de una entidad, que puede ser una persona, una aplicación o un servicio. Comúnmente los sistemas representan una identidad con un objeto abstracto denominado cuenta de usuario que contiene un conjunto de atributos de la entidad que representa. Este objeto posee un nombre (identificador de usuario) que es utilizado para representar el objeto abstracto en el sistema y puede tener atributos adicionales como por ejemplo el nombre completo de la persona que representa, el departamento para el cual trabaja, etc. [1]

2.1.2 Identificación y Autenticación

La Autenticación se puede ver como un proceso de dos fases: (1) Identificación y (2) Autenticación propiamente dicha.

La Identificación es el proceso por el cual una identidad es presentada a un sistema de seguridad. Comúnmente en la forma de un identificador de usuario.

La Autenticación es el proceso de verificar una identidad ante un sistema. Para esto la entidad cuya identidad se intenta validar, debe presentar pruebas al sistema para que este pueda verificar su identidad. Estas pruebas se denominan credenciales.

La verificación de una entidad frente a un sistema se puede realizar:

- Indicando algo que se sabe, por ejemplo una contraseña.
- Mostrando algo que se tiene, por ejemplo un *badge* o un *token*.
- Permitiendo que el sistema "mida" o determine algo que se es, por ejemplo huellas digitales, ADN, reconocimiento de iris.

Algunos sistemas combinan estos enfoques. Por ejemplo, los cajeros automáticos suelen utilizar una tarjeta magnética y además requieren que el usuario ingrese un número de identificación personal (PIN). Es recomendable siempre usar al menos dos de estos métodos, porque algo que una persona tiene puede ser objeto de un robo, algo que se sabe puede ser descifrado al enviarlo a través de Internet, pero es más difícil que alguien logre acceso sobre los dos elementos. [1] [2]

2.1.3 Autorización

La autorización es el proceso que verifica si una identidad previamente autenticada, tiene realmente permiso para ejecutar la acción que está solicitando realizar sobre cierto recurso.

La autorización se logra previamente asignando permisos de acceso para realizar una determinada acción, como ser de lectura, escritura o borrado a identificadores de usuario o grupos, sobre los objetos que se quiere controlar. Luego en el momento en que el usuario intenta acceder al recurso para realizar una acción, se verifica que el usuario o grupo al que pertenece el usuario tenga asociado el correspondiente privilegio sobre el recurso y sólo en caso afirmativo se permite el acceso. [2]

2.1.4 Trazabilidad

Los usuarios son responsables de sus acciones en un sistema informático. Los mismos pueden ser autorizados para acceder a cierto recurso, y si acceden efectivamente, el sistema operativo o aplicación necesita proveer una traza de auditoría que permite mantener información histórica sobre cuándo y cómo fueron accedidos los recursos. Por otro lado, si un usuario trata de acceder a un recurso que no está autorizado, se necesita una traza de auditoría para determinar un intento de violación al sistema de autorización y ejecutar alguna acción al respecto.

Trazabilidad es el proceso de mantener trazas de las acciones que los usuarios realizan sobre un sistema. La trazabilidad puede ser útil desde una perspectiva de seguridad para determinar los accesos autorizados y no autorizados, así como también de las autenticaciones exitosas y no exitosas. [1]

2.1.5 No Repudio

El No repudio es la garantía de que el remitente tiene un comprobante de que el destinatario recibió el mensaje, y que el destinatario tiene un comprobante de la identidad del remitente, y por lo tanto ninguno de los dos puede luego, negar haber procesado la información. A través de las firmas digitales se puede garantizar el no repudio. [3]

2.1.6 Confidencialidad de la información

Es la propiedad de que la información sensible no sea revelada a individuos, entidades o procesos no autorizados. [3]

2.1.7 Integridad de la información

Es la propiedad de que la información sensible no ha sido modificada o eliminada sin autorización. [3]

2.1.8 Disponibilidad de la información

Es la propiedad de que la información sea accesible y usable por las entidades con la debida autorización. [3]

2.1.9 Cifrado Simétrico

En un cifrado simétrico el remitente y el destinatario del mensaje, deben tener una clave compartida que debe ser secreta para todas las demás partes. El remitente cifra el mensaje con dicha clave y el destinatario la usa para descifrar el mensaje. [4]

2.1.10 Cifrado Asimétrico o de clave pública

En este tipo de cifrado se tienen dos claves, una para cifrar y otra para descifrar. Lo que se cifra con una de las claves, solamente puede descifrarse con la otra. Es computacionalmente imposible obtener la clave de descifrado a partir de la de cifrado. De esta forma una de las claves puede hacerse pública. El remitente cifra con la clave privada y el destinatario realiza el descifrado con la clave pública. [4]

2.1.11 Firma Digital

La firma digital es un mecanismo criptográfico que permite al receptor de un mensaje o documento firmado digitalmente verificar la autenticidad de la identidad de la entidad originadora y la integridad de la información recibida.

El uso de cifrado asimétrico sirve como firma digital. Se realiza el encriptado del mensaje o documento con la clave privada y luego alcanza con verificar que es desencriptado correctamente con la clave pública. De esta forma se verifica la identidad del firmante y la integridad de los datos.

Debido a que los algoritmos de encriptación asimétricos son lentos, se suele encriptar un hash del mensaje. [4]

2.1.12 XML Signature

XML signature es un estándar recomendado por la W3C, que consiste de una serie de elementos XML que pueden ser embebidos en un documento XML, permitiendo autenticar al remitente del mismo así como también garantizar la integridad del documento.

En la Figura 2 se muestra un ejemplo de firma digital con XML Signature.

```
<ds:Signature>
  <ds:SignedInfo>
    <ds:CanonicalizationMethod Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
    <ds:SignatureMethod  Algorithm= "http://www.w3.org/2000/09/xmldsig#hmac-sha1"/>

    <ds:Reference URI="#MsgBody">
      <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
      <ds:DigestValue>LyLsF0Pi4wPU...</ds:DigestValue>
    </ds:Reference>
  </ds:SignedInfo>

  <ds:SignatureValue>DJbchm5gK...</ds:SignatureValue>

  <ds:KeyInfo>
    <wsse:SecurityTokenReference>
      <wsse:Reference URI="#MyID"/>
    </wsse:SecurityTokenReference>
  </ds:KeyInfo>
</ds:Signature>
```

Figura 2: Ejemplo de firma XML

El elemento *Signature* encapsula la firma digital. Contiene tres sub-elementos: *SignedInfo*, *SignatureValue* y *KeyInfo*. El elemento *SignedInfo* contiene información sobre qué es lo que se firma y cómo se firma. Este elemento especifica el algoritmo de transformación que se realiza a la información antes de realizar la firma en el elemento *CanonicalizationMethod* y el método utilizado para calcular el valor de la firma en el elemento *SignatureMethod*. El elemento *SignedInfo* también contiene referencias a los elementos XML que se firman (*Reference*) que incluyen el método utilizado para calcular el hash (*DigestMethod*) y el valor del hash (*DigestValue*). El resultado del cálculo de la firma se indica en el elemento *SignatureValue*. El elemento *KeyInfo* es un elemento opcional que indica la clave que ha de utilizarse para validar la firma. En algunos casos la clave pública se encuentra en el *token* de seguridad por lo que en este elemento puede incluirse una referencia al mismo. [5] [6] [7] [8]

2.2 Tecnologías de Web Services

2.2.1 Definición

Un Web Service es una aplicación de software que puede ser descripta, publicada, descubierta, coordinada y configurada a través de artefactos XML, con el propósito de permitir el desarrollo de aplicaciones distribuidas e interoperables.

La tecnología de web services se basa principalmente en tres estándares que cumplen una función específica en el desarrollo de dichas aplicaciones. Se utiliza el Simple Object Access Protocol (SOAP) como protocolo de mensajería basado en XML, el Web Service Description Language (WSDL) y el Universal Description Discovery and Integration (UDDI) para describir y descubrir servicios respectivamente. [9] [10] [11]

En las siguientes sub secciones se describen más en detalle los diferentes estándares de Web Services.

2.2.2 Estándares básicos de Web Services

En esta sección se describen los estándares básicos en los que se apoya la tecnología de Web Services.

2.2.2.1 Simple Object Access Protocol (SOAP)

El SOAP es un protocolo para el intercambio de información estructurada en un ambiente descentralizado y distribuido. Utiliza las tecnologías XML para definir un framework de mensajería y un formato de mensaje. No limita el protocolo a utilizar para el intercambio de dichos mensajes, pudiéndose utilizar diferentes opciones como por ejemplo HTTP o FTP. Es muy común utilizarlo con HTTP, lo que habilita a utilizar SOAP en ambientes donde los mensajes pueden ser filtrados por firewalls.

El mensaje SOAP es un documento XML formado por un *envelope* que contiene una sección de encabezados (*header*) y un cuerpo (*body*). En el encabezado se envía información de control o de proceso para indicar al receptor como procesar el mensaje, por ejemplo aspectos de seguridad o transaccionalidad. El cuerpo del mensaje SOAP es el área donde viaja la información en formato XML específica de la aplicación (el *payload*) que está siendo intercambiada.

El estándar también provee una forma para enviar errores en un mensaje SOAP, mediante una sección *Fault* que se envía dentro del cuerpo del mensaje.

En la Figura 3 se puede ver un ejemplo que invoca la operación de consulta de stock (getStock) de un servicio pasando por parámetro el tipo de producto (ítem). [9] [10] [11] [12]

```
<env:Envelope
      xmlns:env="http://www.w3.org/2003/05/soap-envelope"
      xmlns:pdg="http://pdg.lins.inco.fing.edu.uy/webservices/samples>
   ...
<env:Body>
   <pdg:getStock>
     <item>pencil</item>
   </pdg:getStock>
</env:Body>
</env:Envelope>
```

Figura 3: Ejemplo de mensaje SOAP

2.2.2.2 Web Services Description Language (WSDL)

WSDL es un lenguaje para la descripción de interfaces de servicios basado en XML. Un documento WSDL describe cómo invocar un web service y provee información de los tipos de datos intercambiados, los mensajes de solicitud y respuesta de las operaciones, los protocolos utilizados y la ubicación del servicio. El WSDL se divide en dos partes, una denominada descripción abstracta o definición, y la otra, descripción concreta o implementación.

La descripción abstracta contiene los elementos *type*, *message* y *portType*, los cuales especifican las descripciones funcionales y a nivel de datos de las operaciones que provee el Web Service. La descripción abstracta puede ser referenciada por varias implementaciones.

La descripción concreta especifica la ubicación del servicio y el protocolo a utilizar. Contiene los elementos *binding* y *port*. Los elementos binding indican a través de qué protocolos interactuar con el servicio, por ejemplo SOAP sobre HTTP. Los elementos *port* proveen una dirección específica a través de la cual el servicio puede ser invocado. [9] [10] [11]

En la Figura 4 y la Figura 5 se presentan la descripción WSDL abstracta y concreta, respectivamente, de un servicio denominado *WsStockService* el cual tiene una operación denominada *getStock* que recibe como parámetro un String y devuelve como respuesta un valor entero.

```
<definitions name='WsStockService'
        targetNamespace='http://pdg.lins.inco.fing.edu.uy/webservices/samples/'
        xmlns='http://schemas.xmlsoap.org/wsdl/'
        xmlns:soap='http://schemas.xmlsoap.org/wsdl/soap/'
        xmlns:tns='http://pdg.lins.inco.fing.edu.uy/webservices/samples/'
        xmlns:xsd='http://www.w3.org/2001/XMLSchema'>
  <types>
    <xs:schema targetNamespace='http://pdg.lins.inco.fing.edu.uy/webservices/samples/'
          version='1.0'
          xmlns:tns='http://pdg.lins.inco.fing.edu.uy/webservices/samples/'
          xmlns:xs='http://www.w3.org/2001/XMLSchema'>
      <xs:element name='getStock' type='tns:getStock'/>
      <xs:element name='getStockResponse' type='tns:getStockResponse'/>
      <xs:complexType name='getStock'>
        <xs:sequence>
          <xs:element minOccurs='0' name='item' type='xs:string'/>
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name='getStockResponse'>
        <xs:sequence>
          <xs:element minOccurs='0' name='num' type='xs:int'/>
        </xs:sequence>
      </xs:complexType>
    </xs:schema>
  </types>

  <message name='WsStock_getStock'>
    <part element='tns:getStock' name='getStock'></part>
  </message>
  <message name='WsStock_getStockResponse'>
    <part element='tns:getStockResponse' name='getStockResponse'></part>
  </message>

  <portType name='WsStock'>
    <operation name='getStock'>
      <input message='tns:WsStock_getStock'></input>
      <output message='tns:WsStock_getStockResponse'></output>
    </operation>
  </portType>
  ...
</definitions>
```

Data types transmitidos

Mensajes intercambiados

Port type con una operación

Figura 4: Ejemplo descripción abstracta de un WSDL


```
<binding name='WsStockBinding' type='tns:WsStock'>
  <soap:binding style='document'
                transport='http://schemas.xmlsoap.org/soap/http'/>

  <operation name='getStock'>
    <input>
      <soap:body use='literal'/>
    </input>
    <output>
      <soap:body use='literal'/>
    </output>
  </operation>
</binding>

<service name='WsStockService'>
  <port binding='tns:WsStockBinding' name='WsStockPort'>
    <soap:address
          location='http://localhost:8080/WebServiceSamples/WsStock'/>
  </port>
</service>
```

Figura 5: Ejemplo de descripción concreta de un WSDL

2.2.2.3 Universal Description, Discovery and Integration (UDDI)

UDDI es un estándar que provee la infraestructura para clasificar, catalogar y administrar web services con el propósito de permitir el descubrimiento y consumo de los mismos.

A través de la UDDI las empresas pueden publicar y descubrir información sobre otros negocios y los servicios que estos proveen. El registro mantiene información de las empresas o entidades de negocio y para cada una de ellas, la lista de servicios que proveen. Para cada servicio mantiene la información referente al acceso al mismo para poder invocarlo.

Las interfaces definidas por UDDI son expuestas como Web Services descriptos a través de WSDL. [9] [10] [11]

2.2.2.4 MTOM y XOP

XML-Binary Optimized Packaging (XOP) [13], es una especificación que define una forma optimizada de serializar un conjunto de información XML que contiene cierto tipo de contenido codificado en base 64, permitiendo extraer y empaquetar el contenido en paquetes MIME (*Multipurpose Internet Mail Extensions*) para transportar de forma optimizada (decodificado) y referenciando las distintas partes en el XML, por medio de URIs.

Message Transmission Optimization Mechanism (MTOM) [14] es un estándar de la W3C que define un mecanismo de optimización abstracto para la trasmisión de mensajes SOAP, con el fin de ser utilizado para enviar archivos binarios en mensajes SOAP.

SOAP define una forma de estructurar un mensaje utilizando XML. Para el envío de un archivo no XML, por ejemplo un binario, éste se codifica en base 64 (Figura 6). Esto implica trasformar el binario, tomando palabras de 6 bits y convertirlo en un carácter ASCII, utilizando solo 64 caracteres para ello. Dado que esto aumenta el tamaño original del archivo en un 33%, para archivos grandes resulta muy costoso. Las dos especificaciones MTOM/XOP se relacionan para describir una forma optimizada para serializar los mensajes SOAP con contenido no XML, utilizando MIME Multipart/Related lo cual permite enviar el

archivo como partes relacionadas fuera del *envelope* del mensaje SOAP sobre http. MTOM utiliza el XOP dentro del mensaje SOAP conteniendo la referencia a las partes MIME (Figura 7).

MTOM/XOP posibilita el envío de archivos grandes en su formato original (JPGE, PNG, GIF, etc.) y realizando streaming. [15] [16]

```
... other transport headers ...
Content-Type: text/xml; charset=UTF-8

<?xml version="1.0" encoding="UTF-8"?>
   <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
      <soapenv:Header/>
      <soapenv:Body>
         <sendImage xmlns="http://org/apache/axis2/jaxws/sample/mtom">
            <input>
               <imageData>R0lGOD1  ... more base64 encoded data ...  KTJk8giAAA7</imageData>
            </input>                                          elemento <xsd.base64Binary>
         </sendImage>
      </soapenv:Body>
   </soapenv:Envelope>
```

Figura 6: Mensaje SOAP con datos contenido en el documento codificado en base 64

```
... other transport headers ...
Content-Type: multipart/related; boundary=MIMEBoundaryurn_uuid_0FE43E4D025F0BF3DC11582467646812;
type="application/xop+xml"; start="<0.urn:uuid:0FE43E4D025F0BF3DC11582467646813@apache.org>";
start-info="text/xml"; charset=UTF-8

--MIMEBoundaryurn_uuid_0FE43E4D025F0BF3DC11582467646812                    Parte msg SOAP
content-type: application/xop+xml; charset=UTF-8; type="text/xml";
content-transfer-encoding: binary
content-id:
   <0.urn:uuid:0FE43E4D025F0BF3DC11582467646813@apache.org>

<?xml version="1.0" encoding="UTF-8"?>
        <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
           <soapenv:Header/>
           <soapenv:Body>
              <sendImage xmlns="http://org/apache/axis2/jaxws/sample/mtom">
                 <input>
                    <imageData>
                       <xop:Include xmlns:xop="http://www.w3.org/2004/08/xop/include"
                            href="cid:1.urn:uuid:0FE43E4D025F0BF3DC11582467646811@apache.org"/>
                    </imageData>
                 </input>
              </sendImage>
           </soapenv:Body>
        </soapenv:Envelope>                                               Parte msg SOAP
--MIMEBoundaryurn_uuid_0FE43E4D025F0BF3DC11582467646812
content-type: text/plain
content-transfer-encoding: binary                          Referencia URI al archivo
content-id:
        <1.urn:uuid:0FE43E4D025F0BF3DC11582467646811@apache.org>

... binary data goes here ...
--MIMEBoundaryurn_uuid_0FE43E4D025F0BF3DC11582467646812--
```

Figura 7: Mensaje SOAP con uso de MTOM

2.2.3 Estándares avanzados de Web Services

En esta sección se detallan algunos de los estándares avanzados de Web Services, que son referenciados en el resto del documento. Estos estándares forman parte del conjunto denominado WS-*.

2.2.3.1 WS-Security

WS-Security es un estándar de la familia WS-* de la OASIS, que propone un conjunto estándar de extensiones a SOAP que pueden ser utilizadas para enviar *tokens* de seguridad como partes del mensaje SOAP y proveer integridad y confidencialidad del contenido de los mismos. El envío de *tokens* de seguridad permite autenticar el origen. Además define cómo utilizar XML Signature y XML Encryption para firmar y cifrar todo o parte del mensaje. [7] [8] [17]

WS-Security define un conjunto de security tokens y el mecanismo para adjuntarlos, identificarlos y referenciarlos dentro del mensaje. Soporta la utilización del *usernameToken* que permite especifiar el nombre de usuario y opcionalmente una contraseña. También permite adjuntar *tokens* en formato binario, el *BinarySecurityToken*, como por ejemplo los certificados X.509 o los tickets Kerberos. Finalmente, mediante los XMLTokens es posible adjuntar *tokens* basados en XML como Security Assertion Markup Language (SAML).

En la Figura 8 se puede ver la estructura general de un mensaje SOAP, con un header WS-Security. [8] [18] [19]

```
<?xml version="1.0" encoding="utf-8"?>
<S11:Envelope xmlns:S11="..." xmlns:wsse="..." xmlns:wsu="..." xmlns:ds="...">

  <S11:Header>
    <wsse:Security xmlns:wsse="...">

      <wsse:BinarySecurityToken
          ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-
token-profile-1.0#X509v3"
          wsu:Id="X509Token"
          EncodingType="...#Base64Binary">
             MIIEZzCCA9CgAwIBAgIQEmtJZc0rqrKh5i...
      </wsse:BinarySecurityToken>                                    Security token

      <xenc:EncryptedKey>
        <xenc:EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5"/>
        <ds:KeyInfo>
          <wsse:SecurityTokenReference>
            <wsse:KeyIdentifier
                EncodingType="...#Base64Binary"
                ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
x509-token-profile-1.0#X509v3">
                    MIGfMa0GCSq...
            </wsse:KeyIdentifier>
          </wsse:SecurityTokenReference>
        </ds:KeyInfo>

        <xenc:CipherData>
          <xenc:CipherValue>d2FpbmdvbGRfE0lm4byV0...
          </xenc:CipherValue>
        </xenc:CipherData>

        <xenc:ReferenceList>
          <xenc:DataReference URI="#EncBody"/>
        </xenc:ReferenceList>
      </xenc:EncryptedKey>                                           Clave de cifrado

      <ds:Signature>
        <ds:SignedInfo>
          ...
          <ds:Reference URI="#MsgBody">
            <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
            <ds:DigestValue>LyLsF0Pi4wPU...</ds:DigestValue>
          </ds:Reference>
        </ds:SignedInfo>
        <ds:SignatureValue>DJbchm5gK...</ds:SignatureValue>
          ...
      </ds:Signature>                                                Firma

    </wsse:Security>
  </S11:Header>

  <S11:Body Id="MsgBody">
    <xenc:EncryptedData Id="EncBody">
      <xenc:CipherData>
        <xenc:CipherValue>...</xenc:CipherValue>
      </xenc:CipherData>
    </xenc:EncryptedData>                                            Cifrado
  </S11:Body>

</S11:Envelope>
```

Figura 8: Mensaje SOAP con un header WS-Security

2.2.3.2 WS-Trust

El consumo de web services a través de los sistemas de información muchas veces requiere el intercambio de *tokens* de seguridad para autenticar la identidad del consumidor del servicio. Los consumidores y proveedores están distribuidos en diferentes dominios de seguridad que podrían utilizar diferentes tipos de *token*. Por esta razón la utilización de WS-Security para transmitir la información de autenticación de los usuarios no es suficiente para garantizar la interoperabilidad de la autenticación entre los clientes y los proveedores.

WS-Trust es una extensión de WS-Security que define una estructura y un mecanismo para que el consumidor pueda llevar a cabo en forma transparente la transformación del *token* de seguridad.

WS-Trust define mecanismos para administrar, establecer y evaluar relaciones de confianza entre dominios de seguridad federados, independientemente de los protocolos de seguridad utilizados. Para ello el estándar define el Security *Token* Service (STS) que permite emitir, renovar, validar y cancelar *tokens* de seguridad. Un STS emite *tokens* de seguridad con afirmaciones basadas en pruebas confiables para quienquiera que confíe en él o para un destinatario específico. Dichas afirmaciones deben estar firmadas por el STS para garantizar la procedencia a los servicios que confían en él.

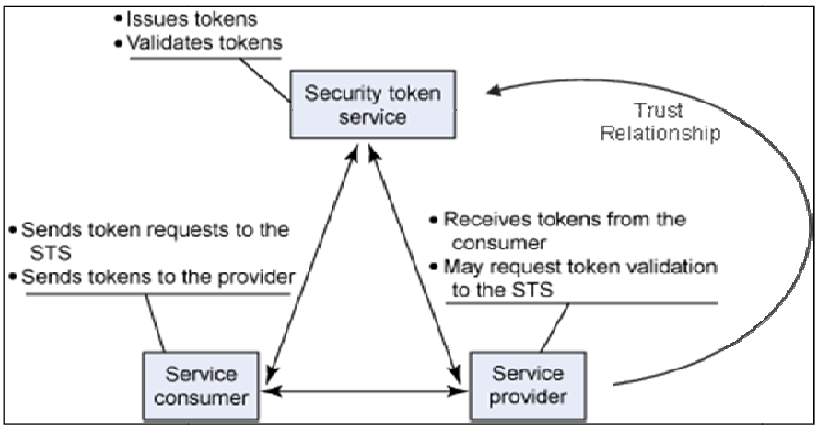


Figura 9: Modelo de seguridad WS-Trust [20]

En la Figura 9 se puede ver el modelo de confianza definido por WS-Trust para un dominio de seguridad. El modelo funciona como se explica a continuación. Existe un servicio que para ser consumido requiere un *token* de seguridad. Los clientes para consumirlo, deben obtener previamente un *token* de seguridad del STS y adjuntarlo en el mensaje de solicitud al servicio. Dado que el servicio sólo puede confiar en el *token* emitido por el STS o consultarle su validez, existe una fuerte relación de confianza a lo que determine el STS.

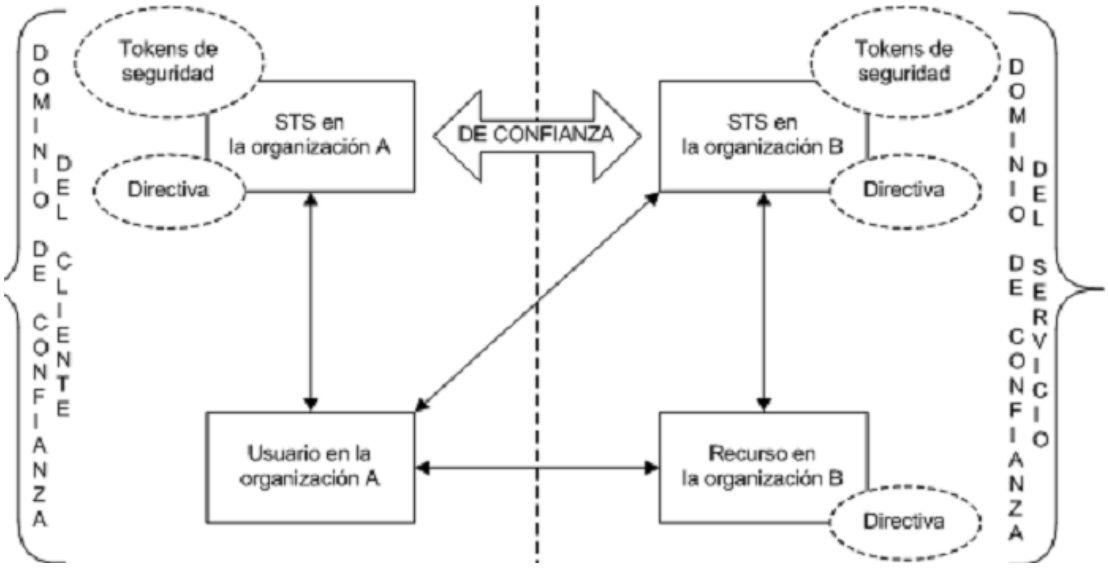


Figura 10: Modelo de Confianza Federado [21]

En la Figura 10 se puede ver el modelo de confianza WS-Trust federado. Cuando el sistema cliente consume un servicio que se encuentra en una organización perteneciente a otro dominio se genera una relación de confianza entre los STS de cada dominio.

En estos casos el usuario del dominio A solicita un *token* al STS de la organización A, presentando las credenciales que usualmente utiliza dentro de la organización. De esta forma el usuario siempre utiliza las mismas credenciales.

Luego envía ese *token* al STS de la organización B solicitando otro *token* para consumir el servicio que se encuentra en la organización B. El STS de B tiene una relación de confianza con el STS de A, confía en la firma del STS A en el *token* recibido y proporciona un nuevo *token* firmado por él para consumir el servicio. El cliente presenta este *token* al servicio dentro de la organización B. Al recibir el *token*, el servicio confía en el *token* emitido por el STS B o consulta la validez del mismo al STS B. [20] [21] [22] [23] [24]

2.3 Estándares de seguridad

2.3.1 Security Assertion Markup Language (SAML)

SAML es un estándar de la OASIS que provee un framework basado en XML para comunicar información de autenticación, autorización y atributos de los usuarios. Esta información es comunicada en aserciones SAML, en las que aplicaciones que trabajan en distintos dominios de seguridad pueden confiar. El estándar define las reglas para solicitar, crear, comunicar y usar las aserciones SAML.

2.3.1.1 SAML Assertions

SAML permite realizar afirmaciones con cierta información de seguridad en forma de declaraciones (*statements*) sobre un sujeto (*subject*). Por ejemplo una aserción SAML puede decir que un sujeto se llama "Juan Pérez" y su dirección de e-mail es “juanperez@mail.com”.

Una aserción contiene información básica obligatoria y opcional que aplica para todas sus declaraciones, y usualmente contiene un sujeto de la aserción (en caso de no estar presente la identidad del sujeto se confirma de otra forma, como por ejemplo un

certificado). También contiene condiciones (*conditions*) que determinan la validez de la aserción.

Hay tres tipos de declaraciones que se pueden incluir en una aserción:

- **Authentication statements:** Son creados por la organización que realizó la autenticación del usuario. Como mínimo describen el medio utilizado para autenticar al usuario y el momento exacto en qué se realizó dicha autenticación.
- **Atribute statements:** Contienen atributos específicos para identificar al sujeto, como por ejemplo el nombre, la dirección de correo electrónico o el grupo al que pertenece.
- **Authorization decision statements:** Contienen información sobre lo que el sujeto está autorizado a hacer, por ejemplo "Juan Pérez" está autorizado a comprar cierto elemento.

En la Figura 11 se muestra un ejemplo de aserción SAML.


```
<saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" Version="2.0"
IssueInstant="2005-01-31T12:00:00Z">
  <saml:Issuer Format=urn:oasis:names:SAML:2.0:nameid-format:entity>
    http://idp.example.org
  </saml:Issuer>
  <saml:Conditions
    NotBefore="2005-01-31T12:00:00Z" NotOnOrAfter="2005-01-31T12:10:00Z">
  </saml:Conditions>
  <saml:AuthnStatement
    AuthnInstant="2005-01-31T12:00:00Z" SessionIndex="67775277772">
    <saml:AuthnContext>
      <saml:AuthnContextClassRef>
      urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport
      </saml:AuthnContextClassRef>
    </saml:AuthnContext>
  </saml:AuthnStatement>
  <saml:AttributeStatement>
    <saml:Attribute Name="urn:oid:2.5.4.42" FriendlyName="givenName">
      <saml:AttributeValue xsi:type="xs:string">Juan</saml:AttributeValue>
    </saml:Attribute>
    <saml:Attribute Name="LastName">
      <saml:AttributeValue xsi:type="xs:string">Pérez</saml:AttributeValue>
    </saml:Attribute>
  </saml:AttributeStatement>
</saml:Assertion>
```

Figura 11: Ejemplo de Aserción SAML [25]

2.3.2 eXtensible Access Control Markup Language (XACML)

XACML es un estándar de OASIS basado en XML que define un lenguaje de políticas de acceso y un lenguaje solicitud/respuesta (*request/response*) para la toma de decisiones de control de acceso. El lenguaje de políticas es utilizado para describir los requerimientos generales de control de acceso, y tiene algunos puntos de extensión para la definición de nuevas funciones, tipos de datos, lógica de la combinación, etc. El lenguaje de solicitud/respuesta permite realizar consultas para determinar si una cierta acción puede o no, realizarse e interpretar el resultado. Las respuestas siempre incluyen uno de los siguientes cuatro valores: Permit, Deny, Indeterminate (ocurrió algún error o falta algún valor que hace que la decisión no pueda ser tomada), Not Applicable (la petición no puede ser respondida por este servicio).

A continuación se detalla el modelo de flujo de información XACML y los componentes que intervienen. [26]

Modelo de flujo de Información [27]

La descripción del flujo comienza con la definición de algunos conceptos clave en el flujo XACML.

Policy Administration Point (PAP)

Es la entidad del sistema que crea las políticas o conjuntos de políticas de acceso.

Policy decision point (PDP)

Es la entidad del sistema que evalúa las políticas aplicables y emite una decisión de autorización.

Policy enforcement point (PEP)

Es la entidad del sistema que ejecuta el control de acceso, realizando solicitudes de decisión de autorización y haciendo cumplir las decisiones tomadas.

Policy information point (PIP)

Es la entidad del sistema que actúa como fuente de atributos.

Context handler

Es la entidad del sistema que convierte las solicitudes de decisión del formato nativo de solicitud a la forma canónica XACML y convierte las decisiones de autorización en la forma canónica XACML al formato nativo de respuesta. En la Figura 12 se presenta el flujo de información XACML.

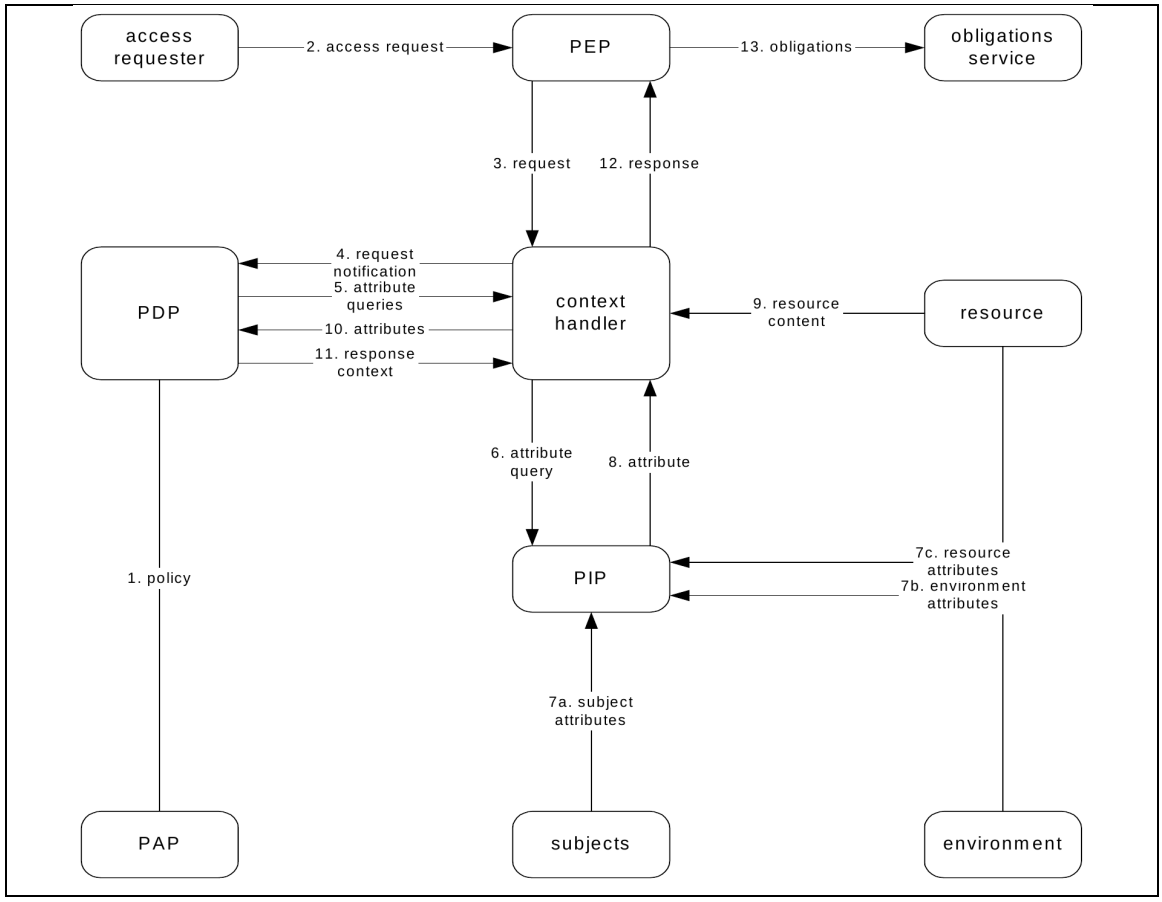


Figura 12: Flujo de información XACML [27]

1. El Administrador de políticas crea las políticas y conjuntos de políticas en el PAP y los deja disponibles para el PDP.
2. El solicitante de acceso (*access requester*) envía una solicitud de acceso al PEP.
3. El PEP envía la solicitud de acceso al *context handler* en su formato nativo, incluyendo en forma opcional atributos del sujeto, recurso, acción y entorno.
4. El *context handler* construye una solicitud XACML y se la envía el PDP.
5. El PDP solicita atributos adicionales del sujeto, recurso, acción y entorno al *context handler.*
6. El context handler solicita los atributos a un PIP.
7. El PIP obtiene los atributos solicitados
8. El PIP retorna los atributos solicitados al *context handler*.
9. Opcionalmente el *context handler* puede incluir el recurso en el contexto.
10. El *context handler* envía los atributos solicitados y opcionalmente el recurso al PDP. El PDP evalúa las políticas.
11. El PDP retorna la respuesta de contexto al *context handler* incluyendo la decisión de autorización.
12. El *context handler* traduce la respuesta de contexto al formato de respuesta nativo del PEP. El *context handler* retorna la respuesta al PEP.
13. El PEP hace cumplir las obligaciones (operaciones especificadas en las políticas que deben ser realizadas por el PEP al momento hacer cumplir la decisión de autorización) que implican las políticas que se están controlando.
14. Si la decisión indica que el acceso es permitido, el PEP permite el acceso al recurso; de lo contrario el acceso es denegado.

2.4 Service Oriented Architecture (SOA)

La computación orientada a servicios (Service Oriented Computing, SOC) es un paradigma de computación que utiliza servicios como elementos fundamentales para dar soporte al desarrollo rápido y de bajo costo de aplicaciones distribuidas en ambientes heterogéneos. [28] Los Servicios son entidades de software autónomas, auto-contenidas e independientes de la plataforma, que proveen funcionalidades de negocio, tienen una interfaz pública y pueden ser descubiertas, invocadas y combinadas de forma dinámica.

