



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

**INTEROPERABILIDAD DE BASES DE DATOS. CASO DE
ESTUDIO: EXPEDIENTE CLÍNICO ELECTRÓNICO**

T E S I S

QUE PARA OPTAR POR EL GRADO DE:
MAESTRO EN INGENIERÍA (COMPUTACIÓN)

P R E S E N T A:
ESTEBAN ARRANGÓIZ ARECHAVALA

DIRECTOR DE TESIS:
DR. JAVIER GARCÍA GARCÍA
FACULTAD DE CIENCIAS, UNAM

MÉXICO, D. F. JULIO 2014



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Resumen

Posgrado en Ciencia e Ingeniería de la Computación

Maestro en Ingeniería (Computación)

Interoperabilidad de Bases de Datos. Caso de estudio: Expediente Clínico Electrónico

por [Esteban Arrangóiz Arechavala](#)

La interoperabilidad entre sistemas heterogéneos facilita el acceso a la información y mejora los procesos que involucran la comunicación entre diferentes sistemas. En el ámbito de la salud la interoperabilidad ayuda a brindar una mejor y más eficiente atención al paciente. Este trabajo presenta una metodología para implementar la interoperabilidad de bases de datos heterogéneas, con el caso de estudio de sistemas de información de la salud en México empleando el estándar HL7v3. La metodología puede ser aplicada a cualquier tipo de bases de datos que requieran implementar la interoperabilidad (de intercambio de datos) sin alterar sus estructuras ni afectar su funcionamiento existente. Además, se presentan dos casos en los que se utilizó la metodología presentada para implementar satisfactoriamente la interoperabilidad entre dos sistemas; el primer caso, entre un Sistema de Información Hospitalaria y un Sistema de Información de Laboratorios; y el segundo caso, se realizó para la comunicación de una instancia deportiva con un sistema de Información Hospitalaria.

Índice general

Resumen	i
Índice de Figuras	v
1. Introducción	1
1.1. Objetivo	3
1.2. Trabajos relacionados	4
1.3. Descripción de capítulos	5
2. Marco teórico	7
2.1. Bases de datos heterogéneas	7
2.2. Integración de bases de datos	7
2.2.1. Metodología de diseño Bottom-Up	8
2.2.2. Emparejamiento de esquemas	9
2.2.3. Integración de sistemas heterogéneos	10
2.2.4. Mapeo de esquemas	11
2.2.5. Interoperabilidad	11
2.2.5.1. Interoperabilidad Sintáctica	12
2.2.5.2. Interoperabilidad Semántica	12
2.2.5.3. Interoperabilidad del negocio	13
2.3. Estándares y la interoperabilidad	15
2.3.1. HL7	16
2.3.2. HL7v2.x	18
2.3.3. Definiciones de HL7	20
2.3.4. Mensajes HL7	21
2.3.5. Reglas básicas de construcción de mensajes	22
2.3.6. Object Identifier (OID)	23
2.3.7. Modelo de Información de Referencia (RIM)	23
2.3.8. HL7v3	23
2.4. LOINC	24
2.5. SNOMED-CT	25
2.6. NEMA	26
2.7. XML	27
2.7.1. XSD	29

2.7.2. Espacios de nombres	29
2.8. Servicios Web	29
3. Expediente Clínico Electrónico y la situación actual en México	31
3.1. Expediente Clínico Electrónico	31
3.2. Historia de Salud Personal	33
3.3. Situación actual del ECE en México	33
4. Implementación	39
4.1. Metodología de implementación	39
4.1.1. Especificación de la información de intercambio	41
4.1.2. Identificación del estándar	41
4.1.3. Realizar mapeo de campos	42
4.1.4. Especificación de la interfaz de intercambio	42
4.1.5. Generar los mensajes	43
4.1.6. Implementar la interfaz de intercambio	44
4.2. Estructura de mensajes	44
4.2.1. Solicitud de laboratorio	44
4.2.2. Resultados de laboratorio	49
4.3. Experimentación	52
4.3.1. Comunicación de un HIS con un LIS	53
4.3.2. Institución educativa	55
5. Conclusiones y trabajos futuros	59
A. Evidencia de resultados	61
B. Ejemplo de un mensaje (XML) de solicitud de laboratorio	65
C. Ejemplo de un mensaje (XML) de resultados de laboratorio	67
D. Código de Java que genera los mensajes de solicitud de estudios de laboratorio	69
Abreviaciones	74
Bibliografía	77

Índice de figuras

2.1. Interoperabilidad en los sistemas de salud	14
3.1. Diferentes módulos del ECE	32
4.1. Metodología de implementación	40
4.2. Flujo de datos de la solicitud en la implementación, desde el HIS hasta el LIS	55
4.3. Flujo de datos de resultados en la implementación, desde el LIS hasta el HIS	56
A.1. Evidencia de resultados en Grupo GC. Hoja 1	62
A.2. Evidencia de resultados en Grupo GC. Hoja 2	63
A.3. Carta de participación en Grupo GC.	64

Capítulo 1

Introducción

El ímpetu en el crecimiento de la información en las grandes organizaciones y la necesidad intrínseca de su manejo eficiente y adecuado, entre otras cosas, ha provocado el desarrollo de software a medida, la adquisición de software comercial o la adopción de software libre. En el mejor de los casos, la solución de software que se elige es global, esto es, abarcando a todas las áreas de la organización. Sin embargo, normalmente se ha optado por crear soluciones heterogéneas, es decir, independientes para cada área o sección [1].

A diferencia de los sistemas centralizados, que basan su construcción en la visión global del negocio de las organizaciones, los sistemas heterogéneos se han ido adoptando para dar solución a las necesidades individuales de áreas específicas de las organizaciones a lo largo del tiempo [2]. Sin embargo, estas soluciones individuales no se han pensado para solventar los requisitos actuales de interoperabilidad, integración y escalabilidad. Parecería ser que en algunas organizaciones la adopción de sistemas de información, con el paso de los años, ha sido guiada por la intuición, la moda, o las políticas de la autoridad en turno, en lugar de ser orientado por una perspectiva y planeación futurista del negocio y beneficios de la organización.

Dado que las necesidades de los clientes de cada organización aumentan paulatinamente en lo que respecta a la facilidad del acceso a la información y agilidad en los servicios, más aún en las organizaciones que prestan servicios sociales como es el caso de la salud, un sistema integral se vuelve fundamental. Desafortunadamente, el costo que esto implica tanto monetario como operativo y la pérdida en la inversión de sistemas existentes, hacen compleja la opción de la creación de un sistema integral nuevo [3]. Por lo tanto, la interoperabilidad e integración de sistemas heterogéneos

existentes en las organizaciones ha llegado a constituirse en una alternativa muy prometedora. Sin embargo, las políticas de las organizaciones, particularmente las de la salud, parecen influir en el desarrollo de esta iniciativa de interoperabilidad. Es por esto que se torna necesario un enfoque que ofrezca viabilidad a la integración de los sistemas heterogéneos.

La interoperabilidad entre diferentes bases de datos es un reto que tienen muchos servicios de información al buscar compartir datos entre sistemas heterogéneos [2]. Cuando estos sistemas han sido desarrollados de manera separada, sin una planeación que se vislumbrara al interactuar entre ellos, la intercomunicación debe poder realizarse sin que se requiera hacer cambios en sus estructuras ni en su funcionamiento. Es posible compartir información entre estos sistemas heterogéneos mediante la integración de bases de datos que se comuniquen con protocolos establecidos y así funcionen correctamente compartiendo sus datos.

En los sistemas de salud se tiene este problema al requerir compartir datos entre las entidades que proporcionan recursos o datos involucrados en el cuidado de la salud [4], como pueden ser Sistemas de Información Hospitalaria (HIS, por sus siglas en inglés), Sistemas de Información de Laboratorios (LIS, por sus siglas en inglés), Sistema de Archivado y Transmisión de Imágenes (PACS, por sus siglas en inglés), etc. Para estandarizar a nivel internacional esta comunicación se crearon los estándares HL7, los cuales contienen los requerimientos generales para comunicar sistemas de información de la salud [4].

Como se sabe, cada empresa o institución que requiere un sistema de información lo implementa según sus necesidades y recursos. Así ha ocurrido en las diferentes instituciones que componen el sector salud de México, cada una ha utilizado un sistema de información que mejor se ajusta a sus necesidades o posibilidades [3], lo que ha dado lugar a que existan sistemas que funcionan aceptablemente bien de manera interna pero que no están desarrollados con la capacidad de comunicarse fuera de su entorno. Esto viene a ser una limitante para todas las personas involucradas con el uso de esta información, que en este caso son los médicos y pacientes del sistema de salud, porque al cambiar de institución (ya sea por interés propio o necesidad), se pierde la información que se había acumulado en la institución anterior, lo cual acarrea dificultades para poder dar una buena atención médica si no se cuenta con el historial clínico [4].

Los HIS han crecido grandemente en los últimos años, pero cada uno por su camino. Lo que se requiere en México, y en cualquier país, para mejorar la atención médica es integrar todos estos sistemas y así evitar que se pierdan datos de los pacientes y que se cuente con ellos en el momento que se requieran [5]. Un sistema de Expediente Clínico Electrónico (ECE) centralizado a nivel nacional sería demasiado costoso y difícil de realizar [5], además de que se desecharían las inversiones realizadas en los avances de cada institución. La interoperabilidad es la opción más sencilla, de bajo costo y eficaz que presenta una solución viable, además de que facilitaría el acceso a la información y mejoraría la seguridad y calidad del cuidado del paciente sin importar su ubicación [6]. Existen estudios que revelan que la interoperabilidad entre sistemas médicos pueden reducir costos de integración en un 30% [7].

1.1. Objetivo

El objetivo de este trabajo es proponer una metodología de implementación para lograr la interoperabilidad entre bases de datos de sistemas heterogéneos (en particular de los sistemas de información de la salud), como pueden ser entre un Sistema de Información Hospitalaria y un Sistema de Información de Laboratorios, o entre un HIS y un Sistema de Información Radiológica. Para esto, se realizó un análisis de la situación actual de la interoperabilidad entre instituciones de salud en México y se muestran dos casos de estudio de la comunicación implementada, en la cual se utilizó la metodología propuesta empleando el estándar HL7 en su tercera versión (HL7v3). Estas implementaciones se realizaron con miras a resolver la falta de comunicación entre diferentes HIS y otros sistemas de información de la salud en México. Los dos casos de estudio son: 1) comunicación de un HIS con un LIS y 2) comunicación de un HIS con los datos de laboratorio de una institución de actividades deportivas de una institución educativa. A continuación se mencionan los objetivos que, una vez alcanzados, permitieron llegar al objetivo principal:

Objetivos:

- Proponer una metodología que ayude a implementar la interoperabilidad entre bases de datos heterogéneas que requieren mantener sus estructuras y funcionamiento original.

- Presentar dos casos de estudio de integración de los sistemas de instituciones de salud en México que puedan servir como propuesta para las bases de una guía de implementación de éstos.
- Realizar una investigación exhaustiva sobre la integración de bases de datos, en particular en el sistema de salud de México, y conocer las investigaciones más recientes del tema.

1.2. Trabajos relacionados

Existen trabajos que han tratado el problema de interoperabilidad entre sistemas médicos heterogéneos [7], [8], [9], [10], [11], [12]. En estos trabajos se toca el tema de interoperabilidad pero en ninguno se propone una metodología de solución como la presentada en este trabajo. Por ejemplo, Liu et al. [8] muestran una arquitectura que permite comunicar diferentes tipos de Registros Electrónicos de Pacientes utilizando la tecnología OGSA-DAI (Open Grid Services Architecture - Data Access and Integration). Su trabajo permite comunicar únicamente bases de datos relacionales, lo cual presenta la limitante de no poder comunicar cualquier tipo de sistema manejador de bases de datos. Además de que su solución está realizada en Java, por lo que se requiere que los sistemas que se quieren comunicar tengan la capacidad de integrar esta arquitectura de solución que requiere la ejecución de código Java. En contraste, la metodología presentada en este trabajo no trae estas restricciones, sino que muestra una serie de pasos que se deben realizar en los sistemas a comunicar sin importar la naturaleza de éstos en cuanto a lenguaje de programación y estructura de las bases de datos.

En el trabajo de Yuksel et al. [7] utilizan mensajes HL7 para estandarizar la comunicación entre dispositivos médicos del Modelo de Información de Dominio (DIM, por sus siglas en inglés) ISO/IEEE 11073. El trabajo logra interoperabilidad de manera estandarizada entre dispositivos que utilizan el modelo ISO/IEEE 11073. Esa es la limitante y contraste principal con el presente trabajo.

Cuenca et al. [9] mencionan una arquitectura de comunicación entre los diferentes componentes de un HIS utilizando como base el estándar HL7v2. Este trabajo se ha implementado en Colombia pero no presenta una metodología de implementación que sirva como guía para lograr la interoperabilidad en cualquier sistema. Un caso

muy similar es el trabajo de Hussain et al. [10], en el que muestran una implementación de mensajería HL7 que es parte del subsistema de transporte de un proyecto de HLH (Health Life Horizon) en la escuela de Ingeniería Eléctrica y Ciencias de la Computación (NUST SEECS, por sus siglas en inglés) de Pakistán. Este caso es una particular solución de interoperabilidad utilizando la versión 2 del estándar HL7 y servicios web, pero tampoco propone una metodología general para lograr la interoperabilidad de cualquier sistema.

En el trabajo de Ryan et al. [12] se discute el problema de la falta de interoperabilidad en los sistemas de salud. Analizan principalmente la interoperabilidad sintáctica, para la cual mencionan que es posible lograrla mediante el uso de SNOMED-CT y de mensajes HL7. Su trabajo muestra cómo se pueden ocupar los estándares en terminología médica para la interoperabilidad sintáctica, aunque para realizar la implementación de la transferencia de la información sólo mencionan que se puede lograr al combinar el uso de los términos de SNOMED-CT con la mensajería HL7, sin dar una guía sobre cómo lograrlo. Este trabajo aborda esta misma necesidad de interoperabilidad brindando una propuesta para implementarla y lograr que sea operacional en sistemas ya existentes.

1.3. Descripción de capítulos

El presente trabajo se estructura de la siguiente manera:

En el capítulo 2 se presenta el marco teórico del trabajo realizado. En el capítulo 3 presentamos el panorama actual del Expediente Clínico Electrónico en México. El capítulo 4 presenta la metodología propuesta para lograr la implementación de la interoperabilidad entre sistemas heterogéneos y los casos de estudio realizados en sistemas reales. Por último, el capítulo 5 describe las conclusiones y alcances identificados en el impacto de la implementación de la metodología presentada.

Capítulo 2

Marco teórico

2.1. Bases de datos heterogéneas

Desde la década de 1970, eran muy utilizadas las bases de datos centralizadas y se comenzó a ver que los usuarios necesitaban consultar y manejar información de diferentes bases de datos y las aplicaciones también requerían utilizar bases de datos independientes. Éstas son creadas de manera autónoma y de la misma forma es realizada su administración, por lo que difieren tanto física como lógicamente. Cada una de ellas tiene su esquema propio, expresado en su modelo de datos y accedido por su propio lenguaje de consultas definido por su creador [2].

El desarrollo del cómputo distribuido ha aportado las bases teóricas para el acceso remoto de elementos separados físicamente. Sin embargo, no es suficiente lograr la comunicación a distancia, sino que se requiere que estos elementos separados puedan interactuar entre ellos y funcionar en conjunto, sin que cada uno de ellos pierda su autonomía y de una manera que sea transparente para el usuario y las aplicaciones, esto es, hacerlos interoperables [2].

2.2. Integración de bases de datos

La integración de bases de datos trata de crear una base de datos que contenga a varias. La nueva base de datos es formada por un conjunto de éstas que existían independientes una de otra [13].

En la integración de bases de datos, el diseño bottom-up es apropiado en sistemas multibase de datos [13]. En este diseño, un conjunto de bases de datos existe, y la tarea involucra integrarlas en una sola. El punto de partida del diseño bottom-up es el esquema conceptual local (LCS, por sus siglas en inglés) de cada una de las bases de datos existentes. El proceso consiste en la integración de las bases de datos locales con sus esquemas (locales) en una base de datos global con su esquema conceptual global (GCS, por sus siglas en inglés).

La integración de bases de datos puede ser de manera física o lógica. En la primera, al ser integradas las bases de datos originales, se materializa una que contiene los datos de todas ellas. Éstas son conocidas como almacenes de datos (DW, por sus siglas en inglés). En el segundo caso, el GCS es completamente virtual y no se materializa. Ese es el caso de las integraciones lógicas de datos o sistemas multibase de datos. En ellas los datos residen en las bases de datos transaccionales y el GCS provee una integración virtual para consultarlas. En este caso, el GCS puede no ser la unión de los esquemas conceptuales locales [13]. Las consultas de los usuarios son presentadas en el esquema global y después se descomponen y envían a las bases de datos transaccionales locales para su procesamiento de la manera en que se realiza en sistemas estrechamente integrados. Las principales diferencias son la autonomía y heterogeneidad potencial de los sistemas locales. Estos sistemas son principalmente de solo lectura ya que mantener las actualizaciones globales es bastante difícil debido a la autonomía de los sistemas manejadores de bases de datos de las bases de datos transaccionales [13].

2.2.1. Metodología de diseño Bottom-Up

El diseño bottom-up involucra el proceso por el cual la información de las bases de datos participantes pueden ser integradas (física o lógicamente) para formar una sola multibase de datos. El GCS se define como la integración de las partes de los esquemas conceptuales globales. En este caso, el diseño bottom-up involucra tanto la generación de esquemas conceptuales globales como el mapeo de los esquemas conceptuales locales al global [14].

El diseño bottom-up ocurre en dos pasos generales: la traducción del esquema y la generación de esquemas. En el primer paso, los componentes del esquema de la base de datos son traducidos a una representación canónica intermedia. El uso de una representación canónica facilita el proceso de traducción al reducir el número de traductores que necesitan ser escritos. En este sentido, la elección del modelo

canónico es importante. Como regla se debe seleccionar aquel que es suficientemente expresivo para incorporar los conceptos disponibles en todas las bases de datos que serán integradas más adelante. Algunas alternativas que pueden ser utilizadas son: el modelo entidad-relación, el modelo orientado a objetos, o un grafo que pueda ser simplificado a un árbol. La elección del modelo relacional como la representación de datos canónica no afecta de ninguna manera fundamental la discusión de los mayores problemas de la integración de datos [14].

Evidentemente, el primer paso del diseño bottom-up, el de traducción, será necesario solamente si las bases de datos que componen el sistema son heterogéneas y los esquemas locales fueron definidos utilizando diferentes modelos de datos.

En el segundo paso del diseño bottom-up, los esquemas intermedios son usados para generar un GCS. Este proceso de generación de esquemas consiste de los siguientes pasos [14]:

1. El “*schema matching*” o emparejamiento de esquemas, que consiste en determinar las correspondencias sintácticas y semánticas entre los elementos traducidos de los LCS o entre los elementos individuales de los LCS y los elementos de los GCS predefinidos.
2. Integración de los elementos comunes de los esquemas en un GCS si no se ha definido alguno.
3. El “*schema mapping*” o mapeo de esquemas, que es el proceso de determinar cómo se trasladan los elementos de cada uno de los LCS a los otros elementos del GCS.

2.2.2. Emparejamiento de esquemas

El emparejamiento de esquemas determina cuáles conceptos de un esquema son equivalentes a los de otro esquema [13]. Las equivalencias que son definidas o descubiertas durante este proceso se especifican como un conjunto de reglas donde cada regla identifica una correspondencia entre dos elementos, un predicado que indica cuándo se tiene la correspondencia, y un valor de similitud entre los dos elementos identificados en la correspondencia. Una correspondencia puede simplemente identificar que dos conceptos son similares o puede ser una función que especifica que un concepto puede ser derivado por un cálculo a partir de otro. El predicado es una condición que califica

la correspondencia especificando cuando se tiene ésta. El valor de similitud para cada regla puede ser especificado o calculado.

Las correspondencias pueden ser tanto descubiertas como especificadas. Aunque se quisiera automatizar este proceso, hay muchos factores que lo complican [13]. El más importante de éstos es la heterogeneidad de esquemas, que, de manera general, se refiere a las diferencias en la manera en que los fenómenos del mundo real son capturados en diferentes esquemas.

Los algoritmos de emparejamiento de esquemas tratan tanto con la estructura heterogénea como con la semántica heterogénea entre los esquemas emparejados. Los conflictos estructurales ocurren en cuatro posibles formas [13]: conflictos de tipo, conflictos de dependencia, conflictos entre llaves y conflictos de comportamiento. Los conflictos de tipo ocurren cuando el mismo objeto es representado por un atributo en un esquema y por una entidad (relación) en otro. Los conflictos de dependencia ocurren cuando diferentes modos de relación (por ejemplo uno-a-muchos contra uno-a-uno) son utilizados para representar la mismo en diferentes esquemas. Los conflictos entre llaves ocurren cuando se tienen disponibles diferentes llaves candidato y las llaves primarias elegidas no son las mismas en los esquemas. Los conflictos de comportamiento están implícitos en el mecanismo empleado para el modelado del esquema (por ejemplo, que al borrar el último artículo de un catálogo implique el borrado de ese catálogo en un esquema y en el otro esquema sí se mantiene el catálogo).

Las diferencias estructurales entre esquemas son importantes pero su identificación y solución no es suficiente. El emparejamiento de esquemas tiene que tomar en cuenta la semántica de los esquemas conceptuales. Esto se conoce como heterogeneidad semántica. Básicamente se refiere a las diferencias entre las bases de datos que relacionan el significado, interpretación y el uso previsto de los datos.

2.2.3. Integración de sistemas heterogéneos

Como se mencionó anteriormente, para lograr la integración de bases de datos existen dos pasos importantes: el emparejamiento de esquemas, y el mapeo de esquemas. Para lograr la correspondencia entre las bases de datos se requiere conocer y analizar las estructuras de las bases de datos existentes, lo cual es difícil cuando se está hablando de sistemas que pertenecen a entidades que tal vez no comparten ninguna estructura

y que seguramente fueron creadas por diferentes desarrolladores de software, cada uno con su metodología, lenguaje y estructura. La integración de bases de datos es solo una parte del problema más general que es la interoperabilidad [14]

Así que si contamos con un conjunto de sistemas heterogéneos que no se pueden (o es difícil) cambiarlos estructuralmente para hacerlos homogéneos, se requiere una alternativa que permita conservar los sistemas tal cual están operando y agregar ciertas funcionalidades para lograr la integración de estos sistemas. En este trabajo se propone una alternativa de solución orientada a servicios, la cual permite hacer sencilla dicha integración al proponer una metodología de implementación para lograr que interactúen o se comuniquen los sistemas sin importar la tecnología con la cual fueron desarrollados.

Llevando este tema al ámbito de la salud, la interoperabilidad entre los sistemas de información de la salud de manera eficiente, segura, oportuna y portable, es un factor que puede mejorar notablemente la atención al cuidado de la salud que se brinda a la población, además de reducir los costos administrativos y de operación que tienen las instituciones involucradas [4].

2.2.4. Mapeo de esquemas

Una vez que se tiene el emparejamiento de esquemas, es necesario realizar un proceso para identificar cómo se van a mapear los datos de una base de datos a otra, de manera que se conserve la integridad de los datos. Este proceso, llamado “*schema mapping*” o mapeo de esquemas, se basa en las equivalencias determinadas en el emparejamiento de esquemas para hacer que los datos alojados en una base de datos sean transferidos a otra, incluyendo las consultas explícitas que obtienen los datos en el origen y las posibles transformaciones que se deben aplicar a los datos para ser almacenados en su destino.

2.2.5. Interoperabilidad

La IEEE define la interoperabilidad como la habilidad que posee un sistema o producto para trabajar con otros sistemas o productos sin requerir un esfuerzo especial por parte del cliente [15]. Esta habilidad permite comunicar sistemas sin importar

que sus estructuras sean diferentes o incluso que sean sistemas legados (software que ha quedado anticuado pero continúa siendo utilizado por el usuario).

En diversas oportunidades se ha reconocido la importancia de la interoperabilidad, por ejemplo, en el año 2005 el Departamento de Salud de Estados Unidos destacó que la falta de interoperabilidad es uno de los mayores impedimentos en el uso de los registros médicos electrónicos [16].

Para que haya una interoperabilidad total, desde el sistema más simple al más complejo y desde el nivel de datos al de aplicaciones, existen varios requerimientos que deben ser satisfechos [11]:

- El intercambio de información sea posible (interoperabilidad sintáctica o funcional).
- La información intercambiada se pueda utilizar (interoperabilidad semántica).
- Existen reglas de negocio definidas que permiten comprender cuál es la información que se va a intercambiar y qué detonará el intercambio (interoperabilidad de negocio).

2.2.5.1. Interoperabilidad Sintáctica

La interoperabilidad sintáctica, llamada también operativa o funcional, es la habilidad que tienen dos o más sistemas o componentes para intercambiar información entre ellos. Esto es, se refiere a la estructura de una comunicación para lograr que los sistemas se comuniquen utilizando una sintaxis “entendible” entre ellos.

2.2.5.2. Interoperabilidad Semántica

La interoperabilidad semántica contiene el significado de lo que se comunica, es decir, sería semejante a lo que es un diccionario para un idioma. Para lograr esta interoperabilidad, una solución que se propone para emplear en Sistemas de Información Médica es utilizar estándares de terminologías internacionales como SNOMED (*Systematized Nomenclature of Medicine*), LOINC (*Logical Observation Identifiers Names and Codes*) o documentos clínicos estándares como los CDA (*Clinical Document Architecture*).

Si no se tiene interoperabilidad semántica, aunque los datos sean transferidos a otro sistema, no habría seguridad de que puedan ser utilizados por el receptor si éste no les puede dar un significado. Muchos de los estándares que están disponibles actualmente en el área de la salud, cumplen con ambos tipos de interoperabilidad, sintáctica y semántica.

2.2.5.3. Interoperabilidad del negocio

Para alcanzar una interoperabilidad al 100 % se requiere comprender las prioridades y la persistencia de los datos en escenarios muy variados. En un HIS por ejemplo, es necesario contemplar todas las necesidades que tienen las diferentes áreas que participan como consulta interna, urgencias, internación psiquiátrica, rehabilitación, emergencias, farmacia, enfermería, etc. Lograr esto requeriría tener información de cada escenario que se pueda presentar en cada una de las áreas para entender todas las reglas de negocio de los distintos elementos que participan en la comunicación. Con verdadera interoperabilidad, esto es, aquella que cumpla con los 3 tipos (semántica, sintáctica y de negocio), los datos podrían ser intercambiados dentro de una misma institución o con otras, compartidos con el público en general y hasta utilizados para sistemas de toma de decisiones a nivel nacional.

Habiendo mencionado lo anterior, se debe tener claro que la interoperabilidad requiere estándares por varias razones, para intercambiar datos (escribir con la misma sintaxis) y además para entender y hacer uso de los datos intercambiados (“hablar” el mismo idioma a través de diccionarios o terminologías médicas).

A su vez es importante mencionar que para evitar que se rompa la cadena de interoperabilidad en los diferentes niveles de agregación (local, regional y nacional), es necesario adoptar estándares en todos estos niveles. De no hacerlo así, nunca se logrará un sistema nacional totalmente integrado. El consenso sobre qué estándares utilizar y cómo utilizarlos juega un rol fundamental, ya que no existe un axioma sobre cómo llegar a la interoperabilidad [17].

En el mismo sentido, para que la interoperabilidad sea posible con cualquier sistema es necesario utilizar estándares internacionales [18]. En el ámbito de la salud existen los estándares HL7, una de las alternativas de integración de servicios de información de la salud, y también el método más empleado [9]. Este es el estándar que fue utilizado para implementar la solución descrita en este trabajo.