La implementación del paradigma SOC, involucra el desarrollo de Arquitecturas Orientadas a Servicios (SOAs). Una SOA es una forma lógica de diseñar un sistema de software para proveer servicios a usuarios finales, aplicaciones u otros servicios a través de interfaces que pueden ser publicadas y descubiertas. [9] Para esto es necesario contar con las

correspondientes tecnologías de middleware que permitan hacer posible el descubrimiento, utilización y combinación de servicios interoperables para dar soporte a los procesos de negocio.

Los principios de diseño de una SOA son independientes de una tecnología específica. En particular SOA especifica que todas las funcionalidades que brinde una aplicación basada en SOA son provistas como servicios. Es decir, los servicios de SOA incluyen todas las funcionalidades y procesos de negocio que componen la aplicación, así como cualquier otra funcionalidad relacionada al sistema que sea necesaria para dar soporte a una aplicación basada en SOA. [28]

Una arquitectura SOA define tres roles principales, el Proveedor de Servicios, el Registro de Servicios y el Consumidor de Servicios (o cliente). El Consumidor de servicios es una aplicación, módulo de software o servicio, que busca servicios en el registro y ejecuta las funcionalidades de negocio que estos proveen. El proveedor de servicios es una entidad accesible desde la red que acepta y ejecuta los pedidos de los consumidores y es responsable de publicar sus servicios en el registro. Por último el Registro de Servicios proporciona mecanismos para que los consumidores puedan buscar y descubrir servicios.

Las SOAs facilitan las tareas del desarrollo de aplicaciones distribuidas como su integración, procesos de negocio y reuso de sistemas legados. Además, SOA provee la flexibilidad y agilidad que los usuarios de negocio requieren, permitiendo definir servicios de granularidad gruesa que pueden ser combinados y reutilizados para resolver necesidades de negocio, actuales o futuras. [29]

Los principios de diseño de una SOA no restringen el uso de una determinada tecnología. Sin embargo los Web Services se han convertido en la tecnología preferida para implementar una arquitectura SOA, dado que basan su desarrollo en infraestructura y estándares ampliamente difundidos como HTTP, SOAP, and XML. [28]

2.5 Interoperabilidad en Sistemas de Información Sanitarios

La interoperabilidad según el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) es definida como la habilidad de dos o más sistemas o componentes para intercambiar información y utilizar la información intercambiada. [30]

La interoperabilidad en salud según define la Sociedad de Sistemas de Información y Gestión en Sanidad (HIMMS de sus siglas en inglés) [31]: es la capacidad que diferentes sistemas de tecnología de la información o aplicación de software tienen, para comunicarse, intercambiar datos y utilizar la información que se ha intercambiado. Los estándares y el esquema de datos intercambiado deben permitir que los datos compartidos por médicos, laboratorios, hospitales, farmacia y el paciente sean independientes de la aplicación o proveedor. Es decir, interoperabilidad significa la capacidad de los sistemas de salud para trabajar juntos dentro y fuera de la frontera de la organización con el fin de prestar una mejor asistencia para el paciente y la comunidad.

Existen tres niveles de interoperabilidad en los sistemas de salud: fundacional, estructural y semántica. [31]

Interoperabilidad fundacional, permite el intercambio de datos entre los sistemas de salud y no se requiere de la interpretación de estos por el sistema receptor de la información.

Interoperabilidad estructural, es un nivel intermedio donde se define la estructura o formatos de los datos intercambiados. Tiene una finalidad clínica u operacional donde el significado de los datos se conservara sin alteraciones. Este tipo de interoperabilidad

define la sintaxis del intercambio de datos, asegurando la interpretación a nivel de campos por parte de los sistemas de información.

Interoperabilidad semántica, es el nivel más alto y en el cual dos o más sistemas o elementos tienen la capacidad de intercambiar información y utilizar la información que se ha intercambiado. Este nivel se aprovecha tanto de la estructura y codificación de los datos, incluyendo el vocabulario para que los sistemas puedan interpretar los datos.

La Interoperabilidad entre sistemas y equipos es reconocida como un factor clave para mejorar la calidad de la atención médica.

En salud existen diferentes tipos de sistemas, así como estándares y protocolos utilizados, lo que lleva a que interoperar sistemas de salud sea una tarea compleja y costosa. Para poder identificar y resolver estos problemas se realizó un relevamiento de estándares y perfiles, tomando como referencia un conjunto de soluciones y trabajos realizados a nivel internacional.

Los estándares y perfiles más relevantes considerados en el contexto de este proyecto corresponden a los definidos por organizaciones relacionadas a salud e interoperabilidad, como son: HL7 (Health Level Seven Internacional), OASIS (Organization for the Advancement of Structured Information Standars), W3C (World Wide Web Consortium), DICOM (Digital Imaging and Communication in Medicine) e IHE (Integrating the Healthcare Entreprise).

2.5.1 Health Level Seven Internacional (HL7)

HL7 es una organización sin fines de lucro, dedicada a la definición de estándares acreditados por ANSI (American National Standards Institute [32]), para intercambiar, integrar, compartir y recuperar información electrónica de salud que soporte la práctica clínica, la administración, la realización y la evaluación de los servicios de salud. [33]

2.5.1.1 Reference InformationModel (RIM)

El RIM es el modelo de información de referencia definido por HL7. Es un diagrama representado en UML que posee más de 70 clases, y representa todos los datos clínicos (dominios que pueden ser por ejemplo: clínico, asistencial, administrativo, logístico etc.). Es un modelo compartido por todos los dominios que HL7 define en el ámbito de la Salud [33].

En la Figura 13, se muestran las seis clases fundamentales del RIM así como también las relaciones entre ellas:

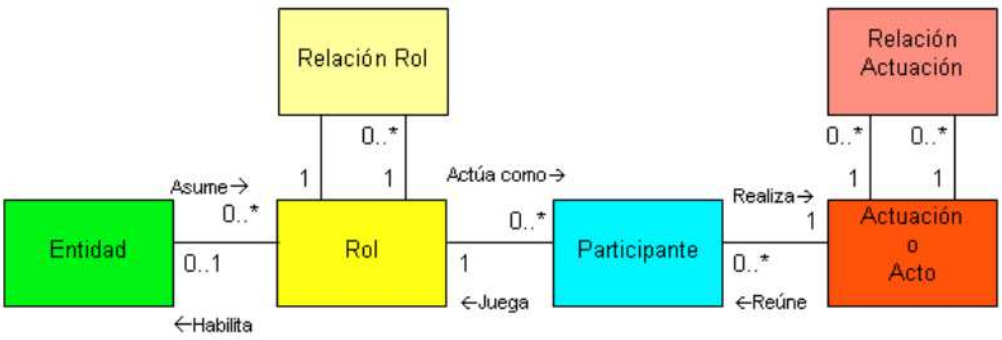


Figura 13: Clases y relaciones base del RIM [34]

2.5.1.2 Clinical Document Architecture (CDA)

CDA es la arquitectura clínica de documentos definida por HL7. Es un estándar basado en XML para el marcado de documentos, que especifica la estructura y semántica de documentos clínicos con el propósito de facilitar su intercambio en un entorno de interoperabilidad. Su utilización en conjunto con el RIM y vocabularios codificados (LOINC [35], SNOMED [36], etc.), convierte a los documentos clínicos, en objetos interpretables por multitud de aplicaciones y transferibles a través de cualquier medio electrónico [33] [34].

La Figura 14 muestra la estructura general de un documento CDA:

Figura 14: Estructura general de un documento CDA

La estructura general de un documento CDA se divide en dos elementos principales que son el encabezado y el cuerpo. El encabezado contiene los metadatos que describen el documento, contiene datos como por ejemplo: el autor, la fecha de creación, el tipo de documento, la información del paciente, la información del médico etc.; y el cuerpo que contiene la información clínica y que dependiendo del nivel de interoperabilidad del mismo puede solamente contener datos binarios, o bloques descriptivos.

Un documento CDA posee tres niveles de interoperabilidad, a saber:

- **Nivel 1**: incluye el encabezado del CDA más un cuerpo que se compone de datos binarios sin estructurar, como por ejemplo un PDF, DOC o incluso una imagen digitalizada
- **Nivel 2**: incluye el encabezado del CDA más un cuerpo representado en formato XML con bloques descriptivos. Cada sección se identifica con un código
- **Nivel 3**: incluye el encabezado del CDA más un cuerpo en XML con bloques descriptivos y entradas codificadas. La sección debe ser codificada con todo el poder de RIM mediante terminologías tales como LOINC [35], SNOMED [36], etc.

En la Figura 15 se muestra un de un ejemplo de CDA:

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<?xml-stylesheet type="text/xsl" href="CDA_EstiloSaludUy.xsl"?>
<!-- CDA de Laboratorio
     Fecha : Marzo 24 de 2014
     Autor : Salud.uy
-->
<ClinicalDocument xmlns="urn:hl7-org:v3" xmlns:voc="urn:hl7-org:v3/voc"
                  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                           xsi:schemaLocation="urn:hl7-org:v3 CDA.xsd">
   <typeId root="2.16.840.1.113883.1.3" extension="POCD_HD000040"/>                Encabezado
   <id root="2.16.858.2.10004427.67430.150748.1"/>
   <code code="26436-6" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"
   <title>CDA LABORATORIO SALUD.UY</title>
   <effectiveTime value="201403201048"/>
   <confidentialityCode code="N" codeSystem="2.16.840.1.113883.5.25"/>
   <languageCode code="es-UY"/>
   <versionNumber value="1"/>
   <recordTarget>
      <patientRole>
         <id root="2.16.858.1.858.68909.12345678"/>
         <patient>
            <name>
               <given>Juan Alberto</given>
               <family>Perez Lopez</family>
            </name>
            <administrativeGenderCode code="1"
                 codeSystem="2.16.858.2.10000675.69600" displayName="Masculino"/>
            <birthTime value="19670328"/>
         </patient>
       </patientRole>
   </recordTarget>
   <author>
      ...
   </author>
   <custodian>
       ...
   </custodian>
   <component>                                                                     Cuerpo
      <structuredBody>
         <component>
            <section>
               <code code="7574" codeSystem="2.16.840.1.113883.10.1.1.3"/>
               <title>HEMOGLOBINA</title>
               <text>
                  <paragraph> Eritrocitos = 100</paragraph>
                  <paragraph> Globulos rojos  = 12</paragraph>
                  <paragraph> Globulos blancos  = 20 </paragraph>
                  <paragraph/>
               </text>
            </section>
         </component>
        <component>
          <section>
            <title>PSA</title>
               <text>
               <paragraph>Se oserva : ------   </paragraph>
               </text>
            </section>
         </component>

      </structuredBody>
   </component>
</ClinicalDocument>
```

Figura 15: Ejemplo de CDA

2.5.2 Perfiles IHE

IHE (Integrating the Healthcare Entreprise) [37] [38] es una organización que resuelve distintas problemáticas de interoperabilidad en el área de la salud a través de la definición de perfiles [39] que recomiendan el uso de distintos estándares. Cada perfil se describe en base a un conjunto de actores y transacciones que se basan en estándares de la industria [40] [41]. Estos perfiles son clasificados en categorías según el dominio clínico para el cual aplica el perfil, como por ejemplo Radiología, Cardiología, Farmacia, etc. [42] Particularmente hay un conjunto de problemáticas de infraestructura comunes a todos los dominios clínicos que se agrupan en el dominio de Infraestructura en Tecnologías de Información (ITI - Information Technologies Infrastructure), que son los utilizados para el propósito de este proyecto. [43]

Los perfiles de IHE, hacen a un sistema interoperable ya que las soluciones definidas en estos perfiles se basan en la aplicación coordinada de estándares y en la definición de actores y transacciones que los utilizan. Los perfiles hacen a la solución interoperable, más escalable y con menor costo de mantenimiento frente a soluciones hechas a medida.

A continuación se presenta una reseña de los perfiles más relevantes para el proyecto y que son referenciados a lo largo del documento. Se presenta una descripción del perfil, junto a los actores y transacciones que éste define, y que se pueden encontrar en el marco técnico correspondiente al dominio infraestructura publicados por IHE (IHE IT Infrastructure Technical Framework [43]). Las transacciones entre los actores aparecen referenciadas en el documento, con el formato IHE como por ejemplo: PIX Query [ITI-9] correspondiente al Marco técnico sección 9 (ITI TF-2a: 3.9).

2.5.2.1 Patient Identifier Cross-Referencing (PIX)

El perfil PIX resuelve el problema de identificación de pacientes en varios dominios, manteniendo las referencias cruzadas entre éstos. Permite notificar a los sistemas interesados sobre cambios en los datos de los pacientes, realizar consultas con el identificador del paciente de cualquier dominio y así obtener el identificador cruzado correspondiente en otro dominio para el cual se requiere consultar información. El perfil se plantea con tres actores y las transacciones necesarias entre ellos para cumplir con los requerimientos de interoperabilidad. [39]

La Figura 16 muestra los actores y transacciones definidos por IHE para este perfil. Una descripción más detalla se puede ver en el Apéndice A.

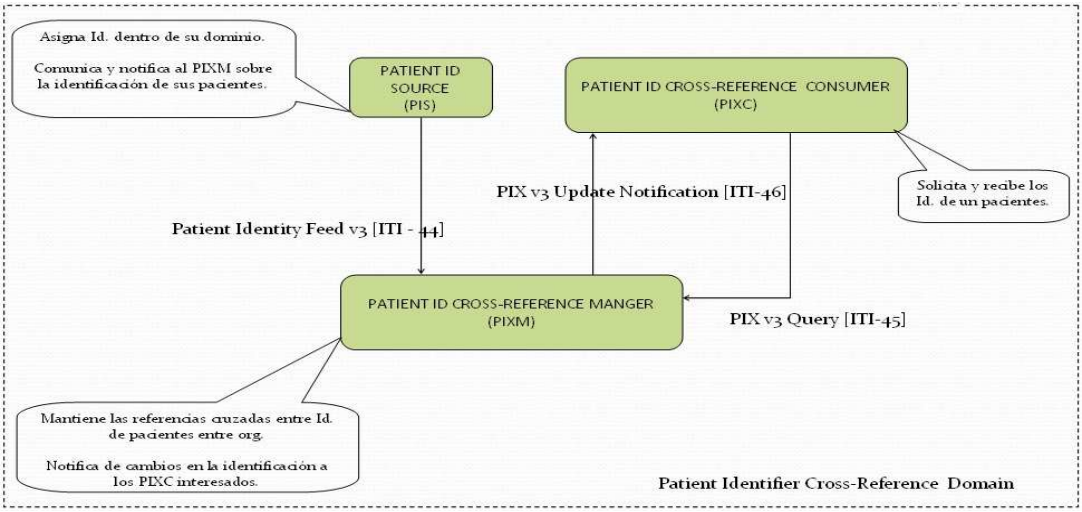


Figura 16: Actores y transacciones perfil PIX (adaptación [39])

Una configuración particular del PIX, donde se considera a uno de los dominios de identificación como "Dominio Maestro" y para el cual se asocia su fuente de identificación (Master PIS) con el *PIX Manager*, se conoce como Índice Maestro de Pacientes o por sus siglas en inglés: MPI (Master Patient identifier) o EMPI (Enterprise MPI). Es decir, en el dominio de identificación de cruzamiento cross se maneja la identificación dada en el dominio maestro. [39]

2.5.2.2 Cross Enterprise Document Sharing - (XDS.b)

El perfil XDS permite que distintas organizaciones puedan compartir documentos dentro de un dominio de afinidad, facilitando la registración, distribución y acceso a los registros de salud del paciente a lo largo de todo el dominio. Dos conceptos importantes dentro del perfil son: Dominio de afinidad y Documento XDS.

Un **dominio de afinidad** implica que las organizaciones estén de acuerdo en trabajar juntas en un entorno de cooperación definido, utilizando un conjunto común de políticas de seguridad, confidencialidad y a su vez compartiendo una infraestructura común de comunicaciones. Algunos ejemplos de dominio de afinidad XDS son:

- Una comunidad de cuidados, soportada por Organizaciones de salud con el fin de brindar asistencia a todos los pacientes en una determinada región.
- Grupo de especialidades o cuidados orientados a enfermedades puntuales.
- Una federación regional compuesta por hospitales locales y sus proveedores de salud.
- Instalaciones gubernamentales

Algunas de las principales políticas de carácter técnico (ITI TF-1 Appendix K) a considerar para garantizar la interoperabilidad son:

- El formato de documentos que se aceptarán para la registración.
- Los diferentes conjunto de valores referentes al vocabulario y esquemas de codificación que se utilizarán para representar la información. Por ejemplo, aquellos que se enviarán en los metadatos del documento.
- El dominio de identificación del paciente utilizado para registrar los documentos.

Un **documento XDS** es una composición de información clínica que contiene las observaciones y servicios brindados para fines de intercambio, dando el soporte para las siguientes características: Persistencia, Custodia del territorio, autenticación y la integridad. Estas características se definen en la especificación de Arquitectura de documento Clínica (CDA de sus siglas en ingles) de HL7, que como se explicó antes, define la arquitectura de un documento clínico representado en XML, que permite tener distintos niveles de estructura para representar la información.

Un documento XDS puede ser legible por humanos (con la aplicación correspondiente). En cualquier caso, se debe cumplir con el estándar que define su estructura, el contenido y la codificación. Es decir, los tipos de documentos a compartir deben estar previamente acordados, y si bien el perfil permite compartir tanto archivos de texto, como pdf, imágenes etc., se recomienda el uso del estándar CDA de HL7.

Los **Metadatos** en XDS, representan un conjunto de atributos que identifican al documento y que permiten ser registrados en el *Registry*, para posteriormente ser consultados y encontrados.

Estos atributos pueden ser obligatorios, generados por el repositorio o recomendados.

Algunos de los metadatos almacenados pueden ser sobre:

- El paciente: Dominio de afinidad, id, nombre, fecha de nacimiento.
- El origen del documento: autor, institución, autenticador legal.
- Identificación del documento: id, repositorio URI, identificador único, fecha de creación, fecha de comienzo y finalización del acto clínico, título, tamaño, etc.
- Clasificación del documento: clase, tipo, formato, tipo mime, códigos médicos, nivel de confidencialidad.

En la Figura 17 se muestran los actores y transacciones definidos por IHE para este perfil. En el Apéndice A se puede encontrar una descripción más detallada.

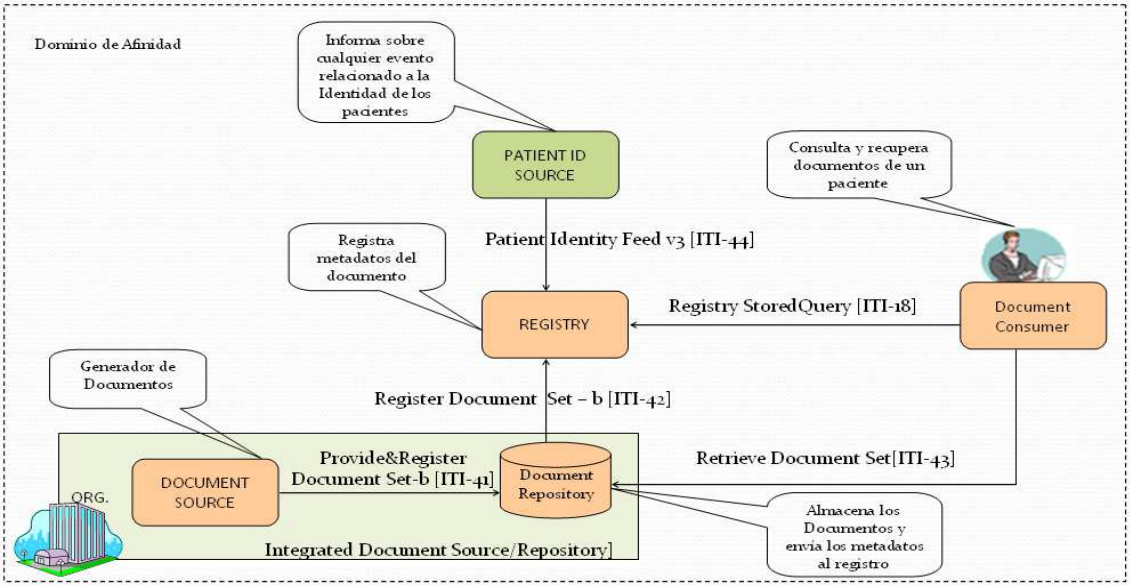


Figura 17: Actores y transacciones perfil XDS (adaptación [39])

El perfil permite tener varios repositorios que están compartiendo información a través de un mismo registro, donde está la información de identificación del documento (metadatos) que permitirá la realización de distintas consultas, para posteriormente recuperar la información médica. Esto hace a la solución escalable, ya que distintas organizaciones pueden compartir fácil y rápidamente documentos. Permite un acceso de forma simple ya que la búsqueda se realiza en el registro centralizado, además de poder mantener los datos de identificación del paciente asociados al documento separados de los clínicos. Es decir la información que se encuentra en el repositorio no identifica directamente al paciente.

Los estándares referenciados son ebRS/ebRIM de ebXML [44] (Electronic Business using extensible Markup Language) en el cual está basado el perfil para compartir documentos, que consiste en un modelo Registro / Repositorio en el cual los documentos se almacenan en el repositorio y en el registro la información (atributos del documento) que facilita su

búsqueda y recuperación. Utiliza como mecanismo de comunicación servicios web con soporte MTOM/XOP para optimizar el envío y recuperación de archivos. Todas las transacciones siguen los estándares de web services.

2.5.2.3 Cross Community Access (XCA)

El **perfil XCA** define un mecanismo para intercambiar información entre comunidades, aceptando las transacciones de consultas y recuperación de documentos. Permite independizar cómo cada comunidad resuelve internamente el manejo de la información, ya que podrían no poseer los mismos tipos de sistemas o de representación, para gestionar la información.

En XCA, una **comunidad** se define como un conjunto de organizaciones que se han puesto de acuerdo para trabajar en conjunto, utilizando las mismas políticas y estableciendo un mecanismo común para compartir información de salud.

El ***homeCommunityId*** es un identificador único y global que identifica a una comunidad. Este identificador es utilizado para obtener los "Web Services endpoint" de los servicios que proveen acceso a los datos de dicha comunidad.

En la Figura 18 se puede ver los actores y transacciones de este perfil. El actor que una comunidad debe implementar es el *Initiating Gateway* para atender solicitudes desde dentro de la comunidad y preguntar a las demás comunidades. La interacción con otras comunidades es a través del actor *Responding Gateway* que es quien recibirá las peticiones de documentos del exterior. Las transacciones entre estos actores corresponden a la consulta (Cross *Gateway* Query) y recuperación (Cross *Gateway* Retrieve) de documentos basadas en el perfil XDS [39]. Si bien el perfil no limita cómo resolver la gestión de documentos dentro de la comunidad, explica cómo se resolvería si la comunidad implementa XDS. En la Figura 16 se muestra cómo es la solicitud desde una comunidad XDS, con otras dos comunidades XDS y No XDS. En el apéndice 2 se puede encontrar una descripción más detallada de los actores y transacciones del perfil XCA.

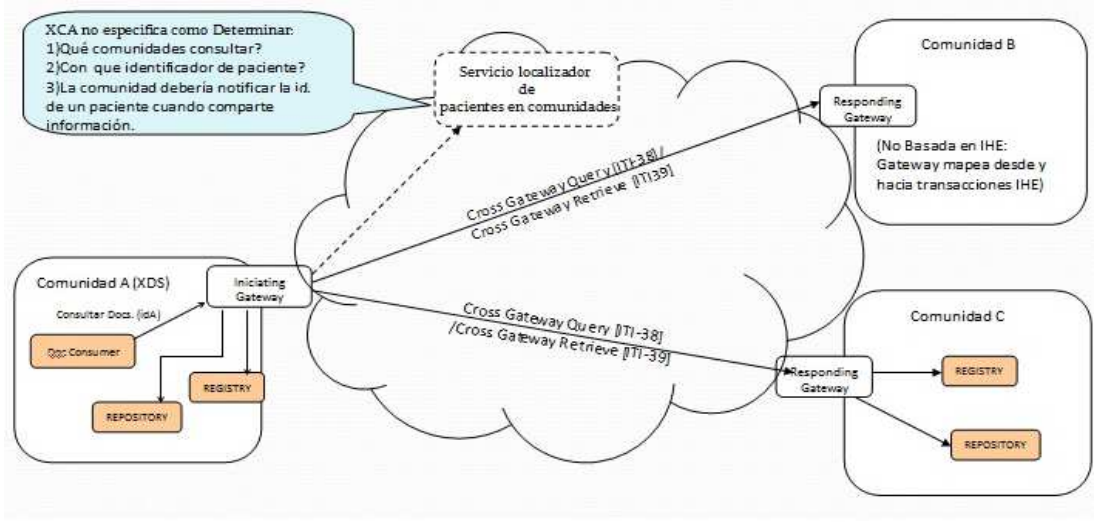


Figura 18: Actores y transacciones perfil XCA (adaptación [39]).

Ante una consulta local dentro de una comunidad surgen los problemas de qué comunidades y con qué identificador de paciente consultar. Esto no es solucionado por el perfil, pero plantea algunas alternativas:

1. Contar con un PIX a nivel de comunidades que al igual que uno de organizaciones pueda ser consultado con el identificador del paciente y éste devuelva las comunidades donde hay información.
2. Consultas por atributos demográficos del perfil PDQ (*Patient Demografic Query*) a las comunidades, lo que permite determinar el identificador correspondiente en ésta, para luego realizar la *Cross Gateway Query*.
3. Otra opción es el perfil XCPD (Cross-Community Patient Discovery) [45], que permite el descubrimiento de pacientes entre comunidades. Este perfil es sugerido sólo si los datos demográficos e identificatorios del paciente se encuentran federados en las comunidades, ya sea por alguna política o decisión regional por la cual no se quiere centralizar la información. (El perfil no se detalla en este documento)

2.5.2.4 Cross Enterprise User Assertion (XUA)

El perfil XUA provee un mecanismo para llevar a cabo la autenticación de transacciones entre Web Services en ambientes federados en un contexto de salud. De esta forma se permite al receptor tomar decisiones de acceso y realizar una correcta auditoría de las transacciones.

En un ambiente integrado por varias organizaciones es probable que se den transacciones entre dos organizaciones que mantienen sus propios directorios de usuario independientes. Mediante la Federación de la Identidad, se puede intercambiar la identidad de un usuario independientemente del tipo de directorio de usuario utilizado. El proveedor de Identidad es quien se encarga de realizar el mapeo de los atributos del directorio en el contexto del consumidor, a los atributos en el contexto del proveedor.

En la Figura 19 se puede ver los actores y la transacción del perfil reflejados en una transacción de consulta Registry Stored Query del perfil XDS.

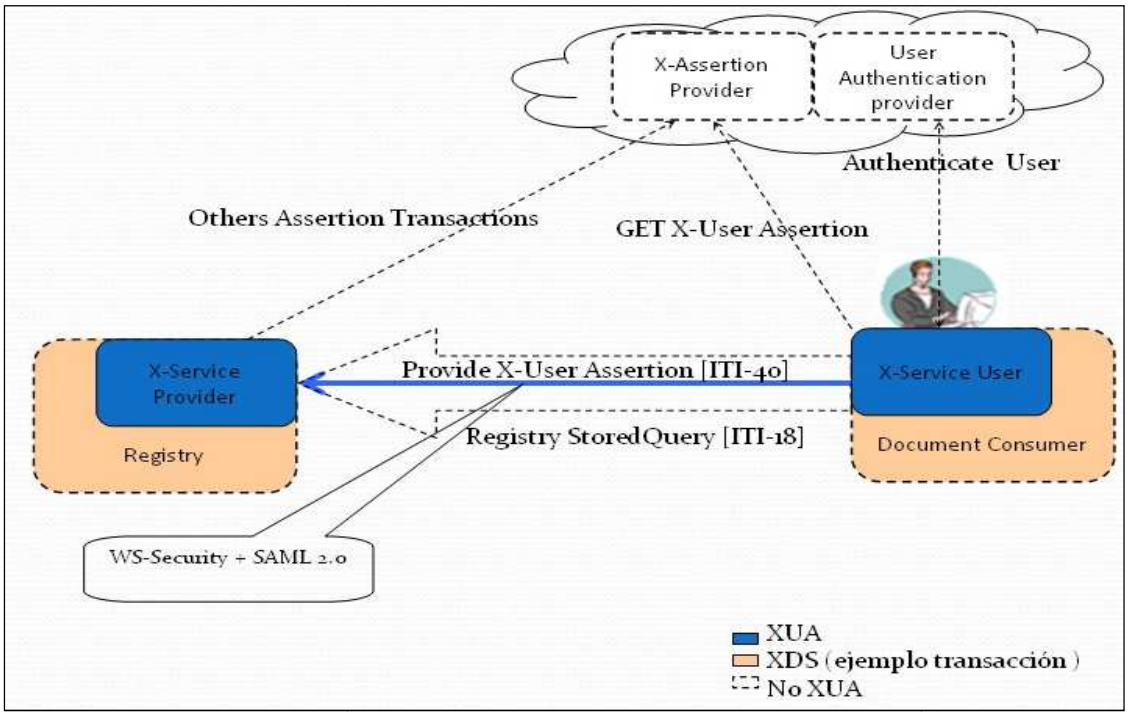


Figura 19: Actores y Transacciones del perfil XUA (adaptación [39]).

El perfil XUA específica utilizar aserciones de identidad SAML 2.0 con WS-Security para dar soporte a la identidad federada. En la Figura 19 se muestra los actores definidos por el perfil: usuario del servicio (*X-Service User*) y el proveedor (X-Service Provider), donde el primero provee su identidad a través de una aserción (Provide X-User Assertion [ITI-40]) SAML utilizando WS-Security como se muestra en la Figura 20.

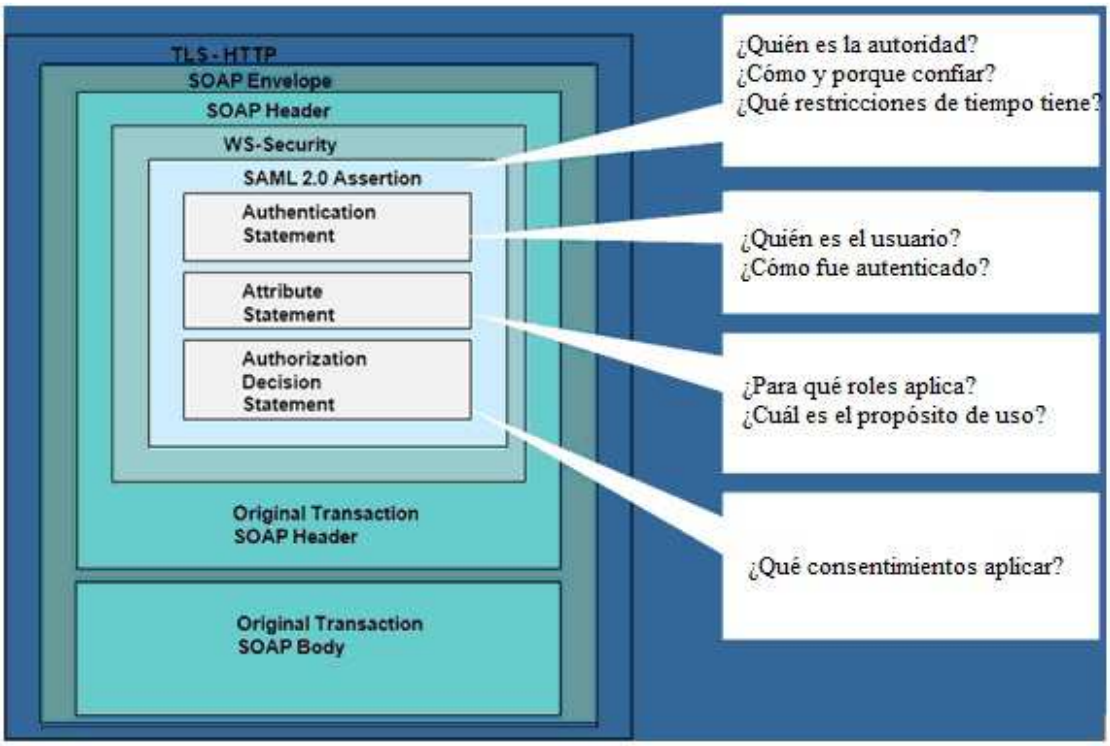


Figura 20: XUA Codificada en Web Service utilizando SAML [46]

El método de autenticación y el método para obtener la aserción de identidad están fuera del alcance de este perfil, pudiéndose utilizar por ejemplo WS-Trust.

2.5.2.5 Consistent Time (CT)

Este perfil especifica que se deben mantener sincronizados los relojes de todos los sistemas y computadoras conectadas a una red. Para ello sugiere el uso de los protocolos NTP (Network Time Protocol) y SNTP (Simple Network Time Protocol). Tiene una **única transacción** (*Maintain Time [ITI-1]*) en que un cliente consulta la fecha hora a un servidor de tiempo.

2.5.2.6 Audit Trail and Node Authentication (ATNA)

Un Dominio de Seguridad se puede ver como un conjunto de nodos seguros (más abajo se especifican las condiciones de un nodo seguro) que interactúan a través de una red.

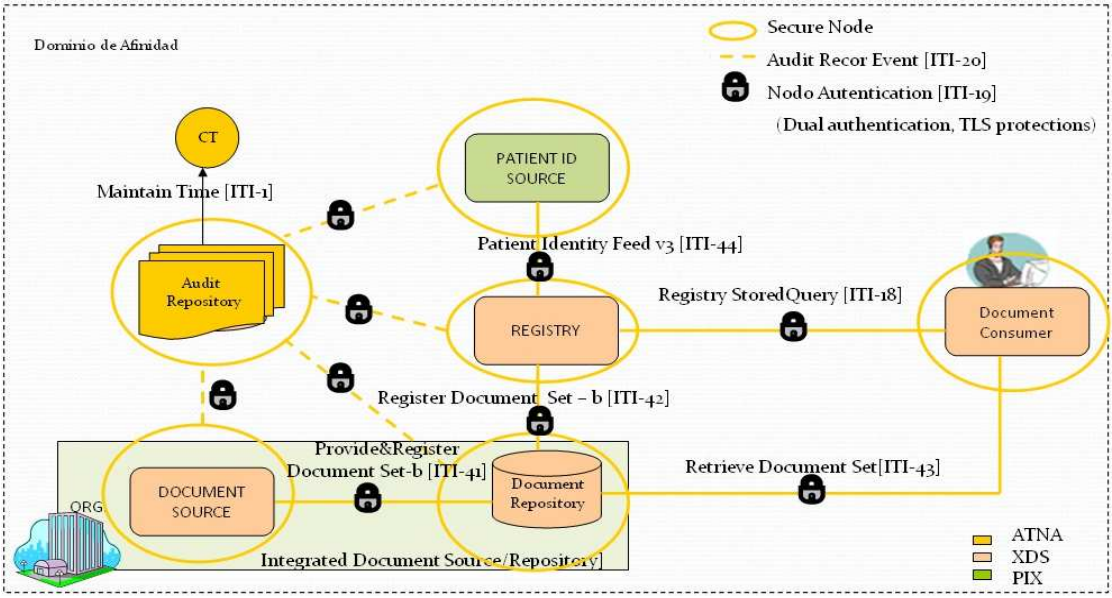


Figura 21: Actores y Transacciones del perfil ATNA con XDS (adaptación [39]).

En la Figura 21 se puede ver que los distintos actores (ejemplo XDS) son un nodo seguro (*secure node*), esto implica que se debe realizar controles de acceso sobre los usuarios que se conectan localmente, típicamente esto es Autenticación y Autorización. Además cada nodo es responsable de registros de auditoría para trazar eventos de seguridad como se especifica en la transacción correspondiente a *Record Audit Event* utilizada para la comunicación con el repositorio de auditoria *Audit Repository*. La comunicación con otro nodo seguro debe ser mediante doble autenticación como se especifica en la transacción *Authenticate Node* mediante el intercambio de certificados. Por otro lado, también deberá soportar la configuración para la autenticación y la seguridad a nivel físico de una red, en lugar de la doble autenticación.

El método específico para determinar si un nodo está autorizado a realizar transacciones no es definido por el perfil. Podría estar por ejemplo basado en determinados atributos del certificado o alguna lista de control de acceso.

2.5.2.7 Basic Patient Privacy Consents integration profile (BPPC)

El BPPC es un perfil básico que brinda la posibilidad definir mecanismos para registrar y gestionar el consentimiento de un paciente sobre las políticas de privacidad que serán aplicadas a sus documentos clínicos. Por ejemplo, permite que en un dominio de afinidad u organización se puedan marcar los documentos publicados, con los consentimientos que autorizan su acceso. Para su representación se crea un documento CDA en el cual se registran: las políticas de privacidad consentidas por el paciente, el texto descriptivo que expresa el consentimiento y sus transcripciones a reglas en formato XML, además de la posibilidad de capturar la firma escaneada de la persona. El documento se identifica de forma única con un OID *(Object Identifier Definition)* y se marca de forma especial en los metadatos para indicar que el documento es del tipo BPPC. Al ser un documento más del paciente, tiene como características, que se registra en el XDS, puede variar en el tiempo y trasladarse con los documentos a los que aplica.

Este perfil BPPC complementa el XDS, ya que posibilita el desarrollo y puesta en práctica de varias políticas de privacidad en un dominio de afinidad. Esta integración entre los

perfiles permite informar al paciente sobre las políticas de privacidad a consentir para que aplique en sus documentos, que puedan ser registradas de forma legible para una persona y procesables para una máquina, brindando el soporte necesario junto a los actores de XDS para realizar el control de acceso.

En los casos en que los pacientes no especifiquen políticas se debe contar con una política por defecto.

Como indica su nombre, este perfil está pensado para especificar políticas de acceso básicas. Las políticas permitidas son estáticas y por tanto no permiten excepciones dinámicas al momento de que un paciente las acepte.

En la Figura 22 se muestra un documento BPPC representado con CDA donde el paciente consintió la política 9.8.7.6.5.4.3.2.1 con su firma. En este caso la firma es escaneada pero también podría ser firmado digitalmente. [39] [47]

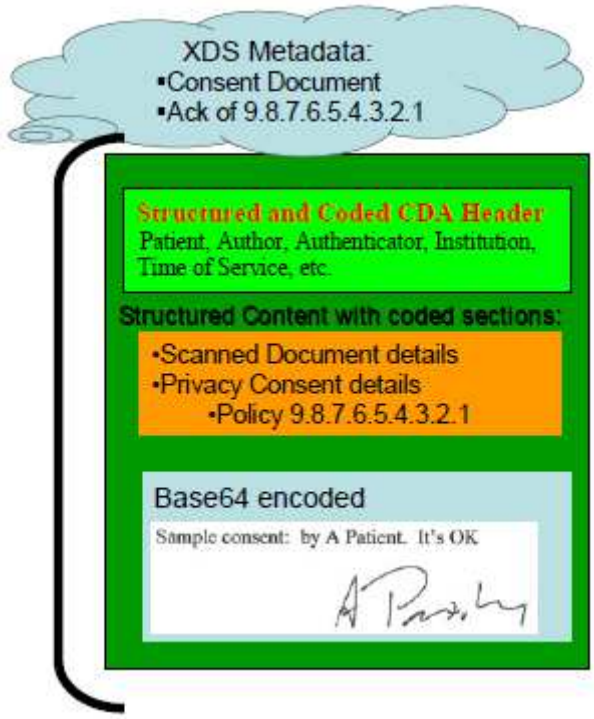


Figura 22: Representación gráfica de un consentimiento con CDA. [39]

2.5.3 Perfiles XSPA-OASIS

La OASIS (Organization for the Advancement of Structured Information Standards) [48] definió un conjunto de perfiles basado en estándares propios para intercambiar atributos de seguridad y privacidad de forma interoperable para el ámbito de la salud entre organizaciones y dentro de las mismas. Es así que se definieron perfiles que extienden los estándares SAML, WS-Trust y XACML para describir cómo extender un *token* SAML en base atributos de salud definidos por el perfil (XSPA-SAML), cómo pedir y obtener el *token* según (XSPA-WSTrust) y una forma de realizar la autorización en base a la aplicación de políticas y reglas definidas por la organización (XSPA-XACML). Los perfiles no abordan particularmente la integridad y la confidencialidad en la transmisión de la información, pero recomienda la utilización de protocolos de transmisión seguros y confiables. En las siguientes subsecciones se describen estos perfiles.

2.5.3.1 *Cross-Enterprise Security and Privacy Authorization (XSPA) Profile of Security Assertion Markup Language (SAML) for Healthcare Version 1.0*

Este perfil describe un framework para proveer interoperabilidad en el control de acceso. La interoperabilidad es alcanzada utilizando aserciones SAML para manejar el vocabulario para intercambiar la información del usuario.

El perfil se centra en la definición del *token* de seguridad a intercambiar mediante la utilización de aserciones de SAML 2.0, para luego realizar controles de acceso y auditoría. Especifica el conjunto de atributos que pueden contener las aserciones. En la Tabla 1 se lista el conjunto mínimo de atributos para cumplir con el perfil. [49]

Identificador	Valores válidos
urn:oasis:names:tc:xacml:1.0:subject:subject-id	Es el nombre del usuario como lo requiere la Ley de responsabilidad, portabilidad y seguridad médica. (HIPAA)
urn:oasis:names:tc:xpsa:1.0:subject:organization	Es el nombre de la organización a la que pertenece el usuario.
urn:oasis:names:tc:xspa:1.0:subject:organization-id	Es el identificador único de la organización que se facilita para consumir
urn:oasis:names:tc:xacml:2.0:subject:role ASTM E1986-98 (2005) Structured Role Value	Roles del personal médico con diferentes niveles para el control de acceso.
urn:oasis:names:tc:xspa:1.0:subject:functional-role	Opcional. Representa el rol funcional, puede incluir atributos personalizados relacionados con la funcionalidad de aplicación acordado por las partes en un intercambio
urn:oasis:names:tc:xacml:1.0:action:action-id HL7 Permission Catalog Resource Action Value	Opcional. Este perfil especifica el catálogo de permisos para acciones de RBAC-HL7. Append, Create, delete, read, Update, Execute. Es un subconjunto mínimo para interoperabilidad.
urn:oasis:names:tc:xspa:1.0:subject:hl7:permission	Opcional. El permiso es definido por la ANSI INCITS para compatibilidad con RBAC y representa la autorización requerida para acceder a un recurso protegido.
urn:oasis:names:tc:xspa:1,0:subject:purposeofuse	El prepósito de uso permite determinar de acuerdo al servicio de políticas si un usuario cumple o excede con la autorización según el control de acceso. TREATMENT, PAYMENT, OPERATIONS, EMERGENCY, SYSADMIN, MARKETING, RESEARCH, REQUEST, PUBLICHEALTH

urn:oasis:names:tc:xacml:1.0:resource:resource-id	Identificador único del Recurso definido y controlado por el servicio de la organización. En salud es el id. del paciente.
urn:oasis:names:tc:xspa:1.0:resource:hl7:type	Opcional. Soporte para la interoperabilidad de acciones y objetos referida a [HL7-PERM] y sus OID.

Tabla 1: Vocabulario mínimo exigido para cumplir con la conformidad del perfil.

No restringe la forma en que el *token* es obtenido, pero se podría utilizar WS-Trust para ello. Tampoco aclara cómo debe realizarse la aplicación de las políticas de seguridad y privacidad, para lo cual se podría utilizar XACML [50].

2.5.3.2 *Cross-Enterprise Security and Privacy Authorization (XSPA) profile of WS-Trust Healthcare Profile Version 1.0*

Este perfil describe un *Framework* para proveer interoperabilidad en el control de acceso. La interoperabilidad es alcanzada utilizando el protocolo WS-Trust para intercambiar información del usuario mediante un *token* de seguridad.

El perfil se centra en la forma de obtener el *token* de seguridad para que luego el proveedor del servicio pueda realizar controles de acceso y auditoría. Si bien especifica el vocabulario para el intercambio de los atributos, no obliga a utilizar un *token* de seguridad SAML. Tampoco aclara cómo debe realizarse la aplicación de las políticas de seguridad y privacidad, para lo cual se podría utilizar, por ejemplo XACML. [51]

2.5.3.3 *Cross-Enterprise Security and Privacy Authorization (XSPA) Profile of XACML v2.0*

Las organizaciones, incluyendo las que tienen que ver con el cuidado de la salud, necesitan un mecanismo para intercambiar políticas de seguridad y privacidad de forma interoperable. Este perfil describe cómo utilizar XACML para proveer dicha funcionalidad de una forma interoperable.

Describe varios mecanismos para administrar y aplicar políticas de autorización para controlar el acceso a información protegida dentro o fuera de los límites de una organización. Se administran políticas de seguridad, privacidad y directivas de consentimiento.

Este perfil puede ser usado en coordinación con otros estándares como SAML para el intercambio de los atributos a través de un *token* y WS-Trust para obtenerlo.

Este perfil especifica el uso de XACML 2.0 para promover la interoperabilidad dentro de una comunidad de salud, proporcionando la semántica y vocabularios comunes para realizar la solicitud de una política de forma interoperable, manejar el ciclo de vida de una política y la aplicación de la misma. [50]

2.5.4 Resumen

A continuación se presenta en la Tabla 2, el resumen de los perfiles para interoperabilidad en sistemas de salud mencionados en el documento.

Organización	Perfil/Estándar	Descripción
HL7	CDA R2	Define la arquitectura de un documento clínico que permite varios niveles de interoperabilidad. Cuenta con un encabezado que contiene metadatos que clasifican al documento y un cuerpo que puede contener desde una imagen, texto descriptivo y partes estructuradas, codificadas con vocabularios clínicos como SNOMED, LOINC, etc.
IHE	PIX HL7 v3	Gestión de la identidad de pacientes en diferentes dominios. Permite consultas y notificaciones cruzadas entre los dominios de identificación.
IHE	PDQ HL7 v3	Permite obtener los identificadores de un paciente consultando por atributos demográficos.
IHE	XDS.b	Compartir documentos clínicos entre organizaciones dentro de un dominio de afinidad. Los documentos son almacenados en repositorios y sus atributos de identificación enviados a un registro centralizado para facilitar la búsqueda de los consumidores permitiendo realizar consultas a éste y recuperar luego del repositorio.
IHE	XCA	Intercambio de documentos de salud entre distintas comunidades. Permite formar parte de una visión más amplia para compartir documentos a nivel de la región o país.
IHE	XUA	Comunicar aserciones sobre la identidad de un usuario, aplicación, sistema autenticado, en transacciones fuera de la organización, que permita conocer información sobre el consumidor de un servicio y así se puedan realizar auditoría y control de acceso por el servicio proveedor.
IHE	CT	Mantener sincronizados todos los sistemas y computadoras con un tiempo común. Necesario para auditoria y autorización.
IHE	ATNA	Define mecanismos para garantizar la confidencialidad, integridad de la información y determinar responsabilidades.
IHE	BPPC	Permite capturar los consentimientos que un paciente acepta sobre las políticas de privacidad que deben aplicar en sus documentos.
OASIS	XSPA - SAML	Describe un conjunto de atributos que pueden emplearse para definir un *token* de seguridad que permita comunicar la información necesaria para realizar el control de acceso en sistemas de información Sanitarios.
OASIS	XSPA-WS TRUST	Describe un mecanismo para autenticar entre organizaciones y/o sistemas de información sanitarios cumpliendo con los requisitos de información necesarios para que el proveedor del servicio pueda realizar el control de acceso.

OASIS	XSPA – XACML	Describe varios mecanismos para administrar y aplicar las políticas de autorización para realizar el control de acceso a la información sanitaria.

Tabla 2: Resumen de perfiles

2.6 Contexto Nacional

En esta sección se definen los conceptos del contexto nacional que se relacionan al proyecto.

2.6.1 Agencia de Gobierno Electrónico y Sociedad de la Información (AGESIC)

AGESIC [52] es un organismo que depende de la Presidencia de la República Oriental del Uruguay (unidad ejecutora 010 dentro del inciso 02). Funciona con autonomía técnica.

Tiene como objetivo procurar la mejora de los servicios al ciudadano, utilizando las posibilidades que brindan las Tecnologías de la Información y las Comunicaciones (TIC).

Entre sus actividades permanentes se encuentran:

- Definir y difundir la normativa informática, fiscalizando su cumplimiento.
- Analizar las tendencias tecnológicas.
- Desarrollar proyectos en Tecnologías de la Información y las Comunicaciones.
- Asesorar en materia informática a las instituciones públicas del Estado.
- Capacitar y difundir en materia de Gobierno Electrónico, apoyando a la transformación y transparencia del Estado.

2.6.2 Unidad Nacional de Asignación de Identificadores de Objetos (UNAOID)

2.6.2.1 ObjectIdentifierDefinition (OID)

Un OID o identificador de objeto, es una secuencia de números que se define de acuerdo a una organización jerárquica, establecida por la Organización Internacional de Normalización (ISO). Está basado en la Norma ISO/IEC 8824-1, Information Technology – Abstract Syntax Notation One (ASN.1) – Specification of basic notation.Los OID son utilizados en una gran cantidad de protocolos por ejemplo LDAP, SNMP, CIP etc.

Un ejemplo de OID para Uruguay sería: 2.16.858.0.0.1, que representa a los entes autónomos del Estado Uruguayo.

2.6.2.2 UNAOID

UNAOID es la unidad nacional de asignación de identificadores de objetos, se trata de funciones asignadas a AGESIC otorgadas por la ISO e ITU, a partir de setiembre de 2009 [53].

La UNAOID define como Objeto todo tangible o intangible, técnicamente viable de ser identificado como unidad, capaz de conformar grupos y por ende de contabilizarse.

La UNAOID, a través de estos identificadores, busca proporcionar una base estandarizada para la interoperabilidad de los sistemas de información en la Administración Pública, así como en el resto de sectores de la sociedad, por otro lado propende a dotar de una herramienta única para la identificación e inventario, así como un instrumento extraordinario para la efectiva trazabilidad de objetos en su concepción más amplia [53].

2.6.3 Plataforma de Gobierno Electrónico (PGE)

La Plataforma de Gobierno Electrónico (PGE) es construida y mantenida por AGESIC. Es un facilitador para el desarrollo y trámites en línea, un proveedor de servicios transversales y herramientas comunes a los organismos del Estado. También es un medio para instrumentar la interoperabilidad y el intercambio de información entre Organismos, es el contexto tecnológico y legal que permite asegurar que la información intercambiada, cumpla con los requisitos legales y tecnológicos predefinidos [54].

La plataforma tiene como objetivo general facilitar y promover la implementación de servicios de Gobierno Electrónico en Uruguay. Para esto, la PGE brinda mecanismos que apuntan a simplificar la integración entre los organismos del Estado y a posibilitar un mejor aprovechamiento de sus activos.

A nivel tecnológico, se implementó una Arquitectura Orientada a Servicios (ServiceOrientedArchitecture, SOA) a nivel del Estado, la cual se apoya fuertemente en la tecnología de Web Services [54].

En la Figura 23 se muestran los principales componentes y actores participantes de la PGE:

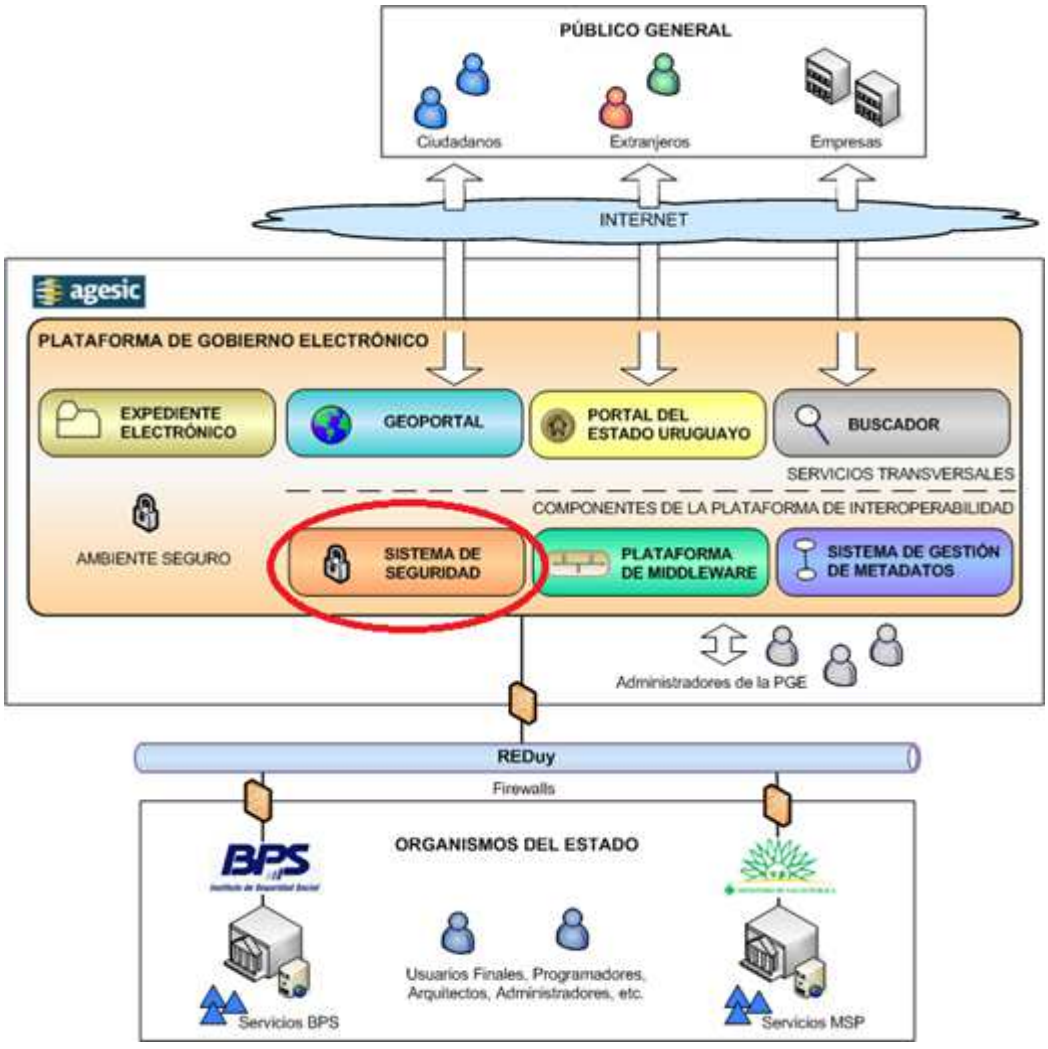


Figura 23: Principales Componentes y Actores de la PGE [54]

Los principales componentes son:

Los Servicios Transversales que son:

- Expediente Electrónico
- Geoportal
- Portal del Estado Uruguayo
- Buscador

La plataforma de interoperabilidad que se compone de:

- Sistema de Gestión de Metadatos
- Plataforma de Middleware
- Sistema de seguridad

Este marco conceptual, se centra en la plataforma de interoperabilidad, más concretamente en su sistema de seguridad.

2.6.3.1 Plataforma de Interoperabilidad - Sistema de Seguridad

La plataforma de interoperabilidad (PDI) brinda los beneficios de su estandarización en seguridad, protocolos, semántica y operativa. Posee un diseño SOA, es decir que brinda la posibilidad de reutilización de servicios, tiene una arquitectura flexible y facilita la integración con sistemas legados.

El sistema de seguridad provee servicios de seguridad al resto de los componentes de la PGE. Brinda mecanismos que permiten realizar auditorías de seguridad en la PGE, aplicar políticas de acceso asociadas a los servicios publicados en la plataforma, y facilitar el acceso seguro de los organismos a la PGE. Dicho sistema de seguridad está compuesto por tres grandes componentes que se muestran en Figura 24:

Figura 24: Sistema de seguridad de la PGE [54]

El Sistema de Auditoría provee las herramientas necesarias para realizar auditorías de seguridad sobre la PGE. Este sistema recolecta información y realiza análisis y reportes de auditoría. El Sistema de Auditoría está implementado por el producto Tivoli Compliance Insight Manager (TCIM).

Los Servicios Periféricos de Seguridad tienen la finalidad de brindar los mecanismos necesarios para facilitar a los organismos el acceso seguro a la PGE. Los dos servicios principales de este componente son: La Autoridad Certificadora, provista por el producto Windows 2003 Server y el servicio de Directorio, implementado principalmente por los productos IBM Directory Server, Tivoli Identity Manager (TIM) y Tivoli Directory Integrator (TDI).

El Sistema de Control de Acceso de la PGE brinda los mecanismos para aplicar políticas de control de acceso sobre los servicios publicados y las aplicaciones disponibles en la PGE. El

control de acceso en la PGE se realiza siguiendo un esquema RBAC, utilizando el rol del usuario que quiere acceder al servicio o aplicación, y las políticas de acceso definidas en la PGE.

El Sistema de Control de Acceso consta de tres componentes como se puede ver en la siguiente Figura 25:

Figura 25: Sistema de control de acceso para servicios de la PGE [54]

El Servicio de *Tokens* de Seguridad (Security *Token* Service, STS) tiene la responsabilidad de emitir los *tokens* de seguridad necesarios para que las aplicaciones cliente puedan invocar a los servicios publicados en la PGE. Este componente soporta el estándar WS-Trust v1.3 y es implementado por el producto Tivoli Federated Identity Manager (TFIM).

El Administrador de Políticas de Seguridad actúa como Punto de Decisión de Políticas (Policy Decision Point, PDP) siendo responsable por tomar la decisión de autorizar, o no, los pedidos de invocación a servicios de la PGE. Este componente es implementado por el producto Tivoli Security Policy Manager (TSPM).

El Firewall XML actúa como Punto de Aplicación de Políticas (Policy Enforcement Point, PEP) de acuerdo a lo que decida el Administrador de Políticas de Seguridad. Este componente está implementado por el producto IBM Websphere Datapower Xi50. A continuación se explica un ejemplo del control de acceso a servicios que se encuentran alojados en la PGE. El control de acceso utiliza un modelo RBAC, a nivel de métodos. Esto implica que cuando un organismo publica un servicio, debe especificar quién tiene acceso a cada método que el servicio contenga. [54]

2.6.3.2 Consumo de Servicios

Cuando una aplicación cliente de un organismo desea invocar a un método de un servicio que se encuentra en la PGE, el flujo que debe realizar es el siguiente:

El cliente:

1. Generar un token de seguridad SAML v1.1 o v2.0, firmado por el organismo cliente, que incluya el rol de usuario con el que se desea acceder al servicio
2. Enviar a la PGE la solicitud de un token de seguridad al STS de la plataforma utilizando WS-Trust

La PGE:

1. Recibe el pedido para el STS
2. Verifica que la firma digital del *token* incluído en el pedido sea de un cliente confiable
3. Verifica que el rol del usuario, incluido en el *token* de seguridad, exista en el directorio
4. Si se verifica todo lo anterior, el STS emite el *token* firmado por la PGE y lo envía al cliente

La Figura 26 muestra un resumen del flujo explicado:

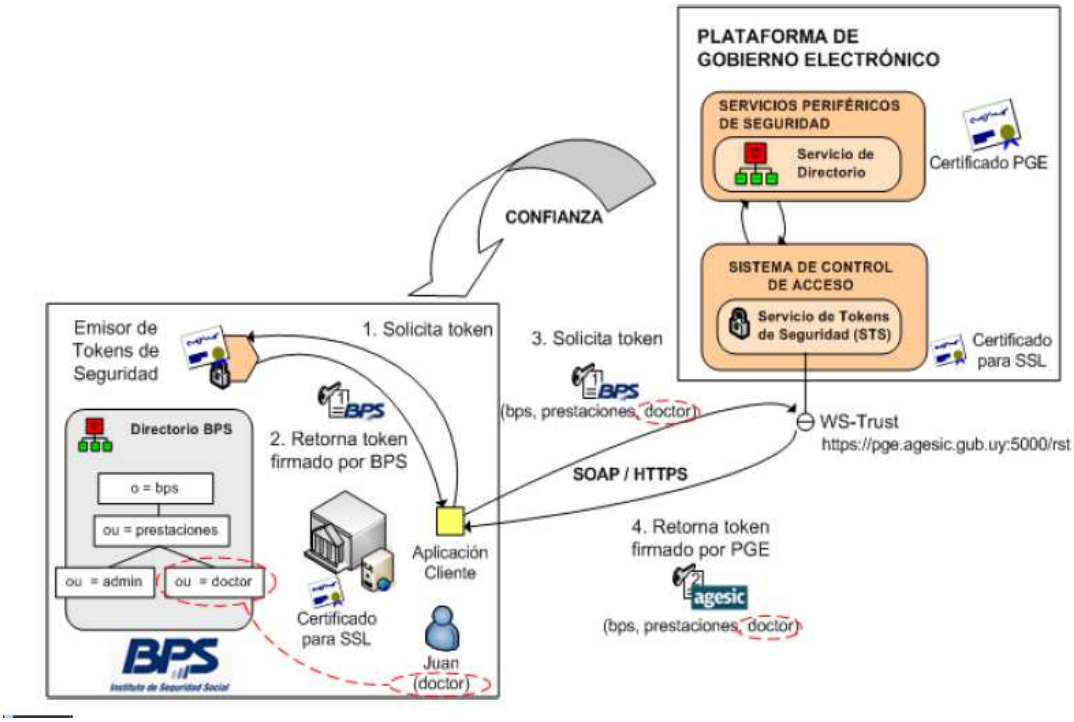


Figura 26: Flujo para autenticación de un servicio a través de la PGE [54]

En el ejemplo de la Figura 26, la aplicación cliente pertenece al BPS (Banco de Previsión Social). La comunicación entre las aplicaciones cliente y el STS de la PGE son HTTPS.

2.6.4 RedUy

El gobierno uruguayo ha ejecutado un proyecto denominado RedUy, que tuvo por objetivo la creación de un medio físico para lograr la infraestructura de conectividad necesaria para una plataforma de Gobierno Electrónico. Se trata de una red de alta velocidad que permite que cada organismo se conecte, de manera segura, con adecuados niveles de servicio, de seguridad informática y alta disponibilidad, con otros organismos que forman parte de dicha red. La conexión es realizada a través de un único enlace, hacia todos los organismos que forman parte de la red. Esto permite que las instituciones interactúen entre sí para intercambiar información con el objetivo de crear una Administración Pública eficiente y centrada en el usuario. [55]

2.7 Trabajos relacionados

Como un primer paso al análisis del problema objetivo se consideraron algunos artículos e implementación de casos reales en proyectos donde se han utilizado algunos de los estándares presentados en las secciones anteriores, así como proyectos académicos relacionados al tema, para poder medir el grado de madurez y aplicabilidad de los mismos.

Historia clínica federada en el Servicio de Salud de Castilla y León (SACYL) [38] [56]: este proyecto surge como una aproximación a la historia clínica federada para servicio público de sanidad que gestiona las prestaciones públicas en la comunidad española de Castilla y León, perteneciente al Sistema Nacional de Salud. Está basado en XDS.b como mecanismo principal para compartir documentos, siendo SACYL el dominio de afinidad donde se tiene:

un único identificador para el paciente, un índice de repositorio (registro) y a las diferentes estaciones clínicas de los hospitales como repositorios. Además como estrategia de implementación se digitalizó la historia clínica en papel usando el perfil XDS-SD. Se proporcionó bibliotecas para aplicaciones proveedoras y consumidoras para facilitar la integración al modelo. En sus conclusiones se encuentra, que han encontrado útiles herramientas que dan soporte a XDS y el uso de CDA por parte de las estaciones clínicas. Todo esto ha repercutido en una disminución notable del tiempo de implantación.

La Informática de la Salud en Europa [57]**:** este artículo habla sobre la interoperabilidad de sistemas de salud en Europa y cómo los perfiles propuestos por IHE han sido adoptados en diversos proyectos de Europa, Estados Unidos y Canadá. El informe se centra en la historia clínica electrónica y en lo que IHE propone como aproximación a la misma basada en XDS y CDA.

Protección de los Datos Personales de la historia clínica en Argentina y Uruguay e IHE XDS (J. Health Inform. 2011 Agosto 3) [58]**:**

Este artículo plantea cómo la ley de Protección de Datos Personales (PDP) en ambos países hacen referencia implícita a la familia de normas de seguridad de la información ISO/IEC 27000 que recomiendan buenas prácticas y aseguran ciertas características relacionadas a un Sistemas de Gestión de Seguridad de la Información (SGSI) desde un punto de vista organizativo y de gestión de estos sistemas. Luego presenta algunos perfiles de IHE y cómo estos cumplen o dan soporte desde el punto de vista tecnológico para la implementación de un SGSI que cumple con los principios de las ISO/IEC 27002 (buenas prácticas en SGSI), ISO/IEC 27779 (Gestión de la seguridad en Salud, utilizando ISO/IEC 27002), y PDP de Argentina y Uruguay. Se realiza un planteo donde se determina qué conjunto de datos serían necesarios para aplicar PDP a la historia clínica, qué perfiles de IHE cumplirían con estas necesidades y qué flexibilidad se brinda para registrar los consentimientos del paciente lo cual es una cuestión fundamental para la PDP, y cómo éste da soporte para los casos de uso más comunes. Concluye cómo los perfiles XDS y BPPC son una alternativa a considerar para aplicar de PDP.

El artículo nos proporciona la posibilidad de considerar los perfiles de IHE para la solución sin entrar en detalle en la ley de protección de datos personales Uruguaya.

Control de Acceso a la Historia Clínica Electrónica (HCE) [59]**:**

Es un proyecto de grado de la Facultad de Ingeniería realizado en el 2011. Aquí se plantea cómo realizar un control de acceso a HCE utilizando perfiles de IHE y OASIS. El proyecto se centra en la representación, creación y almacenamiento de consentimientos de pacientes, y el estudio de políticas de privacidad, considerando para ello principalmente los estándares CDA y XACML respectivamente.

Este proyecto al igual que el anterior, nos aporta mayor detalle sobre el cumplimiento de la legislación uruguaya y la representación de consentimientos a través del uso del perfil BPPC (usando el estándar CDA), y políticas XACML.

3 Análisis

En este capítulo se describe el análisis realizado para definir la solución propuesta. En primer lugar se especifica en bajo nivel el caso de estudio utilizado para guiar el análisis, se definen los requerimientos y sus principales problemáticas asociadas. En segundo lugar se presenta el análisis de las soluciones por requerimientos y cómo los estándares estudiados pueden resolver las problemáticas identificadas, y por último se presentan dos alternativas de solución y las conclusiones del análisis.

3.1 Análisis de Requerimientos

Como se mencionó en la introducción de este documento, se definió un Caso de Estudio que fue utilizado como guía para lograr concretar los objetivos definidos en el marco del proyecto realizado. A continuación se describe el caso de forma detallada.

Un Paciente visita a su médico de cabecera luego de haberse realizado diferentes estudios (informes, placas, etc.) prescriptos por éste. Los estudios fueron realizados en distintas organizaciones de salud, en cada una de estas instancias de atención el paciente fue identificado de forma diferente de acuerdo a los mecanismos de identificación utilizados por la organización donde se realiza el estudio. El paciente ha consentido políticas de privacidad que aplican a su información clínica, cuando es ingresado por primera vez al sistema.

El caso de estudio plantea tres situaciones relacionadas con la atención médica:

- Un paciente se atiende en distintas organizaciones de salud para realizarse diferentes estudios.
- El médico tratante consulta los documentos clínicos (informes, placas, etc.) que se realizó el paciente.
- El paciente consiente diferentes políticas de privacidad para el acceso a su información.

El caso de estudio consta de dos escenarios que se detallan a continuación.

3.1.1 Escenario 1

Este escenario representa el registro de la atención médica del paciente en el sistema de salud de la institución en Montevideo:

1. El paciente concurre al médico en la mutualista que es socio en Montevideo porque está continuamente con la nariz congestionada y tiene eventuales dolores de cabeza
2. La doctora en Montevideo, lo atiende y le indica tomar antialérgicos y analgésicos, pero si el malestar persiste, deberá realizarse una placa de los senos faciales porque podría ser que sea propenso a tener sinusitis
3. A través del sistema informático, la doctora de la mutualista registra los hallazgos y el tratamiento indicado al paciente
4. El paciente no requiere pasar por farmacia porque ya tiene los medicamentos indicados

3.1.2 Escenario 2

Este escenario representa la consulta de información médica del paciente desde el sistema de salud de Paysandú hacia el sistema de salud en Montevideo:

1. Unos días después el mismo paciente viaja a Paysandú y tiene que consultar un médico por un fuerte dolor en la cabeza que tiene hace dos días y no se cura con analgésicos ni antialérgicos, concurre a una mutualista en Paysandú.
2. El paciente indica que se estuvo tratando en Montevideo por alergia nasal, y que le habían mencionado algo de los senos frontales que no recuerda bien.
3. La doctora de Paysandú consulta la historia clínica del paciente a través del sistema médico de la mutualista.
4. La doctora, realiza su diagnóstico con mayor precisión luego de haber consultado la documentación disponible en la mutualista de Montevideo.

3.1.3 Requerimientos Identificados