Kamel propuso algunos de los requerimientos y objetivos que se deben cumplir y proveer al interoperar con bases de datos heterogéneas [19]:

- Transparencia de distribución. El usuario debe poder acceder a un número de bases de datos diferentes de la misma forma que lo hace para acceder a una sola.
- Transparencia heterogénea. Los usuarios deben acceder a otros esquemas de la misma forma que acceden a la base de datos local (utilizando un modelo y lenguaje familiar).
- Los sistemas de base de datos y las aplicaciones existentes no deben requerir ser modificados.
- La adición de nuevas bases de datos debe lograrse fácilmente en el sistema.
- Las bases de datos se deben poder acceder para consultas y actualizaciones.
- El desempeño de los sistemas heterogéneos debe ser similar al de los sistemas distribuidos homogéneos.

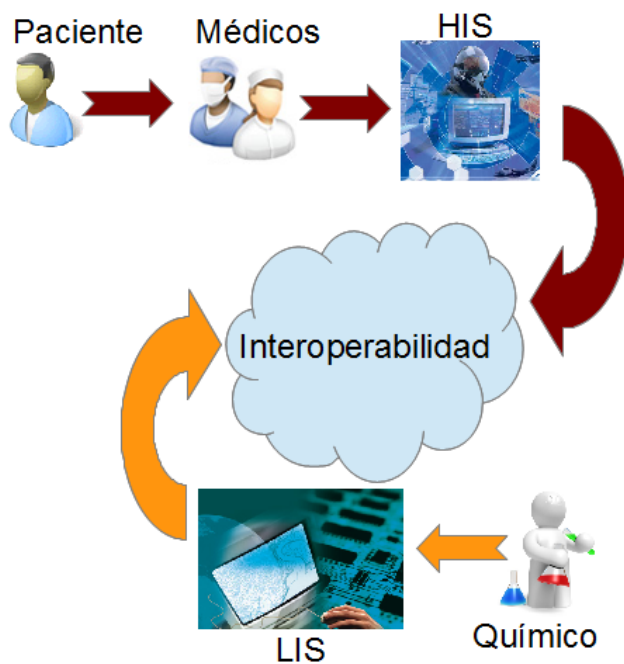


FIGURA 2.1: Interoperabilidad en los sistemas de salud

2.3. Estándares y la interoperabilidad

Originalmente estándar significaba (en el idioma inglés): bandera, color, pancarta; de allí el nombre estándar(te). El principal significado moderno que le siguió fue “lo que es establecido por la autoridad, la costumbre o el consentimiento general”. En este sentido se utiliza como sinónimo de norma, esto es, una serie de reglas y definiciones que especifican cómo llevar a cabo un proceso o producir un producto.

En 1987 un reporte técnico de la Organización Internacional de Estándares ¹ (ISO, por sus siglas en inglés) estableció que cualquier intercambio significativo de palabras depende de la existencia previa de acuerdo de reglas sintácticas y semánticas. En la informática de las ciencias de la salud, donde el énfasis está puesto en la manipulación y transmisión de información, los estándares son muy requeridos, pero todavía no alcanzan una gran madurez.

Eventualmente, la diversidad crea ineficiencia o impide efectividad, es entonces que se necesita emplear estándares. Por ejemplo, en el caso de la atención sanitaria, las instituciones de salud fueron diseñadas como unidades independientes donde los pacientes recibían atención médica en diferentes niveles (primero, segundo y tercero, esto es, dependiendo del grado de especialización requerido) sin ningún tipo de comunicación ni coordinación efectiva. La falta de acoplamiento entre los distintos escenarios y participantes del sistema genera ineficiencias en el cuidado de la salud. Se tiene una necesidad de integrar, consolidar y coordinar la información de los pacientes, incorporando técnicas organizacionales como indicadores de calidad y ofreciendo herramientas avanzadas como sistemas de apoyo a la toma de decisiones, lo cual exige la implementación de estándares.

Existen cuatro mecanismos básicos para el desarrollo de estándares:

- *Ad-hoc*: Es el caso en el que un grupo de personas u organizaciones interesadas se ponen de acuerdo en las especificaciones del estándar. Las especificaciones son informales pero aceptadas por las partes. Un ejemplo es el caso de cómo inició el estándar DICOM (*Digital Imaging and Communications in Medicine*) que fue producto del acuerdo de la ACR (*American College of Radiology*) y la NEMA (*National Electrical Manufacturers Association*).

¹Enlace al documento: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=16549, consultado el 6 de abril de 2014

- De facto: Esto ocurre cuando una empresa domina el mercado y marca lo que viene a ser el estándar. La empresa Microsoft es un buen ejemplo ya que algunos códigos de programación web son exclusivos del navegador Internet Explorer de Microsoft y sólo se visualizan correctamente con ese navegador, esto es, no son compatibles con otros navegadores.
- De Juri o Mandato Gubernamental: En este caso, una agencia del gobierno crea un estándar y legisla su uso. En México y otros países de América Latina un ejemplo podría ser el formulario de certificado de defunción.
- Consenso: Un grupo de voluntarios representantes de las partes interesadas en el estándar desarrollan el mismo en forma abierta. Éste es el caso de la creación del estándar HL7.

2.3.1. HL7

HL7 International [20] es una de las organizaciones más importantes en estándares de intercambio de información en sistemas de información de la salud. Surgió en Estados Unidos a finales de la década de los 80's como una iniciativa para lograr un HIS integrado a nivel nacional, interconectando distintos sistemas específicos. Se puede decir que la solución surgió a partir de una problemática de conectividad entre sistemas. Esta iniciativa se autodenominó Health Level 7. El nombre “*Level Seven*” hace referencia al nivel más alto del modelo de comunicación OSI (*Open Systems Interconnection*) de ISO (*International Standards Organization*), el nivel de aplicación. Este nivel indica la definición de los datos que se transmiten, los tiempos de transferencia y los avisos de ciertos errores que pueden ocurrir en las aplicaciones.

HL7 es un estándar internacional aprobado por la ANSI (*American National Standards Institute*), el cual tiene como objetivo principal crear una familia de estándares para el intercambio de información entre aplicaciones médicas que eviten la necesidad de programar interfaces ad-hoc y el mantenimiento de éstas. Con esa premisa se diseñó la familia de estándares que proporciona una sintaxis para la comunicación de mensajes entre aplicaciones informáticas (enfocadas a manejar información relativa a la salud) definiendo la forma de estructurar la información, principalmente por medio de mensajes estructurados.

La amplia difusión de los estándares creados han dado forma a lo que es hoy HL7: una organización con mas de 2300 miembros a nivel mundial con filiales en Canadá,

China, México, Alemania, India, Japón, Corea, Reino Unido, Argentina, Brasil, Chile, Uruguay, España, entre muchos otros.

El Capítulo HL7 México [21] es la organización sin fines de lucro que representa a *HL7 International* en México. Promueve el uso de estos estándares por medio de programas de difusión, capacitación y certificación. De esta manera se busca que los afiliados (organizaciones, estudiantes o cualquier persona interesada) apliquen los conocimientos sintetizados por cientos de expertos en proyectos de interoperabilidad, intercambio de información, expediente clínico electrónico, sistemas de imagen y laboratorio, entre otros. La NOM-024-SSA3-2012 [22] considera estándares HL7 para regular de los Sistemas de Información de Registro Electrónico para la Salud en México.

La NOM-024-SSA3-2012 [22] establece los criterios bajo los cuales se debe generar, procesar, conservar, interpretar y asegurar el intercambio de información entre Sistemas de Información de Registro Electrónico para la Salud, entre los que se encuentran los de Expediente Clínico Electrónico. También estipula los mecanismos mediante los cuales se emiten especificaciones técnicas para los posibles escenarios de intercambio y para el diseño de este tipo de sistemas. Para lo cual se basan en los estándares HL7.

Como organización, HL7 tiene una estructura y procedimientos formales basados en la búsqueda del consenso y el balance de intereses entre los distintos sectores que representa (empresas de software, financiadores de la salud, estados nacionales, universidades, prestadores de salud, consultores, etc.). Gracias a su Comité Internacional permite armonizar y discutir las necesidades locales del estándar y la adaptación en distintas partes del mundo, como en México.

HL7 se basa en la mensajería, utilizando un modelo de “eventos detonadores”, en el que el sistema emisor envía un mensaje específico al sistema receptor, con una subsiguiente respuesta por parte del receptor. HL7 define la estructura de cada uno de estos mensajes. En otras palabras, logra el acuerdo entre el emisor y el receptor.

La versión 1.0 de estos mensajes surgió en septiembre de 1987. Actualmente existen varios mensajes definidos según diferentes “eventos detonadores”. Estos eventos fueron multiplicándose con el correr de los años y al llegar a la versión 2.3, en marzo de 1997, el espectro de eventos se había expandido, proporcionando una sintaxis estándar para intercambiar información relacionada con:

- Información administrativa (admisión, pases de cama, de servicio, altas).

- Información financiera (gastos, facturación, proveedores).
- Prescripciones electrónicas y datos clínicos.
- Gestión de información médica.
- Gestión de calendario (citas o turnos).
- Interconsultas.
- Otras funciones permitían comunicar registros orientados a problemas, así como denuncias de efectos adversos, informes de inmunizaciones y otras funciones de registros de pacientes

Desde 1994 la organización HL7 está acreditada como SDO (Standards Developing Organization) por el ANSI (American National Standards Institute), acreditando luego cada una de sus versiones como estándar nacional. La versión HL7 v2.4 fue aceptada como un estándar del ANSI en octubre de 2000, incluyendo mensajes para eventos de laboratorio, gestión de aplicaciones y de personal. La versión HL7 más implementada es la HL7 v2.5 y está ampliamente soportada por las aplicaciones de los HIS actuales (compatibles con HL7 v2.x, claro). Utiliza mensajes en código ASCII puro y un texto plano sin formato (no estructurado, como los son los documentos XML). Esta versión tiene pocas restricciones semánticas, es decir que no especifica qué tipo de información debe contener cada parte de un mensaje, esto es, no define el modelo subyacente y el vocabulario de forma tan estricta como la versión HL7 v3. Ésta es una de las principales diferencias entre HL7 versión 2.x (2.x significa cualquier versión 2) y la versión 3.

Como lo menciona Cuenca et al. [9] Al utilizar el estándar HL7 los sistemas de información de la salud pueden alargar su vida útil, ya que tendrían la capacidad de integrarse con sistemas de información de otras entidades nacionales e internacionales.

2.3.2. HL7v2.x

La segunda versión del estándar HL7 transformó la industria de la interoperabilidad en salud, iniciando en los Estados Unidos desde los años 90's y se hizo popular a nivel mundial siendo adoptado como un estándar avalado por el *American National Standards Institute* (ANSI) [9]. Las primeras implementaciones se realizaron con la

versión 2.1. Para la versión 2.5 se mandó por la legislación de Estados Unidos darle un uso significativo. La 2.8 es la última actualización de esta segunda versión.

La versión 2.x se basa en cadenas de caracteres para transportar la información, lo cual hace sencilla su implementación, ya que fue desarrollado de una forma ad-hoc, pero con varias limitaciones, como es la validación de los datos que contiene y carece de una estructura orientada a objetos. Esta versión tampoco define la forma en que se deben enviar los mensajes, si debe ser en forma directa a través de TCP/IP, a través de archivos, a través de un puerto serial o a través de servicios web). La versión 2.x “recomienda” algunas opciones al respecto, pero ninguna de ellas es una normativa definida en el estándar.

HL7 v2.x está organizado en capítulos, donde cada uno de éstos tiene un propósito específico. Por ejemplo, el capítulo 2 define cómo se codifican y decodifican los mensajes. Cada capítulo representa y resume el trabajo de un comité técnico, formado por miembros de HL7 interesados sobre un área del sistema de información en salud, en la cual crean un grupo de trabajo. El producto final del grupo, es un capítulo con la información requerida para implementar mensajes referidos al área en cuestión y referencias al contenido ya definido en otros capítulos. Por ejemplo, en el capítulo de “Agendamiento - Turnos” uno de los segmentos que se utilizan es el de pacientes, que no está especificado dentro del capítulo de agendamiento sino en el capítulo de ADT (Admisión Discharge Transfer). El estándar entonces, en esa área, especifica el contenido propio y específico del manejo de agendas, como consultar turnos disponibles etc.

Para poder usar el estándar debe conocerse profundamente el capítulo 2 (Control) de la versión, lo que permitirá entender con facilidad el resto de los capítulos, ya que el resto de los capítulos contiene elementos requeridos del capítulo 2 para armar mensajes referidos a un área específica.

Una descripción paso a paso, de modo simplificado, de la trayectoria de un mensaje es la siguiente:

- Un evento disparador hace que el Sistema 1 envíe un mensaje al Sistema 2.
- El mensaje es recibido por el Sistema 2, el cual envía respuesta al Sistema 1.
- Finalmente, el Sistema 1 recibe la respuesta.

Viéndolo con un ejemplo real: Al internarse un paciente en el hospital, el sistema de admisión (que se encarga de administrar las camas del hospital) envía un mensaje al sistema de facturación para dar a conocer la ubicación del paciente recién internado y solicitar la apertura de una hoja de facturación de ese episodio. Este mensaje es recibido por el sistema de facturación y le envía un mensaje confirmando el arribo e indicando el número de cuenta corriente asignado al paciente.