En esta sección se presentan los requerimientos funcionales y no funcionales identificados en el marco del proyecto.

3.1.3.1 Requerimientos Funcionales

Los requerimientos funcionales definidos son:

RQF1: La solución debe permitir identificar el mismo paciente en las distintas organizaciones donde es atendido.

RQF2: La solución debe permitir que la información sea compartida (accedida y visualizada) entre los sistemas de las diferentes organizaciones, teniendo en cuenta que los pacientes están registrados con distintos identificadores.

RQF3: La solución debe contemplar el cumplimiento de las leyes de Privacidad de la Protección de Datos Personales de la información.

3.1.3.2 Requerimientos no funcionales

Los requerimientos no funcionales definidos son:

RQNF1: Compartir la información entre las organizaciones a través de la PGE.

RQNF2: Reutilizar los recursos existentes en las organizaciones de salud.

RQNF3: La solución debe cumplir con los estándares y buenas prácticas de la industria de la salud.

3.1.4 Principales Problemáticas y Desafíos a resolver

En esta sección se describen las problemáticas identificadas durante la etapa de análisis de requerimientos. En la Tabla 3 se listan las problemáticas y su asociación a los requerimientos.

Requerimientos	Problemáticas
RQF1: La solución debe permitir identificar el mismo paciente en las distintas organizaciones donde es atendido.	**PR1:** Poder identificar el mismo paciente desde distintas organizaciones. **PR2:** Notificación de cambios en la identificación del paciente.
RQF2: La solución debe permitir que la información sea compartida (accedida y visualizada) entre los	**PR3:** La información clínica del paciente se encuentra distribuida.

sistemas de las diferentes organizaciones, teniendo en cuenta que los pacientes están registrados con distintos identificadores.	**PR4:** Formatos heterogéneos de la información intercambiada.
RQF3: La solución debe contemplar el cumplimiento de las Leyes de Privacidad de la información.	**PR5:** Garantizar seguridad y privacidad. **PR6:** Registrar los consentimientos del paciente. **PR7:** Identificación de la información amparada bajo los documentos de consentimiento (qué políticas de privacidad aplican a que documentos) **PR8:** Disociación de la información clínica, de los datos que identifican al paciente.
RQNF1: La solución debe compartir la información entre las organizaciones, a través de la PGE.	**PR9:** Cómo realizar la integración con la PGE. **PR10:** Encontrar la forma de reutilizar recursos existentes en la PGE.
RQNF2: Reutilizar los recursos existentes en las organizaciones de salud.	**PR11:** Diversidad de estándares y protocolos en cada organización. **PR12:** Reutilizar recursos y existencia de escasos recursos en las organizaciones.
RQNF3: La solución debe cumplir con los estándares y buenas prácticas de la industria de la salud.	**PR13:** Variedad de estándares, asimilación. Componer todo en una solución integral.

Tabla 3: Requerimientos y problemáticas asociadas

A continuación se presenta la justificación de cada problemática y su asociación al requerimiento correspondiente.

PR1 - Poder identificar el mismo paciente en distintas organizaciones

Cada organización identifica a sus pacientes de forma particular, y un paciente puede atenderse en más de una organización (por ejemplo: realizarse estudios de laboratorio en una organización distinta a la que pertenece). Resulta entonces necesario, contemplar un mecanismo que solucione este cruzamiento de información de identificación del paciente, que eventualmente pueda ser diferente debido a errores al momento de registrarla.

Por ejemplo, hay organizaciones que utilizan el documento de identidad, otras que utilizan un número de socio, también existe el caso en que el paciente es originario de otro país y el formato de su número de documento de identidad no coincide con el uruguayo, etc. Es por esto que se plantea como problemática en este proyecto, el hecho de que un mismo paciente puede llegar a estar registrado en un sistema con un tipo de identificador y en otro sistema con otro tipo distinto al primero.

PR2 - Notificación de cambios en la identificación del paciente

Dado que la información de identificación puede ser modificada, por ejemplo al corregir un error, es necesario que los sistemas que la referencian sean notificados de estos cambios.

PR3 - La información clínica del paciente se encuentra distribuida

Al momento de consultar información clínica de un paciente, es necesario contar con un mecanismo que resuelva la localización, dentro de las diferentes organizaciones, donde el paciente fue asistido.

PR4 - Formato de la información intercambiada

La información compartida debe estar representada en un formato que pueda ser interpretado por todos los actores involucrados (usuarios, sistemas, etc.), dado que de otra manera se dificulta su interpretación y entendimiento resultando en un bajo nivel de interoperabilidad entre los sistemas.

PR5 - Garantizar la seguridad y privacidad

Debido a que la información que es compartida se encuentra catalogada como información sensible, es necesario poder realizar controles de seguridad y acceso. Dichos controles deberán garantizar el cumplimiento de las leyes definidas por el Estado Uruguayo, y los consentimientos autorizados por los pacientes. Esta problemática consiste en dar solución a las siguientes sub-problemáticas:

- Definición de controles de autenticación en el sistema
- Definición de controles de autorización para acceder a las funcionalidades
- Definición de control de acceso a la información clínica basado en los consentimientos del paciente
- Definición de mecanismos de auditoria (trazabilidad y auditabilidad de las transacciones)
- Integridad de los documentos (implica poseer un sistema de archivado con enmienda por edición de un nuevo documento en lugar de una modificación aplicada)

PR6 – Registrar los consentimientos del paciente

Salvo en casos muy excepcionales, los pacientes siempre son libres de permitir el acceso a su información clínica personal o no. Resulta necesario encontrar una solución para crear, modificar, mantener y consultar, los consentimientos del paciente.

PR7 - Identificación de la información amparada bajo los documentos de consentimiento

Se debe encontrar solución a una representación de la asociación entre los consentimientos del paciente y la información clínica que le pertenece. La solución debe brindar un mecanismo que asegure, que para cualquier información clínica que pueda ser consultada, sus consentimientos asociados pueden ser debidamente identificados para luego aplicar las políticas de acceso que corresponda.

PR8 - Disociación de la información clínica, de los datos que identifican al paciente

La información generada en una atención médica, involucra tanto datos personales, como clínicos del paciente. Debido a las restricciones de confidencial, resulta necesario que dichos datos se registren separados, de forma tal que no se pueda asociar directamente un diagnóstico con el paciente al cual pertenece.

PR9 - Compartir la información entre las organizaciones, a través de la PGE.

Dado que parte importante del objetivo principal de este proyecto es compartir la información a través de la PGE, resulta necesario analizar dicha integración. Es decir, lograr una integración manteniendo la SOA y estándares de interoperabilidad utilizados por la PGE, a la vez que compatibilizar con los estándares de Salud utilizados.

PR10 - Identificar el reuso de recursos existentes en la PGE

Es necesario entender en profundidad la solución definida para la plataforma de interoperabilidad de la PGE. Debe ser posible adquirir el entendimiento suficiente como para poder evaluar la reutilización o no, de componentes de la misma.

PR11 - Diversidad de estándares y protocolos en cada organización

Los sistemas utilizados por las distintas organizaciones de Salud para registrar la información referente a la atención médica, utilizan una diversidad de estándares y protocolos que difieren entre sí. Es necesario contemplar este hecho a la hora de definir la solución propuesta, de forma tal que la misma sea flexible y contemple la reutilización de dichos sistemas.

PR12 - Reutilizar recursos y existencia de escasos recursos en las organizaciones

La solución deberá contemplar a aquellas organizaciones que tienen bajos recursos. La integración a la solución por parte de dichas organizaciones deberá ser lo más fácil posible. Además es deseable tener en cuenta que también existen organizaciones que sí poseen recursos (por ejemplo infraestructura) y que sería deseable reutilizar.

PR13 - Variedad de estándares, asimilación. Componer todo en una solución integral

Hoy en día existe una gran variedad de estándares definidos, tanto para salud como para interoperabilidad y comunicación. Es una problemática en sí misma la asimilación de dichos estándares de forma tal de lograr la mejor solución. Los estándares deberán ser compatibles con aquellos utilizados por parte de la PGE, y con los cuales hay que interactuar.

3.2 Análisis de Estándares y Soluciones

En esta sección se presenta el análisis de los requerimientos, y cómo los diferentes perfiles y estándares estudiados, resuelven las distintas problemáticas que éstos presentan. Al final se describen dos alternativas de solución basada en los perfiles y estándares analizados. Finalmente en la Tabla 4 se presenta un resumen de requerimientos, problemáticas y estándares utilizados para su resolución.

RQF1 – Identificación cruzada de pacientes.

Para poder intercambiar información sobre pacientes entre distintas organizaciones es necesario conocer el identificador en cada una de ellas incluso mantener consistente la información ante eventuales cambios en sus datos demográficos.

PR1: Poder identificar al mismo paciente en distintas organizaciones.

La identificación de los pacientes fue analizada con el perfil PIX de IHE, para ello deberá contarse con el componente *PIX Manager* alimentado por las organizaciones con las cuales se comparta información. Cada organización que tenga la necesidad de contar con información de otra, podrá consultar el PIXM (*PIX Manager*) y éste le devolverá la información de identificación del paciente correspondiente a esa organización.

Este perfil da soporte a la opción de realizar cruzamiento de identificadores mediante la configuración de un *Master Patient Index* (MPI). El MPI es el dominio maestro de identificadores, y es global a todos los dominios de identificación. Existen dos posibles configuraciones para el MPI, una de ellas define el dominio de una organización en particular como MPI, y la otra define uno nuevo que sea independiente de los ya existentes en las organizaciones. La desventaja de elegir la primera opción es que, si por alguna razón, la organización que posee el dominio elegido como MPI no comparte más información surge la necesidad de resolver cómo se manejarán los identificadores de los nuevos pacientes que se ingresen en el sistema. Esto sucede porque ya no se contaría con el dominio de referencia que fue elegido como MPI.

PR2: Notificación de cambios en la identificación del paciente.

El perfil PIX también brinda la opción de que una organización se registre con el *PIX Manager* para ser notificada, en caso de que otra modifique la información de identificación de uno de sus pacientes, por ejemplo al corregir un dato demográfico.

RQF2 - Compartir información entre organizaciones

La información a compartir, es cualquier información clínica que se encuentre involucrada en la atención de pacientes por parte de las distintas organizaciones de Salud. Algunos ejemplos de ésta información pueden ser: Diagnóstico de atención, prescripción médica, órdenes para exámenes de laboratorio, detalles de admisión de pacientes en las instituciones, resumen de internación, resumen de alta etc.

PR3 - La información clínica del paciente se encuentra distribuida.

Para que dos sistemas informáticos que comparten información puedan realizar un intercambio exitoso, e interpretar la información de la misma manera, fueron analizados los perfiles XDS y XCA definidos por la IHE y explicados en el Marco Conceptual. Ambos solucionan la problemática específica de compartir la información de salud teniendo en cuenta que se encuentra distribuida.

El perfil XDS define un modelo de registro (*Registry*) / repositorio (R*epository*) que brinda la posibilidad a cualquier organización, de encontrar y acceder a toda la información clínica del paciente sin la necesidad de tener en cuenta qué organización crea y mantiene dicha información.

Esta separación permite la disociación entre los datos de identificación y los datos clínicos del paciente, lo que resulta transparente si se utiliza este perfil, dado que los primeros se guardan en el *Registry* y los segundos en el *Repository*. Este perfil permite fácilmente compartir información independiente del contenido, es decir XDS es neutral en cuanto al contenido y tipo de información, solo los consumidores y fuentes de documentos, son los encargados de procesar la información. Al ser centrado en los documentos permite realizar consultas pre-establecidas en el *Registry* y su posterior recuperación de documentos de los *Repositorios*, los cuales principalmente estarán en las organizaciones que son "dueñas" de sus documentos. Es un modelo escalable para cualquier tipo de organización, Hospital, Laboratorio, Clínica, etc.

Dado que el *Registry* está centralizado y los *Repositories* federados, brinda una solución que es agnóstica del tipo de documentos que se comparten. Además como sus transacciones están basadas en estándares de Web Services, este perfil se puede combinar con otros para manejar distintos tipos de documentos, por ejemplo: imágenes XDS-I, laboratorio XDS-LAB.

El perfil XCA es otra opción, que también define un mecanismo para compartir documentos, pero está enfocado en el intercambio más allá de la frontera de su dominio. Define un actor *Gateway* con transacciones para consulta y recuperación de documentos entre distintas comunidades. Una comunidad generalmente maneja los datos de identificación de sus pacientes y los formatos de codificación definidos, solamente dentro de la comunidad, lo cual se presenta como una nueva problemática al momento de intercambiar información con otras comunidades. La ventaja del modelo de comunidades XCA, es la capacidad de incorporarse, o formar parte de una región o nación reutilizando sus sistemas de información a un menor costo de integración. Está fuera del alcance de este proyecto analizar la implicancia interna de la comunidad, en términos de integración con otras comunidades, en caso de no contar con XDS. XCA presenta un nivel mayor de abstracción, de manera tal que por un lado, es más flexible que XDS en términos de las exigencias de aspectos particulares (por ejemplo, no exige la implementación de actores como *Registry* y *Repository*). Y por otro lado, al ser una solución más flexible, dificulta la posibilidad de garantizar la accesibilidad de toda la información existente. En el intercambio de información entre comunidades, XCA puede ser pensado como un "XDS que tiene su *Registry* federada". Implementando un actor y dos transacciones, el perfil XCA permite unirse a una solución más grande de interoperabilidad en salud (región, nación, etc.), de manera sencilla. Esta tarea puede ser muy fácil si se cuenta con toda la información requerida por las transacciones (principalmente los metadatos definidos en el *Registry* de XDS). Esto facilita mucho la posibilidad de reutilizar recursos de software que las organizaciones poseen y utilizan actualmente en sus procesos de atención a los pacientes.

Por otro lado, la desventaja de este perfil es que no brinda una solución para el aseguramiento de garantizar la privacidad y confidencialidad de la información de los pacientes. Esto sucede porque no exige la definición de políticas comunes entre las comunidades, sólo lo define "intra" comunidad. Tampoco se puede garantizar con certeza que en una solución basada en XCA, toda la información que posee la organización se encuentra accesible para compartir. Esto se debe a que el registro de documentos está federado, y es responsabilidad de cada comunidad, entonces puede suceder que la comunidad tenga definidas políticas más restrictivas con respecto al acceso de la información.

PR4 - Formatos heterogéneos de la información intercambiada.

En el marco de este proyecto no se define un formato específico para la representación de la información que se comparte. De todas maneras se tuvo en cuenta la importancia de que dicha representación sea estándar, ya que todos los sistemas informáticos involucrados en el intercambio, deben ser capaces de crear e interpretar dicha información con la menor ambigüedad posible. Por otro lado, un documento XDS es información clínica que puede estar estructurada o no, que está identificada, es legible, puede ser persistida, está custodiada y su acceso es autenticado. Se analizó el estándar HL7 CDA R2, porque contiene estas características bien definidas y especificadas. En su estructura de cabezal y cuerpo, el cabezal permite registrar la información necesaria para autenticación e integridad del documento, y el cuerpo que puede estar estructurado en secciones y que puede contener desde imágenes, texto narrativo y entradas codificadas en vocabularios clínicos. Además el estándar es lo suficientemente general, como para luego poder instanciarlo en cualquier tipo de documento clínico.

RQF3- Soporte para el cumplimiento de las leyes de privacidad

El requerimiento de contemplar el cumplimiento de las leyes de privacidad de la información involucra varias problemáticas. Por un lado el control de acceso a las funcionalidades por parte de organizaciones autorizadas para dicho fin. Por otro lado el control de acceso a la información clínica de los pacientes, que a su vez, está relacionada a los consentimientos del paciente y al cumplimiento de los mismos. Los consentimientos se componen de un conjunto de políticas que el paciente consiente. A continuación se describe en profundidad el análisis de estas problemáticas.

PR5 - Garantizar la seguridad y privacidad.

Para solucionar esta problemática fueron analizados los perfiles XUA, XSPA los cuales permiten definir los atributos necesarios para comunicar la identidad y necesidades de acceso a la información por parte del consumidor de un servicio, y que el proveedor pueda validar su autenticidad, y autorización al recurso solicitado.

Autenticación y control de acceso a las funcionalidades

La mayoría de los perfiles de IHE analizados, exigen la utilización del perfil ATNA. En dicho perfil se definen controles de acceso básicos para usuarios y nodos, así como también un modelo de auditoría. Es por esto que la solución debe garantizar el cumplimiento del mismo, para, a su vez, cumplir con el resto de los perfiles. El ATNA además especifica que todos los computadores que integren el dominio de seguridad que define, deben tener sus relojes sincronizados. Esto es lo que especifica el perfil CT, mediante la invocación a un servidor de tiempo con el protocolo NTP o SNTP.

Control de acceso a la información clínica del paciente

Para realizar los controles de acceso propios de salud, se deberán ejecutar los controles basados en las políticas definidas a nivel del dominio. Para llevar a cabo esto, es necesario definir un *token* de seguridad para autenticar y autorizar a las organizaciones. Por lo tanto cuando una organización desee consumir un servicio en el que interviene información de salud, debe enviar dicho *token* para poder realizar el control de acceso.

El control de acceso en salud tiene tres aspectos principales:

- Las políticas de control de acceso: Una vez definidas a nivel de dominio, se implementan con base en el perfil XSPA-XACML de la OASIS. Dicho perfil define la utilización de XACML como lenguaje de políticas para el control de acceso.
- Comunicación de los atributos de identidad del usuario: Una vez definidas las políticas, y por tanto qué información del usuario es necesaria para aplicarlas, se debe definir cómo comunicarla. La base de los atributos para la construcción de políticas y la forma de comunicación de los mismos se basa en el perfil XSPA-SAML de OASIS y el XUA de la IHE, ambos trabajan con SAML. La obtención del *token* se basa en el perfil XSPA-WS-Trust, en el que se indica utilizar el estándar WS-Trust para obtenerlo.
- Registro de las políticas consentidas por el paciente que se desarrolla en la siguiente sección.

Con respecto al control de acceso a la información clínica, los perfiles XSPA proponen un modelo de permisos basado en el rol, el propósito de uso, organización, localidad y los permisos especiales que pueda tener el usuario. Todos los atributos recomendados por XUA están incluidos en el perfilado que realiza XSPA, a excepción de la opción de enviar el identificador de un documento que provea autorización sobre el recurso solicitado al solicitante. El conjunto de perfiles XSPA además de especificar cómo representar las políticas de forma interoperable con XACML, también define la utilización de SAML 2.0 para intercambiar atributos de identidad de los usuarios y WS-Trust 1.3 como mecanismo de obtención del *token*. Define también un conjunto de atributos que pueden utilizarse. Por otro lado los atributos recomendados por el perfil XUA pueden utilizarse para intercambiar la identidad del usuario siendo a la vez compatible con los definidos por XSPA. Estos perfiles definen por tanto, el vocabulario de atributos a utilizar en el control de acceso.

Trazabilidad

El perfil ATNA especifica que todas las transacciones entre actores IHE deben ser logueadas. Para ello define el actor con el rol de repositorio de auditorías, el *Audit Record Repository*. Cada transacción genera al menos dos registros de log, uno por cada lado de la transacción.

La IHE define qué eventos deben registrarse, en su marco técnico. El envío de los registros se realiza a través del protocolo SYSLOG [60], sobre TLS [61].

Los mensajes de log se envían con formato XML definidos mediante un esquema XML, de acuerdo a la especificación realizada en la RFC 3881 [62]. Es un formato particular para los eventos relacionados a las transacciones IHE, por lo que no sería compatible con el sistema de auditoría actual de la PGE. Es por esta razón que las soluciones IHE deben contar con un actor *Audit Record Repository.*

El registro de las transacciones puede ser centralizado o distribuido. En el caso centralizado implica tener un único *Audit Recrod Repository* a nivel de solución de integración, donde todos los componentes IHE que intervengan deben enviar sus logs. En el caso distribuido, cada organización decide mantener un *Audit Record Repository* para manejar sus registros internos, además de tener uno centralizado. En estos casos la IHE indica que si alguno de los actores de dicha organización interacciona con otro

de otra organización que integra la solución, si bien se van a generar los registros en los repositorios de cada organización, también se deberán enviar copias de dichos registros al repositorio centralizado de la solución de integración. Estas copias son enviadas por los propios repositorios de auditoría.

PR6 - Registrar los consentimientos del paciente.

El registro de consentimientos se resuelve con el perfil BPPC de la IHE, guardando un documento con los identificadores de las políticas consentidas y la firma del paciente escaneada, en un Repositorio. Esto implica que en el momento en que el paciente realiza el consentimiento, se crea un documento CDA donde se registran las políticas consentidas, así como también la firma escaneada del paciente. Además en el metadato XDS *eventCodeList* se cargan todos los identificadores de las políticas que el paciente consintió. El conjunto de políticas que se pueden consentir, son las definidas para un determinado dominio de afinidad, y a su vez administradas por el Repositorio de políticas XACML perteneciente al dominio.

PR7 - Identificación de la información amparada bajo los documentos de consentimiento.

Más allá de la necesidad de capturar el consentimiento del paciente, se requiere para realizar el control de acceso correspondiente, poder relacionar los documentos a los consentimientos. Una alternativa es asociar el identificador del documento de un consentimiento al momento de registrar un documento, en esta opción los consentimientos del paciente podrían aplicar a documentos específicos. Otra opción es que los consentimientos sean globales al paciente, es decir el paciente acepta las políticas que aplican a todos sus documentos. Al momento de realizar el control de acceso, es necesario obtener los metadatos de los documentos BPPC desde *Registry* de XDS, para luego extraer los identificadores de políticas a ser aplicadas. Además como las políticas pueden aplicar a un documento en particular, se deben evaluar teniendo en cuenta los metadatos del cada documento que se desea consultar.

PR8 – Disociación de la información clínica, de los datos que identifican al paciente.

La disociación entre los datos de identificación del paciente y los datos clínicos, es transparente si se utiliza el perfil XDS para compartir, dado que los primeros se guardan en el *Registry* y los segundos en el *Repository*.

RQNF1 – Integración con la PGE

Las interacciones de los distintos sistemas de las organizaciones para compartir información de salud, deben realizarse a través de la PGE. Para resolver esta problemática, por un lado se analizó, la utilización del componente de seguridad que ésta define, en conjunto con los perfiles XUA y XSPA, y por otro lado la capacidad de infraestructura que ésta brinda.

PR9: Cómo realizar la integración con la PGE.

La elección de perfiles y estándares para resolver las problemáticas de los requerimientos funcionales, condiciona la facilidad con la que la solución se pueda integrar a la plataforma de interoperabilidad (PDI) definida en la PGE. Los requisitos que la PDI plantea es el uso de

Web Services, WS-Security, SAML 2.0 y WS-Trust. Las organizaciones de salud para interactuar con ella, deberán respetar dichos estándares. Lo que implica generar un *token* de seguridad SAML que contenga los atributos requeridos por la PDI, y a su vez los propios de salud. El perfil XSPA - WS-Trust recomienda la utilización de WS-Trust 1.3 para la obtención y autenticación del *token*. Por lo antedicho se puede diseñar un modelo de autenticación y obtención del *token* de seguridad, basado en los estándares SAML 2.0 y WS-Trust cumpliendo simultáneamente con los perfiles XSPA-SAML, XSPA-WS-Trust y XUA. Este modelo es compatible 100% con el actual funcionamiento de la PDI definida en la PGE, ya que como se mencionó, ésta utiliza SAML 2.0 y WS-Trust 1.3. Se debe definir un solo *token* de seguridad que permita realizar tanto el control de acceso que actualmente se realiza en la PDI de la PGE, y también los nuevos controles asociados a la información de salud. Esto implica extender el *token* actual, con los atributos de salud específicos definidos los perfiles XUA y XSPA. Luego de realizado en análisis de todo lo anterior, el *token* de salud propuesto se compone de los siguientes atributos:

Atributo	**Ejemplo**
Servicio a consumir	http://192.168.40.190:9000/IngresarAtención
Rol	uid=rolPruebaDoctor,cn=agesic
Nombre de Usuario	Juan
Rol estructural (ASTM 1986 Structured Role)	Physician
Propósito de Uso	TREATMENT
Permisos HL7 (Append, create, delete, read, update, etc.)	Read

Tabla 4: Token de Salud

Los primeros cuatro datos se corresponden al *token* actual utilizado por la PGE, los restantes son los datos de la extensión definida para acceso a salud.

PR10 - Identificar el reuso de recursos existentes en la PGE.

Un primer re-uso es el STS de la plataforma, el cual deberá ser capaz de generar el nuevo *token* extendido definido para salud, en base a los atributos que serán enviados en un *token* firmado por la organización consumidora del servicio. Por lo tanto el flujo de autenticación y autorización actual se seguirá manteniendo, en base a los mismos atributos, ignorando los atributos de salud que el *token* pueda contener. Como el flujo de autenticación con la PDI es el mismo que se realiza actualmente, no se necesitan nuevos componentes de software para llevar adelante esta tarea con el nuevo *token* de salud.

RQNF2 – Utilización de recursos existentes en las organizaciones

La problemática del re-uso de recursos existentes está relacionada, en el marco de este proyecto, directamente al hecho de que los sistemas existentes puedan integrarse de la manera más sencilla y rápida posible, a la solución definida para compartir información de salud a través de la PGE. Esto se relaciona principalmente con los perfiles XDS, XCA y PIX analizados arriba.

PR11 - Diversidad de estándares y protocolos en cada organización.

Al utilizar perfiles de IHE para compartir información, la implementación y/o instalación de un producto que cumpla con el actor correspondiente de IHE vuelve transparente su integración a la plataforma. Sin embargo, es posible que la organización tenga que realizar transformaciones en la información que maneja para poder registrar la información en la nueva solución de interoperabilidad.

PR12 - Reutilizar recursos y existencia de escasos recursos en las organizaciones.

Como ya se vio en la sección que trata acerca del tema de compartir información, los perfiles analizados para abordar dicha problemática son XDS y XCA de IHE, ambos perfiles tienen sus ventajas y desventajas y como veremos más adelante en la sección donde se plantean las alternativas de solución uno de ellos es más flexible que el otro en torno a este tema.

La problemática de la migración de la información existente a compartir, está por un lado relacionada a la reutilización de los sistemas existentes, ya analizada anteriormente en esta misma sección, y por otro lado también con la información que actualmente se encuentra representada solamente en papel, como veremos más adelante esta problemática no fue abordada en este proyecto y se desarrollará el conocimiento relevado en la sección de Trabajo Futuro de este mismo documento.

RQNF3 – Utilización de estándares y buenas prácticas de la industria.

Un aspecto importante al momento pensar en una solución general, interoperable y escalable, sin reinventar la rueda, es el relevamiento de estándares y buenas prácticas en la industria de la salud.

En actualidad existe una variedad de estándares en salud, como ser estándares de infraestructura (XDS, PIX) de contenido (HL7 CDA, ISO 13606) y de terminología (LONIC, SNOMED, etc.). Esto implica la existencia de una cantidad de transacciones complejas y de distintos tipos, difícil de abordar y comprender por si solas.

PR13 - Variedad de estándares, asimilación. Componer todo en una solución integral

El análisis realizado en este proyecto está basado en perfiles de IHE, OASIS y W3C, lo que permite lograr una solución interoperable dado que las soluciones definidas en estos perfiles se basan en la aplicación coordinada de estándares (DICOM, HL7, WS-*, SAML, TLS, etc.). Esta coordinación se logra por medio de la definición de actores y transacciones, lo que permite que un sistema con sólo implementar las transacciones del actor correspondiente, pueda integrarse fácilmente a la solución definida. Los perfiles hacen que una solución sea más interoperable, escalable y tenga menor costo de mantenimiento frente a soluciones hechas a medida.

A continuación y a modo de resumen, se presenta en la Tabla 4 cómo las problemáticas fueron cubiertas por los distintos perfiles y estándares utilizados

Requerimientos	Problemáticas	Estándar/Perfil
RQF1: La solución debe permitir identificar el mismo paciente en las distintas organizaciones donde es atendido.	**PR1:** Poder identificar al mismo paciente desde distintas organizaciones. **PR2:** Notificación de cambios en la identificación del paciente.	PIX
RQF2: La solución debe permitir que la información sea compartida (accedida y visualizada) entre los sistemas de las diferentes organizaciones, teniendo en cuenta que los pacientes están registrados con distintos identificadores.	**PR3:** La información clínica del paciente se encuentra distribuida. **PR4:** Formato de la información intercambiada.	XDS, XCA, HL7 CDA R2, PIX
RQF3: La solución debe contemplar el cumplimiento de las Leyes de Privacidad de la información.	**PR5:** Garantizar seguridad y privacidad. **PR6:** Registrar los consentimientos del paciente. **PR7:** Identificación de la información amparada bajo los documentos de consentimiento **PR8:** Disociación de la información clínica, de los datos que identifican al paciente.	PIX, XDS, BPPC, XSPA, XUA, ATNA, CT, SAML, XACML, WS-Turst, WS-Security
RQNF1: La solución debe compartir la información entre las organizaciones, a través de la PGE.	**PR9:** Cómo realizar la integración con la PGE. **PR10:** Identificar la reutilización de recursos existentes en la PGE.	SOAP, WS-Trust, SAML, XACML
RQNF2: Reutilizar los recursos existentes en las organizaciones de salud.	**PR11:** Diversidad de estándares y protocolos en cada organización. **PR12:** Reutilizar recursos y existencia de escasos recursos en las organizaciones.	XDS, XCA, PIX
RQNF3: La solución debe cumplir con los estándares y buenas prácticas de la industria de la salud.	**PR13:** Variedad de estándares, asimilación. Componer todo en una solución integral.	N/A

Tabla 5: Perfiles utilizados para resolver las problemáticas

3.3 Alternativas de Solución

En esta sección se describen las dos alternativas de solución analizadas previamente a la definición de la solución propuesta.

3.3.1 Dominio de Afinidad UY

La primera alternativa de solución se denomina “Dominio de Afinidad UY”, está basada en el perfil XDS de IHE, y en ella se define un dominio de afinidad a nivel de Uruguay. Esto implica que todas las organizaciones de salud que deseen compartir información, acuerden un entorno de cooperación, y establezcan un único conjunto común de políticas de seguridad y confidencialidad sobre la información. También deberán compartir una infraestructura común de comunicaciones.

En la Figura 27 se muestran los distintos componentes involucrados en la solución. Los actores XDS, el componente MPI/PIXM y finalmente los componentes de la PGE. Las flechas punteadas indican los destinatarios finales de los mensajes y las flechas continuas el flujo de información que es realizado a través de la PGE.

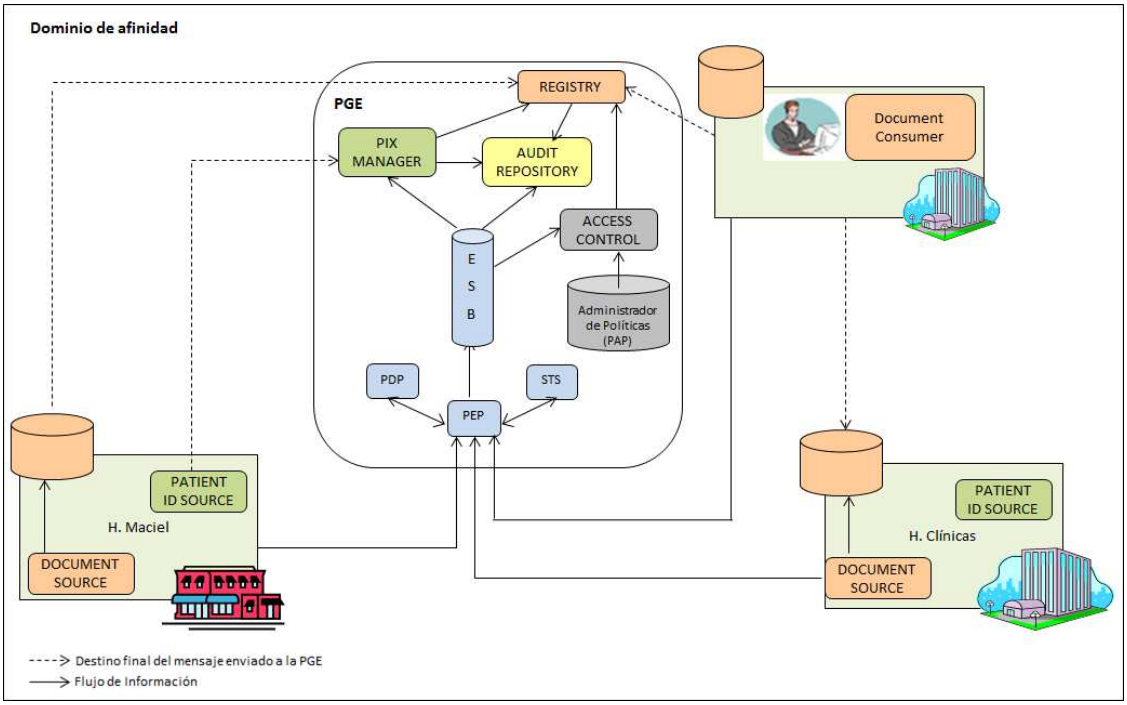


Figura 27: Diagrama estático de la solución Dominio de Afinidad UY

Esta solución utiliza el perfil PIX definido por IHE para resolver los cruzamientos entre identificadores, y las notificaciones de cambios en los datos de identidad del paciente. La configuración PIX que se utiliza es la de MPI, que mapea los distintos identificadores de un mismo paciente, con un único identificador global, el correspondiente al dominio de afinidad. Es decir que el MPI es independiente de los dominios de identificación existentes en las organizaciones. El componente MPI/PIX se encontrará alojado en la PGE.

Esta solución está basada en el perfil XDS de IHE; el actor *Registry* se encontrará centralizado a nivel de Uruguay y estará alojado en la PGE, al mismo tiempo que los distintos actores *Repository* se alojarán en las organizaciones de salud.

El mecanismo de autenticación y autorización de acceso a las funcionalidades de salud, será el mismo que se realiza actualmente en la PGE, y que está explicado en el Marco Conceptual de este documento.

Para conformar un dominio de afinidad XDS, es necesario poder establecer las políticas comunes de seguridad y control de acceso a la información clínica. Tanto el componente encargado de administrar las políticas, como el encargado de ejecutar el control de acceso, se alojarán en la PGE. La representación de las políticas se realiza en base a los perfiles XSPA, como se explicó en el análisis de estándares, y para ejecutar el mecanismo de control se utilizará el *token* extendido para salud, definido en análisis del requerimiento de integración con la PGE. Incorporar esta forma de control de acceso al Sistema de Seguridad que actualmente tiene la plataforma, implicaría que el PDP además de las políticas, y los atributos que vienen en el *token* de salud, utilice información de los metadatos de los documentos. El control puede realizarse solamente en función de los

atributos del *token*, sino que se debe consultar al *Registry* para obtener los metadatos de los documentos de consentimientos, para luego a partir de un metadato particular (*eventCodeList*) obtener los identificadores de las políticas a aplicar. Esto ocasiona que el PDP de la plataforma deba conocer cómo interactuar con el *Registry* XDS, y además conocer qué metadatos utiliza, cómo se implementa el perfil BPPC. También debería conocer la definición de los metadatos de cada documento para extraer los datos que se consultan allí para aplicar las políticas (por ejemplo el tipo de documento). De esta forma el Sistema de Seguridad quedaría altamente acoplado a la implementación particular que se está realizando de Plataforma de Salud y perdería el carácter más genérico que posee actualmente. Por esta razón la solución, no modifica el control de acceso actual sino que agrega un nivel más específico de control, el referente a información de salud. Dicho control queda encapsulado dentro de la órbita de la Plataforma de Salud (Access Control). Para cumplir con las exigencias de trazabilidad del perfil ATNA la solución cuenta con un repositorio de auditorías (*Audit Repository*) centralizado en la PGE, donde deben registrarse todas las transacciones realizadas entre actores IHE.

3.3.2 Comunidades

La segunda alternativa de solución se denominó “Comunidades” dado que está basada en el perfil XCA de IHE. En esta solución son las comunidades las que comparten información, y lo hacen de manera estandarizada. Cada comunidad está compuesta por varias organizaciones que han acordado un mecanismo para compartir información y políticas comunes a utilizar, dentro de la comunidad.

En la Figura 28 se muestran los distintos componentes involucrados en la solución. Por un lado se muestran los componentes pertenecientes al perfil XDS que podría ser la solución interna de una comunidad, el perfil PIX y finalmente los componentes de la PGE. Las flechas indican el flujo de información que es realizado a través de la PGE.

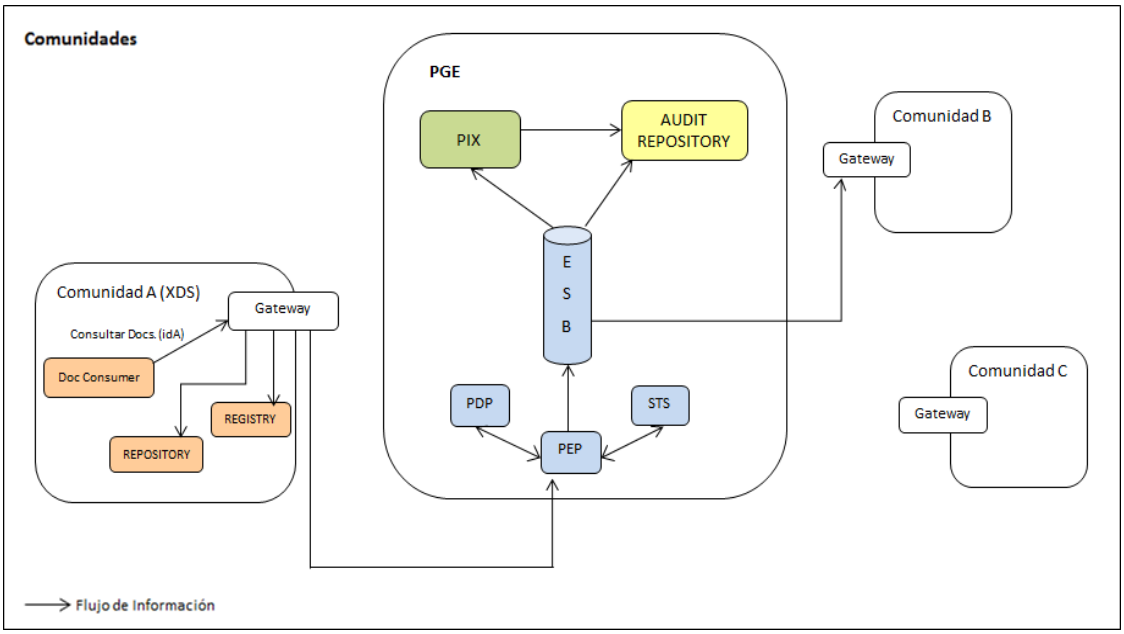


Figura 28: Diagrama estático de la solución de Comunidades

Cada comunidad, independientemente de la solución utilizada internamente para compartir información de salud, implementa el actor *Gateway* definido en el perfil XCA

que será el único punto de entrada/salida para compartir información de salud, desde y hacia la comunidad.

La identificación de los pacientes es modelada con el perfil PIX de IHE como se explicó en el análisis de dicha problemática, pero con la salvedad de que en esta solución se define un PIX a nivel de comunidades, es decir que el cruzamiento de identificadores es a través de los identificadores del paciente y las comunidades. Dicho componente *PIX Manager* estará alojado en la PGE. La interacción entre comunidades es realizada a través de la PGE, es decir que la autenticación y el control de acceso a las funcionalidades, es igual que el definido para la otra alternativa de solución. Si bien en esta solución (como se explicó en el análisis de perfiles y estándares), no es posible garantizar el control de acceso de la información de salud dentro de las comunidades, se exige que el *token* de seguridad utilizado para las interacciones sea el *token* extendido de salud definido para la otra alternativa de solución. También se exige que las políticas definidas y utilizadas, sean las mismas que las de la otra alternativa. Esto se debe a que es necesario lograr compatibilidad. De la misma forma y dado el contexto para el cual se define la solución (Plataforma de interoperabilidad en salud para Uruguay), se exige que la representación de las políticas de seguridad sea también igual que la utilizada en la otra alternativa de solución. Con respecto a la trazabilidad, esta solución también cuenta con un *Audit Repository,* donde se loguerán las taransacciones enre los Gateways y las que se realicen con el *PIX Manager*.

3.3.3 Resumen de alternativas de Solución y Conclusiones

Esta sección presenta un resumen de las alternativas analizadas en forma tabular, detallando algunos aspectos para ser utilizadas como criterios de comparación.

Alternativa/Criterio	Dominio de Afinidad UY	Comunidades
Identificación de pacientes	La información de identificación del paciente se encuentra centralizada. Cruzamiento de ids entre organizaciones, junto con un id global	La información de identificación está centralizada. Cruzamiento de ids entre comunidades, junto con id global.
Integración con la PGE	Mayor integración	Menor integración
Registro de consentimientos	Registro de consentimientos centralizado en la PGE	El registro de los consentimientos es responsabilidad de cada comunidad
Registro de documentos	Centralizado	Federado en comunidades.
Soporte a organizaciones que no cuenten con la infraestructura necesaria	Incorporación de repositorios centralizados	Es necesario que se implemente un actor *Gateway*
Control de acceso de documentos de salud	Control de acceso salud centralizado en la PGE. El Estado puede asegurar el cumplimiento de la ley	Control de acceso salud en cada comunidad. El Estado no puede asegurar el cumplimiento de la ley
Reutilizar los recursos existentes en las organizaciones de salud	Solución más acoplada. Instalación de repositorio.	Favorece la reutilización de sistemas de organizaciones que ya trabajan en conjunto.