2.3.3. Definiciones de HL7

- **Mensaje:** Un mensaje es toda la unidad de datos transferido entre sistemas en una sola transmisión. Es una serie de segmentos en una secuencia definida, con un tipo de mensaje y un disparador de evento.
- **Segmento:** Un segmento es un grupo lógico de campos de datos. Los segmentos definidos en un mensaje pueden ser requeridos u opcionales, puede aparecer una vez o tal vez sea permitido que se repitan. Cada segmento tiene nombre y es identificado por un ID de segmento, un código único de 3 caracteres.
- **Campo:** Un campo es una cadena de caracteres. Cada campo se identifica por el segmento en el que se encuentra y su posición dentro del segmento, por ejemplo PID-5 es el quinto campo del segmento PID. Campos de datos opcionales pueden ser omitidos. Cuando un campo es requerido, opcional o condicional en un segmento, se especifica en las tablas de atributos de segmentos. Los indicadores son: R=Requerido, O=Opcional y C=Condicional en el disparador de evento o en algún otro campo(s). La definición del campo debe definir cualquier condicionante para el campo: X=No se usa en este disparador de evento, B=Permanece para permitir compatibilidad con versiones previas de HL7. El tamaño máximo del campo se especifica como información normativa. Al exceder ese tamaño especificado no se debe considerar un error.
- **Componente:** Un componente es uno de un grupo lógico de elementos que abarca los contenidos de un campo codificado o compuesto. En un campo que contiene varios componentes, no todos los componentes requieren tener algún valor.
- **Campos nulos y vacíos:** El valor nulo es transmitido como una marca de dos comillas dobles (“”). Un campo con valor nulo es diferente a un campo vacío. Un campo vacío no debe sobrescribir datos previamente escritos en el campo,

mientras que el valor nulo significa que cualquier valor previo en el campo debe ser sobrescrito.

- **Tipo de dato:** Un tipo de dato restringe el contenido y formato de los datos del campo. Los tipos de datos se identifican por un código de dos o tres letras. Algunos tipos de datos son codificados o tipos compuestos con muchos componentes. El tipo de datos aplicable es listado y definido en cada definición de campo.
- **Delimitadores:** Los valores delimitadores se especifican en MSH-2 y son utilizados en el resto del mensaje. Las aplicaciones deben usar este delimitador especificado para analizar el mensaje completo. Los delimitadores recomendados para inmunizar los mensajes son: <CR>=Finalizador de segmento; |=Separador de campos; ^=Separador de componentes; &=separador de subcomponentes; ~=Separador de repeticiones; y \= carácter de escape.
- **Sintaxis del mensaje:** Cada mensaje es definido en una notación especial que enumera los identificadores de tres letras de los segmentos, en el orden en que aparecerán en el mensaje. Las llaves, {}, indican que uno o más segmentos que se encuentran dentro de las llaves pueden repetirse, y los corchetes, [], indican que el grupo de segmentos que encierran es opcional.
- **Segmentos Z:** Todos los tipos de mensaje, códigos de disparadores de eventos, y códigos de ID de segmentos que comienzan con Z son reservados para mensajes definidos localmente. Estos códigos no pertenecen al estándar HL7.

2.3.4. Mensajes HL7

Los mensajes HL7 son definidos como la unidad atómica de datos transferida entre sistemas [23]. Se componen de un grupo de segmentos en una secuencia definida. Estos mensajes especificados en el estándar, indican cuáles son los datos necesarios para que la comunicación sea correcta y suficiente según el tipo de información que van a contener. Cada mensaje contiene un código de tres caracteres que identifica el tipo de mensaje, el cual define el propósito de éste.

El protocolo HL7 representa un gran avance en la implementación de soluciones de interoperabilidad de sistemas en el ámbito de la salud. La versión 2.x establece el modelo de datos al que se deben ajustar los mensajes que sirven para comunicar

los diferentes sistemas de información en el marco de la salud. Este protocolo es un estándar internacional que trata de abarcar todos los requerimientos de intercambio de información que puedan surgir a nivel mundial, pero obviamente no abarca la totalidad de ellos y tampoco puede cubrir todas las particularidades que se dan en todos los países.

2.3.5. Reglas básicas de construcción de mensajes

A continuación se presentan algunas reglas o recomendaciones que comúnmente se emplean para la codificación de mensajes que serán enviados utilizando la versión 2.x de HL7:

- Codificar cada segmento en el orden especificado en el formato de mensaje abstracto.
- Colocar el segmento ID al inicio del segmento.
- Preceder cada campo de datos con un separador de campo.
- Codificar los campos de datos en el orden y tipo de datos especificados en la tabla de definición de segmentos.
- Terminar cada segmento con un finalizador de segmento.
- Componentes, subcomponentes o repeticiones que no tienen valor al final de un campo no necesitan ser representados por separadores de componentes.

Las siguientes son algunas recomendaciones para codificar mensajes recibidos:

- Si un segmento de datos que debería existir no aparece, tratarlo como si todos los campos de datos que incluye no estuvieran presentes.
- Si un segmento de datos aparece pero no debería existir, ignorarlo ya que no es un error.
- Si al final del segmento de datos se encuentran campos de datos que no deberían aparecer, ignorarlos; esto no es un error.

2.3.6. Object Identifier (OID)

Un OID es un identificador global único de ISO (*International Organization for Standardization*). Existen múltiples maneras para representar este identificador. En HL7 se ha establecido representarlo por la forma que consiste de únicamente números y puntos, por ejemplo "2.16.840.1.113883.3.1". Los OIDs son caminos en una estructura de árbol, en la que el primer número de la izquierda representa la raíz y el último número de la derecha representa una hoja.

2.3.7. Modelo de Información de Referencia (RIM)

Modelo fundamental para especificar la construcción de mensajes HL7. De este modelo se derivan todos los demás modelos de información y especificaciones de mensajes. Describe potencialmente cualquier dominio del ámbito sanitario.

2.3.8. HL7v3

Hablar de un sólo estándar, en realidad es muy genérico, actualmente HL7 y la versión 3, son una familia de estándares. Algunos de estos productos son:

- *HL7 V3 Messaging*: Es un estándar de mensajería, el reemplazo de la versión 2.x.
- *HL7 CDA: Clinical document architecture*. Estándar de marcado de documentos clínicos, que trataremos algunos párrafos más adelante.
- *HL7 SPL: Structured Product Labeling*: Estándar de marcado para etiquetado de medicamentos.
- *GELLO*: Lenguaje estándar para expresión de reglas para soporte de decisiones clínicas.

Con este conjunto de herramientas y metodologías HL7 genera un proceso para definir todo tipo de artefactos de interoperabilidad (mensajes, documentos, reglas) en lugar de solamente, como en la era de v2.x, los mensajes. La justificación de la nueva versión, surge como respuesta de la creciente utilización en EEUU y el resto del mundo de la versión 2 y los problemas que derivaron del uso masivo, los intentos de localización

(adaptar el estándar a las necesidades de cada país), y el intento de proyectar el éxito de la V2.x a instalaciones regionales o nacionales, en las que puedan interactuar más de dos sistemas a la vez. Además, la versión 3 de HL7 pone el foco en la semántica de los mensajes, reconociendo la tendencia global y la necesidad creciente de generar sistemas de información integrados regionalmente (ciudades, provincias, países). De este modo incorpora definiciones estándares de servicios requeridos para implementar este tipo de interoperabilidad compleja.

La versión 3.0 del estándar está orientada a objetos (un paradigma de programación) y basa su estructura en un RIM (*Reference Information Model*), desarrollado por HL7. El RIM, es un modelo de información de referencia que define las entidades involucradas en los mensajes, las personas, instituciones, procedimientos, etc. [24]. Actualmente no es un modelo completo del sistema de salud pero se perfila a serlo, ya que intenta agrupar todas las áreas y escenarios posibles, casos de uso, actores, eventos, interacciones, etc., de tal forma que represente toda la información para especificar un mensaje HL7. Esto transforma el RIM en algo más que un estándar de intercambio de información, ya que busca lograr no sólo interoperabilidad semántica, sino también la representación terminológica del contenido (en vocabularios de referencia) y no sólo su intercambio [12]. EL RIM fue aprobado como un estándar del ANSI en 2003 [25].

El RIM ha sido utilizado o tenido en cuenta en varias iniciativas de sistemas de información en salud (tanto comerciales como locales) para desarrollar sus modelo de datos. EL RIM de HL7 V3 y sus metodologías asociadas promueven y facilitan la interoperabilidad semántica, al definir cada elemento de datos y su relación con los otros (modelo semántico).

Un cambio importante en la versión 3 es el uso de XML (*Extensible Markup Language*). A diferencia de la versión 2.x, donde los mensajes se implementan en simple ASCII con tabulaciones, el HL7v3 usa para expresar los mensajes el estándar XML y sus derivados, dando mucha mayor versatilidad a los mismos.

2.4. LOINC

El catálogo LOINC (*Logical Observation Identifiers Names and Codes*) es una base de datos con nombres y códigos estandarizados para la identificación de resultados de laboratorio, observaciones clínicas y observaciones de estudios diagnósticos [26].

Su propósito es facilitar el intercambio y almacenamiento de resultados, como la hemoglobina en sangre, potasio en suero, o signos vitales, para el cuidado clínico, administración de resultados e investigación. Fue desarrollada por la universidad de Indiana. Cada registro corresponde a un solo resultado o prueba [27]. El registro incluye los siguientes campos ²:

- Componente o analito, como: hemoglobina o potasio.
- Característica medida, por ejemplo: una concentración total, actividad enzimática.
- Medición del tiempo, por ejemplo: si la medida es una observación en un momento del tiempo, o una observación integrada sobre una duración extendida del tiempo (orina de 24 horas).
- El tipo de muestra, por ejemplo: orina; sangre.
- El tipo de escala, por ejemplo: si la medida es el ordinal cuantitativo (de una medida verdadera), nominal (por ejemplo: E. coli; Estafilococo áureo), o narrativa (por ejemplo: narración de los resultados de radiografías).
- Cuando sea relevante, el método usado para producir el resultado u otra observación.

2.5. SNOMED-CT

SNOMED-CT es un conjunto de términos médicos estandarizados que provee códigos, términos, sinónimos y definiciones utilizadas en los documentos clínicos. Utilizado estos términos estandarizados se aporta precisión al representar fenómenos clínicos y mejora la interoperabilidad semántica de sistemas heterogéneos, al permitir la diversidad lingüística. Desarrollado inicialmente por el Colegio Americano de Patólogos (CAP), incluye términos de todo el espectro de la medicina, incluso de veterinaria.

SNOMED CT es el vocabulario más rico que existe para describir hallazgos clínicos, enfermedades, procedimientos, sustancias, etc. Sus características principales son:

- Enfoque composicional.

²Información obtenida de la guía de usuario LOINC, disponible en: <https://loinc.org/international/spanish/loincmanualespanol-02-09-2004.pdf>, consultado el 7 de abril de 2014

- Funcionalidades de vocabulario de interfase y de referencia.

El enfoque composicional se refiere a que permite la combinación de términos (pulmón + inflamación), o que se agreguen modificadores a un concepto (severo, leve, inicio brusco, etc.).

SNOMED CT es una excelente nomenclatura para utilizar como vocabulario de referencia, por su nivel de detalle y calidad de las relaciones semánticas. Pero, además, SNOMED CT incluye funcionalidades de interfase, es decir que para cada concepto tiene varias descripciones posibles, que pueden ser utilizadas como vocabulario de usuarios (también llamados de interfase) en una implementación.

Una de las características fundamentales de SNOMED CT es que no se limita a proveer una lista válida de términos, sino que cada término está definido por sus relaciones con el resto, y estas relaciones son entendibles por un sistema informático. Estas son algunas relaciones posibles que se utilizan para describir enfermedades o hallazgos clínicos: Sitio del Hallazgo, Asociado con, Después, Agente causal, Debido a, Morfología Asociada, Severidad, Curso, Episodicidad, Proceso patológico, Etiología, Hallazgo, Seguido de, etc. Por ejemplo, dentro de SNOMED CT hay una relación entre el concepto “Diabetes” y el concepto “Pié diabético”, mediante un atributo “Debido a” (“Pié diabético” - “Debido a” - “Diabetes”). Estas relaciones semánticas brindan información para que los sistemas puedan “razonar” al aplicar sistemas de soporte de la decisión.

2.6. NEMA

La Asociación Nacional de Fabricantes Eléctricos (NEMA, por sus siglas en inglés) ha desarrollado una familia de estándares que tienen la finalidad de regular la protección de los equipos eléctricos y la comunicación de datos médicos. Esta asociación fue fundada en 1926 y tiene su sede en Virginia, Estados Unidos, pero también cuenta con oficinas en Beijing y en la Ciudad de México ³.

En esta familia de estándares existe un conjunto de éstos que se enfocan en las condiciones operativas de los equipos, de manera que los usuarios puedan conocer el grado de protección mecánica con el que cuentan y así seleccionar el que mejor se ajuste a sus necesidades de acuerdo al ambiente en el que va a operar el equipo.

³Fuente: <https://www.nema.org/>, consultado el 6 de abril de 2014

La NEMA también tiene una división de imágenes diagnósticas y sistemas de terapia, la cual es líder en creación de estándares para imágenes médicas, como son rayos X, medicina nuclear, tomografía axial computarizada (CAT, por sus siglas en inglés), imagen de resonancia magnética (MRI, por sus siglas en inglés) y ultrasonido. Esta división publica los estándares internacionales de Imágenes Digitales y Comunicaciones en Medicina (DICOM, por sus siglas en inglés), que define el formato para que las imágenes médicas digitales puedan ser intercambiadas con la calidad y los datos necesarios para el uso clínico, como puede ser el nombre del paciente, instrumento utilizado, motivo del procedimiento, etc. Permite a los usuarios adquirir, desplegar, almacenar, recuperar, mover o imprimir imágenes digitales entre diferentes instrumentos, computadoras o incluso hospitales. Desde su primera publicación, en 1993, DICOM ha revolucionado la práctica de la radiología, permitiendo que sean reemplazadas las películas de rayos X con un conjunto de datos totalmente digital [28].

2.7. XML

XML es la abreviación de lenguaje de marcado extensible (*Extensible Markup Language*). Este lenguaje fue diseñado por el consorcio W3C [29] con el objetivo principal de tener un medio universal para compartir y transportar datos entre diferentes aplicaciones y plataformas. Es un lenguaje de marcas como HTML, pero a diferencia de éste los programadores crean sus propias etiquetas personalizadas, permitiendo así definición, transmisión, validación e interpretación de datos entre aplicaciones y entre organizaciones.

El lenguaje XML describe una clase de objetos conocidos como documentos XML. En éstos documentos se almacenan datos de manera estructurada, sin especificar ningún formato de visualización; las aplicaciones son las que, al utilizar los documentos XML proporcionan la presentación de los datos que se encuentran en éstos. De esta manera se mantiene la estructura de los datos sin que tenga alteraciones por la presentación que se les otorgue en la aplicación.

Un documento XML correctamente formado es aquel que cumple con las reglas de sintaxis del lenguaje, que son:

- Todo documento XML debe tener un elemento raíz.

- Las etiquetas XML se inician con el carácter “<” y terminan con el carácter “>”, por ejemplo: “<elemento>”.
- Las etiquetas de cierre se inician con los caracteres “</” terminan con el carácter “>”, por ejemplo: “</elemento>”.
- Todos los elementos XML deben tener su etiqueta de cierre, excepto los elementos vacíos, los cuales inician con el carácter “<” y terminan con los caracteres “/>”, por ejemplo: “<elemento atributo=“valorAtributo”/>”.
- Las etiquetas son sensibles a mayúsculas. La etiqueta “<elemento>” es diferente a la etiqueta “<Elemento>”.
- Los elementos XML deben estar anidados apropiadamente, incluyendo sus etiquetas de cierre.
- Los valores de los atributos deben escribirse entre comillas.
- El uso de los caracteres “<” y “&” está reservado para el lenguaje. Para utilizarlos en los datos o texto del documento deben sustituirse por su referencia de entidad (“<” y “&”, respectivamente).
- Los comentarios se escriben igual que en HTML, entre los caracteres “<!--” y “-->”.
- Los espacios en blanco se respetan, no se truncan como suceden en HTML.

Algo que hace muy poderoso a este lenguaje es la gran aceptación y uso que ha tenido a nivel internacional. Muchos programadores y empresas lo han utilizado satisfactoriamente para crear interfaces para bases de datos, programación, aplicaciones de oficina, aplicaciones para móviles entre otros. Debido al amplio uso que se le ha dado existen herramientas que ayudan a transferir los datos de diferentes tipos de plataformas a un documento en XML y también en la dirección inversa, es decir, para incorporar los datos desde un XML hacia una aplicación. En el caso del lenguaje Java existe la herramienta XJC [30], la cual es parte del kit de desarrollo de Java (*Java Development Kit* [31]), que realiza la creación automática de clases correspondientes a la estructura del documento XML que se le indique, mediante un archivo XSD.

2.7.1. XSD

Una ventaja de los documentos XML es que, además de que permiten corroborar que estén correctamente contruidos, también es posible validarlos contra un archivo que pone las reglas para la estructura que debe tener en un cierto entorno y así determinar si cumple con los requisitos establecidos. Estos archivos de validación, llamados XSD, contienen las indicaciones de los elementos que pueden contener los XML, cuántas veces se pueden repetir, sus atributos e incluso el orden en que deben aparecer. Con estas validaciones podemos saber cuando un documento XML tiene la estructura y los datos que esperamos recibir o enviar. Esto es en especialmente útil para que los mensajes cumplan con la estructura establecida por HL7, ya que la tercera versión de este estándar especifica la estructura de los mensajes mediante el uso de documentos XML y su correspondiente XSD.

2.7.2. Espacios de nombres

Internet es un espacio donde se encuentran muchos puntos de información. Las referencias URI (*Uniform Resource Identifier*) nos permiten identificar a cada uno de estos puntos de manera única ⁴. En los documentos XML se emplean los espacios de nombres para clasificar los nombres de elementos y atributos utilizados en ellos. Estos espacios de nombres son conjuntos de nombres, identificados por una referencia URI, que se utilizan en documentos XML como tipos de elemento y nombres de atributo.

2.8. Servicios Web

Un Servicio Web (WS, por sus siglas en inglés) permite el intercambio de información entre sistemas brindando una interfaz de servicio sin que cada uno de éstos tenga que conocer detalles de aquellos con los que se está comunicando, es decir, permite la independencia de plataformas y de aplicaciones. Estos servicios se ofrecen mediante un servidor Web a otros sistemas que solicitan consumirlos mediante los protocolos Web. Los WSs utilizan el lenguaje XML como el mecanismo de estandarización adoptado para la codificación e intercambio de datos. También emplean otros estándares como son: SOAP (*Simple Object Access Protocol*, para establecer el

⁴Fuente: <http://www.w3.org/Addressing/>, consultado el 6 de abril de 2014

intercambio), HTTP (*Hypertext Transfer Protocol*, para la transferencia de los datos), WSDL (*Web Services Description Language*, contiene la descripción de los requisitos funcionales necesarios para establecer una comunicación), WS-Security (*Web Service Security*, protocolo de seguridad), entre otros. Estos estándares fueron ocupados en la implementación de los casos de estudio presentados; XML para almacenar los datos del mensaje que se envía, SOAP para implementar el servicio Web en el servidor, WSDL para establecer las funcionalidades del WS disponibles.

Capítulo 3

Expediente Clínico Electrónico y la situación actual en México

3.1. Expediente Clínico Electrónico

El expediente clínico es una herramienta fundamental para la práctica del médico en los ámbitos clínico, educativo, de investigación, administrativo, estadístico, legal e histórico. Incluye el relato patográfico individual del paciente, es decir, el registro de la situación real de salud del paciente a lo largo de un período, incluyendo las etapas y resultados de los estudios realizados. A su vez integra los datos necesarios para formular los diagnósticos y establecer un tratamiento para el paciente y consigna la evolución del caso hasta su solución parcial o total. Para esto se utiliza un conjunto de documentos escritos, gráficos e imagenológicos o de cualquier otra índole que se integran por diferentes secciones donde el personal de salud registra, anota y certifica las acciones correspondientes a su intervención.

Los expedientes clínicos recopilan la información necesaria para la atención a la salud de los pacientes. Para realizar la atención se precisa información de los pacientes en las áreas preventiva, médica, administrativa y social. El Expediente Clínico Electrónico (ECE) es el medio electrónico en el cual el personal de salud registra el expediente clínico de los pacientes.

Con el avance de la medicina y la incorporación de las tecnologías de la información y comunicación se inició el uso de ECE. Fue Weed que desde 1967 [32], paralelamente

con el desarrollo de su modelo de expediente clínico orientado en problemas, conceptualizó el ECE o expediente clínico computarizado, y hasta el día de hoy múltiples esfuerzos tanto públicos, como privados y de software libre buscan desarrollar un ECE.

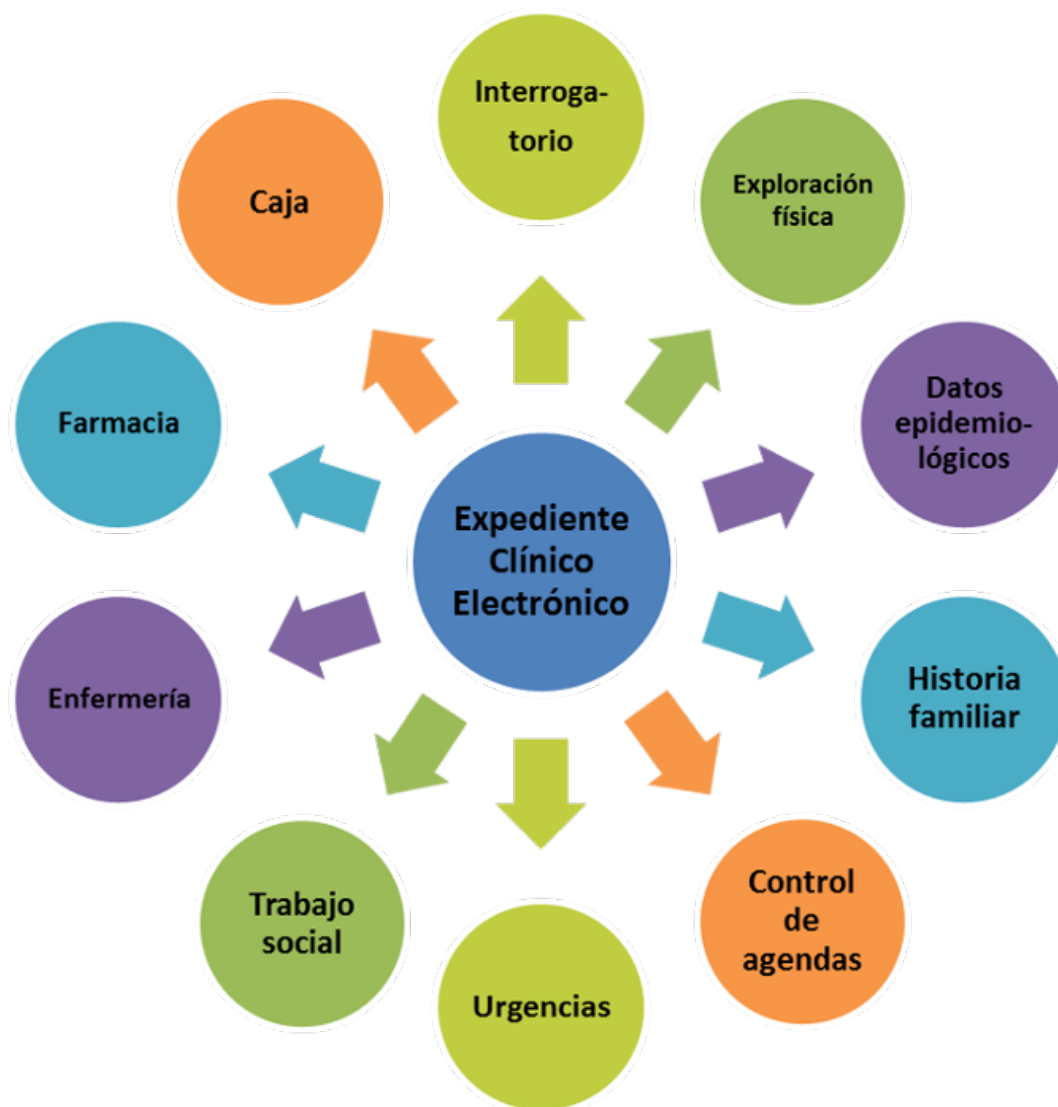


FIGURA 3.1: Diferentes módulos del ECE

El no disponer de la información clínica esencial en el momento en que se necesite es una de las principales fuentes de errores de los profesionales de la salud. Éstos se podrían prevenir, según Salgado [33], con el acceso a una historia clínica única disponible en línea para su consulta en cualquier nivel de atención del sector.

3.2. Historia de Salud Personal

La Historia de Salud Personal (PHR, por sus siglas en inglés) es una aplicación electrónica que permite a las personas acceder, administrar y compartir su propia información de salud con aquellos a quienes se les concede autorización, en un entorno privado, seguro y confidencial. Es importante mencionar que este repositorio de información es mantenido por el paciente, a diferencia del ECE que es mantenido por las instituciones de salud, como son los hospitales. El PHR otorga la responsabilidad, el control y pleno acceso al contenido a la persona titular, así como la posibilidad de compartirlo con quien requiera. La información de un PHR puede incluir datos de signos o síntomas reportados por el paciente, resultados de laboratorio, datos de dispositivos electrónicos como monitores de signos vitales, horarios de medicamentos, etc.

Los PHR abren la posibilidad de que una persona porte su historia clínica a donde quiera que vaya, siempre y cuando la transporte consigo misma. Esto podría ser poco práctico si estamos pensando en medios físicos como papeles o dispositivos electrónicos que puedan ser olvidados o que ocupen un espacio importante que represente una carga para la persona. Esto pronto podrá ser una realidad debido a los avances tecnológicos que se han dado. Ahora es posible almacenar una gran cantidad de información en pequeños dispositivos que podrían ser casi imperceptibles a simple vista. En el momento en el que se permita implementar pequeños chips en el organismo de una persona será posible llevar a todos lados el PHR sin ningún impedimento de portabilidad. Esto daría lugar a que en cualquier lugar se tenga disponible el PHR de las personas y así utilizarlo en el momento que se requiera, beneficiando enormemente la atención médica que pueda necesitarse en una emergencia.

3.3. Situación actual del ECE en México

En México sigue predominando el expediente clínico en papel, y generalmente presenta algunas desventajas como: mala legibilidad, desorganización especialmente en pacientes con enfermedades múltiples o crónicas, un gran consumo de papelería y espacio para el archivo clínico. La información clínica esencial no se encuentra disponible para el personal médico, y en algunas ocasiones es la fuente principal de errores médicos que pueden ser prevenidos con información clínica accesible y precisa obtenida en los expedientes clínicos [22]. Otras limitantes son que sólo puede ser

utilizado por una persona a la vez, generalmente está incompleto, algunos reportes se pueden perder con relativa facilidad, la información no se comparte entre distintas instituciones o entre los diferentes niveles de atención de una misma institución y es inseguro porque no se puede rastrear donde estuvo, quien lo consultó y modificó o bien copiar y plagiar.

En la actualidad muchas escuelas y facultades de Medicina, así como instituciones de salud en México, carecen de un sistema de ECE. Según la NOM-024-SSA3-2012 [22] el ECE es el medio en el cual el personal de salud, registra, anota y certifica su intervención, relacionada con el paciente con arreglo a las disposiciones sanitarias. Éste permite la gestión de un único registro de salud longitudinal de cada paciente en un formato digital. Se ha demostrado que el uso del ECE favorece la calidad de la atención del paciente diabético [34].

El sistema de salud en México funciona como un ente granulado, de tal manera que aún dentro de un mismo hospital, el intercambio de información entre los diferentes departamentos existe sólo en papel, aunque existan sistemas electrónicos o de informática en los respectivos departamentos [4].

Existe en el país una Norma Oficial referente al Expediente Clínico (NOM-168-SSA1-1998), la cual fue publicada en el año de 1998 y después, en el año 2003 se modificó para que incluyera y validara la posibilidad de la existencia de un ECE. Por lo tanto, como lo afirma Vázquez et al. [4] esta norma debe ser la base para la creación de un ECE estándar para todo México. Más recientemente se publicó la norma NOM-024-SSA3-2012 [22],

Según un reporte de Notimex en enero de 2013, muchas de las clínicas y hospitales de México todavía utilizaban máquinas de escribir para elaborar los expedientes clínicos ¹. Esto es de llamar la atención, pero existe una gran diferencia en las tecnologías a las que tienen acceso las diferentes instituciones que conforman el sistema de salud del país.