Disociación de la información	El modelo XDS garantiza la	Las comunidades son responsables de

clínica, de los datos que identifican al paciente	disociación de la información clínica de los datos identificatorios.	garantizar la disociación de la información.
Interoperabilidad	Mayor interoperabilidad dado que existe consenso en la información a compartir.	Menor, debido a que las comunidades pueden definir otro esquema de representación para la información.

Tabla 6: Resumen de alternativas

A continuación se realiza la comparación de las alternativas propuestas en base a los criterios listados en la Tabla 5.

Identificación de pacientes

En ambas alternativas la identificación queda resuelta con un *PIX Manager* centralizado en la PGE. De esta forma el Estado puede tener el control de la información de identidad de los pacientes. Sin embargo hay una diferencia que es que en el "Dominio de Afinidad UY" los identificadores que se mantienen en el *PIX Manager* son los identificadores reales de las organizaciones de salud, mientras que en la alternativa de "Comunidades" son los identificadores de las Comunidades. Esto hace que la alternativa "Dominio de Afinidad UY" permita un mayor control por parte del Estado, de cómo se identifican los pacientes.

Integración con la PGE

Ambas alternativas realizan las integraciones a través de la PGE, reusando el Sistema de Seguridad y la PDI. Sin embargo en lo que refiere a infraestructura, si bien la alternativa de comunidades tiene dos componentes en la PDI, la alternativa del Dominio de Afinidad UY está casi completamente soportada por la infraestructura de la PGE, a excepción de los repositorios.

Registro de documentos

La alternativa de un dominio de afinidad tiene un registro de documentos centralizado en la PGE. Esto permite al Estado tener identificada toda la información relacionada a los documentos compartidos así como garantizar la disponibilidad del registro para realizar consultas.

Soporte a organizaciones que no cuenten con infraestructura

Si bien ambas soluciones implican que las organizaciones deban integrarse a través de Web Services con la PGE también requieren la implementación de actores que en algunos casos puede ser costosa. Para la alternativa 1 las organizaciones deberían disponer de al menos un repositorio de documentos. Este caso podría resolverse brindando en la PGE uno o varios repositorios centralizados donde las organizaciones registren los documentos que desean compartir. En el caso de las comunidades se debe implementar un *Gateway*, no habría una alternativa para evitar esto.

Control de acceso de documentos de salud

Las dos alternativas reutilizan el Sistema de Seguridad de la PGE y por tanto se garantiza un control de acceso básico con RBAC. Con respecto al control de acceso sobre los documentos clínicos, en la alternativa 1 se dispone de un componente especializado para

este fin en la PGE. Por lo tanto se puede garantizar el control de acceso en salud a nivel gobierno, verificando el cumplimiento de las leyes. Las comunidades por otro lado realizan el control de acceso internamente. Si bien se les exige que cumplan con las leyes de privacidad, no se puede garantizar.

Registro de consentimientos

La alternativa uno registra los documentos de consentimientos en el registro de documentos centralizados y realiza el control de acceso basado en los mismos. En la alternativa dos, las comunidades son responsables de esto, en tanto que es parte de lo necesario para contemplar las leyes de privacidad del Uruguay. Por lo tanto en la alternativa uno se garantiza claramente que los consentimientos serán tenidos en cuenta para el control de acceso, no así, en la dos.

Disociación de la información clínica, de los datos que identifican al paciente

En la alternativa uno, el modelo XDS de registro - repositorio garantiza la disociación de los datos clínicos de los datos que identifican al paciente. Las comunidades deben resolver esto internamente.

Reutilizar los recursos existentes en las organizaciones de salud

La alternativa uno, basada en un dominio de afinidad XDS es una solución más acoplada donde las organizaciones que la integran deben trabajar bajo un conjunto de reglas comunes. Esto podría significar cambios en los procedimientos internos de las organizaciones. Implica también un compromiso de infraestructura, ya que contar con al menos un repositorio XDS con los documentos que deseen compartir y además deben cumplir con las reglas de disponibilidad, y capacidades de respuesta que se definan a nivel de dominio.

Las comunidades pretenden ser un modelo más desacoplado, intentando aprovechar el camino realizado por organizaciones que ya tengan una forma de compartir documentos establecida respetando estándares y las leyes del Uruguay. En esos casos el hecho de definir una comunidad permite implementando un *Gateway* comenzar a compartir información sin necesidad de cambiar sus proceso internos y mantener total control sobre los documentos que posee.

Conclusiones

De lo anterior, se pude concluir que la alternativa uno, “Dominio de Afinidad UY” presenta una solución integral a la hora de compartir documentos con una fuerte participación del Estado a través de la PGE. De esta forma el Estado es responsable del control de acceso sobre los servicios y sobre los documentos, garantizando el cumplimiento de las leyes de privacidad tomando en cuenta los consentimientos informados del paciente. Al ser responsable de la información de identificación del paciente y el registro de documentos, puede garantizar la disponibilidad y confiabilidad de esta información. Sin embargo, pueden existir organizaciones que ya estén trabajando de forma cooperativa bajo un conjunto de procedimientos y políticas de acceso comunes utilizando los estándares del mercado. En estos casos, la integración al dominio de afinidad podría implicar desechar una solución en funcionamiento con años de trabajo y recursos consumidos. Para contemplar estas situaciones es que elegimos una solución de compromiso, híbrida entre

las dos alternativas. Es decir que la solución tendrá un dominio de afinidad a nivel país expuesto como comunidad (la “Comunidad UY”) y se permitirá la creación de nuevas comunidades para capitalizar soluciones de integración existentes que ya resuelven muchos de los aspectos que involucran un dominio de afinidad.

En el siguiente capítulo se detalla el diseño de la solución elegida.

4 Diseño de la Solución

La solución elegida para implementar una plataforma de integración de servicios en el área de la salud a través de la PGE y poder intercambiar documentos, es un híbrido de las dos alternativas presentadas en el capítulo anterior. Esta solución, que a partir de ahora denominamos “Híbrida”, define un modelo de comunidades que abarca todo el Uruguay y en particular una comunidad a nivel de gobierno denominada Comunidad UY que implementa internamente un Dominio de Afinidad. En las siguientes secciones se describe el diseño de la solución Híbrida, presentando una descripción general, vista de componentes con sus dependencias y cómo se resolvieron los requerimientos planteados en el capítulo 3. Por último se muestran las interacciones entre componentes para la consulta y recuperación de documentos.

4.1 Descripción General

A continuación se realiza una descripción general de la solución. En la Figura 29 se presenta un diagrama de bloques que brinda una visión macro de los componentes que intervienen.

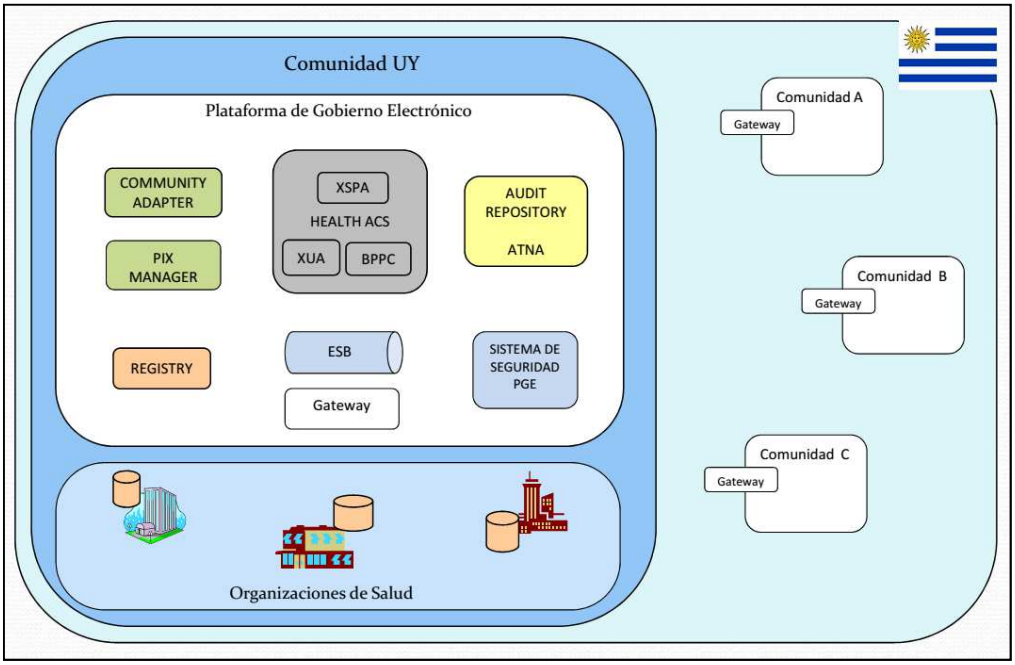


Figura 29: Diagrama de bloques de la solución

La construcción del Dominio de Afinidad para la Comunidad UY se realiza con base en la alternativa 1. Por lo tanto la PGE contará con un *Pix Manager*, un actor registro (*Registry*) de documentos, un componente responsable de realizar el control de acceso propio de salud (*Health ACS*) y un repositorio de auditorías (*Audit Repository*). Además cada organización podría mantener uno o más repositorios con los documentos que desee compartir. Estos repositorios deben estar configurados para registrar los metadatos de los documentos en el registro de la PGE. A estos actores se agrega un *Gateway* para terminar de definir la Comunidad UY. Esto modifica las interacciones internas del Dominio de Afinidad. En particular las consultas de documentos en lugar de ser dirigidas al registro, serán dirigidas al *Gateway*. De esta forma la Comunidad UY queda lista para comenzar a

compartir documentos con otras comunidades que puedan conformarse en el Uruguay. La implementación de un modelo de comunidades para compartir información a través de la PGE, implicar resolver dos aspectos principales, determinar las comunidades que conocen un determinado paciente y cómo lo identifican. Esto puede resolverse agregando un nuevo *Pix Manager* a la solución, para cruzar los identificadores de los pacientes en las distintas comunidades. Sin embargo como se detalla más adelante en este mismo capítulo, se puede utilizar el mismo *PIX Manager* para resolver las necesidades de la Comunidad UY a la vez que las de las comunidades, utilizando un componente adaptador (*Community Adapter*).

Con respecto a la seguridad y privacidad, la solución reutiliza el Sistema de Seguridad de la PGE para la autenticación y para realizar un control de acceso básico RBAC. El control de acceso propio de salud se debe realizar en cada comunidad. En particular, dentro de la Comunidad UY se cuenta con un componente especializado para éste fin. Cabe aclarar que se exigirá a cada comunidad realizar el control de acceso en base al perfil XSPA utilizando políticas XACML, y que el conjunto de políticas a aplicar sea el definido por la Comunidad UY (en tanto que es la comunidad a nivel país). Más adelante en este capítulo se detallan los mecanismos de seguridad utilizados. Como parte del requerimiento de contemplar las leyes de privacidad el Uruguay, en la Comunidad UY se registra el consentimiento informado de los pacientes a través del perfil BPPC. Los consentimientos serán utilizados para realizar el control de acceso.

4.2 Vista de Componentes

En esta sección se describen los componentes que intervienen en la solución y sus dependencias. En la Figura 30 se presenta una vista de los componentes que intervienen en la solución.

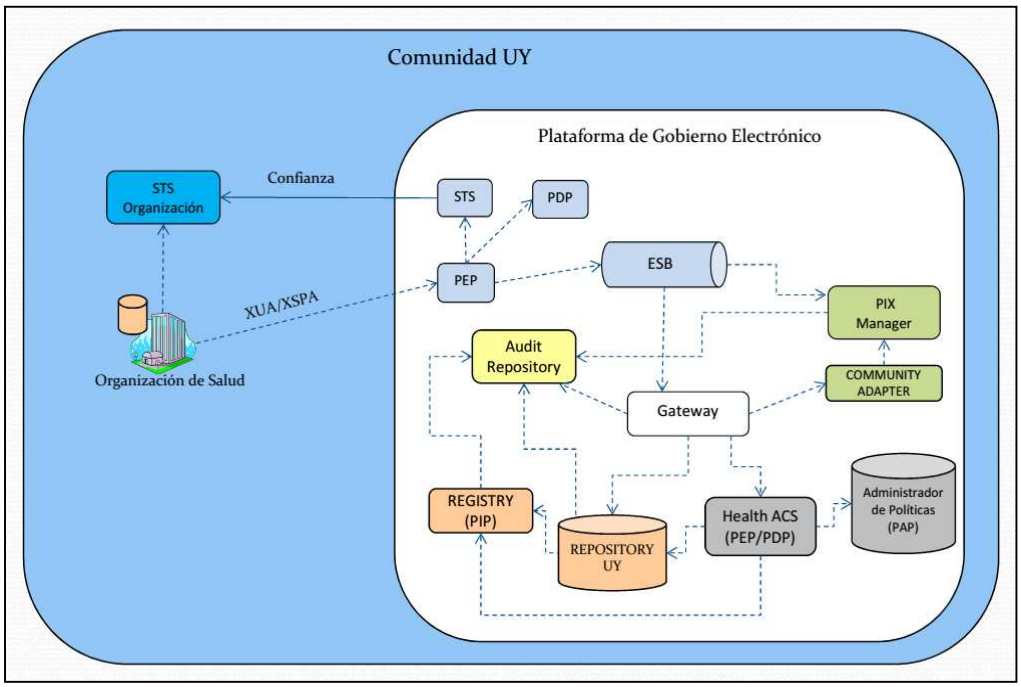


Figura 30: Vista de componentes

PIX Manager

Es el actor *PIX Manager* del perfil PIX, configurado como MPI. Mantiene los datos de identificación de todos los pacientes. Resuelve el cruzamiento de identificadores entre organizaciones, y entre comunidades. Se configura como identificador global, el correspondiente a la Comunidad UY.

Community Adapter

Este componente adapta el *PIX Manager* para resolver el cruzamiento de identificadores entre comunidades. Es utilizado por los gateways de las comunidades para determinar, en qué comunidades es conocido un determinado paciente y obtener su identificador. Realiza una consulta al *PIX Manager* y posteriormente filtra el resultado quedándose solamente con los identificadores de pacientes que corresponden a dominios de identificación que representan comunidades. Para poder realizar esto deberá conocer los identificadores de todas las comunidades que tengan registrados identificadores en el *Pix Manager*. Por esta razón se mantendrá un registro de identificadores de comunidad en una base de datos.

Cada nueva comunidad que se integre deberá ser registrada por procedimiento en la tabla de identificadores de comunidades.

Gateway

Es el actor *Gateway* del perfil XCA. Es responsable de resolver las consultas de documentos internas y desde otras comunidades. En este último caso las reenvía al Registro de documentos para obtener los metadatos a responder. Las consultas de documentos realizadas desde dentro de la comunidad deben ser dirigidas al *Gateway*. Luego éste, mediante una consulta al Community Adapter, determina si existen datos en otras comunidades y en tal caso realiza una consulta a las correspondientes comunidades. Además consulta el Registro de la comunidad UY a través del componente de control de acceso Health ACS. Una vez obtenidos los metadatos de otras comunidades y de la Comunidad UY consolida la información y se la devuelve al consumidor que la solicitó. También es responsable de recibir y responder solicitudes de recuperación de documentos, desde dentro de la Comunidad UY, como desde otras comunidades.

Administrador de políticas

Esta solución se basa en el conjunto de perfiles XSPA, que como se indicó en secciones anteriores especifica la utilización de XACML como lenguaje para representar las políticas de control de acceso. Este componente, tiene el rol de *Policy Administration Point (PAP)* y funciona como un repositorio centralizado de políticas de control de acceso a información de salud, para la Comunidad UY. Las políticas son referenciadas por un identificador, y el repositorio ofrece la posibilidad de consultarlas a través del mismo.

***Repository* UY**

Es un Repositorio de documentos XDS integrado a la Comunidad UY, centralizado en la PGE y administrado como un componente más de la misma, con el propósito de:

- Brindar soporte a organizaciones que no sean capaces de instalar su propio repositorio porque no poseen la infraestructura necesaria, pero les interesa compartir documentos a través de la Comunidad UY.

- Ofrecer un servicio a las organizaciones para compartir documentos a demanda a través de la Comunidad UY.

Registry

Implementa el actor *Registry* del perfil XDS. Contiene los metadatos de los documentos que se encuentran almacenados en los repositorios de las organizaciones pertenecientes a la Comunidad UY y en el *Repository* UY mantenido por la comunidad.

Health ACS

Este componente es responsable de filtrar todas las consultas de documentos realizadas al Registro XDS, ejecutando el control de acceso propio de Salud. Actúa como Punto de Decisión de Políticas (*Policy Decision Point, PDP*) de Salud, ya que en base a los atributos recibidos en el *token* de salud (*token* PGE + atributos Salud) y las políticas consentidas por el paciente, toma la decisión de autorizar o no la solicitud de documentos.

Luego, en base a la decisión tomada, actúa como Punto de Aplicación de Políticas (Policy Enforcement Point, PEP).

Para determinar los identificadores de las políticas que el paciente consintió, solicita los documentos BPPC del mismo al Registro de documentos, que actúa como Punto de Información de Políticas (*Policy Information Point, PIP*). Luego, con los identificadores obtenidos, consulta el Repositorio de políticas XACML para obtener las políticas a ser aplicadas.

Eventualmente podría pasar que el paciente haya consentido un conjunto de políticas, que restringe el acceso a un subconjunto de los documentos consultados. En este caso el Health ACS realizará un filtrado de los metadatos recibidos, retornando solamente los que son autorizados para el solicitante.

Audit *Repository*

Este componente implementa el actor Repositorio de Auditorías del perfil ATNA. Es un repositorio centralizado donde se registran todas las transacciones realizadas entre todos los actores IHE. Se registran al menos dos registros por transacción, uno por cada lado de la misma.

La IHE define qué debe registrarse en cada transacción, en su marco técnico. El envío de los registros se realiza a través del protocolo SYSLOG [60], sobre TLS [61].

Los mensajes de log se envían con formato XML definidos mediante un esquema XML, de acuerdo a la especificación realizada en la RFC 3881 [62].

4.3 Identificación de Pacientes en organizaciones y comunidades

Al tratarse de una solución híbrida, el intercambio de documentos se produce entre organizaciones y comunidades. Por esta razón, el requerimiento de identificar al mismo paciente en diferentes organizaciones se extiende para incluir también las comunidades. El actor *PIX Manager* del perfil PIX, resuelve el cruzamiento de identificadores entre distintos dominios de identificación. En este caso el conjunto de dominios de identificación está compuesto por la unión de todas las organizaciones pertenecientes a la Comunidad UY y todas las comunidades del Uruguay que se integren a la solución. Como además, dentro de

la Comunidad UY se implementa un Dominio de Afinidad, es necesario considerar un identificador cómo global dentro de la Comunidad UY. Esto es lo que resuelve la configuración MPI del perfil PIX, en la que se define uno de los dominios de identificación como global. En esta solución se plantea que el identificador global sea el de la Comunidad UY. Todos los documentos que se registren dentro de la Comunidad UY, deben ser registrados con este identificador. La solución Híbrida dispondrá de un único *PIX Manager* centralizado en la PGE que registrará identificadores tanto de organizaciones como de comunidades, configurado como MPI, tomando como identificador global, el de la Comunidad UY. En la Figura 30 se pueden ver las interacciones que se realizan con el *PIX Manager*, tanto desde organizaciones como desde comunidades.

4.4 Consulta de documentos: Interacciones Community Adapter

Los gateways (incluyendo el de la Comunidad UY) actuando con el rol de *Initiating Gateway,* ante una consulta de una organización, deben poder determinar qué comunidades consultar y qué identificador enviar en la solicitud. Mediante una consulta al *PIX Manager* se puede determinar los dominios de identificación que conocen al paciente y con qué identificador lo referencia. Esta información no alcanza, dado que es necesario saber si los dominios de identificación representan organizaciones o comunidades. Para resolver esto la solución Híbrida cuenta con un componente denominado *Community Adapter*, que adapta el *PIX Manager*, para ser utilizado por los gateways de las diferentes comunidades. El *Community Adapter*, filtrará las consultas al *PIX Manager* realizadas por los gateways, devolviendo solamente los identificadores que corresponden a dominios de identificación de comunidad, cumpliendo el rol de un *PIX Manager* a nivel de comunidades. Como dentro de la Comunidad UY se implementa un Dominio de Afinidad XDS, el *Gateway* además de consultar las correspondientes comunidades, consulta el registro de documentos de la comunidad UY.

En la Figura 31 se pueden ver las interacciones que se realizan con el Community Adapter.

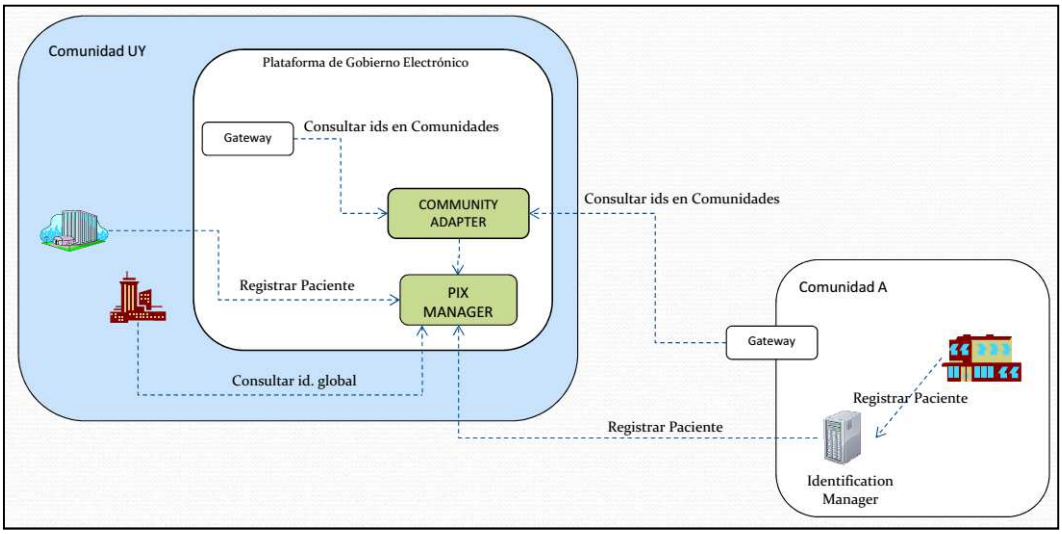


Figura 31: Interacciones con Community Adapter y PIX Manager

4.5 Mecanismos de Seguridad y Privacidad

La solución reutiliza el Sistema de Seguridad de la PGE para llevar adelante la autenticación y autorización, y se basa además en los perfiles XSPA y ATNA. El perfil XSPA define la utilización de SAML como *token* de seguridad, WS-Trust como mecanismo de autenticación y XACML como lenguaje de políticas de acceso. Estos estándares son utilizados por el Sistema de Seguridad de la PGE lo que permite adaptar los mecanismos actuales para realizar el control de acceso en salud. Además XSPA especifica el conjunto de atributos en los que se debe basar el control de acceso. Por lo tanto se extiende el *token* SAML actual de la plataforma para incluir estos atributos.

El perfil ATNA especifica requerimientos básicos de seguridad en el canal de comunicación con el protocolo TLS, autenticación de usuarios, control de acceso básico a nivel de aplicaciones, seguridad en los Web Services con WS-Security, y trazabilidad de las transacciones. Todas las interacciones en el marco esta solución se llevan adelante a través de la PGE. Esto implica el uso de la Red UY con canal seguro y seguridad en los servicios con WS-Security. Además se reutiliza el Sistema de Seguridad de la PGE donde quedan cubiertos los requerimientos de autenticación de usuarios y control de acceso básico a nivel de aplicaciones, solicitados por el perfil ATNA.

Registro de Consentimientos en la Comunidad UY

El registro de las políticas consentidas por el usuario se realiza a utilizando el perfil BPPC. Esto implica que en el momento que el paciente realiza el consentimiento, se crea un documento CDA donde se registran las políticas consentidas, así como también la firma escaneada del consentimiento del paciente. Además en el metadato XDS “eventCodeList” se cargan todos los identificadores de las políticas que el paciente consintió. El conjunto de políticas que el paciente puede consentir, son las definidas a nivel de comunidad y administradas por el Repositorio de políticas XACML.

Autenticación

La autenticación será llevada adelante por Sistema de Seguridad de la PGE utilizando el *token* extendido. Esto aplica tanto para las interacciones entre organizaciones dentro de la Comunidad UY, como las interacciones entre los gateways de las distintas comunidades.

Autorización

La autorización será llevada adelante en dos etapas. Primero se ejecutará la autorización basada en RBAC que realiza actualmente el Sistema de Seguridad de la PGE y luego la autorización de salud. Esta segunda etapa debe realizarse dentro de cada comunidad bajo un conjunto común de políticas de acceso definidas por el Estado Uruguayo acorde con las leyes de privacidad y utilizando los atributos y estándares especificados en el perfil XSPA como son SAML, WS-Trust y XACML. Esto implica también que cada comunidad debe tener registrados los consentimientos informados por los pacientes.

Dentro de la Comunidad UY, se tiene un componente especializado para el control de acceso en salud, el Health ACS. El Health ACS realizará el control de acceso en cada consulta documentos tomando en cuenta los consentimientos registrados para el paciente al cual pertenece la información clínica. Por lo que el Health ACS deberá aplicar solamente las políticas que el paciente consintió. La solución cuenta con el componente Administrador de Políticas, responsable de crear las políticas XACML y de proveer

interfaces de consulta y recuperación. Por lo antedicho se puede apreciar que el Health ACS cumple con los roles de PEP y PDP, el Administrador de Políticas cumple con el rol de PAP y que el *Registry* cumple con el rol de PIP en el flujo de control de acceso con XACML.

Trazabilidad

Con respecto a la trazabilidad, el perfil ATNA define un actor repositorio de registros de auditoría (*Audit Repository*), un formato para los registros de log, y que el intercambio de los mismos deber ser a través del protocolo SYSLOG [60]. Por lo tanto para cumplir con este perfil, la solución Híbrida cuenta con un *Audit Repository,* donde se deben registrar las transacciones realizadas. Cada transacción debe generar al menos dos logs, uno de cada lado de la misma. Es decir que las Organizaciones enviarán registros de log al *Audit Repository*.

4.6 Reutilización de los sistemas existentes en las organizaciones

Toda la solución está basada en perfiles y estándares tanto para resolver problemas propios de integraciones de sistema de salud o sistemas en general. En particular los perfiles de IHE hacen más sencillas las integraciones dado que definen sus interacciones en base a un conjunto de actores y transacciones. Por lo tanto a la hora de integrarse basta con tener implementados los correspondientes actores.

Esta solución apunta a que las organizaciones que deseen compartir información se integren a la Comunidad UY. Esto implica que deben contar un actor Repositorio, lo cual para algunas puede ser complicado, por escasez de recursos o conocimiento. Por esta razón se contará con un repositorio centralizado en la PGE, el Repositorio UY. De esta forma se facilita a las organizaciones la integración a la Comunidad UY. Este repositorio puede ser utilizado también como medio para compartir información a demanda entre organizaciones. Eventualmente esto puede escalarse y se podrían tener varios repositorios centralizados organizados por ejemplo por dominio de información.

Esta solución apunta a que las organizaciones se unan a la Comunidad UY, para que de esta forma el Estado Uruguayo pueda garantizar la seguridad y la privacidad de la información relacionada al cuidado de la salud, de todas las personas que se atienden o atendieron en organizaciones de salud dentro del Uruguay. Sin embargo se propone una solución híbrida que, en ciertas situaciones, favorece el reuso. Para crear una comunidad se debe contar con un actor *Gateway* que el responsable del intercambio de documentos de la comunidad. Este actor exige dos transacciones, una de consulta y una de recuperación que involucran el intercambio de metadatos. De contar con la información que representan esos metadatos se podría adaptar un sistema existente para interactuar con el *Gateway* logrando de esta forma la reutilización de dicho sistema. Las integraciones por comunidades son apropiadas en casos donde se tiene, de facto, una comunidad. Estos pueden ser casos de integración por región o por un conjunto de organizaciones que ya se encuentran organizadas bajo un conjunto de estatutos y políticas de acceso a la información comunes.

4.7 Integración con la PGE

La plataforma de gobierno electrónico, da soporte a una arquitectura SOA a nivel país, y todas las integraciones que se realizan a través de misma, son a través de Web Services y sus estándares asociados. A su vez la solución que se describe en este capítulo, para una

plataforma de integración de servicios de software que intercambian información de salud, brinda interfaces Web Services para las integraciones tanto entre sus componentes, como hacia los consumidores externos[1].

La información de salud, requiere de un control de acceso más restrictivo que el que es realizado por la PGE. Por esta razón la solución propuesta complementa el control de acceso de al PGE, con uno particular de salud a través de un componente especializado, el Health ACS.

Por lo antedicho subiendo el nivel de abstracción, se puede ver la solución propuesta, como un componente cerrado que permite la interoperabilidad de la información de salud entre organizaciones y comunidades realizando la integración a través de Web Services, donde la autorización es realizada por el Sistema de Seguridad de la PGE. Es así que la plataforma de Salud, forma parte de la Plataforma de Interoperabilidad de la PGE. En la Figura 32 se muestra la Plataforma de Salud, enmarcada dentro de la PGE.

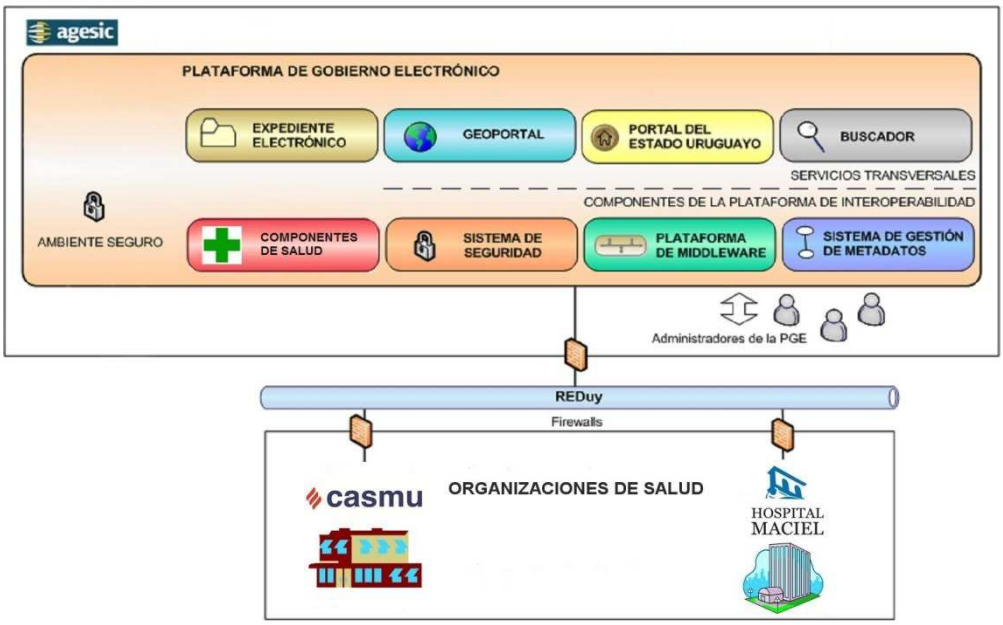


Figura 32: Plataforma de Salud en la PGE

4.8 Utilización de estándares y buenas prácticas

La solución está basada en los perfiles de integración de la IHE. Estos perfiles resuelven problemáticas de integración propias de salud basados en estándares y buenas prácticas del mercado. También se utilizan perfiles especializados en seguridad de la información de salud de la OASIS. Se utilizaron también estándares de organizaciones como OASIS, W3C y HL7.

4.9 Flujo de consulta y recuperación

A continuación se presenta un flujo de interacción con la plataforma de integración de salud, en el que una organización de salud, la Organización A de la Comunidad UY consulta

[1] Cabe aclarar que el *Audit Repository* brinda una interfaz SYSLOG para el registro de logs de acuerdo con el perfil ATNA.

los documentos de un paciente y elige uno para recuperar. Existen documentos en la Comunidad UY y en otra comunidad, la Comunidad B.

Los pasos ejecutados por el sistema de la Organización A son:

1. Consulta el identificador global del paciente (Figura 33)
2. Consulta de documentos (Figura 34 y Figura 35)
3. Recuperación de un determinado documento (Figura 36)

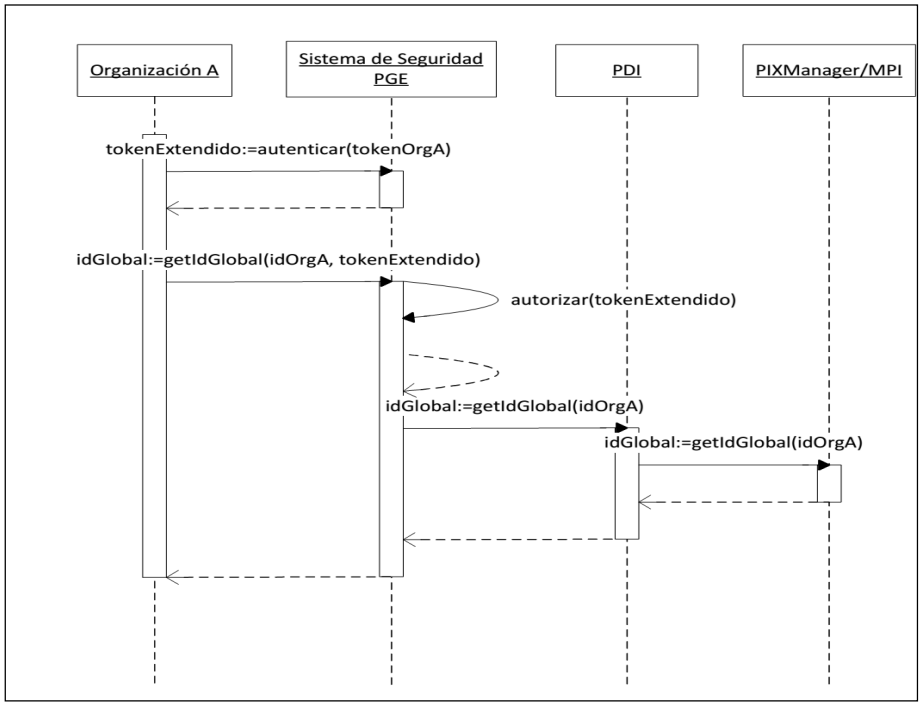


Figura 33: Consultar el identificador global de un paciente

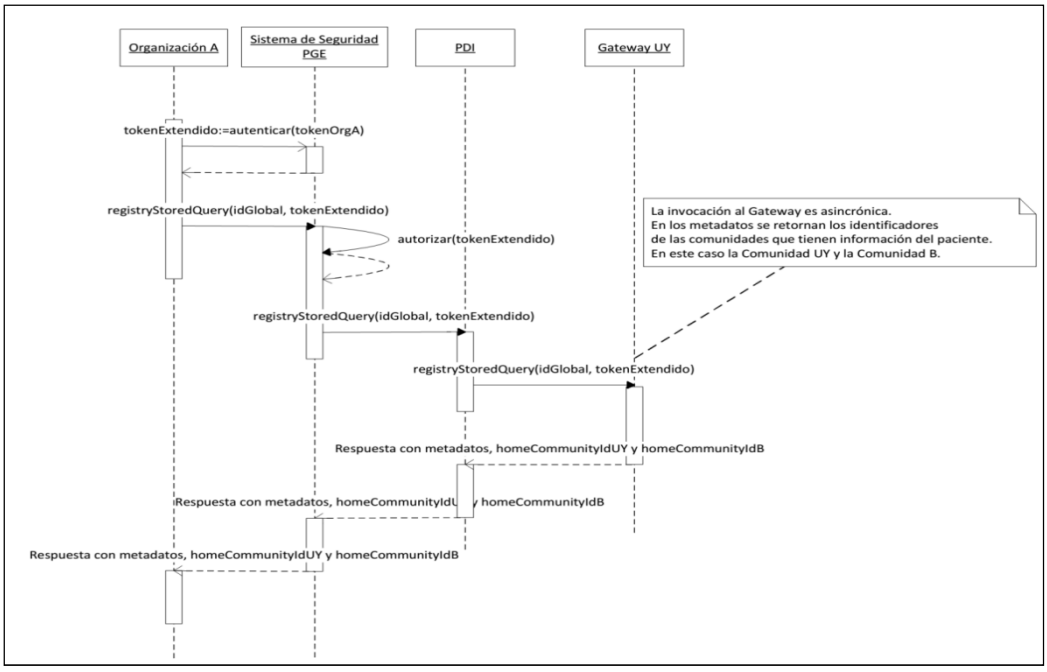


Figura 34: Consulta de documentos

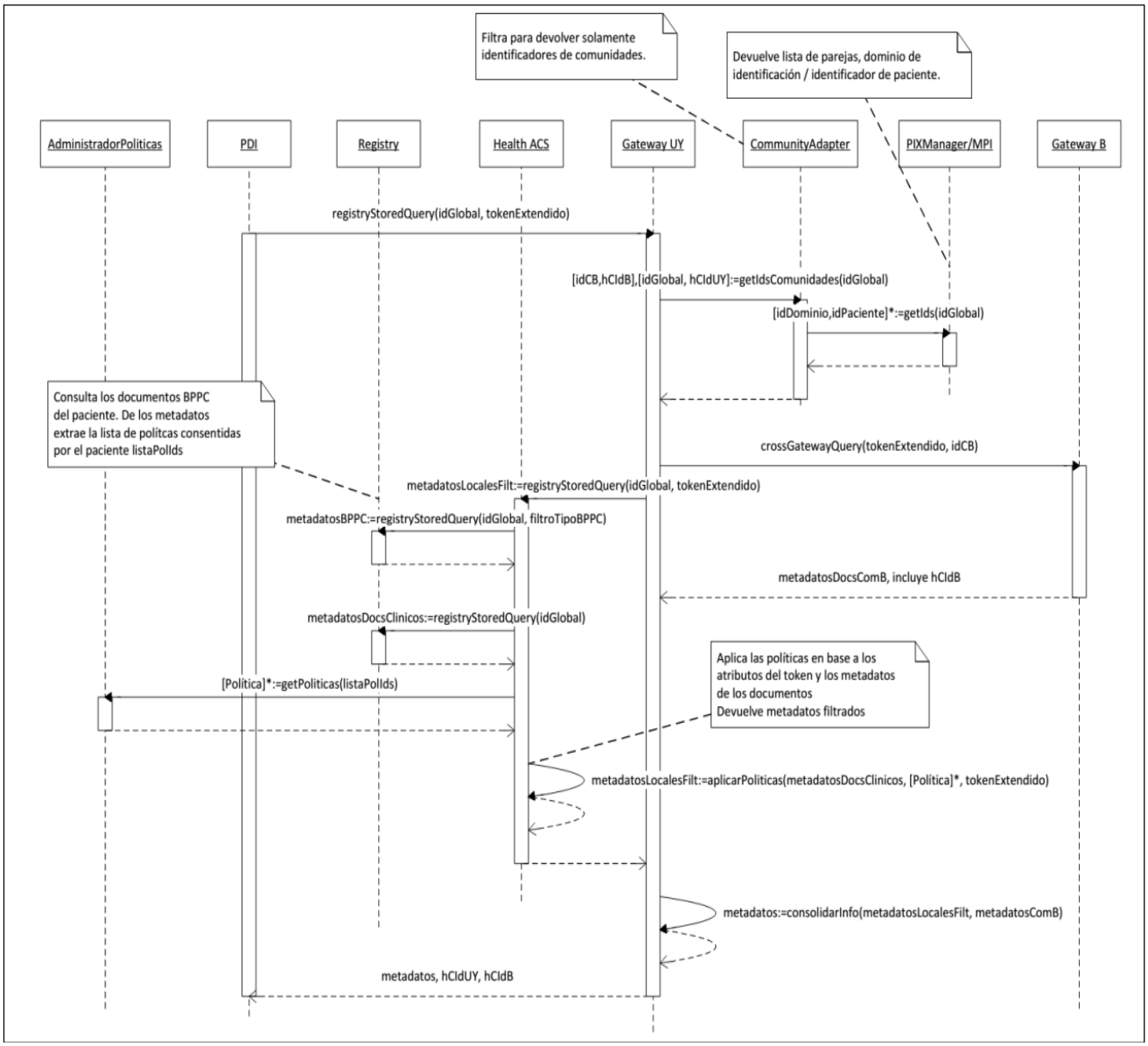


Figura 35: Consulta de documentos con detalle

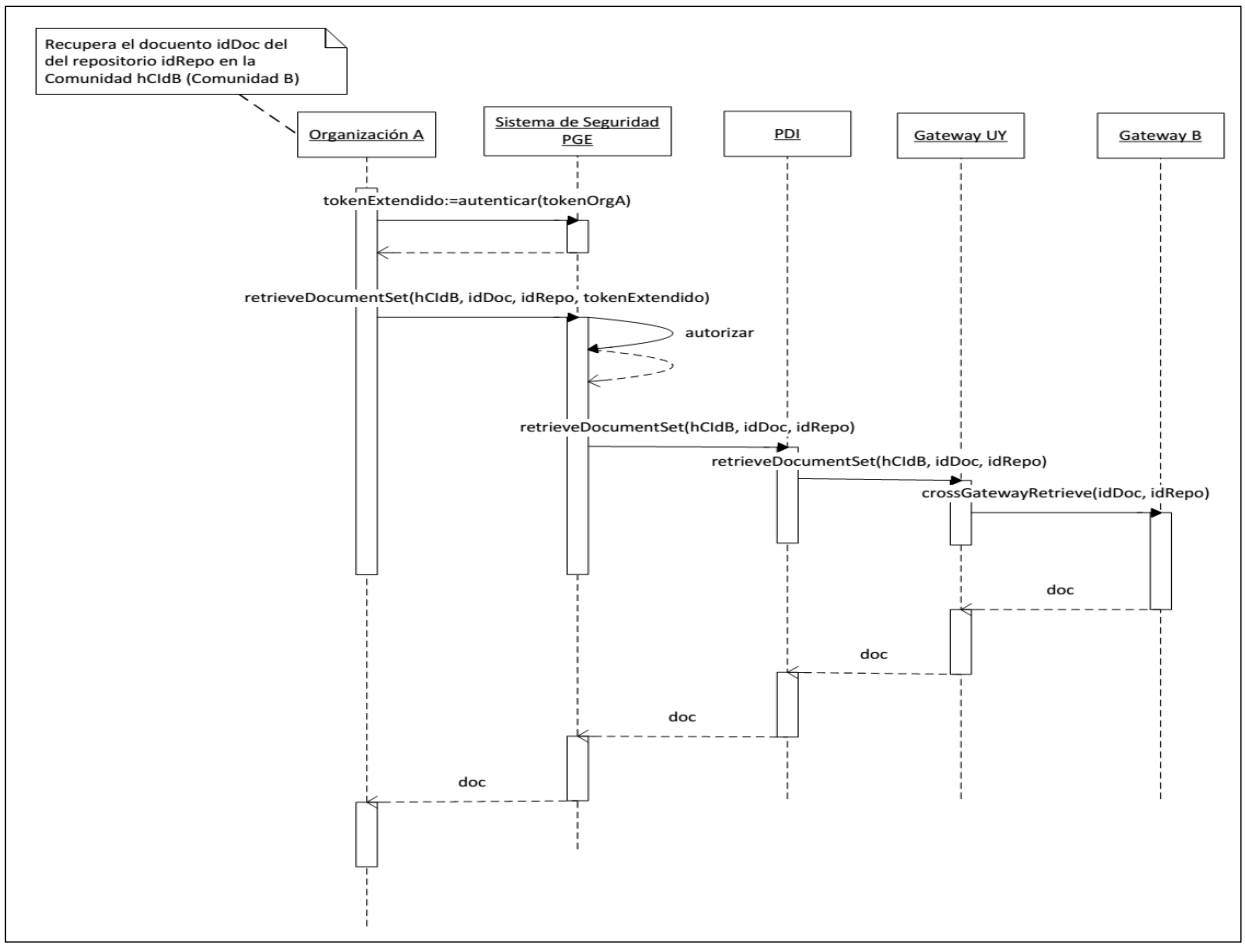


Figura 36**: Recuperación de un documento**

5 Escenarios de uso e implementación

En este capítulo, se presenta qué parte de la solución se implementó, cómo se definió su alcance en base a las necesidades del cliente AGESIC [52] y tiempo para el desarrollo. Se describe el escenario a validar, la instanciación de la solución propuesta en el capítulo anterior y su relación con el prototipo, mostrando una vista de diseño más detallado de los componentes involucrados. Se realiza un listado de las herramientas utilizadas y su correspondiente configuración, así como los casos de prueba realizados para validar el escenario planteado. Por último, se comentan los desafíos técnicos, problemáticas y dificultades encontradas.

5.1 Objetivo

El objetivo planteado que valida parte de la solución propuesta, contemplando las necesidades actuales del cliente y la utilización de herramientas *open source*, implica:

- Implantar un modelo XDS con varios repositorios
- Integrar un producto MPI/PIX.
- Compartir documentos, analizar su estructura y representación en la herramienta

5.2 Alcance

Consideraciones de la implementación

A la hora de acordar el prototipo con el cliente se plantearon dos escenarios de implementación:

Escenario 1: Implementación de un prototipo para validar la solución global

Escenario 2: Explotación de las capacidades de XDS+EMPI dentro de una única comunidad.

Se priorizó la implementación del XDS y por razones de tiempo se descartó el escenario 1, debido a la cantidad de componentes e interacciones.

Los criterios utilizados para tomar tal decisión fueron los siguientes:

- No era viable el desarrollo de un prototipo para el escenario 1 en los plazos del proyecto.
- Mayor interés por parte del cliente, en poder profundizar en el uso del XDS+EMPI.

Alcance escenario 2

En la Figura 37 se muestra resaltado en “L” los componentes de la propuesta planteada en el Capítulo 4, que se consideran para la etapa de prototipo. Los componentes son: un MPI/PIX, el registro XDS (*Registry*) y varios repositorios XDS (*repository*) federados en las organizaciones.

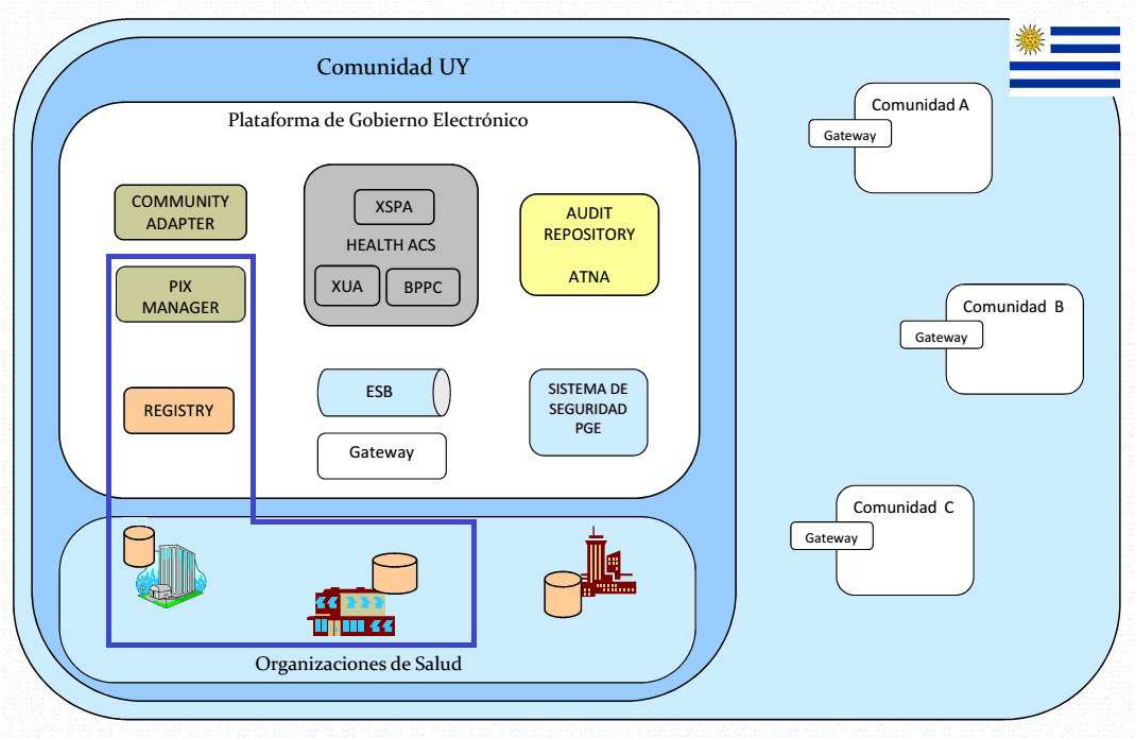


Figura 37: Definición del alcance basado en la solución propuesta

Teniendo en cuenta el escenario 2 se definieron, en conjunto con el cliente, los siguientes requisitos para el prototipo:

- Contar con un dominio de afinidad denominado “Salud UY” y dos dominios de identificación correspondiente a organizaciones.
- Manejar un registro XDS centralizado para compartir documentos en Salud UY.
- Disponer de repositorios federados en las organizaciones que comparten documentos en el dominio.
- Contar con un Índice maestro de pacientes (EMPI/PIX) para manejar los distintos identificadores de un paciente en las organizaciones que comparten información en el dominio Salud UY.
- Definir un componente “Proxy” que permita la comunicación entre Salud UY y las organizaciones a través de la PGE.
- Determinar qué información sería necesaria intercambiar, para soportar la comunicación tomando en cuenta el perfil XSPA.

5.3 Escenarios de Uso

En función de los objetivos y requisitos planteados anteriormente, se define el contexto donde participan varias organizaciones que comparten documentos a través de un modelo XDS en el dominio de afinidad denominado “Salud Uy”. Se cuenta con un registro de documentos (*Document Registry*) y un manejador de referencias cruzadas para los identificadores de pacientes (PIX) que se encuentran centralizados y gestionados bajo dicho dominio. Cada una de las organizaciones contiene los repositorios que almacenan los documentos (*Document Repository*) que se comparten a través del registro centralizado.

Existen varios dominios de identificación, uno por cada organización y otro correspondiente al dominio Salud UY. Este es el identificador bajo el cual se "agrupan" todos los documentos compartidos.

La comunicación entre las organizaciones y Salud UY es mediante *web services* a través de la PGE utilizando los mecanismos de seguridad que provee actualmente.

Escenario de Análisis:

Teniendo en cuenta la descripción anterior del contexto, a continuación se detalla un escenario de análisis donde se comparten documentos:

En una primera instancia, el paciente concurre a su médico tratante para consultar problemas de una alergia. Durante la atención médica se genera un documento con el resumen del acto clínico y otro correspondiente a un estudio realizado por el mismo médico, quien además le indica otros estudios más avanzados.

En segunda instancia, el paciente visita otra organización sanitaria para consultar un especialista y realizarse el estudio indicado. Durante la atención se generan nuevos documentos que son compartidos por la organización.

Finalmente el paciente visita al médico tratante, éste consulta la información de los estudios realizados, obteniendo información relacionada a los documentos (Autor, organización, fecha, tipo, etc.) compartidos en Salud UY, seleccionando el médico cuál recuperar para su visualización.

En cada una de estas atenciones el paciente fue identificado de forma diferente de acuerdo a los mecanismos de identificación utilizados por la organización donde se realiza el estudio. Los documentos generados y compartidos en cada una de las organizaciones son registrados en el registro central de Salud UY con el identificador del paciente de este dominio.

5.4 Implementación del escenario

En esta sección se detallan las herramientas utilizadas y la configuración necesaria para cumplir con cada uno de los componentes de la solución instanciada y poder implementar el prototipo que da soporte al escenario de uso planteado.

Con el fin de hacer la solución a prototipar más viable y dado que el interés está en el funcionamiento de la solución XDS+PIX, el prototipo se simplifica no considerando el flujo através de la PGE, el componente Proxy, ni la aplicación web consumidora de los servicios XDS-PIX. La simulación de los consumidores *XDS Consumer*, *PIX Consumer y PIX Source* se realiza directamente con herramientas para testeo que soportan la invocación a servicios HL7 y SOAP. Una discusión y comparación del prototipo original con la solución presentada en el Capítulo 4 pude verse en el Apéndice B.

En la Figura 38 se presenta el diagrama que refleja la solución probada, indicando las transacciones entre cada uno de los componentes y las herramientas utilizadas, mostrando con línea punteada los que no fueron representados en el prototipo.

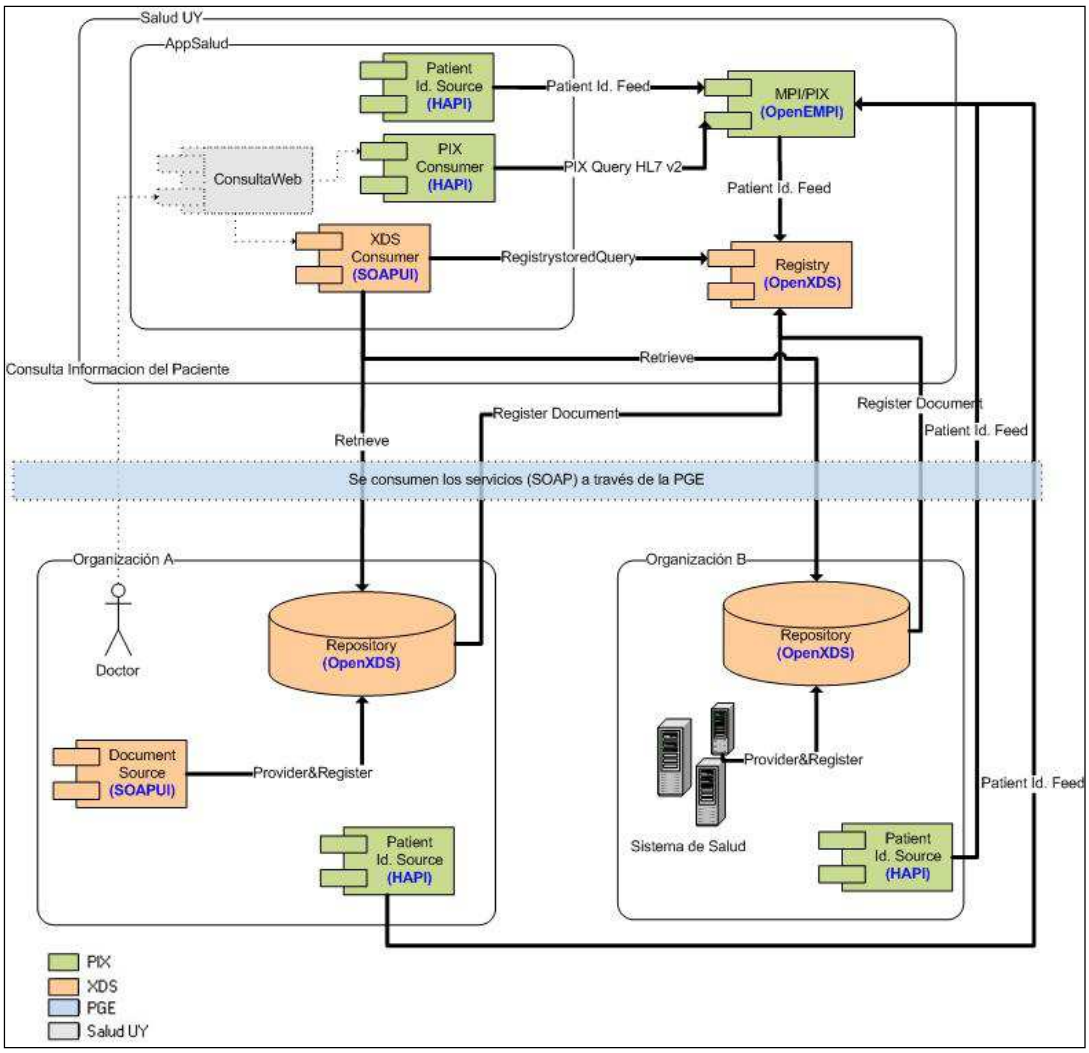


Figura 38: Solución probada en el prototipo

5.4.1 Descripción de Herramientas Utilizadas

OpenXDS 1.0.1 – Open Cross Enterprise Document Sharing [63]**:** proyecto de código abierto, que implementa los actores correspondientes al repositorio y registro del perfil XDS.b, así como también el *Gateway* de inicio y respuesta del perfil XCA. Utilizado para instalar dos repositorios y un registros XDS.

OpenEmpi 2.2.7 – Open Master Patient Index [64]**:** proyecto de código abierto, que implementa un Índice Maestro para identificación de pacientes en toda la empresa, manteniendo un repositorio con toda la información para la identificación de pacientes. Certificado en los perfiles de IHE PIX/PDQ para las implementaciones HL7 v2 y HL7 v3. Es utilizado para instalar el MPI/PIX.

SoapUI 4.6.4 [65]**:** herramienta con interfaz gráfica para pruebas funcionales SOA y de web services tanto SOAP como REST. Se utiliza como cliente para interactuar con el registro y repositorio XDS.

Hapi Testpanel 2.0.1 [66]**:** software libre para editar, transmitir, recibir y validar mensajes HL7. Utilizado como consumidor del OpenEmpi para enviar mensajes HL7 v2 al componente PIX.

JBOSS - 4.2.3.GA [67]**:** servidor de aplicaciones de código abierto, utilizado para el deploy de OpenEmpi.

Postgres 9.1 [68]**:** base de datos de código abierto, que da soporte al modelo de datos de los productos OpenEmpi y OpenXDS.

Para la Instalación del registro XDS en el dominio de "Salud UY" se instala el producto OpenXDS en su versión standalone. Se configura en el archivo IheActor.xml los actores *XdsRegistry, XdsRepository* (permite la opción de configurar el *Gateway* del perfil XCA). En los archivos de conexiones *XdsRegistryConnections.xml y XdsRepositoryConnections.xml* se configuran los *endpoint* de los servicios, que utilizan los actores definidos en IheActors.xml. En el archivo *XdsRegistryConnections.xml* se deberá especificar el dominio de identificación con el cual se registrarán los documentos. La instalación de OpenEMPI se realiza en un servidor de aplicaciones JBOSS y su configuración es similar a OpenXDS.

En el Apéndice 5 se encuentra de forma más detallada la configuración necesaria de los productos de salud, sin entrar en detalle en la instalación o configuración del servidor de aplicaciones o base de datos, ni tampoco en las herramientas de prueba utilizadas como consumidores, dado que no están directamente relacionadas con el objetivo del prototipo.

5.5 Casos de Prueba

A continuación se presenta una serie de casos para verificar el escenario en el prototipo implementado.

Caso 1	Registro del paciente en el dominio Salud UY
Objetivo	Dar de alta el paciente en el PIX
Descripción	El paciente es ingresado al sistema con CI
Resultado	Verificación de que el paciente fue registrado correctamente en el PIX

Caso 2	Registro del paciente en la organización A
Objetivo	Asociar un nuevo identificador del paciente en el PIX
Descripción	La organización A ingresa el paciente a su sistema, para poder realizar los estudios que su médico le solicita.
Resultado	El paciente se asoció correctamente con el identificador de Salud UY.

Caso 3	Registro de documentos.
Objetivo	Registrar documentos de una organización asociados a una carpeta.
Descripción	El paciente realiza una consulta al médico en la organización A. Durante el acto clínico se genera dos documentos A1 y A2 correspondientes a un resumen médico y un estudio sobre alergias, este último queda asociado a la carpeta de tratamientos de alergias.
Resultado	Los documentos A1 y A2 quedaron asociados en el registro al envío correspondiente y A2 quedó bajo la carpeta de Tratamiento de Alergias

Caso 4	Registro del paciente en un segunda organización B
Objetivo	Asociar el nuevo identificador del paciente en el PIX pero enviando solo los datos de identificación con un error menor.
Descripción	La organización B ingresa el paciente al sistema indicando solo los datos nombre y apellido, cambiando la última letra del nombre.
Resultado	El PIX identificó correctamente al paciente y lo vinculó a los identificadores de la organización A y Salud UY.

Caso 5	Registro de documentos en organización B.
Objetivo	Registro de documentos asociándolos a una carpeta existente en otra organización y una carpeta propia.
Descripción	El paciente visita una organización externa a la que pertenece para realiza estudios más avanzados sobre sus problemas de alergia. Como resultado del acto clínico, se generan dos documentos: A1 como informe del especialista y que es guardado bajo la carpeta Pacientes externos (PacExt), y B2 correspondiente al nuevo estudio guardado en la carpeta de tratamiento de alergias (TratAlerg)
Resultado	El B1 y B2 quedaron asociados al envío correspondiente y que el B1 quedó bajo la carpeta Paciente Externo (PacExt) y el B2 quedó asociado a la carpeta tratamiento de alergias (TratAlerg).

Caso 6	Consultar el identificación del paciente en SaludUY
Objetivo	Obtener el identificador del paciente en Salud Uy conociendo solo el identificador de la organización A.
Descripción	El médico tratante, en la organización A, quiere consultar los documentos del paciente que se comparten en Saluy UY. Solo se conoce el identificador del paciente en la Organización A de donde el paciente es socio.
Resultado	Se obtiene el identificador correspondiente al dominio Salud UY.

Caso 7	Consulta de documentos
Objetivo	Poder consultar la información de todos los documentos del paciente que se registraron en diferentes organizaciones utilizando el identificador de Salud UY.
Descripción	El paciente vuelve a consultar a su médico tratante. El médico consulta los estudios del paciente compartidos en Salud UY
Resultado	Verificar que la consulta traiga todos los metadatos correspondientes a los cuatro documentos compartidos

Caso 8	Recuperar un documento
Objetivo	Obtener un documento específico dado su Id
Descripción	El médico recupera para su visualización el último estudio realizado.
Resultado	Verificar la correcta recuperación del documento generado en B con el identificador B2.

En la Figura 39 se muestra el orden de ejecución y el flujo de mensajes entre los componentes prototipados utilizando las diferentes herramientas.

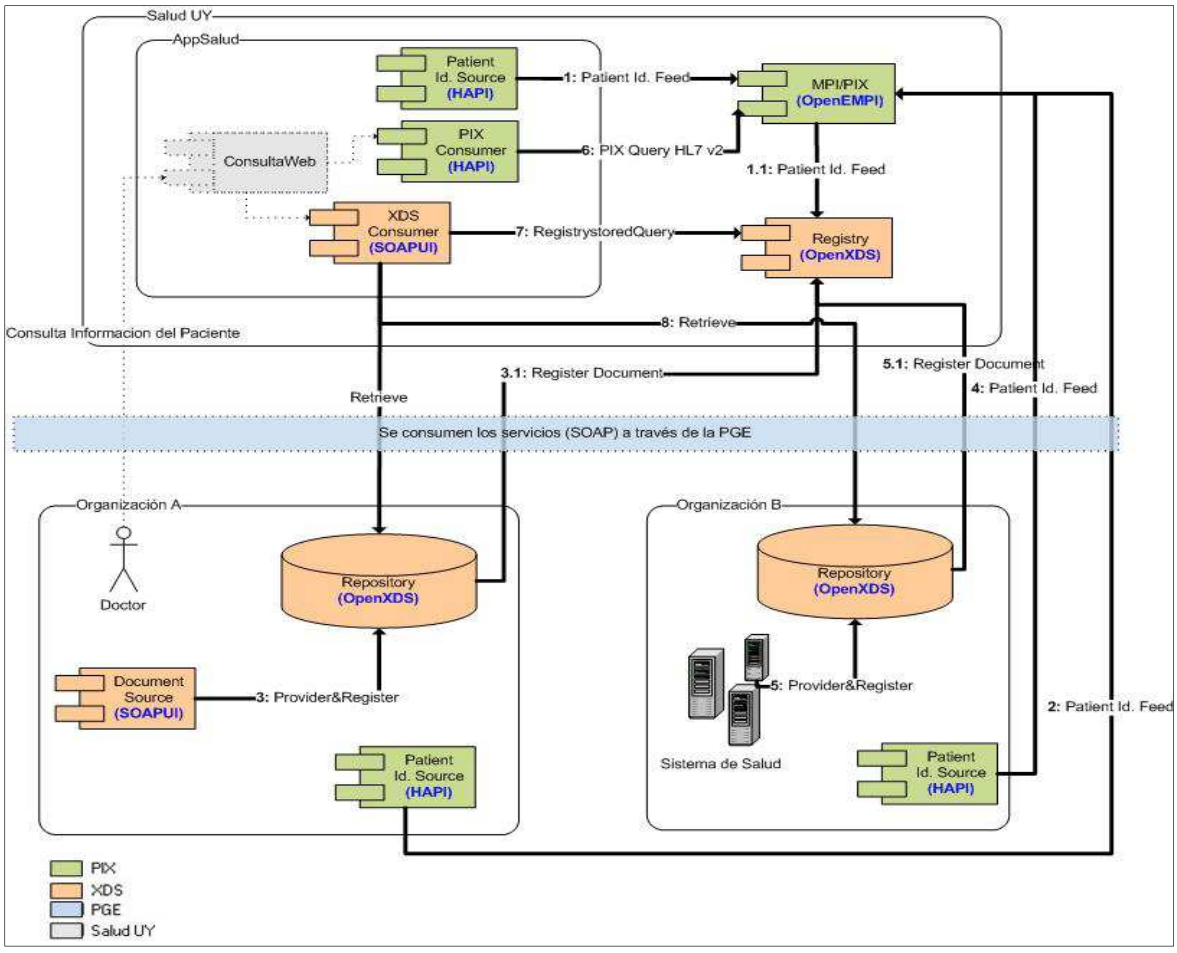


Figura 39: Flujo de ejecución de casos de prueba

5.6 Desafíos Técnicos

Presentado el escenario e instanciación de la solución, los desafíos planteados fueron:

- Instalar e integrar productos OpenEMPI y OpenXDS
- Configurar múltiples repositorios en un dominio XDS.
- Consumir los servicios y ejecución del escenario de prueba.
- Envío de documentos CDA a los repositorios.
- Uso de HL7 v2: para la comunicación con el *PIX Manager* y de librerías HL7 para implementación de un cliente.
- Analizar metadatos: provistos por OpenXDS en relación al perfil XDS, su representación en ebXML y como lo implementa OpenXDS

Instalar e integrar productos OpenEMPI y OpenXDS

La instalación de OpenEMPI presentó algunos problemas con la base de datos ya que había incompatibilidad entre el formato de fechas instalado en la base, y el manejado por OpenEmpi. Esto no permitía la ejecución de scripts de creación para armar el ambiente. La solución consiste en cambiar el atributo datestyle = ISO, MDY en el archivo postgresql.config.

Al momento de realizar el deploy de la aplicación en el servidor JBOSS, se presenta un problema por falta de espacio en memoria, que es solucionado al modificar las opciones de configuración del servidor para la variable JAVA_OPTS, como se muestra en la Figura 40:

```
set OPENEMPI_HOME=C:\servers\openempi-2.2.7
set JAVA_OPTS=%JAVA_OPTS% -Dprogram.name=%PROGNAME% -Dopenempi.home=%OPENEMPI_HOME%
set JAVA_OPTS=%JAVA_OPTS% -Xms128m -Xmx512m -XX:PermSize=312m -XX:MaxPermSize=312m
    -Dopenempi.home=%OPENEMPI_HOME%
```

Figura 40: Ejemplo de run.conf

En la integración entre OpenEMPI (actor *PIX Manager*) y OpenXDS (actor *XDS Registry*) se presentaron problemas referentes a la comunicación realizada desde el *PIX Manager* con el *XDS Registry*. Al momento de dar de alta un paciente en el PIX, éste no notifica al registro XDS, por lo cual a éste último, no le llega el identificador del paciente y esto provoca que luego, al enviar un documento relacionado a dicho paciente, la transacción falle. El problema se resuelve configurando el dominio global en archivo *PiXPdqClientesDomains.xml* igual al dominio de afinidad indicado en la propiedad *XDSAffinityDomainPatientIdetifier* que se encuentra en la definición de la conexión del Registro XDS del archivo de conexiones *PixConsumerConnections.xml* y que es el mismo que está definido en *XdsRegistryConnections.xml* (ver sección anterior) de OpenXDS.

Configurar múltiples Repositorios en un dominio XDS

Este punto no presentó dificultades, la configuración de un ambiente con dos repositorios consiste en la instalación de dos instancias de OpenXDS, configurando sólo los actores

correspondientes al repositorio *XdsRepository* y la conexión al registro *XdsRegistry* para el dominio Salud UY como se explicó en la sección anterior. Además se deberá indicar en el archivo de propiedades *openxds* la identificación del repositorio.

Envío de documentos CDA a los repositorios

El inconveniente que se presentó fue al enviar documentos utilizando el mecanismo MTOM/XOP que establece la IHE. Para ello se utilizó la herramienta SoapUI como cliente, con el rol de actor *Document Source* para el envío de documentos al repositorio XDS. En las pruebas realizadas se utilizó, tanto codificación en Base64 como MTOM/XOP. El problema consistía en que si bien el contenido del archivo se guardaba correctamente, al momento de recuperar el documento del repositorio XDS, este se devolvía "corrupto" (codificado en HEXA y con caracteres especiales al principio). La incompatibilidad de OpenXDS estaba dada por la versión de PostgreSQL utilizada y la representación de salida para el tipo bytea que utiliza en versiones posteriores a la 9. Es decir, el valor que se define en la variable *bytea_output* que es el que determina en que codificación es devuelto el contenido del campo de la Base de Datos donde se guarda el archivo, y que es del tipo bytea no era el mismo. La solución, es definir esta propiedad con el valor compatible para OpenXDS, o sea *bytea_output = escape*.

Consumir los servicios y ejecución del escenario de prueba

En este punto el desafío se presentó al consumir los servicios SOAP de OpenXDS, y con el envío de mensajes HL7 v2.x al PIX instalado con OpenEmpi. Como se mencionó en la sección anterior, como actor Document Consumer se utilizó la herramienta SoapUI, para poder enviar las solicitudes a los servicios de XDS por lo cual había que contar con los archivos wsdl para generar el proyecto y realizar el envío de mensajes. A partir de los WSDL de OpenXds no se podían generar los request por lo que se tuvo que utilizar los WSDL publicados por IHE para generar el proycto en el SOAPUI para que soportara el envío de mensajes a OpenXDS. El otro desafío estaba en utilizar la versión HL7 v2.x que planteó el cliente (en contraposición a HL7 v3 que usa WS y que se había estudiado) y es la que se quería validar ya que es la utilizada por la mayoría de los componentes definidos. Como actor PIX Consumer se utilizó la herramienta gráfica Hapi TestPanel. El desafío era entender los segmentos definidos en HL7v2.x y los códigos para mensajes de alta y consulta al PIX. Además un problema que se presento era el no enviar el mensaje en la versión correcta que esperaba el servicio, por ejemplo v 2.3.1 para alta de pacientes y 2.5 para consultas PIX Query. Luego de la puesta a punto de los consumidores XDS - PIX y resuelto el problema para el envío de documentos se ejecutaron las pruebas obteniendo los resultados esperados.

Uso de HL7 v2

Si bien para emular el actor PIX Consumer se utilizó la herramienta Hapi testPanel para realizar altas de pacientes y consultas al *PIX Manager*, se logró desarrollar un cliente java que recibe el identificador de un paciente en el dominio de "Salud UY" y devuelve la lista de identificadores en los demás dominios donde está identificado el paciente. Para el desarrollo del cliente se utilizó la librería HAPI v2.2 (API de java para mensajería HL7 2.x) de código abierto que permite el desarrollo de aplicaciones HL7, permitiendo el parseo, envío y recepción de mensajes HL7 2.x. La librería es utilizada en otros productos open source de salud como OpenXDS y OpenEMPI.

Analizar Metadatos

El cliente plantea la necesidad de comprender como el producto OpenXDS modela los metadatos del ebRIM exigidos por el perfil XDS. Para llevar adelante esto, lo primero que se realizó fue un relevamiento y estudio de la documentación disponible sobre el ebRIM para entender y comprender los objetos que intervienen y cómo se relacionan.

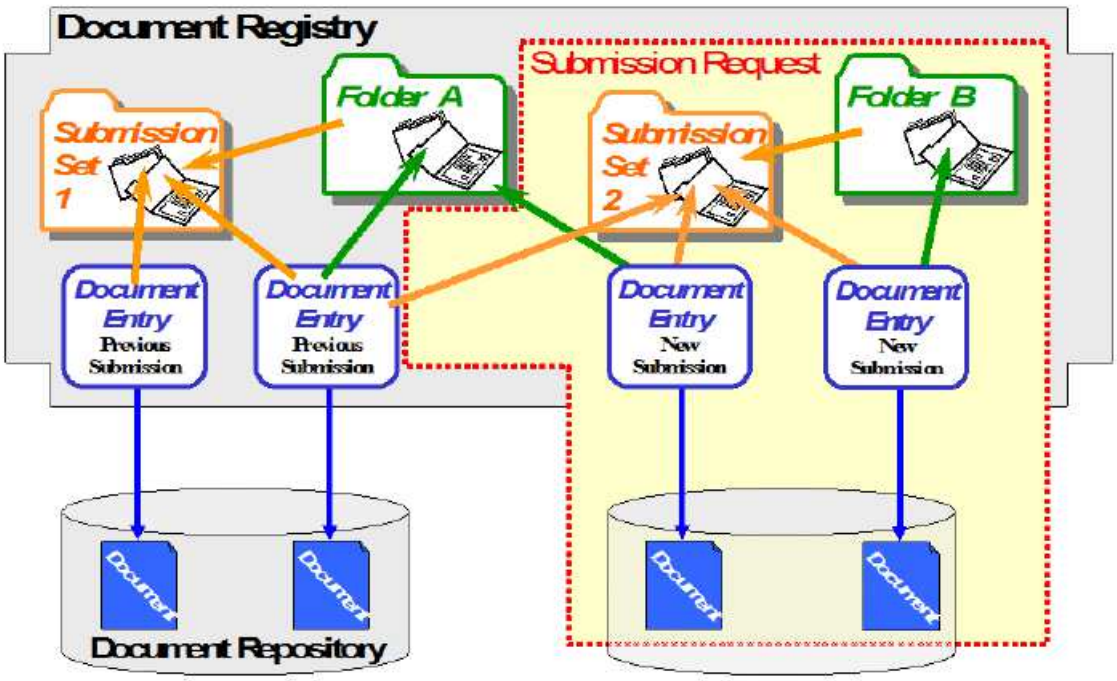
Como resultado de esta investigación se generó un documento MetadatosXDS que se adjunta como anexo a este documento.

Luego se pasó a instalar el OpenXDS y estudiar el modelo de datos que implementa, y de esta forma establecer la correspondencia entre los objetos de dicho modelo y los del ebRIM. Con el estudio de la representación de ebRIM y el modelado de datos utilizado por la herramienta, se pudo realizar diferentes consultas sobre datos enviados previamente al repositorio, que ayudaron a comprender la relación entre los distintos objetos contenidos en el envío de un documento al repositorio. Más detalle sobre esta representación se puede encontrar en el apéndice C1.

A continuación se plantea un caso a modo de ejemplo, donde se muestra la representación de los documentos y metadatos asociados.

La publicación de documentos en un repositorio XDS se da siempre a través del envío de un conjunto de datos, el cual puede contener uno o más documentos, que a su vez pueden estar agrupados.

En la Figura 41 se muestra un ejemplo de representación en el registro XDS con el envío de dos documentos y el uso de dos carpetas. Inicialmente el registro cuenta con la información de dos documentos, de los cuales uno se encuentra asociado a la carpeta A. El siguiente envío consta de dos nuevos documentos, uno es colocado en la carpeta A



existente y el otro en una nueva carpeta B.

Figura 41: Ejemplo de un envío a un registro XDS [47]

En resumen tenemos que los metadatos utilizados por los perfiles definidos para compartir documentos están caracterizados por tres tipos de objetos: Submission Set, Folder y Document Entry; y dos tipos de asociaciones: HasMember y Relationship, que tienen su representación en clases ebRIM y están implementadas en OpenXDS como se muestra en la Tabla 7.

Document Sharing Object/Association	ebRIM class
DocumentEntry	rim:ExtrinsicObject
SubmissionSet	rim:RegistryPackage
Folder	
MemberOf	rim:Association
Relationship	

Tabla 7: Representación Metadatos en ebRIM [47]

5.7 Evaluación de Herramientas

OpenEMPI/OpenPIXPDQ

OpenEmpi es un índice maestro de pacientes de código abierto, es decir gestiona la identidad del paciente. Para ello manteniene en un registro central con la información demográfica de todos los pacientes, elimina entradas duplicadas, mantiene el registro de los proveedores de salud que ha visitado cada paciente y es capaz de notificar eventos a otros interesados.

Como **características** principales, se tiene que está basado en una arquitectura orientada a servicios (SOA), con un conjunto de servicios débilmente acoplados. Cada servicio se expone a otro a través de una interfaz bien definida. Resuelve la coincidencia de registros de paciente por medio de algoritmos, presenta un algoritmo determinista y otro probabilista, también permite ser extendido por medio del desarrollo de un algoritmo propio.

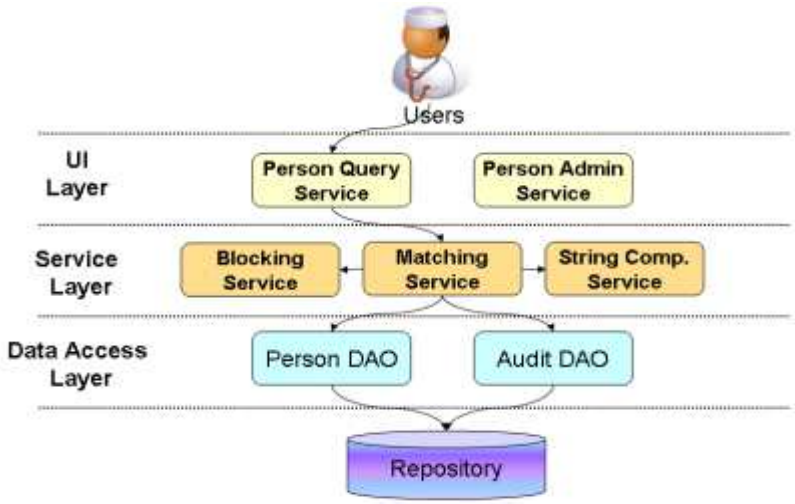


Figura 42: Arquitectura OpenEmpi basada en principios SOA y Capas [69]

La **integración** con OpenEmpi puede ser por medio de una API JAVA, Web Service REST, Interfaz IHE PIX/PDQ (que es la utilizada en este proyecto).

La **Instalación** es sencilla ya que cuenta con un módulo war el cual se despliega en un servidor de aplicaciones y con lo cual ya queda disponible para su uso. Cuenta con una consola administrativa web sencilla donde se permite agregar y realizar búsquedas de personas, realizar carga de registros desde archivos, arrancar y detener el servidor PIX/PDQ, etc.

OpenEmpi fue certificado en el Connectathon 2013 por el cumplimiento de perfiles IHE PIX/PDQ tanto para las implementaciones que utilizan HL7 v2 y como HL7 v3.

Al desplegar la aplicación war, esta cuenta con el componente OpenPIXPDQ (versión que se mantiene con OpenEmpi) que levanta un servidor PIX/PDQ que se conecta por medio de un *adapter* con el EMPI.

A continuación se muestra la arquitectura de alto nivel del OpenPIXPDQ y su conexión EMPI.

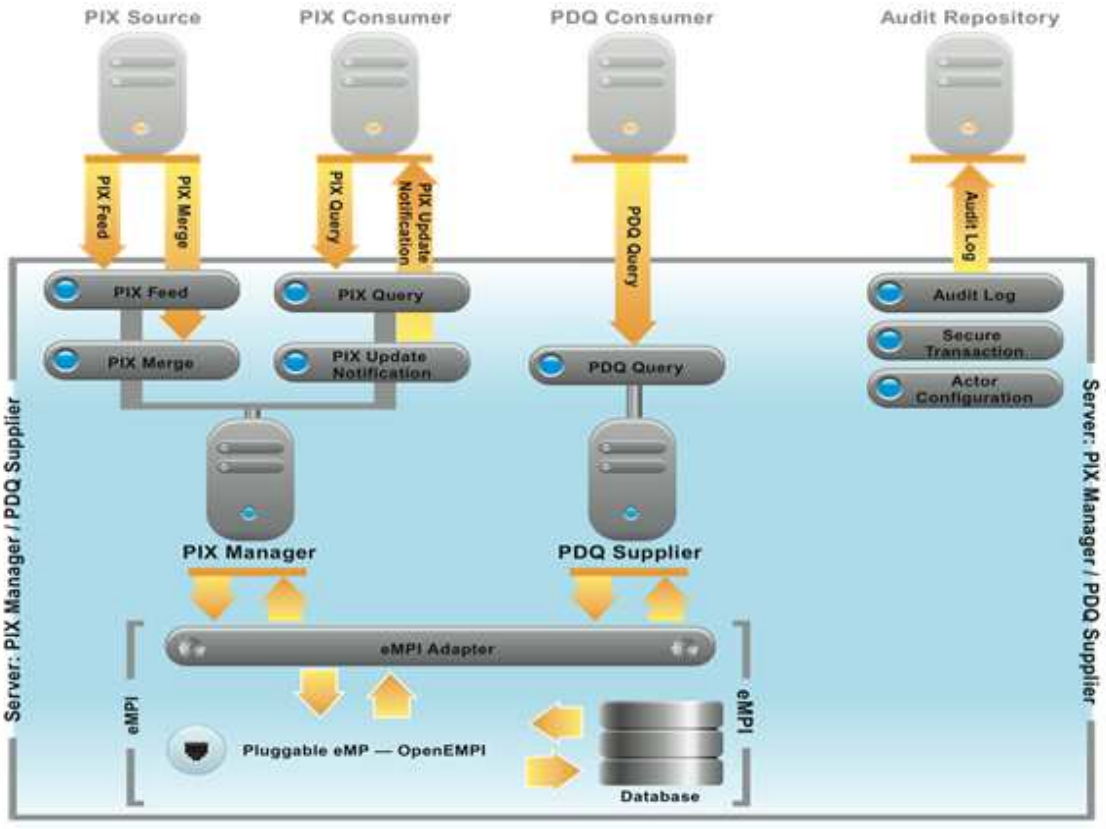


Figura 43: Arquitectura alto nivel OpenPIXPDQ [70]

La versión estable utilizada en este proyecto es la 2.2.7 (2013), pero actualmente existe una beta 3.0 a ser liberada a fines del 2014.

El proyecto cuenta tanto foros de desarrollo, como de usuarios, ambos activos y brinda la posibilidad de descargar el código fuente vía SVN.

La **documentación** referente a OpenEmpi es suficiente, pero en cuanto a la configuración para la integración mediante interfaz PIX/PDQ no presenta información. Por el contrario,

ésta debe consultarse en el sitio de OpenPIXPDQ, el cual no está lo suficientemente activo, o debe recurrirse a los foros de OpenEmpi.

OpenXDS

Es un proyecto de código abierto que implementa los actores de IHE correspondientes al Registro y Repositorio de documentos del perfil XDS.b y los actores para consulta y recuperación de documentos entre comunidades (initiating Gateway y Responding Gateway) del perfil XCA.

Algunas de las **características** principales del Registro y Repositorio, se pueden ver en la arquitectura de alto nivel presentada en las Figuras 44 y 45:

El Registro OpenXDS admite las siguientes transacciones:

- Un mensaje ADT desde una fuente de identificación de pacientes por medio de la transacción PIX Feed [ITI-8]
- Consultas para Control de Acceso al componente que ejecuta el Control de acceso (reservado para el futuro)
- El registro de documentos por parte de un repositorio por medio de la transacción Register Document Set –b [ITI-42]
- Consulta de documentos por medio de la transacción RegistryStoredQuery [ITI18]
- Envío de logs de auditoría por el registro a un repositorio de auditorías (Record Audit Event [ITI-20])

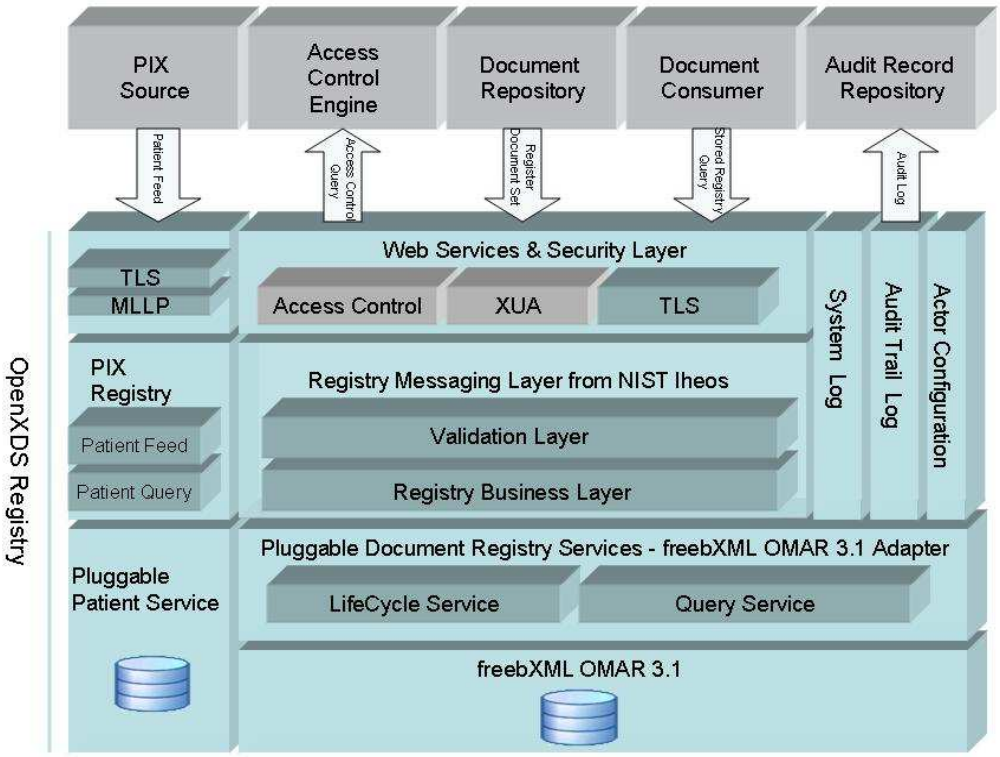


Figura 44: Arquitectura de alto nivel Registro OpenXDS [63]

El repositorio OpenXDS soporta las siguientes transacciones.

- Consultas para Control de Acceso al componente que ejecuta el Control de acceso (reservado para el futuro)
- Proveer y registrar documentos desde fuentes documentales (Provide&Register Document Set-b [ITI-41])
- Registrar Documento en el Registro (Register Document Set –b [ITI-42])
- Recuperar un documentos por un consumidor de documentos (Retrieve Document Set [ITI-43]
- Envío de log de auditoría por el registro a un repositorio de auditorías (Record Audit Event [ITI-20]

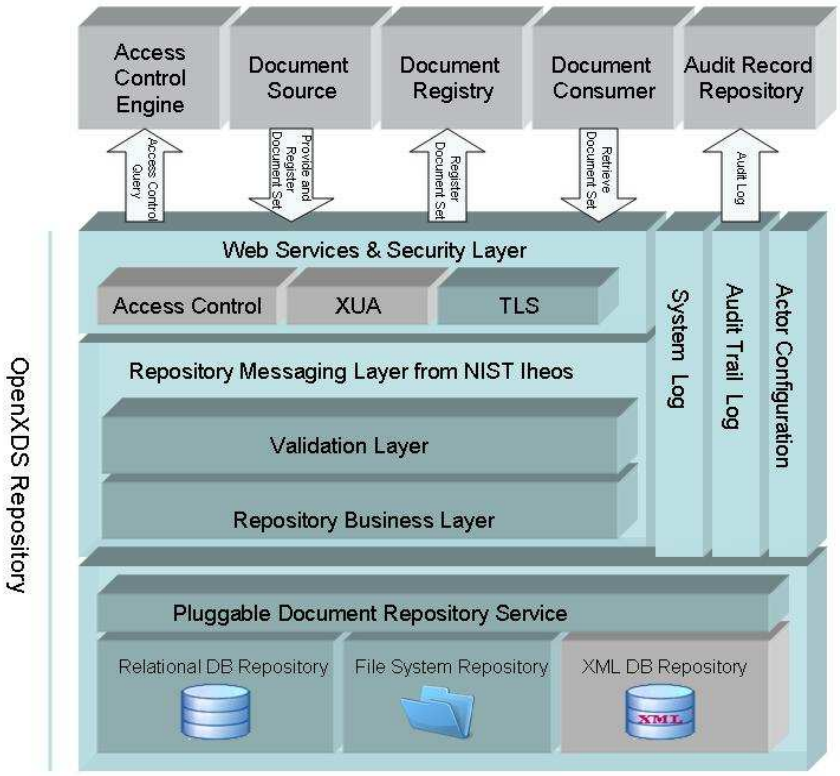


Figura 45: Arquitectura de alto nivel Registro OpenXDS [63]

Es fácilmente instalable, se puede instalar una versión standalone y su configuración es por medio de archivos XML, donde se configuran los actores y las conexiones con demás actores del perfil, pero no resulta trivial.

Se puede integrar varios repositorios al dominio de afinidad, instalando el producto y cambiando la identificación del repositorio y la conexión correspondiente con el registro central.

La documentación en general, es escasa y principalmente contiene información sobre la instalación. No se encuentra documentación sobre funcionalidades de los servicios, para ello tiene como referencia a IHE.

No se pudo generar directamente solicitudes con la obtención directa del archivo de descripción de los servicios (WSDL).

La versión estable utilizada en este proyecto es la 1.0.1 (09/2010), no se encontró información sobre liberación de nuevas versiones.

El proyecto cuenta tanto foros de desarrollo, como de usuarios, ambos con poca actividad. Se brinda la posibilidad de descargar el código fuente vía SVN.

En resumen, Open XDS presenta las herramientas básicas para la instalación del modelo XDS, pudiéndose descargar el código fuente y desarrollar nuevas librerías (jar) basadas en esta herramienta y así facilitar la integración de aplicaciones que van a compartir documentos en un dominio de afinidad XDS.

6 Conclusiones y Trabajo Futuro

En este capítulo se presentan las conclusiones de todo el trabajo realizado, comenzando por un resumen de los conceptos más importantes y relevantes, dando una visión global del contexto del proyecto. Luego, se sigue con una descripción de la solución propuesta y las principales características de diseño. Por último, se plantean las contribuciones, desafíos y trabajos a futuro, considerados en el marco del proyecto.

6.1 Conclusiones

El objetivo principal de este proyecto era proponer mecanismos para integrar e interoperar servicios de salud entre diferentes organizaciones, a través de la PGE y con el menor impacto posible para las organizaciones. Para ello se requirió definir los lineamientos generales del diseño de una plataforma integrada de salud. La mayor parte del proyecto se destinó al estudio de estándares y buenas prácticas en salud, revisión de artículos y de trabajos relacionados a este contexto. Se estudió también el funcionamiento de la plataforma de interoperabilidad de la PGE y en particular sus componentes de seguridad. Se consideraron los perfiles asociados al dominio de infraestructura de IHE, los perfiles de seguridad en salud y de Web Services de OASIS, HL7 v3 y HL7 CDA R2. Se realizó un análisis detallado de los perfiles de IHE y OASIS ya que son los que presentan soluciones a problemas de interoperabilidad conocidos en el ámbito de la salud. Con respecto a los estándares de HL7, se estudió la estructura y representación de los mensajes HL7 versión 2 y versión 3 utilizados por el perfil PIX.

Para definir los mecanismos de privacidad sobre la información clínica, se tomaron en cuenta las leyes de Protección de Datos Personales (PDP), analizadas en trabajos relacionados. [58]

Diseño de la solución y principales características

La solución propuesta define los lineamientos generales para el diseño de una plataforma de interoperabilidad en salud que se encuentra integrada a la PGE. Es un híbrido entre las dos alternativas de solución analizadas. Cuenta con una comunidad a nivel país, denominada Comunidad UY, que está definida como un dominio de afinidad XDS cuyo registro se encuentra centralizado en la PGE, y permite compartir documentos que se encuentran distribuidos en repositorios que están federados en las organizaciones. De esta manera el Estado, dentro de la comunidad UY, puede garantizar la privacidad de la información relacionada al cuidado de la salud de todas las personas que se atienden o atendieron en las organizaciones pertenecientes a dicha comunidad. La solución también permite la creación de una Comunidad, distinta a la Comunidad UY, para integrarse a la Plataforma de Salud. Esto es, que la creación de una Comunidad queda limitada a casos en que eventualmente exista un conjunto de organizaciones que tienen definida una forma de trabajo en común, es decir que funcionan conceptualmente como una comunidad. Cada comunidad que se forme, será responsable de realizar el control de acceso a la información, garantizando la privacidad de la misma, utilizando un mismo conjunto de políticas de acceso definidas a nivel de Estado y también aplicadas por la Comunidad UY.

La solución está basada en un conjunto de perfiles de la IHE y OASIS particulares de salud y en estándares HL7 para definir los lineamientos para una plataforma de salud. Estos estándares y la solución propuesta son compatibles con la PGE. En particular se utilizaron WS-Trust, SAML, XACML, XSPA-SAML, XSPA-WS-Trust, XSPA-XACML y perfiles de la IHE.

Contribuciones

Las Principales contribuciones de este proyecto son:

- El análisis realizado de los distintos estándares y perfiles de salud e interoperabilidad de sistemas, junto con el estudio y verificación de compatibilidad de éstos con la plataforma de Interoperabilidad de la PGE.
- La definición de los lineamientos del diseño de una solución, que se presenta integrada con la PGE, basada en perfiles y centrada principalmente en compartir (XDS, XCA) documentos que son la base para la historia clínica electrónica.
- Una solución a la identificación cruzada de pacientes (PIX), que permite el registro de los documentos compartidos sin tener que cambiar la identificación propia de cada organización.
- El análisis de la aplicabilidad de los perfiles de seguridad respecto a las problemáticas de autenticación, autorización y auditoría (XUA, XSPA, ATNA).
- La definición de un mecanismo para realizar el control de acceso en salud (XSPA, BPPC, *token* extendido para salud).
- Una solución que es flexible respecto a la integración con otras agrupaciones o "comunidades" con sistemas de salud en funcionamiento (XCA), para permitir el intercambio de información con un dominio de afinidad global apoyado sobre la PGE, con un mayor nivel de interoperabilidad e integración.
- Una solución basada en perfiles IHE que permite, una vez que se entiende su funcionamiento y se logra cierta experiencia y conocimiento, asegura escalabilidad, menores esfuerzos y costos para las integraciones.
- Un prototipo que integra dos perfiles de IHE (XDS, PIX) centrales en este proyecto, con la utilización de herramientas Open Source, que permitió realizar distintas pruebas de concepto y viabilizar su uso en un contexto más reducido.
- La formación de estudiantes en temas actuales para el país y el mundo, como la Plataforma de Gobierno Electrónico del Estado y las TI correspondientes al área de salud.

Desafíos y Problemas técnicos encontrados

Un gran porcentaje del tiempo invertido en este proyecto fue dedicado a la etapa de análisis. Esto se debe en gran medida a que el estudio y análisis de los estándares y buenas prácticas, implicó enfrentarse a una documentación densa y difícil de entender y asimilar. Si bien los marcos técnicos de IHE detallan los perfiles, describiendo sus Actores y transacciones basados en estándares, muchas veces resulta difícil poder lograr una idea cerrada de la solución que se plantea. En ocasiones quedan algunos puntos abiertos (dando flexibilidad) o referencia a otros perfiles que es necesario conocer para su entendimiento. Durante el transcurso del proyecto se fueron madurando los diferentes conceptos y perfiles utilizados, así como se logró entender los marcos técnicos de IHE. En muchos casos se tuvieron que rever sobre la marcha, diferentes conceptos y propuestas en las que se venían trabajando. En general no se encontró información detallada de proyectos o incluso ejemplo de los perfiles estudiados. La principal validación de que los perfiles son viables en nuestro contexto fue el considerar analíticamente al detalle sus transacciones y los estándares referenciados.

Los problemas técnicos con herramientas son los presentados en el capítulo 5. No se presentaron grandes complicaciones que no se resolvieran con investigación o búsquedas en foros. A pesar de ello no todos se resolvieron de forma rápida debido a no estar clara la documentación o actividad del foro. De todas formas, dentro de las funcionalidades probadas, no quedaron problemas sin resolver.

6.2 Trabajos a futuro

A lo largo del análisis se detectaron varias problemáticas, que dado el alcance definido para el proyecto, no fueron abordadas y se proponen como líneas de trabajo futuro a seguir en el contexto de este proyecto, las cuales se describen a continuación.

Impactos intracomunidad

Identificar, analizar y resolver los impactos que provocaría en una comunidad que no implementa un dominio de afinidad XDS, el compartir información a través de un *Gateway*.

Implementación del Control de acceso

Un aspecto importante para compartir documentos en salud, es el control de acceso teniendo en cuenta los consentimientos dados por el paciente. Este proyecto presenta una solución viable para su realización, que dada su complejidad, la indisponibilidad de la plataforma para realizar pruebas, el alcance del proyecto y el tiempo acotado, no se llegaron a implementar. La definición de un conjunto de políticas reales utilizando XACML y la implementación de este componente sería un aporte importante para continuar en el camino de este proyecto.

Compartir información con el exterior.

Otro aspecto para desarrollar a futuro, es analizar qué impactos tendría la solución si se requiere compartir información con comunidades extranjeras, teniendo en cuenta que el alcance de esta solución es a nivel de país, que la localización de pacientes y el acceso a documentos compartidos se resuelve utilizando perfiles de IHE como XDS, XCA y PIX. Esto facilita la comunicación con el exterior ya que estos perfiles dan soporte a un mecanismo para compartir información (XCA) y para el descubrimiento de pacientes (XCPD) entre comunidades.

Migración de la información

Una problemática no atacada y posible línea de extensión de este trabajo, es cómo migrar información de sistemas legados o que no se encuentra digitalizada. IHE define algunas soluciones a estos problemas, por ejemplo a través del perfil XDS-SD que permite pasar de un documento en papel a un documento XDS utilizando el estándar HL7 CDA R2 con el contenido escaneado. Así como también abordar transformaciones de la información a documentos XDS mediante el uso de un middleware. También sería interesante analizar cómo adherir a la solución el perfil RID (Retrieve Information for Display) de IHE, para que la información intercambiada pueda ser visualizada por los usuarios finales de los sistemas que se encuentran en las organizaciones.

7 Glosario

A

AGESIC – Agencia para el desarrollo de la Gestión de Gobierno Electrónico y la sociedad de la información.

ANSI – American National Standars Institute

ATNA – Audit Trail and Node Authentication

B

BPPC – Basic Patient Privacy Consents integration profile

C

CDA – Clinical Document Arquitecture

D

DICOM – Digital Imaging and Communication in Medicine

H

HL7 – Health Level Seven

I

IHE – Integrating the Healthcare Enterprise

L

LOINC – Logical Observation Identifiers Names and Codes

O

OASIS – Organization for the Advancement of Structured Information Standards

OID – Object Identifier

P

PDI – Plataforma de Interoperabilidad

PIX – Patient Identifier Cross-Referencing

PGE – Plataforma de Gobierno Electrónico

R

RID – Retrieve Information for Display

RIM – Reference Information Model

S

SAML – Security Assertion Markup Language for Healthcare

SUEIIDISS – Sociedad Uruguaya de Estandarización, Integración e Intercambio de Datos Información de Servicios de Salud

U

UNAOID – Unidad Nacional de Asignación de Identificadores de Objetos

X

XACML – eXtensible Access Control Markup Language
XCA – Cross Community Access
XDS – Cross Enterprise Document Sharing
XSPA – Cross Enterprise Security and Privacy Authorization
XUA – Cross Enterprise User Assertion
XML – Extensible Markup Language

W

W3C –World Wide Web Consortium

8 Referencias

[1] D. Todorov, Mechanics of User Identification and Authentication: Fundamentals of Identity Management, Auerbach Publications, 2007.

[2] «The Need for Authentication and Authorization,» [En línea]. Available: http://www.redbooks.ibm.com/Redbooks.nsf/RedbookAbstracts/tips0266.html?Open. [Último acceso: Junio 2014].

[3] «Glossary of Key Information Security Terms, Revisión 2,» [En línea]. Available: http://nvlpubs.nist.gov/nistpubs/ir/2013/NIST.IR.7298r2.pdf. [Último acceso: Junio 2014].

[4] «Fundamentos de Seguridad Informáticas, Transparencias del curso 2014,» [En línea]. Available: https://eva.fing.edu.uy/pluginfile.php/56857/mod_resource/content/2/Introduccion%20a%20la%20criptografia.pdf. [Último acceso: Junio 2014].

[5] «Understanding XML Digital Signature,» [En línea]. Available: http://msdn.microsoft.com/en-us/library/ms996502.aspx. [Último acceso: Junio 2014].

[6] «XML Signature Syntax and Processing,» [En línea]. Available: http://www.w3.org/TR/xmldsig-core. [Último acceso: Junio 2014].

[7] «XML Encryption Syntax and Processing,» [En línea]. Available: http://www.w3.org/TR/xmlenc-core/. [Último acceso: Junio 2014].

[8] «Signature Encryption,» [En línea]. Available: http://w3c.es/Divulgacion/GuiasBreves/Seguridad.

[9] M. P. Papazoglou, Web Services Technologies and Standards, 2007.

[10] L. González, «Plataforma ESB Adaptativa para Sistemas Basados en Servicios, Tesis de Maestría en Informática,» 2011.

[11] N. M. Josuttis, SOA in Practice, The Art of Distributed System Design, O'REILLY, 2007, 2007.

[12] «SOAP,» [En línea]. Available: http://www.w3.org/TR/soap. [Último acceso: Junio 2014].

[13] «W3C - XOP,» [En línea]. Available: http://www.w3.org/TR/soap12-mtom/#XOP. [Último acceso: Junio 2014].

[14] «W3C - MTOM,» [En línea]. Available: http://www.w3.org/TR/soap12-mtom/. [Último acceso: Junio 2014].

[15] «IBM – Knowledge Center (Mecanismo de optimización de transmisión de mensajes),» Junio 2014. [En línea]. Available: http://www-01.ibm.com/support/knowledgecenter/SS7K4U_7.0.0/com.ibm.websphere.zseries.doc/info/zseries/ae/cwbs_soapmtom.html?lang=es.

[16] «MTOM en ambiente heterogéneo - Facultad de Ingeniería FING,» 09 2014. [En línea]. Available: www.fing.edu.uy/inco/grupos/lins.

[17] «Encrypting SOAP Messages Using Web Services Enhancements,» [En línea]. Available: http://msdn.microsoft.com/en-us/library/ms996945.aspx#wseencryption_topic3. [Último acceso: Junio 2014].

[18] «Web Services Security: SOAP Message Security Version 1.1.1,» [En línea]. Available: http://docs.oasis-open.org/wss-m/wss/v1.1.1/os/wss-SOAPMessageSecurity-v1.1.1-os.pdf. [Último acceso: Junio 2014].

[19] «Understanding WS-Security,» [En línea]. Available: http://msdn.microsoft.com/en-us/library/ms977327.aspx. [Último acceso: Junio 2014].

[20] «Usar WS-Trust para la transformación token,» [En línea]. Available: http://www.ibm.com/developerworks/ssa/websphere/library/techarticles/1003_chades/1003_chades.html. [Último acceso: Junio 2014].

[21] «Federación, Definición de la seguridad federada,» [En línea]. Available: http://msdn.microsoft.com/es-es/library/vstudio/ms730908%28v=vs.100%29.aspx. [Último acceso: Junio 2014].

[22] «WS-Trust 1.4,» [En línea]. Available: http://docs.oasis-open.org/ws-sx/ws-trust/v1.4/errata01/os/ws-trust-1.4-errata01-os-complete.html. [Último acceso: Junio 2014].

[23] «Understanding WS-Federation,» [En línea]. Available: http://msdn.microsoft.com/en-us/library/bb498017.aspx. [Último acceso: Junio 2014].

[24] F. d. I. -. U. d. l. República, *Introducción al Middleware - Presentación Web Services Avanzados,* 2013.

[25] «SAML,» [En línea]. Available: http://docs.oasis-open.org/security/saml/v2.0/saml-2.0-os.zip. [Último acceso: Junio 2014].

[26] «OASIS,» [En línea]. Available: https://www.oasis-open.org/committees/download.php/2713/Brief_Introduction_to_XACML.html. [Último acceso: Junio 2014].

[27] «eXtensible Access Control Markup Language version 3.0,» [En línea]. Available: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf. [Último acceso: Junio 2014].

[28] M. P. P. Dimitrios Georgakopoulos, Service-Oriented Computing, The MIT Press, Cambridge, Massachusetts, London, England.

[29] «Descripción y guías de uso de la Plataforma de Gobierno Electrónico del Estado Uruguayo,» [En línea]. Available: http://www.agesic.gub.uy/innovaportal/file/1295/1/descripcion_y_guias_de_uso_de_la_plataforma_de_gobierno_electronico_del_estado_uruguayo.pdf. [Último acceso: Junio 2014].

[30] I. o. E. a. E. E. IEEE, IEEE Standard Computer, New York, NY, 1990.

[31] «HIMSS Healthcare Information and Management Systems Society,» [En línea]. Available: http://www.himss.org/library/interoperability-standards/what-is/. [Último acceso: Junio 2014].

[32] «American National Satandards Institute,» [En línea]. Available: http://ansi.org/about_ansi/overview/overview.aspx?menuid=1. [Último acceso: Julio 2014].

[33] «HL7 Internacional,» [En línea]. Available: http://www.hl7.org/implement/standards/rim.cfm accedida en Junio 2014. [Último acceso: Julio 2014].

[34] «HL7 España,» [En línea]. Available: http://www.hl7spain.org/documents/comTec/cda/GuiaElementosMinimosCDA.pdf. [Último acceso: Julio 2014].

[35] «Logical Observation Identifiers Names and Codes,» [En línea]. Available: http://loinc.org/. [Último acceso: Agosto 2014].

[36] «International Health Terminology Standards Development Organization,» [En línea]. Available: http://www.ihtsdo.org/snomed-ct/. [Último acceso: Agosto 2014].

[37] «IHE - Integrating the Healthcare Entreprise,» [En línea]. Available: http://www.ihe.net/. [Último acceso: Junio 2014].

[38] «IHE – España,» [En línea]. Available: http://www.ihe-e.org/. [Último acceso: Junio 2014].

[39] «IHE IT Infrastructure Technical Framework – Vol. 1,» [En línea]. Available: http://www.ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_TF_Vol1.pdf. [Último acceso: Junio 2014].

[40] «IHE IT Infrastructure Technical Framework – Vol. 2a,» [En línea]. Available: http://www.ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_TF_Vol2a.pdf. [Último acceso: Junio 2014].

[41] «IHE IT Infrastructure Technical Framework – Vol. 2b,» [En línea]. Available: http://www.ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_TF_Vol2b.pdf. [Último acceso: Junio 2014].

[42] «IHE Domains,» [En línea]. Available: http://www.ihe.net/IHE_Domains. [Último acceso: Junio 2014].

[43] «IHE – Technical Frameworks,» [En línea]. Available: http://www.ihe.net/Technical_Frameworks/#IT. [Último acceso: Junio 2014].

[44] «ebXML - Electronic Business using extensible Markup Language,» [En línea]. Available: http://docs.oasisopen.org/regrep/v3.0/regrep-3.0-os.zip. [Último acceso: Julio 2014].

[45] «Perfil XCPD - Cross-Community Patient Discovery,» [En línea]. Available: http://www.ihe.net/Technical_Framework/upload/IHE_ITI_Suppl_XCPD.pdf. [Último acceso: Julio 2014].

[46] «Current Published ITI Educational Materials - Security and Privacy Overview (IHE-Security_Privacy_Part1_2012-12-30.pptx),» [En línea]. Available: http://wiki.ihe.net/index.php?title=Current_Published_ITI_Educational_Materials. [Último acceso: Junio 2014].

[47] «IHE IT Infrastructure Technical Framework – Vol. 3,» [En línea]. Available: http://www.ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_TF_Vol3.pdf. [Último acceso: Junio 2014].

[48] «Sitio oficial OASIS,» [En línea]. Available: https://www.oasis-open.org. [Último acceso: Julio 2014].

[49] «Standard XSPA - SAML,» [En línea]. Available: http://docs.oasis-open.org/security/xspa/v1.0/saml-xspa-1.0-os.html. [Último acceso: Julio 2014].

[50] «Standard XSPA - XACML,» [En línea]. Available: http://docs.oasis-open.org/xacml/xspa/v1.0/xacml-xspa-1.0-os.html. [Último acceso: Julio 2014].

[51] «Standard XSPA - WS Trust,» [En línea]. Available: http://docs.oasis-open.org/xspa/ws-trust-v1.0/xspa-ws-trust-profile-os.html. [Último acceso: Julio 2014].

[52] «AGESIC,» [En línea]. Available: http://www.agesic.gub.uy/. [Último acceso: Agosto 2014].

[53] «Unidad Nacional de Asignación de OID,» [En línea]. Available: http://unaoid.gub.uy/. [Último acceso: Julio 2014].

[54] «Plataforma de Gobierno Electrónico,» [En línea]. Available: http://www.agesic.gub.uy/innovaportal/v/452/1/agesic/plataforma_de_gobierno_electronico.html. [Último acceso: Julio 2014].

[55] «Red UY,» [En línea]. Available: http://www.agesic.gub.uy/innovaportal/v/759/1/agesic/REDuy.html. [Último acceso: Julio 2014].

[56] «Aproximación a la historia clínica federada en SACYL: Modelo XDS,» [En línea]. Available: http://aiiscyl.files.wordpress.com/2012/02/2012_xds_sacyl_2012_oti-nataliaperez.pdf. [Último acceso: Junio 2014].

[57] «Los estándares en Sistemas de Información Sanitarios en Europa: la visión de IHE,» [En línea]. Available: http://www.seis.es/documentos/noticias/adjunto/IHE_IS_68_ABRIL08.pdf. [Último acceso: Julio 2014].

[58] «Protección de los Datos Personales de la historia clínica en Argentina y Uruguay e IHE XDS,» [En línea]. Available: http://www.jhi-sbis.saude.ws/ojs-jhi/index.php/jhi-sbis/article/download/166/80. [Último acceso: Julio 2014].

[59] M. Yelen, «Control de Acceso a la Historia Clínica Electrónica - Proyecto de grado 2011 - Fing-UdelaR.,» 2013.

[60] «The Syslog Protocol. (RFC 5424),» [En línea]. Available: http://tools.ietf.org/html/rfc5424. [Último acceso: Julio 2014].

[61] «Transmission of Syslog Messages over TLS (RFC 5425),» [En línea]. Available: https://www.ietf.org/rfc/rfc6587.txt. [Último acceso: Julio 2014].

[62] «Security Audit and Access Accountability Message XML Data Definitions for Healthcare Applications (RFC 3881),» [En línea]. Available: http://tools.ietf.org/html/rfc3881. [Último acceso: Julio 2014].

[63] «Proyecto OpenXDS,» [En línea]. Available: https://www.projects.openhealthtools.org/sf/go/page1046/. [Último acceso: Julio 2014].

[64] «Sitio Oficial OpenEMPI,» [En línea]. Available: http://www.openempi.org/. [Último acceso: Julio 2014].

[65] «Sitio Oficial SoapUI,» [En línea]. Available: http://www.soapui.org/. [Último acceso: Julio 2014].

[66] «Sitio Oficial proyecto Hapi,» [En línea]. Available: http://hl7api.sourceforge.net/. [Último acceso: Julio 2014].

[67] «JBOSS 4.2.3.GA,» [En línea]. Available: http://sourceforge.net/projects/jboss/files/JBoss/JBoss-4.2.3.GA/. [Último acceso: Julio 2014].

[68] «Sitio oficial PostgreSQL,» [En línea]. Available: http://www.postgresql.org/. [Último acceso: Julio 2014].

[69] «OpenEmpi,» [En línea]. Available: https://openempi.kenai.com/OpenEMPIandRecordLinkage.pdf. [Último acceso: Julio 2014].

[70] «OpenExchange - OpenPIXPDQ,» [En línea]. Available: https://www.projects.openhealthtools.org/sf/go/page1043. [Último acceso: Julio 2014].

9 Apéndice A - Perfiles IHE

9.1 A1: PIX Actores y Transacciones

PIX Manager (PIXM)

Es el encargado de mantener las referencias cruzadas entre todos los identificadores de los dominios de identidad involucrados, y comunicar a los interesados de cambios en la identificación de un paciente.

Patient Indentifier Source (PIS)

Es la fuente de datos de identificación responsable de alimentar el *PIX Manager* con los identificadores asignados en el dominio al cual pertenece, incluye la comunicación de eventos relacionado con el alta, actualización y fusiones de pacientes.

PIX Consumer (PIXC)

Se comunica con el *PIX Manager* para obtener la lista de identificadores para un Id. dado. Es notificado por este mismo de cambios en la identificación de pacientes en caso que se haya solicitado.

Transacciones

Patient Identity Feed Hl7 v3 [ITI-44]

La transacción corresponde a la notificación y/o comunicación de información (altas y modificaciones) sobre la identificación de un paciente, es enviada desde el actor PIS hacia el PIXM incluyendo información demográfica que identifica al paciente. Esta información permite al PIXM luego de establecida la identidad del paciente, modificarla o fusionarla.

PIX Query HL7v3 [ITI-45]

Es usada por el actor *PIX Consumer* para obtener una lista de identificadores del paciente, referenciados en el *PIX Manager* y asociado al identificador de paciente que conoce el consumidor.

PIX Update Notification [ITI-46]

Esta transacción involucra al actor PIXM quien es el responsable de proveer notificaciones sobre actualizaciones de la información de identificación de pacientes, a los PIXC que tenga registrados, como interesados en recibir notificaciones.

9.2 A2: Actores y transacciones XDS.b

Actores

Document Source

El actor Fuente de Documentos (Document Source) es la fuente generadora de documentos encargada de enviar/publicar el documento con los metadatos

correspondientes al repositorio (Document *Repository*) donde será almacenado el documento.

Document Repository

El actor Repositorio de Documentos (Document *Repository*) es el responsable de almacenar documento y enviar a registrar los atributos de este (metadatos) en el registro (Document Resgistry), asignándole un identificador único a cada documento para que luego pueda se recuperado.

Documento Registry

El registro de documento, almacena los atributos que identifican al documento enviados por el repositorio. Es responsable de atender las consultas de los consumidores de documentos, devolviendo la información de identificación (metadatos) que coincida con los registros y que permitirá posteriormente la recuperación del documento desde el repositorio.

Patient Identity Source

El actor Fuente de identidad de pacientes, es responsable de comunicar o notificar al registro cambios sobre la identificación de sus pacientes. Manteniendo así la integridad de los documento registrados.

Document Consumer

Es el actor que realiza consultas al registro, muestra el listado de documentos disponibles, recupera los documentos elegidos por usuario y utiliza los documento por ejemplo para visualizarlos en pantalla.

Transacciones

Provide & Register Document Set-b [ITI-41]

Esta transacción describe la interacción entre una fuente de documentos Document Source responsable de enviar los documentos y los metadatos al Repositorio; mientras el repositorio Document *Repository* almacena el documento y agrega información relevante a los metadatos para enviar estos al registro.

Register Document Set-b [ITI-42]

Esta transacción describe la interacción entre el repositorio y el registro para que los documentos sean registrados en este último con los metadatos enviados por el repositorio que fueron generados por la fuente de documentos.

Registry Store Query [ITI-18]

La transacción permite a los consumidores realizar consultas al registro de documentos. Hay dos tipos de consultas, las primarias basadas en la identificación del paciente (findDocuments, findSubmissionSets, getall, etc) y secundarias basada en identificadores

del registro (getDocuments, getFolder, getAsossiations, etc.). El retorno será los metadatos o referencias a uno o más objetos del registro.

Retrieve Documento Set [ITI-43]

Es usada por los actores Document Consumer e Initiating *Gateway* para obtener un documento, que es identificado por una DocumentEntry donde se especifica el id. del documento y del repositorio. Estos datos fueron obtenidos previamente con una consulta *Registry* Stored Query al Docuement *Registry*/Initianting *Gateway*. La respuesta a la transacción es devuelta en formato MTOM/XOP.

Patient identity Feed [ITI-8]

La transacción describe la interacción entre los actores Patient Identity Source y Document *Registry* para mantener consistente la información del paciente. El registro utiliza el identificador provisto por la fuente de identificación para verificar que los identificadores contenidos en los metadatos de los documentos sean correctos. Además de ser necesario, modifica los identificadores en los mismos metadatos.

9.3 A3: XCA Actores y Transacciones

Actores

Initiating Gateway

Es el actor encargado de realizar las transacciones de consulta Cross *Gateway* Query y recuperación *Cross Gateway Retreive* hacia otras comunidades, además de consultar y obtener información dentro de su comunidad.

Responding Gateway

Es el actor encargado de consolidar y armar las respuestas a las transacciones de consulta Cross *Gateway* Query y recuperación *Cross Gateway Retreive*. Cuando es necesario, añade a la respuesta la identificación de su comunidad.

Transacciones

Cross Gateway Query [ITI-38]

Esta transacción describe la interacción entre *Initiating Gateway y Responding Gateway* para la consulta de documentos. En este caso Initiating *Gateway* luego de recibir una solicitud local este se encarga según corresponda de iniciar consultas internamente al registro local y hacia otras comunidades, comunicándose con el correspondiente Responding *Gateway*.

Cross Gatewway Retrive [ITI-39]

Esta transacción describe la interacción entre Initiating *Gateway* y Responding *Gateway* para la recuperación de documentos. En este caso Initiating *Gateway* luego de recibir una solicitud local se encarga de iniciar la recuperación de documentos dentro de su comunidad y desde otras comunidades, comunicándose con el correspondiente Responding *Gateway*.

9.4 A4: ATNA Actores y transacciones

Actores

Nodo seguro (Secure Node)

Un nodo seguro debe realizar controles de acceso sobre los usuarios que se conectan localmente, típicamente esto es Autenticación y Autorización. Además es responsable del registro de auditoría para trazar eventos de seguridad como se especifica en la transacción *Record Audit Event* descripta más abajo.

La comunicación con otro nodo seguro debe ser mediante doble autenticación como se especifica en la transacción *Authenticate Node*. Por otro lado, también deberá soportar la configuración para la autenticación y la seguridad a nivel físico de una red, en lugar de la doble autenticación.

Secure Application

La diferencia entre un Nodo Seguro y una Aplicación Segura es el grado en que el sistema operativo subyacente y el entorno son seguros. El Nodo Seguro incluye todos los aspectos de autenticación de usuario, protección a nivel del sistema de archivos, y seguridad del entorno de operación. Una Aplicación Segura es un producto que no incluye el entorno de operación, solamente proporciona los aspectos de seguridad asociados a la funcionalidad que provee.

Audit Record Repository

Este actor es el que recibe y almacena todos los registros de transacciones que son enviados por los nodos y aplicaciones seguras. Debe soportar los dos mecanismos de transporte para los registros de auditoría detallados en la transacción *Record Audit Event* y control de acceso de usuarios.

El repositorio de auditorías puede ser centralizado a nivel de un dominio XDS, o cada organización podría tener su repositorio para sus registros de transacciones locales, pero debería enviar copias de los registros de transacciones que tienen que ver con las transacciones IHE en el resto del dominio.

Consistent Time (CT)

Se incluyen los actores y transacciones de este perfil dado que para que los registros de auditoría sean útiles es necesario que todos los actores que los realizan tengan sincronizado el horario.

Transacciones

Authenticate Node [ITI-19]

Esta transacción implica la mutua autenticación entre los nodos que van a interactuar, mediante el intercambio de certificados. Adicionalmente el nodo solicitante también debe realizar la autenticación local del usuario. El nodo seguro debe aplicar esta transacción a toda conexión DICOM, HTTP o HL7.

El método específico para determinar si un nodo está autorizado a realizar transacciones no es definido por el perfil. Podría estar por ejemplo basado en determinados atributos del certificado o alguna lista de control de acceso.

En lugar de realizar la doble autenticación de nodos se puede manejar la seguridad a nivel físico. La utilización de seguridad a nivel físico y por procedimientos a veces es aconsejable para los intercambios por ejemplo dentro de en un hospital donde la red es segura y la penalización de la performance por la encriptación no se justifica. Un nodo seguro deberá soportar la autenticación y la seguridad a nivel físico de una red.

Record Audit Event [ITI-20]

Esta transacción es utilizada por un actor IHE para crear un log de auditoría y comunicarse con el Repositorio de Registros de Auditoría. Un log de auditoría es un registro de las acciones que los usuarios realizan sobre los datos. Las acciones pueden ser consultas, visualizaciones, altas, bajas y modificaciones.

Para el transporte de los registros de auditoría se debe utilizar el protocolo Syslog utilizando como métodos de transporte TLS 1.2 (recomendado) o UDP. El actor Repositorio de Auditoría (Audit Record *Repository*) debe soportar los dos métodos, dejando que los Nodos seguros decidan cuál utilizar. Sin embargo como UDP solo permite mensajes de hasta 1024 caracteres y cuando se pasan de ese largo los trunca, la IHE recomienda la utilización de TLS.

10 Apéndice B. Instanciación de la solución: Prototipo original

Dadas las necesidades requeridas por el cliente, el escenario planteado y centrándonos en el dominio de afinidad, en cual se comparten documentos a través de la implementación de una arquitectura XDS, podemos instanciar parte de la solución (ver alcance) presentada en el capítulo 4 para estudiar las interacciones entre los componentes considerados.

La instanciación de esta solución tiene como principal diferencia, el no tener en cuenta las comunidades, por lo que no se cuenta con el componente *Gateway* que resuelve la interacción con las demás comunidades para compartir documentos. Otro componente que no se considera es el Community Adapter, que extiende al PIX en la solución del capítulo 4 para contemplar la identificación de los pacientes en las comunidades como ya se vio en dicho capítulo.

Por otro lado, se define un nuevo componente *Proxy* que estará en cada una de las organizaciones con la responsabilidad de exponer los servicios de consulta XDS. El dominio de afinidad Salud UY contará con un registro XDS centralizado en el cual se registran todos los documentos compartidos por las organizaciones en dicho dominio. Además, se cuenta con el componente MPI/PIX que gestiona las referencias cruzadas para los identificadores de pacientes en las distintas organizaciones. Si bien los documentos compartidos están asociados al identificador del paciente en Salud UY, la consulta se podría realizar con el identificador de la organización origen, y el *PIX Consumer* consultar al *MPI/PIX* para resolver el cruzamiento y obtener el identificador en Salud UY. Una vez obtenido el identificador se puede consultar el Registro de documentos.

En la Figura 46 se presenta una arquitectura basada en la solución del capítulo 4, que cumple con los requerimientos y escenario de uso planteados.

El Proxy además de exponer las interfaces del registro y repositorio XDS, permite la comunicación desde Salud UY hacia las organizaciones y viceversa. Cumple con un rol similar para interactuar con el PIX, por ejemplo en la comunicación de la identificación de un paciente al realizarse una admisión.

Las organizaciones para compartir información en salud UY deben poder registrar los documentos en el registro de dicho dominio, en este caso el Proxy puede ser una opción para ello, resultando de nexo entre el repositorio de la organización y el registro centralizado. En caso de ser un repositorio XDS este podría directamente registrar utilizando el servicio del Registro.

Es decir, una organización internamente, no necesariamente debe cumplir con un modelo XDS, pero sí debe implementar la interfaz que se expone a través del Proxy.

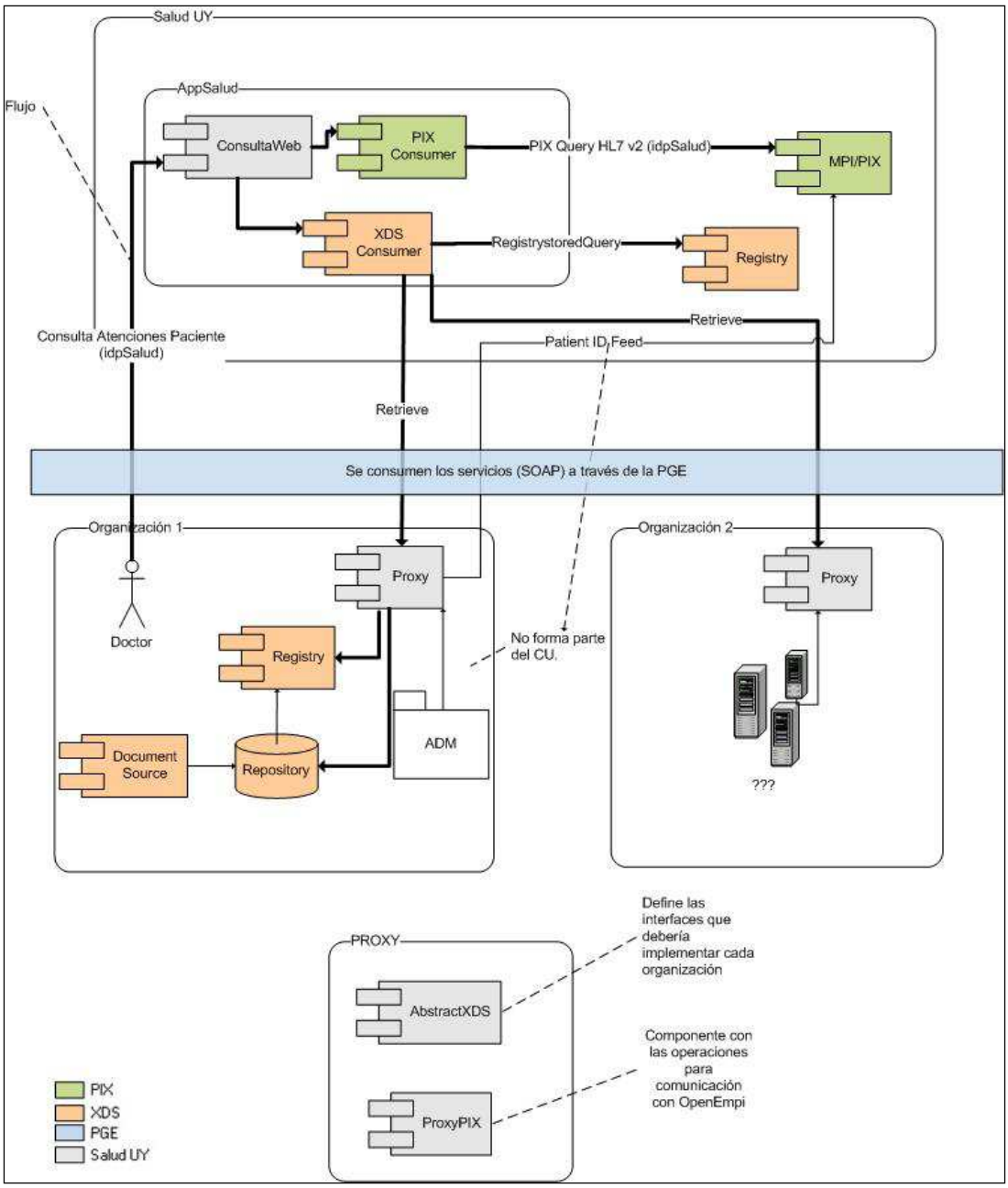


Figura 46: Diagrama de arquitectura basado en la solución del Capítulo 4

10.1 B1: Configuración de los Actores XDS y PIX

En esta sección se presenta la configuración necesaria de los productos OpenXDS y OpenEMPI para el prototipo, presentando la descripción y parte de los archivos correspondientes.

10.1.1 Registro XDS

Para la instalación del registro se instaló OpenXDS configurando los siguientes archivos de configuración.

IheActors.xml: en la Figura 47, se muestra como se especifica el actor "xdsReg" del tipo XdsRegistry y las conexiones que utiliza para atender solicitudes del repositorio y los consumidores, y la comunicación con el PIX. Estas se encuentran definidas en el archivo de XdsRegistryConnections. xml.