Se han realizado esfuerzos por desarrollar ECE por diversas instituciones. Cada una ha tenido buenos avances a nivel institucional, se están instalando en muchas clínicas del país y se está migrando la información del expediente clínico en papel al electrónico. Lo que hace falta y se requiere es que todos estos esfuerzos individuales puedan

¹Fuente: <http://sipse.com/mexico/sector-salud-completa-este-ano-expediente-clinico-electronico-12021.html>, consultado el 6 de abril de 2014

compartir información y poder operar en conjunto para que la atención al paciente sea más completa y eficiente.

Durante el sexenio comprendido entre 2006 y 2012 se diseñó un proyecto de ECE que pretendía que toda la gente pudiera ser atendida en cualquier instancia del sector salud sin necesidad de ser derechohabiente. Una especie de credencial que le permitiría a las personas tener acceso a su historia clínica, incluyendo tomografías, radiografías, cirugías, medicamentos y restricciones, entre otros datos. Según una entrevista realizada a Juan Carlos Reyes, director general de Información de la Secretaría de Salud, el proyecto fue “demasiado ambicioso” y “carente de bases”². El funcionario detalló en la entrevista que para lograr que algo de esa magnitud funcione se requiere primero adaptar la normatividad sobre el manejo de los datos, asignar más presupuesto y tener completo un padrón de beneficiarios detallado y confiable.

El funcionario mencionó que el proyecto fue frustrado de inmediato debido a que las instituciones, hospitales y las clínicas involucradas se declararon incompetentes para sumarse a este proyecto, ya que consideraban fuera de su alcance poder instalar la infraestructura necesaria que contendría toda esa información que además carecía de presupuesto y un sustento legal para intercambiar información sin cometer un delito o atentar contra los reglamentos internos. También indicó que para arrancar el proyecto es necesario contar con un padrón de beneficiarios y realizar modificaciones a las normas NOM-024 [22] y NOM-035 (que establece los criterios para obtener, integrar, organizar, procesar, analizar y difundir la Información en Salud) para garantizar el intercambio de cierta información. Una de las modificaciones que se ya se ha realizado es sobre el repositorio de los datos, ya que la información se alojaría en cada uno de los institutos u hospitales y no en un servidor central. A su vez, la Secretaría de Salud modificó la norma NOM-024 para que todo el sistema público utilizara como identificador de pacientes por medio de la CURP.

En cuanto a los plazos para continuar, indicó: “en 2013 o 2014 vamos a consolidar la base de datos geográficos, de 2014 a 2015 el padrón de los profesionales de la salud; 2015-2016 el despliegue de los ECEs”.

En una entrevista realizada por Notimex, Jaime Cater, director general de Health Digital System (HDS)³, comentó que considera que en el actual gobierno existe una

²Fuente: <http://sipse.com/mexico/proyecto-de-expediente-clinico-electronico-requiere-cirugia-31149.html>, consultado el 6 de abril de 2014

³<http://www.saludhds.com.mx/>, consultado el 7 de abril de 2014

política con miras a modernizar el Sistema Nacional de Salud, aunque lamentablemente muchas clínicas y hospitales del sector público, federal o estatal, utilizan todavía máquinas de escribir para elaborar los expedientes clínicos de los derechohabientes. Estos expedientes son almacenados en gavetas, lo cual genera mayores costos y una inadecuada protección de los datos. Aunque también admitió que existen diferencias significativas en los avances de ECE en las instituciones de salud. Mencionó que en PEMEX, el ISSSTE y el IMSS el desarrollo ha sido mayor ⁴.

Como antecedentes históricos, Gertrudiz [35] menciona que en el año 2005 PEMEX homologó y amplió la cobertura, funcionalidad y soporte de su Sistema Integral de Administración Hospitalaria, en 60 unidades médicas para el primer nivel de atención, en sus módulos de recepción y atención médica con generación de receta electrónica. Para el año 2006 había invertido \$55'200'000 para desarrollar su ECE institucional, con un costo total estimado de del proyecto de \$1'173'000'000.

El IMSS comenzó la operación del proyecto Hospital Digital en mayo de 2006, el cual integra electrónicamente los servicios hospitalarios, incluyendo las áreas críticas y de enfermería. A nivel de dirección, brinda la posibilidad de dar seguimiento a la productividad y actividad diaria del hospital, por medio de tableros de control. Este Hospital Digital representó una inversión de \$259'000'000 [35].

En el 2010 el ISSSTE inició un despliegue para implementar en todo el país el ECE y la operación automatizada de medición del desempeño en la prestación de los servicios de salud a través del Sistema Interno de Evaluación del Desempeño Institucional (SIEDI). Esta iniciativa continúa en desarrollo.

Entre el 2006 y el 2008, la Secretaría de la Defensa Nacional (SEDENA) desarrolló el proyecto del Sistema Informático de Administración Hospitalaria de la Clínica de Especialidades de la Mujer (SIAH-CEM). El sistema fue conformado por 32 módulos que dan soporte a la administración de citas de consulta externa, administración de medicamentos del almacén general, farmacia, salas de hospitalización, trámites administrativos y uso del expediente clínico.

Analizando la situación actual de México, existen muchas iniciativas para desarrollar un ECE en las diferentes instituciones de salud, cada una con sus recursos y obstáculos particulares. Estos proyectos están en desarrollo, pero también es evidente la necesidad de lograr la interoperabilidad entre todos ellos, y finalmente en todo

⁴<http://sipse.com/mexico/sector-salud-completa-este-ano-expediente-clinico-electronico-12021.html> , consultado el 6 de abril de 2014

el sistema de salud. Esto, como ya se expuso, lo han mencionado las autoridades competentes y están en planes para lograrlo. El presente trabajo propone una metodología que ayudaría a lograr implementar esta interoperabilidad que se persigue, y así, mejorar el servicio brindado por el sistema de salud, lo cual tiene miras a tener resultados en beneficio de la población.

Capítulo 4

Implementación

La implementación es una parte muy importante de este trabajo porque es la manera en que se va a aplicar el planteamiento teórico para solucionar un problema real. Este trabajo se implementó en la comunicación entre un HIS y un LIS que operaban por separado y ahora, mediante esta solución ya pueden intercambiar información sin tener que alterar su funcionamiento individual. La metodología propuesta está planificada para poder implementar la interoperabilidad entre sistemas de información que alojan sus datos en una base de datos, sin importar el sistema manejador que utilizan y tampoco la estructura de la base de datos. Los datos que se podrán transmitir son todos aquellos que residen dentro del manejador de bases de datos del sistema. Estos sistemas deben poseer un medio de comunicación permanente que les permita transmitir datos con el sistema al cual requieren compartir información.

4.1. Metodología de implementación

En esta sección presentaremos nuestra metodología para implementar la comunicación entre sistemas heterogéneos, la cual consiste de seis puntos. Primero enlistaremos estos puntos y en seguida detallaremos cada uno de ellos.

1. Especificar la información que se requiere intercambiar (información personal del paciente, historia clínica, resultados de laboratorio, imagenología, etc.).
2. Identificar el estándar de comunicación más apropiado, que contemple tanto la solicitud de información así como la respuesta que se debe obtener. En sistemas

médicos seleccionar el mensaje HL7 específico para la información que se desea intercambiar.

3. Realizar el emparejamiento de esquemas de las bases de datos y el mapeo de esquemas, estableciendo los elementos (opcionales y obligatorios) que serán parte del mensaje que será enviado con los datos.
4. Especificar la interfaz de intercambio de mensajes. Una opción recomendable por su gran potencialidad y sencillez de implementación son los WSs.
5. Desarrollar el código que genere los mensajes con la información extraída de la base de datos.
6. Implementar la interfaz de intercambio de mensajes especificada anteriormente e integrarla con el código que genera los mensajes en el sistema.

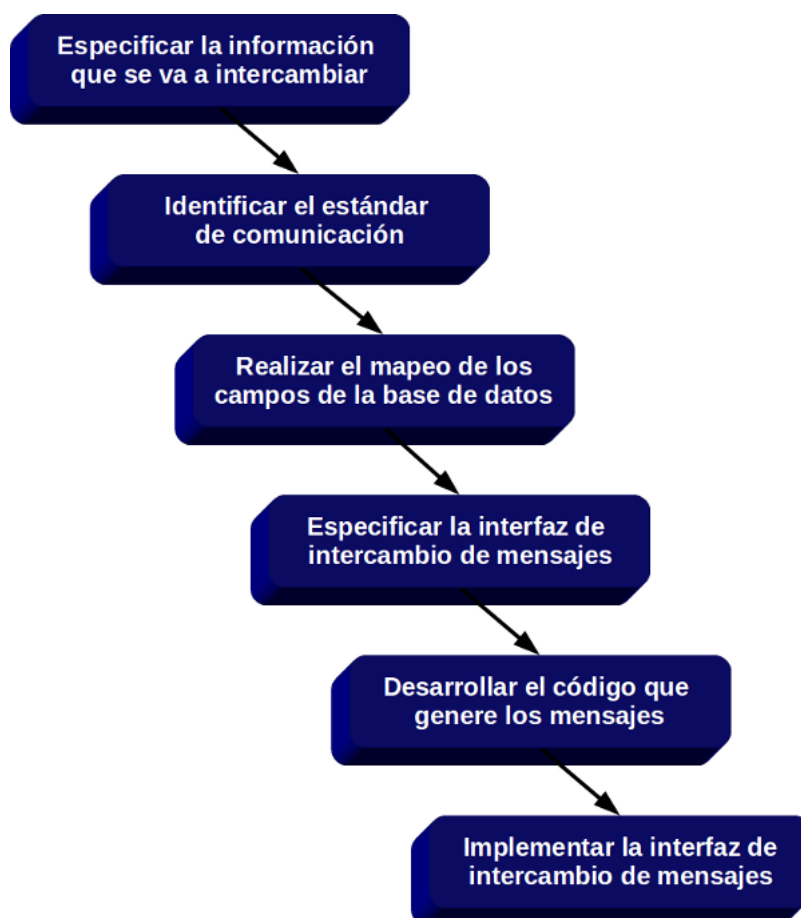


FIGURA 4.1: Metodología de implementación

4.1.1. Especificación de la información de intercambio

Se tiene que tener claro cuál es la información que se va a intercambiar con otros sistemas, no puede quedar abierto a compartir todos los datos para evitar que surjan problemas de seguridad y de eficiencia. Así que es necesario delimitar el dominio de los datos que se quieren compartir, por ejemplo la información personal de un paciente se puede limitar a sólo el CURP, nombre, sexo y fecha de nacimiento. Las características de esta información se deberán especificar en un documento contemplando tanto los campos de la base de datos como el tipo de datos y el rango de valores permitidos. Esto es crucial para lograr una correcta comunicación entre las bases de datos, ya que es posible que el tipo de datos no coincida en los diferentes sistemas y en ese caso sería necesaria una transformación de éstos en la etapa de generación de los mensajes.

4.1.2. Identificación del estándar

Para que exista comunicación se requiere que cada una de las partes puedan entender lo que se está comunicando, esto implica que sepan de qué contexto se trata, cuáles términos existen, cuáles son las reglas del lenguaje de comunicación, etc. Los estándares permiten tener claros todos estos conceptos y así garantizar que los involucrados pueden comprender y utilizar la información que se está intercambiando [10].

La identificación del estándar que vamos a utilizar tiene gran impacto en el alcance que tendrá la comunicación del sistema. Un estándar puede ser tan sencillo y simple como uno quiera (cada institución puede tener su propio estándar), pero tendrá que ser aceptado por todos los involucrados en la comunicación y su sencillez o complejidad dictaminará las posibilidades de comunicación entre los sistemas.

En el caso de sistemas de información de la salud, el estándar HL7 es una opción mundialmente reconocida [20], ya que contempla la gran mayoría de las áreas de la salud que son susceptibles de requerir intercambiar información entre diferentes sistemas, que es justamente el objetivo que se está buscando. Además, algo que hace que este estándar sea una alternativa viable es que es de acceso gratuito (a partir de abril de 2013 [36]), lo cual permite que su disponibilidad no esté limitada para cualquiera que lo desee implementar.

El estándar HL7 pretende abarcar todo el panorama de posibilidades de comunicación en el área de la salud, lo cual pone sobre la mesa un abanico amplio de tipos de

mensajes que podríamos utilizar. Aunado a esto, cada tipo de mensaje tiene muchos elementos que lo componen, algunos obligatorios y otros opcionales, es por esto que no es suficiente con especificar el estándar que se va a emplear para la comunicación, sino que también el tipo de mensaje que va transportar la información y los elementos que se van a ocupar dentro de éste.

4.1.3. Realizar mapeo de campos

Habiendo especificado un estándar, podemos garantizar que los datos que se envían tienen un significado para el sistema que los recibe, es decir, sabe qué es lo que está recibiendo. Teniendo esto claro, es necesario determinar cuáles son los campos equivalentes entre un sistema y otro, es decir, realizar el proceso de “emparejamiento de esquemas”. De esta manera el sistema emisor tendrá claro cuáles son los campos de la base de datos de los que se extraerá la información que será enviada, y el sistema receptor sabrá dónde ubicar los datos que recibe.

Posteriormente se debe realizar el “mapeo de esquemas”, el cual debe especificar la manera en la que se van a trasladar los datos de un sistema a otro. Aquí es donde se pueden emplear los mensajes HL7 para los sistemas de información de la salud. Existen muchos tipos de mensajes definidos en el estándar HL7, orientados a trasladar información médica. Se debe elegir el tipo de mensaje más adecuado, según la información que se requiere enviar, y emplearlo para generar los mensajes que contendrán los datos que viajarán de un sistema a otro. Por ejemplo, en la comunicación de un HIS y un LIS se pueden elegir los mensajes ORU (resultados de observación) para enviar los valores de los estudios de laboratorio registrados en el LIS.

4.1.4. Especificación de la interfaz de intercambio

La interfaz de intercambio será el medio por el cuál va a viajar la información. En este caso, la manera en que se va a transportar el mensaje desde el emisor hasta el receptor. Esta interfaz deberá satisfacer las necesidades de disponibilidad que se tengan, puede ser tan sencillo como una carpeta compartida con la que se adjunte un archivo con el mensaje del emisor para hacerlo llegar al sistema que lo solicita, pero esto no es práctico para resolver el problema que estamos atacando porque requiere de intervención manual.

Una opción recomendable por su potencialidad de comunicación y sencillez de implementación son los WSs. Algo importante que debe cumplir la interfaz de intercambio es que debe ser multiplataforma y para cualquier lenguaje de programación, características que cumplen los WSs. Con esto podemos disponer de un medio de transmisión que no se limitará a la plataforma en la que esté operando el sistema emisor, ni tampoco el lenguaje de programación empleado para implementar la interfaz.

Los WSs quedan publicados en un sistema y son utilizados o “consumidos” por otro. Lo más natural es hacer que el sistema que contiene la información que se va a transmitir sea quien publique el WS y aquel que recibe o solicita la información sea el que lo consume, aunque no es estrictamente de esta manera, ya que es posible hacerlo en el orden inverso haciendo ciertas adecuaciones en la implementación del WS. Debido a esto es necesario especificar claramente cómo va a ser implementado el WS.

La información que se envía a través de la interfaz de intercambio puede ser confidencial, es por eso que se requiere tener mecanismos de seguridad que permitan garantizar la privacidad de los datos que se envían. El consorcio W3C ha creado estándares para proveer integridad y confidencialidad a los documentos XML, como son el *XML Signature* y el *XML Encryption*; los cuales, según Nordbotten [37], son fundamentales para la seguridad en los XML y los WSs.

La programación de los WSs no es parte de los objetivos de este trabajo, es por eso que no se detalla una guía sobre cómo implementarlos en algún lenguaje de programación. Existen tutoriales disponibles para esto, en particular se recomienda consultar uno que provee la empresa Oracle en su página Web ¹.

4.1.5. Generar los mensajes

Habiendo especificado el tipos de mensaje que enviará la información y el medio por el cual va a ser enviado, es necesario crear el código que extraiga la información de la base de datos y la empaquete en el mensaje. La implementación de esta etapa será particular al sistema en el que se esté desarrollando, ya que depende de los esquemas de la base de datos y el lenguaje de programación que utiliza el sistema.

¹Tutorial de implementación de WSs con Java: <http://docs.oracle.com/javasee/6/tutorial/doc/gijti.html>, consultado el 6 de abril de 2014

En Java es recomendable generar una clase correspondiente a cada elemento del mensaje, así se puede crear un objeto con la información que debe contener el elemento. Al crear estas clases es necesario contemplar los elementos que pueden repetirse y los que solamente pueden aparecer una vez en el mensaje. Para lograr esto se puede utilizar la herramienta XJC, la cual es parte del JDK (*Java Development Kit* [31]). Esta herramienta recibe como entrada un archivo XSD y genera las clases correspondientes a cada uno de los elementos indicados en el XSD. A continuación se muestra un ejemplo de cómo se utilizaría esta herramienta utilizando la línea de comandos (en Windows 7):

```
xjc -p mx.nombre.paquete.java archivo.xsd
```

Este comando genera las clases de java correspondientes al archivo XSD (archivo.xsd). Las clases nuevas se crearán dentro del paquete indicado, en este caso el nombre es: “mx.nombre.paquete.java”.

4.1.6. Implementar la interfaz de intercambio

Teniendo el mensaje generado con la información que se va a enviar, es necesario colocarlo en el medio de transporte para que sea enviado. Eso es justamente lo que se debe lograr con la implementación de la interfaz de intercambio.

Utilizando un WS, el sistema que va a enviar la información deberá hacer uso del servicio según haya sido especificado anteriormente. En el caso del envío de una solicitud de estudios de laboratorio por parte de un HIS, éste “consume” un WS del LIS enviando el mensaje con la información. El LIS recibirá la orden y cuando tenga los resultados “consumirá” un WS del HIS para entregar el mensaje con los resultados.

4.2. Estructura de mensajes

4.2.1. Solicitud de laboratorio

Se propone una estructura para los mensajes de solicitud de estudios de laboratorio. En esta propuesta, el documento XSD primero define la estructura del elemento

padre, que incluye todos los elementos de la solicitud. Se puede observar que tiene 5 elementos: “paciente”, “médico”, “capturista”, “lugarAtencion” y “control”. El elemento “paciente” contiene los elementos necesarios para identificar al paciente, como son: “nombre”, “apellido paterno”, “apellido materno”, “cveIdee” (identificador único del paciente que puede ser la CURP), etc. Cada uno de estos elementos se especifican posteriormente en el mismo documento XSD. A continuación se expone el contenido del XSD que contiene esta propuesta. Se procuró insertar comentarios a efecto de resaltar su estructura y sus elementos:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<!-- Elemnto padre solicitudLab -->
  <xs:element name="solicitudLab">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="paciente" minOccurs="1" maxOccurs="1"/> <!-- Datos
          del paciente -->
        <xs:element ref="medico" minOccurs="1" maxOccurs="1"/> <!-- Datos
          del medico -->
        <xs:element ref="capturista" minOccurs="1" maxOccurs="1"/> <!--
          Datos de la persona que captura la informacion -->
        <xs:element ref="lugarAtencion" minOccurs="1" maxOccurs="1"/> <!--
          Datos del lugar donde se atiende al paciente -->
        <xs:element ref="control" minOccurs="1" maxOccurs="1"/> <!-- Datos
          de control y errores -->
      </xs:sequence>
    </xs:complexType>
  </xs:element>
<!-- Ahora se detallan los elementos hijos, iniciando por paciente: -->
  <xs:element name="paciente">
    <xs:annotation>
      <xs:documentation>Datos del paciente.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="origenPaciente" minOccurs="1" type="xs:string"> <
          /xs:element> <!-- Procedencia: consulta externa, urgencias,
          referido, etc. -->
        <xs:element name="fechaAtencion" minOccurs="1" type="xs:date"> </
          xs:element> <!-- Fecha en la que se atiende al paciente -->
```

```

    <xs:element name="cveIdee" minOccurs="1" type="xs:string"> </
      xs:element> <!-- Identificador unico del paciente -->
    <xs:element name="apellidoPaterno" minOccurs="1" type="xs:string">
      </xs:element> <!-- Apellido paterno -->
    <xs:element name="apellidoMaterno" minOccurs="1" type="xs:string">
      </xs:element> <!-- Apellido materno -->
    <xs:element name="nombre" minOccurs="1" type="xs:string"> </
      xs:element> <!-- Nombre(s) -->
  </xs:sequence>
</xs:complexType>
</xs:element>
<!-- En seguida se detallan los elementos de medico -->
<xs:element name="medico">
  <xs:annotation>
    <xs:documentation>Datos del Medico Tratante.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="cveEspecialidad" minOccurs="1" type="xs:string">
        </xs:element> <!-- Clave de la especialidad del medico -->
      <xs:element name="apellidoPaterno" minOccurs="1" type="xs:string">
        </xs:element> <!-- Apellido paterno -->
      <xs:element name="apellidoMaterno" minOccurs="1" type="xs:string">
        </xs:element> <!-- Apellido materno -->
      <xs:element name="nombre" minOccurs="1" type="xs:string"> </
        xs:element> <!-- Nombre(s) -->
      <xs:element name="cedula" minOccurs="1" type="xs:string"> </
        xs:element> <!-- cedula profesional -->
      <xs:element name="cveMatricula" minOccurs="1" type="xs:string"> </
        xs:element> <!-- clave de trabajador -->
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- Ahora los elementos de capturista -->
<xs:element name="capturista">
  <xs:annotation>
    <xs:documentation>Datos del capturista</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="apellidoPaterno" minOccurs="1" type="xs:string">
        </xs:element> <!-- Apellido paterno -->
      <xs:element name="apellidoMaterno" minOccurs="1" type="xs:string">
        </xs:element> <!-- Apellido materno -->
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

        <xs:element name="nombre" minOccurs="1" type="xs:string"> </
            xs:element> <!-- Nombre(s) -->
        <xs:element name="cveMatricula" minOccurs="1" type="xs:string"> </
            xs:element> <!-- clave de trabajador -->
    </xs:sequence>
</xs:complexType>
</xs:element>
<!-- Detalle de los elementos de solicitud -->
<xs:element name="solicitud">
    <xs:annotation>
        <xs:documentation>Datos de la solicitud. </xs:documentation>
    </xs:annotation>
    <xs:complexType>
        <xs:sequence>
            <xs:element name="cveTipoSolicitud" minOccurs="1" type="xs:string">
                </xs:element> <!-- clave para indicar si es solicitud nueva,
                    cancelacion o peticion de estatus -->
            <xs:element name="fechaOrden" minOccurs="1" type="xs:string"> </
                xs:element> <!-- fecha de realizacion de la solicitud -->
            <xs:element name="cveCie10" minOccurs="1" type="xs:string"> </
                xs:element> <!-- CIE-10 del diagnostico presuntivo -->
            <xs:element name="refComplementoDiagnostico" minOccurs="1" type="
                xs:string"> </xs:element> <!-- informacion complementaria al
                    diagnostico presuntivo -->
            <xs:element name="numFolioSolicitud" minOccurs="1" type="xs:string"
                > </xs:element> <!-- numero de folio de solicitud -->
            <xs:element name="cveEstudio" minOccurs="1" type="xs:string"> </
                xs:element> <!-- clave del catalogo LOIN del estudio solicitado
                    -->
            <xs:element name="cvePrueba" minOccurs="1" type="xs:string"> </
                xs:element> <!-- clave del catalogo LOIN de la prueba solicitada
                    -->
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- Elementos de lugarAtencion -->
<xs:element name="lugarAtencion">
    <xs:annotation>
        <xs:documentation>Datos del lugar de atencion. </xs:documentation>
    </xs:annotation>
    <xs:complexType>
        <xs:sequence>

```



```

    <xs:element name="cveLugarAtencion" minOccurs="1" type="xs:string">
      </xs:element> <!-- clave del lugar donde se expide la solicitud
      -->
    <xs:element name="cveTipoLugarAtencion" minOccurs="1" type="
      xs:string"> </xs:element> <!-- clave del tipo de lugar de
      atencion -->
    <xs:element name="cvePresupuestal" minOccurs="1" type="xs:string">
      </xs:element> <!-- clave presupuestal del lugar de atencion -->
  </xs:sequence>
</xs:complexType>
</xs:element>
<!-- Finalmente los elementos de control -->
<xs:element name="control">
  <xs:annotation>
    <xs:documentation>Datos de control de la solicitud. </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="fechaTransaccion" minOccurs="1" type="xs:string">
        </xs:element> <!-- fecha en la que se emite la solicitud -->
      <xs:element name="sistemaSolicitante" minOccurs="1" type="xs:string
        "> </xs:element> <!-- nombre del sistema que realiza la
        solicitud -->
      <xs:element name="idAplicacion" minOccurs="1" type="xs:string"> </
        xs:element> <!-- id de la aplicacion solicitante -->
      <xs:element name="cveRFC" minOccurs="1" type="xs:string"> </
        xs:element> <!-- RFC del proveedor -->
      <xs:element name="cveTipoMensaje" minOccurs="1" type="xs:string"> <
        /xs:element> <!-- clave del tipo de mensaje que se envia -->
      <xs:element name="numContrato" minOccurs="1" type="xs:string"> </
        xs:element> <!-- numero de contrato con el proveedor -->
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Un ejemplo de un documento XML de un mensaje de solicitud de estudios de laboratorio generado con la estructura anterior (XSD), se presenta en el apéndice B.

4.2.2. Resultados de laboratorio

El mensaje de resultados de laboratorio contiene los valores obtenidos en el laboratorio de los estudios que se solicitaron. A continuación se muestra la estructura (documento XSD) propuesta para los mensajes de resultados de laboratorio. La estructura contiene 5 elementos: “resultadoSolicitud”, “jefeServicio”, “quimico”, “estudio” y “control”. El paciente se identifica dentro del elemento “resultadoSolicitud”, específicamente en el elemento “cveIdee”, con su identificador único (en México usualmente la CURP). En “resultadoSolicitud” también se envían los datos de identificación de la solicitud realizada (“numFolioSolicitud”), la fecha en la que se atendió al paciente (“fechaAtencion”) y la fecha en la que los resultados fueron validados por el químico responsable (“fechaValidacion”). Los datos de identificación del químico responsable se encuentran en el elemento “quimico”. Así mismo el jefe de servicio o supervisor se identifica en el elemento “jefeServicio”. El elemento “estudio” contiene los valores que resultaron del estudio aplicado al paciente, identificando la clave LOINC tanto del estudio (“cveEstudio”) como de la prueba (“cvePrueba”) y los valores obtenidos (“numValor”, “numValorMin” y “numValorMax”).