```
<!--Defines the configuration of XDS Registry-->
<ConnectionFile file="XdsRegistryConnections.xml" />

<!--The XDS Registry actor configuration-->
<Actor name="xdsreg" type="XdsRegistry">
  <Description>XDS Registry</Description>
  <Connection type="Server" name="xds-registry"/>
  <Connection type="PixServer" name="pix-registry"/>
</Actor>
```

Figura 47: Ejemplo de configuración del actor XdsRegistry

XdsRegistryConnections.xml: en la Figura 48**, s**e especifica las conexiones *xds-registry* a donde se realizarán las peticiones de consulta (*Registry stored Query [ITI-18]*) por parte de los consumidores. Y la conexión *pix-registry* para la comunicación sobre la identificación de pacientes (*Patient Identity Feed [ITI-8]*), correspondiente al PIX con la *XDSRegistry*, donde además se debe especificar cuál es el dominio de identificación global que será comunicado por el PIX.

```
<StandardConnection name="xds-registry">
  <!--The host name of the Registry Server-->
  <HostName>localhost</HostName>
  <Port>8010</Port>
  <Property name="sourceIds"
            value="1.3.6.1.4.1.21367.2010.1.2, 1.3.6.1.4.1.21367.2009.
  <Includefile name="XdsCodes.xml" />
   <Identifier type="domain" name="GLOBAL">
       <NamespaceId>NIST2010</NamespaceId>
       <UniversalId>2.16.840.1.113883.3.72.5.9.1</UniversalId>
       <UniversalIdType>ISO</UniversalIdType>
   </Identifier>
</StandardConnection>

<StandardConnection name="pix-registry">
      <!--The host name of the PIX Registry Server-->
      <HostName>localhost</HostName>
      <!--The port for PIX Feed transaction-->
      <Port>3602</Port>
      <Identifier name="ReceivingApplication">
          <NamespaceId>PAT_IDENTITY_X_REF_MGR_MISYS</NamespaceId>
          <!-- NamespaceId>MESA_XREF</NamespaceId-->
      </Identifier>
      <Identifier name="ReceivingFacility">
          <NamespaceId>ALLSCRIPTS</NamespaceId>
      </Identifier>
      <!-- If test is true, it will by pass the receiving application
           and receiving facility validation -->
   <Property name="test" value="true"/>
   <Identifier type="domain" name="GLOBAL">
       <NamespaceId>NIST2010</NamespaceId>
       <UniversalId>2.16.840.1.113883.3.72.5.9.1</UniversalId>
       <UniversalIdType>ISO</UniversalIdType>
   </Identifier>
</StandardConnection>
```

Figura 48: Ejemplo de XdsRegitryConnections.xml

10.1.2 Repositorio XDS

Para la Instalación de los repositorios XDS se instalan dos instancias de OpenXDS y se configuran los actores correspondientes al "*XDS Repository*". Se asigna el OID de identificación para cada uno de los repositorios en el archivo de propiedades openxds.

IheActors.xml: en la Figura 49**,** al igual que para el registro**,** se especifica el actor del tipo *XdsRepository* y las conexiones que utiliza, que se encuentran definidas en el archivo de XdsRepositoryConnections.xml.

```
<!--Defines the configuration of XDS Repository-->
<ConnectionFile file="XdsRepositoryConnections.xml" />

<!--The XDS Repository actor configuration-->
<Actor name="xdsrep" type="XdsRepository">
  <Description>XDS Repository</Description>
  <Connection type="Server" name="xds-repository"/>
  <Connection type="Registry" name="xds-registry-client"/>
</Actor>
```

Figura 49: Ejemplo de IHEActors

XdsRepositoryConnections.xml: en la Figura 50 se puede ver la configuración de la conexión *xds-repository* que es donde estará el servicio correspondiente a las operación de envío *(Provide and Register Document Set-b [ITI-41])* y recuperación *(Retrieve Document Set [ITI-43])* de documentos. La conexión *xds-registry-client* el servicio correspondiente para el registro de documentos en la registry *(Register Document Set-b [ITI-42])*

```
<!-- The XDS Repository Actor Connections -->

  <StandardConnection name="xds-repository">
    <!--The host name of the XDS Repository Server-->
    <HostName>localhost</HostName>
    <!--The port-->
    <Port>8020</Port>
        <Includefile name="XdsCodes.xml" />
  </StandardConnection>

  <StandardConnection name="xds-registry-client">
    <!--The host name of the XDS Registry Server-->
    <HostName>localhost</HostName>
    <!--The port-->
    <Port>8010</Port>
    <!--The URL of the XDS Registry web services -->
    <UrlPath>/axis2/services/xdsregistryb</UrlPath>
  </StandardConnection>
```

Figura 50: Ejemplo de XdsRepositoryConnections.xml

10.1.3 MPI/PIX

Posteriormente a la instalación del servidor de aplicaciones JBOSS y base de datos Postgres se deberá instalar el producto OpenEmpi el cual cuenta con la implementación de los perfiles PIX/PDQ certificado para versión HL7 v2 y v3. La configuración de los actores y las conexiones es igual a la de OpenXDS. A continuación se detalla lo necesario para su funcionamiento:

IheActors.xml: en la Figura 51 se muestra la configuración del actor *PixManager* y las conexiones necesarias para sus servicios, correspondiente a las operaciones de registro de identificación y consulta de pacientes (conexión *openempi-pix-manager*). También se

configura la conexión *xds-registry* para la comunicación con el registro XDS. La inclusión del archivo *PixPdqClientDomains.xml* especifica los dominios de identificación que se soportan.

```
    <!--Defines the configuration of local system as PIX Manager-->
    <ConnectionFile file="PixManagerConnections.xml" />
    <!--Defines the configuration of PIX Consumers-->
    <ConnectionFile file="PixConsumerConnections.xml" />

    <!--The PIX Manager actor configuration-->
    <Actor name="pixman" type="PixManager">
      <Description>OpenEMPI PIX Manager</Description>
      <Connection type="Server" name="openempi-pix-manager"/>
    <IncludeFile name="PixPdqClientDomains.xml" />

<!--Define 0 or more PIX Consumers that subscribe to PIX Update Notification -->
      <!--
     <Connection type="PixConsumer" name="openempi-pix-consumer"/>
    <IncludeFile name="PixPdqClientDomains.xml" />
     -->
     <Connection type="XdsRegistry" name="xds-registry-v2"/>
        <IncludeFile name="PixPdqClientDomains.xml" />
    </Actor>
```

Figura 51: Ejemplo de IheActors.xml

PixManagerConnections.xml: en la Figura 52 se muestra como configurar el servicio a donde se comunicarán las transacciones de suministro *(Patient Identity Feed [ITI-8])* y consulta de identificadores de pacientes *(PIX Query [ITI-9])*

```
<!-- The PIX Manager Actor Connections -->
<Configuration>
<StandardConnection name="openempi-pix-manager">
    <!--The host name of the PIX Manager Server-->
    <HostName>localhost</HostName>
    <!--The port for PIX transaction-->
    <Port>3600</Port>
    <Identifier name="ReceivingApplication">
        <NamespaceId>MESA_XREF</NamespaceId>
    </Identifier>
    <Identifier name="ReceivingFacility">
        <NamespaceId>XYZ_HOSPITAL</NamespaceId>
    </Identifier>

    <!-- pixManagerAdapter - required :
        The java adapter class that provides the patient data source for this Pix Manager (required).
        This class must implement org.openhealthexchange.openpixpdq.ihe.IPixManagerAdapter -->
  <Property name="pixManagerAdapter"
            value="org.openhie.openempi.openpixpdqadapter.PixManagerAdapter" />

    <!-- storeLogger - optional :
        The java adapter class that persists messages of this Pix Manager (optional).
        This class must implement org.openhealthexchange.openpixpdq.ihe.log.IMessageStoreLogger -->
    <Property name="storeLogger" value="messageStoreLogger" />

    <!-- Configuration file for each client domains -->
  <IncludeFile name="PixPdqClientDomains.xml" />
</StandardConnection>
</Configuration>
```

Figura 52: Ejemplo de PixManagerConnections.xml

PixManagerConsumer.xml: en la Figura 53 se muestra como se configura la conexión para notificar *(PIX Update Notification [ITI-10])* a los consumidores del *PIX Manager* de cambios en la identificación de pacientes (conexión *openempi-pix-consumer*). También se indica la

conexión necesaria para que el *PIX Manager* notifique *(Patient Identity Feed [ITI-8])* al registro XDS *(xds-registry-v2)* nuevos identificadores de paciente o cambios en uno ya existente. Además se especifica el dominio de afinidad para la identificación de pacientes definido para XDS.

```
<StandardConnection name="openempi-pix-consumer">
    <!--The host name of the PIX Consumer-->
    <HostName>localhost</HostName>
    <Port>3601</Port>     <!--The port for PIX Update Notification-->
    <Identifier name="SendingApplication">
        <NamespaceId>PAT_IDENTITY_X_REF_MGR_MISYS</NamespaceId>
    </Identifier>
    <Identifier name="SendingFacility">
        <NamespaceId>ALLSCRIPTS</NamespaceId>
    </Identifier>
    <Identifier name="ReceivingApplication">
        <NamespaceId>EHR_MISYS</NamespaceId>
    </Identifier>
    <Identifier name="ReceivingFacility">
        <NamespaceId>MISYS</NamespaceId>
    </Identifier>
    <!-- This property is for test purpose, which would not actually
         submit PIX Update Notification. It defaults to false. -->
    <Property name="DoNotNotify" value="false" />
    <IncludeFile name="PixPdqClientDomains.xml" />
</StandardConnection>
<StandardConnection name="xds-registry-v2">
    <!--The host name of the PIX Consumer-->
    <HostName>localhost</HostName>
    <Port>3602</Port>
    <Identifier name="SendingApplication">
        <NamespaceId>PAT_IDENTITY_X_REF_MGR_MISYS</NamespaceId>
    </Identifier>
    <Identifier name="SendingFacility">
        <NamespaceId>ALLSCRIPTS</NamespaceId>
    </Identifier>
    <Identifier name="ReceivingApplication">
        <NamespaceId>EHR_MISYS</NamespaceId>
    </Identifier>
    <Identifier name="ReceivingFacility">
        <NamespaceId>MISYS</NamespaceId>
    </Identifier>
    <Property name="XDSAffinityDomainPatientIdetifier" value="NIST2010" />
    <IncludeFile name="PixPdqClientDomains.xml" />
</StandardConnection>
```

Figura 53: Ejemplo de PixManagerConsumer.xm

PixPDQClientDomains.xml: en la Figura 54 se muestra la configuración de dominios de este archivo. Contiene todos los dominios de identificación soportados por el *PIX Manager*. Particularmente se debe configurar el identificador del dominio global el cual se el único que alimentará el registro XDS.

```
<!-- The Domain of Each PIX and PDQ Client -->

 <Identifier type="globalDomain" name="NIST2010">
    <NamespaceId>NIST2010</NamespaceId>
    <UniversalId>2.16.840.1.113883.3.72.5.9.1</UniversalId>
    <UniversalIdType>ISO</UniversalIdType>
</Identifier>

<Identifier type="domain" name="IHE_GREEN">
    <NamespaceId>IHEGREEN</NamespaceId>
    <UniversalId>1.3.6.1.4.1.21367.13.20.2000</UniversalId>
    <UniversalIdType>ISO</UniversalIdType>
</Identifier>
```

Figura 54: Ejemplo de PixPDQClientDomains.xml

11 Apéndice C: Desafíos técnicos

11.1 C1: Metadatos XDS y Representacion en ebRIM

Tenemos que los metadatos utilizados por los perfiles definidos para compartir documentos están caracterizados por tres tipos de objetos y dos tipos de asociaciones. En la Figura 55 se pueden visualizar los mismos modelados con notación UML para representar la relación entre ellos:

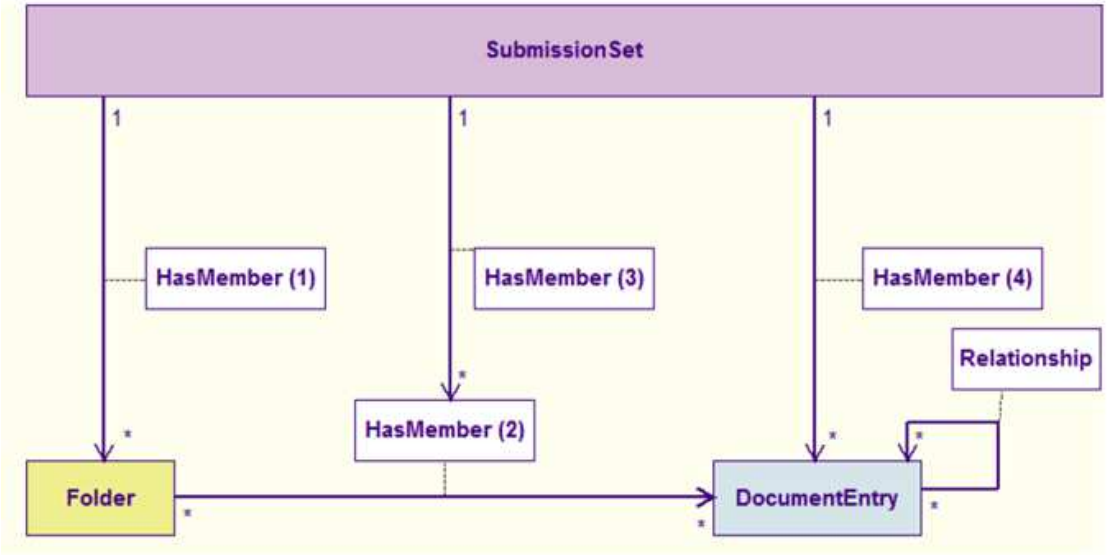


Figura 55: Representación de documentos compartidos y asociaciones [47]

Los tres tipos de objetos son:

- **SubmissionSet:** Representa un conjunto de carpetas, documentos y relaciones que se envían al repositorio juntos.
- **Folder:** Representa una colección de documentos relacionados
- **DocumentEntry:** Representa un documento

Las dos asociaciones son:

- **HasMember:** representa la pertenencia de los objetos. Existen 4 variantes de esta asociación que corresponden a las representadas en diagrama anterior.
- **Relationship:** representa una relación entre DocumentEntries, existen cinco variantes de relación posibles entre entradas de documentos, replace, transform, append, Transform and Replace y signs

Estos objetos a su vez tienen una serie de atributos que se pueden categorizar de acuerdo al propósito específico que se le quiera dar a los documentos.

Alguno de los propósitos de estos metadatos son:

- **Identidad del Paciente** – Atributos que describen el sujeto del documento. Esto incluye id del paciente, nombre y otros datos demográficos relevantes.

- **Procedencia** – Atributos que describen de donde proviene el documento. Esta información depende mucho de las regulaciones sobre los registros médicos. Esto incluye el autor humano, identificación del sistema que lo creó, documentos predecesores, documentos sucesores
- **Seguridad y privacidad** – Atributos que son utilizados para representar las reglas de seguridad y privacidad apropiadas para los documentos
- **Descriptivos** – Atributos que son utilizados para describir los valores clínicos, son expresados específicamente. Estos valores son críticos para los modelos de consultas y permiten *workflows* en todos los modelos de intercambio.
- **Ciclo de vida del objeto** – Atributos que describen el estado actual dentro del ciclo de vida del documento incluyendo las relaciones con otros documentos. Estos datos deberían incluir los clásicos estados del ciclo de vida definidos: *created, published, replaced, transformed y deprecated.*
- **Intercambio** – Atributos que permiten la transferencia de los documentos.

Cada objeto asociación definido en los metadatos XDS se representa con una clase del ebRIM como se muestra en la Figura 56.

Document Sharing Object/Association	ebRIM class
DocumentEntry	rim:ExtrinsicObject
SubmissionSet	rim:RegistryPackage
Folder	
MemberOf	rim:Association
Relationship	

Tabla 8: Representación Metadatos en ebRIM [47]

Como se puede ver en la Figura 56, la DocumentEntry se modela como la clase rim: ExtrinsicObject y las relaciones HasMember y Relationship con la clase rim:Association.

El SubmissionSet y el Folder se representan con la clase rim:RegistryPackage. Debido a que el estándar ebRIM no permite la creación de subclases de la clase RegistryPackage, estos dos objetos son implementados como rim:RegistryPackages y se utiliza una clase rim:Classification para distinguirlos.

Además de las clases ya mencionadas se utilizan otras clases para manejar atributos generales como el nombre (rim:Name) y la descripción (rim:Description) y otras clases de propósito general como rim:Slot donde se mapean atributos propio salud o la clase rim:Classification que se utiliza para clasificar un RegistryObject. También se utiliza una clase para proporcionar información adicional de identificación, la rim:ExternalIdentifier. La clase rim:RegistryObjectList se utiliza en las peticiones y respuestas como contenedor de listas de SubmissionSets, Folders, DocumentEntries y Associations.

El modelo completo de clases ebRIM utilizadas para el XDS se muestra en la Figura 57.

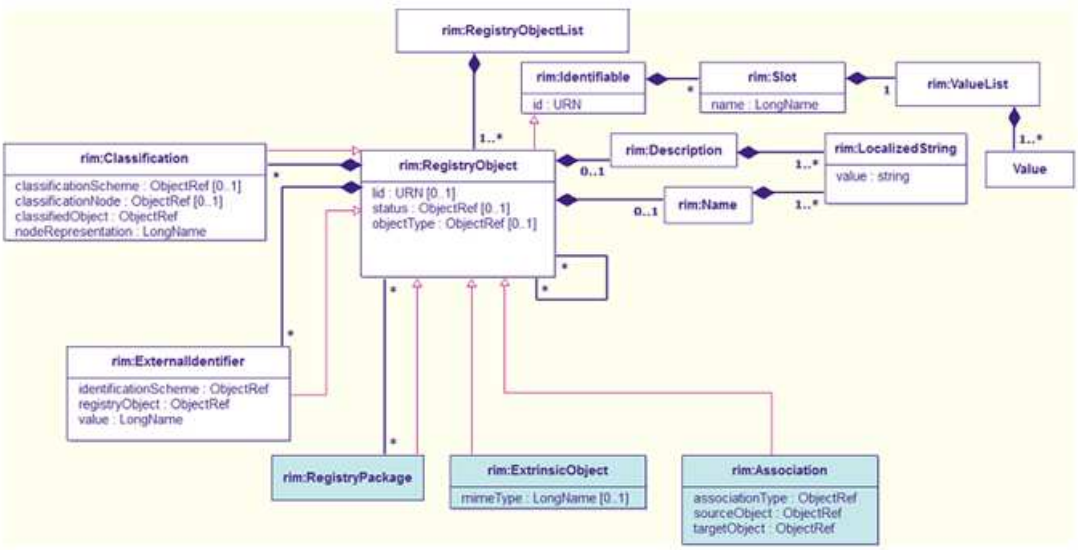


Figura 56: Modelo completo ebRIM [47]

Un análisis más detallado de lo expuesto anteriormente se puede encontrar en el Anexo MetadatosXDS.doc