A continuación se muestra el contenido del XSD que contiene la propuesta de estructura para los mensajes de resultados de laboratorio. Nótese que todo el código está comentado para resaltar su estructura y sus elementos.:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<!-- Elemento raiz: resultadoLab -->
  <xs:element name="resultadoLab">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="resultadoSolicitud" minOccurs="1" maxOccurs="1"/>
          <!-- Datos de los resultados de la solicitud -->
        <xs:element ref="jefeServicio" minOccurs="1" maxOccurs="1"/> <!--
          Datos del jefe de servicio encargado -->
        <xs:element ref="quimico" minOccurs="1" maxOccurs="1"/> <!-- Datos
          del que valida los resultados -->
        <xs:element ref="estudio" minOccurs="1" maxOccurs="unbounded"/> <!--
          Datos del estudio realizado -->
        <xs:element ref="control" minOccurs="1" maxOccurs="1"/> <!-- Datos
          de control del mensaje -->
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```

        </xs:complexType>
    </xs:element>
<!-- Elemento para identificar el estudio solicitado y el paciente -->
    <xs:element name="resultadoSolicitud">
        <xs:annotation>
            <xs:documentation>Datos generales de los resultados de laboratorio.</
            xs:documentation>
        </xs:annotation>
        <xs:complexType>
            <xs:sequence>
                <xs:element name="fechaAtencion" minOccurs="1" type="xs:date"> </
                    xs:element> <!-- fecha de atencion -->
                <xs:element name="cveIdee" minOccurs="1" type="xs:string"> </
                    xs:element> <!-- Identificador unico del paciente -->
                <xs:element name="numFolioSolicitud" minOccurs="1" type="xs:string"
                    > </xs:element> <!-- numero de folio de la orden de solicitud --
                    >
                <xs:element name="fechaValidacion" minOccurs="1" type="xs:string">
                    </xs:element> <!-- fecha en la que se validaron los resultados
                    -->
            </xs:sequence>
        </xs:complexType>
    </xs:element>
<!-- Elemento que identifica al jefe de servicio o responsable: -->
    <xs:element name="jefeServicio">
        <xs:annotation>
            <xs:documentation>Datos del Jefe de Servicio.</xs:documentation>
        </xs:annotation>
        <xs:complexType>
            <xs:sequence>
                <xs:element name="cveMatricula" minOccurs="1" type="xs:string"> </
                    xs:element> <!-- clave de trabajador del jefe de servicio -->
                <xs:element name="apellidoPaterno" minOccurs="1" type="xs:string">
                    </xs:element> <!-- apellido paterno del jefe de servicio -->
                <xs:element name="apellidoMaterno" minOccurs="1" type="xs:string">
                    </xs:element> <!-- apellido matern del jefe de servicio -->
                <xs:element name="nombre" minOccurs="1" type="xs:string"> </
                    xs:element> <!-- nombre del jefe de servicio -->
                <xs:element name="cedula" minOccurs="1" type="xs:string"> </
                    xs:element> <!-- cedula profesional del jefe de servicio -->
            </xs:sequence>
        </xs:complexType>
    </xs:element>
<!-- Datos de la persona que valida los resultados: -->

```

```

<xs:element name="quimico">
  <xs:annotation>
    <xs:documentation>Datos del Quimico que valida.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="cveMatricula" minOccurs="1" type="xs:string"> </
        xs:element> <!-- clave de trabajador del quimico -->
      <xs:element name="apellidoPaterno" minOccurs="1" type="xs:string">
        </xs:element> <!-- apellido paterno del quimico -->
      <xs:element name="apellidoMaterno" minOccurs="1" type="xs:string">
        </xs:element> <!-- apellido materno del quimico -->
      <xs:element name="nombre" minOccurs="1" type="xs:string"> </
        xs:element> <!-- nombre del quimico -->
      <xs:element name="cedula" minOccurs="1" type="xs:string"> </
        xs:element> <!-- cedula profesional del quimico -->
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- Datos de los resultados del estudio -->
<xs:element name="estudio">
  <xs:annotation>
    <xs:documentation>Datos del Estudio realizado.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="cveEstudio" minOccurs="1" type="xs:string"> </
        xs:element> <!-- clave LOINC del estudio -->
      <xs:element name="observacionesEstudio" minOccurs="1" type="
        xs:string"> </xs:element> <!-- observaciones del estudio -->
      <xs:element name="cvePrueba" minOccurs="1" type="xs:string"> </
        xs:element> <!-- clave LOINC de la prueba -->
      <xs:element name="refInterReferencia" minOccurs="1" type="xs:string
        "> </xs:element> <!-- Valor de referencia si es que es generado
        por el sistema de referencia, en formato de texto -->
      <xs:element name="numValor" minOccurs="1" type="xs:string"> </
        xs:element> <!-- Valor del Resultado de la prueba -->
      <xs:element name="numValorMin" minOccurs="1" type="xs:string"> </
        xs:element> <!-- Valor de referencia minimo -->
      <xs:element name="numValorMax" minOccurs="1" type="xs:string"> </
        xs:element> <!-- Valor del referencia maximo -->
      <xs:element name="cveSerieEquipo" minOccurs="1" type="xs:string"> <
        /xs:element> <!-- numero de serie del equipo -->

```

```

    <xs:element name="unidadMedida" minOccurs="1" type="xs:string" </
      xs:element> <!-- unidad de medida utilizado -->
    <xs:element name="inetrpretacion" minOccurs="1" type="xs:string" <
      /xs:element> <!-- interpretacion del estudio -->
    <xs:element name="observacionesPrueba" minOccurs="1" type="
      xs:string" </xs:element> <!-- observaciones de la prueba -->
  </xs:sequence>
</xs:complexType>
</xs:element>
<!-- Datos de control -->
<xs:element name="control">
  <xs:annotation>
    <xs:documentation>Datos del Control.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="cveTipoServicio" minOccurs="1" type="xs:string">
        </xs:element> <!-- Clave del tipo de servicio contratado -->
      <xs:element name="cveRFC" minOccurs="1" type="xs:string" </
        xs:element> <!-- RFC del proveedor -->
      <xs:element name="numAplicacion" minOccurs="1" type="xs:string" </
        xs:element> <!-- numero de aplicacion -->
      <xs:element name="cveTipoMensaje" minOccurs="1" type="xs:string" <
        /xs:element> <!-- clave del tipo de mensaje -->
      <xs:element name="numContrato" minOccurs="1" type="xs:string" </
        xs:element> <!-- numero de contrato -->
      <xs:element name="cvePresupuestal" minOccurs="1" type="xs:string">
        </xs:element> <!-- Clave Presupuestal de la unidad que requirio
          el estudio de laboratorio -->
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Un ejemplo de un mensaje de resultados de estudios de laboratorio es mostrado en el apéndice C.

4.3. Experimentación

La implementación de interoperabilidad se desarrolló utilizando el lenguaje de programación Java, el estándar HL7v3 y WS para la interfaz de mensajes.

En los casos de estudio presentados a continuación, las bases de datos pertenecen a una universidad y a un hospital, ambos de México. En los datos presentados nosotros omitimos información específica para mantener su privacidad.

4.3.1. Comunicación de un HIS con un LIS

Presentamos ahora la solución implementada en un HIS que está operando en una institución pública de salud importante de México. La implementación se realizó para lograr la comunicación de ésta con un LIS, la cual permite que el HIS envíe sus solicitudes de estudios de laboratorio y el LIS las reciba. Cuando el laboratorio tiene los resultados de los estudios envía el mensaje con la información de éstos y es recibida por el HIS para incorporarla a su base de datos. La solución fue lograda siguiendo la metodología de implementación para interoperabilidad entre bases de datos heterogéneas, descrita a continuación:

1. Especificar la información que se requiere intercambiar. Se determinó que la información que se requiere compartir entre los dos sistemas son datos de identificación del paciente (identificador único, nombre completo, fecha de nacimiento y sexo), detalles de la solicitud (folio, clave LOINC de los estudios solicitados y fecha de la solicitud) e identificación del solicitante (identificador del médico, nombre completo del médico y clave única del establecimiento en salud) del lado del HIS. Por parte del LIS la información que se comparte son los resultados de los estudios de laboratorio (clave LOINC del estudio, valores obtenidos en el estudio, nombre completo del químico que valida los resultados y fecha de realización de los estudios). La especificación de esta información quedó plasmada en un documento XSD, el cual se utilizó como base para obtener las clases del lenguaje Java mediante la herramienta XJC:

```
xjc -p mx.com.hl7 archivo.xsd
```

2. El estándar de comunicación seleccionado fue la tercera versión de HL7, con los mensajes ADT, ORM y ORU, debido a que son los tipos de mensajes HL7 más apropiados para comunicar información de estudios de laboratorio.
3. Se realizó un mapeo de los campos de la base de datos hacia los elementos que son parte de los mensajes seleccionados. El sistema manejador de base de datos que se utilizó fue PostgreSQL ².

²<http://www.postgresql.org/>, consultado el 6 de abril de 2014

4. Para la interfaz de intercambio de mensajes se decidió publicar un WS por parte del HIS para enviar las solicitudes de estudios y otro WS por parte del LIS para publicar los resultados de los estudios.
5. El código que genera los mensajes con la información de la base de datos fue desarrollado con el lenguaje Java. Esta programación extrae la información de la base de datos y la coloca en un documento XML con la estructura de los mensajes especificados. El apéndice D contiene el código de Java de una clase que se utilizó para generar los documentos XML de solicitud de estudios de laboratorio.
6. La interfaz de intercambio se implementó en el código de Java. Ocupamos la herramienta JAX-WS para generar los métodos utilizados para hacer uso de los WSs.

Resultados

Los dos sistemas, el HIS y el LIS, trabajan de manera separada y diferente, pero cuando requieren intercambiar información entre ellos ahora es posible mediante esta solución sin tener que alterar su funcionamiento interno. En el apéndice A se muestra el reporte de los resultados de la implementación de esta comunicación.

En la figura 4.2 se muestra un esquema del flujo de datos que va desde que un médico realiza la solicitud de estudios de laboratorio desde el HIS hasta que los datos quedan registrados en la base de datos del LIS. En la figura 4.3 se muestra el esquema de cómo pasan los datos desde que el químico los registra en el LIS hasta que llegan a la base de datos del HIS.

Las dos bases de datos, la del HIS y la del LIS, pueden intercambiar información manteniendo sus estructuras originales y así se tiene un mejor funcionamiento de los sistemas y también se les da un mejor servicio a los usuarios, que en este caso no sólo son los médicos y enfermeras, sino que también los pacientes mismos, quienes han mencionado que se han beneficiado con esta implementación al realizar más fácil y rápidamente los trámites requeridos en los estudios de laboratorio solicitados por su médico.

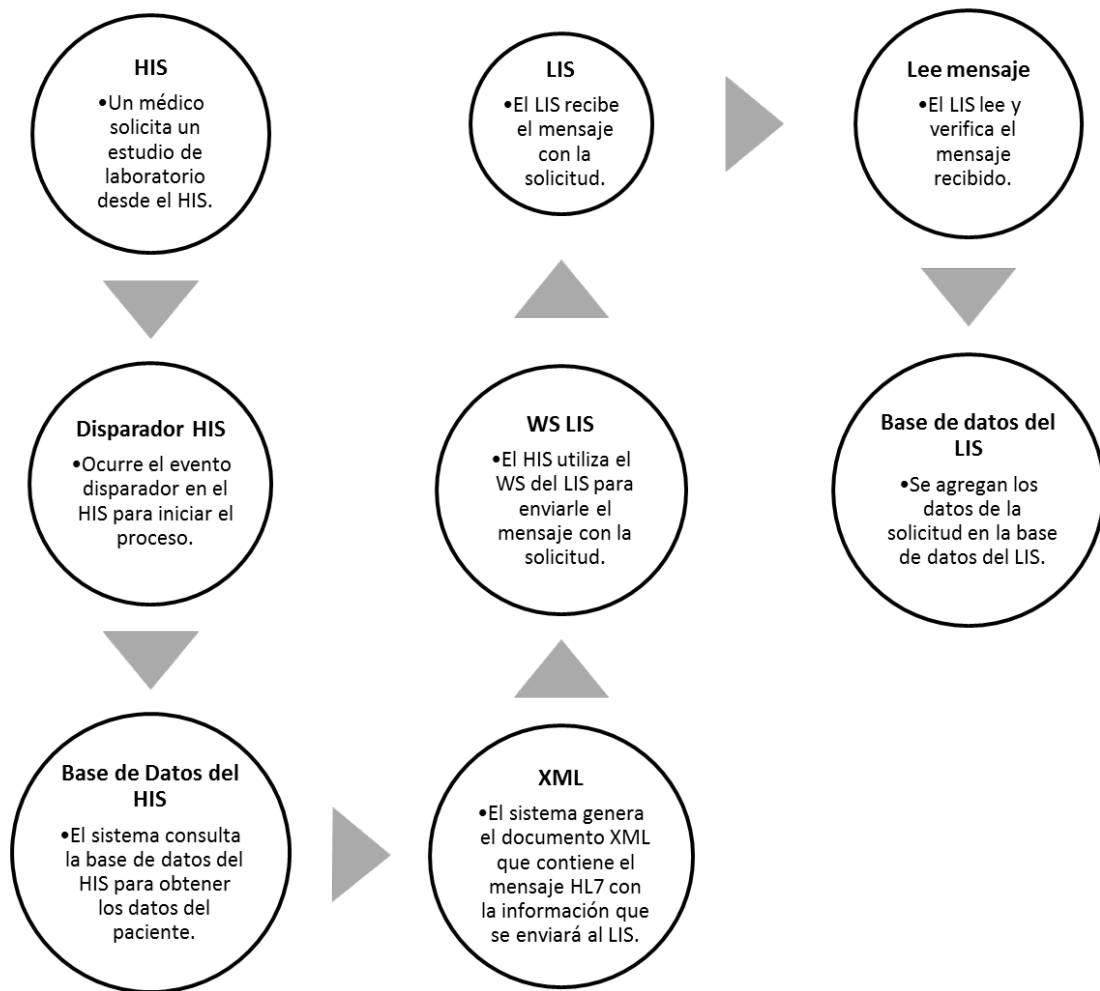


FIGURA 4.2: Flujo de datos de la solicitud en la implementación, desde el HIS hasta el LIS

4.3.2. Institución educativa

Otro caso de estudio se realizó con una base de datos de una instancia deportiva en una institución educativa. El trabajo de investigación se realizó en un sistema que registra los resultados de diferentes pruebas realizadas en un estudio médico. El estudio consta de varias áreas: antropometría, biomecánica, electrocardiografía, espirometría, pruebas bioquímicas, nutrición y psicología. El estudio puede ser realizado a cualquier persona que lo solicite, aunque principalmente se aplica a deportistas. Cuando se realiza el estudio, los resultados quedan registrados en una base de datos y son consultados por los médicos mediante un programa desarrollado con el sistema manejador de bases de datos *SQL Server*.

El problema con el que se encontraron en esta instancia deportiva fue el de lograr que los datos pudieran ser consultados por otros sistemas de información, como por

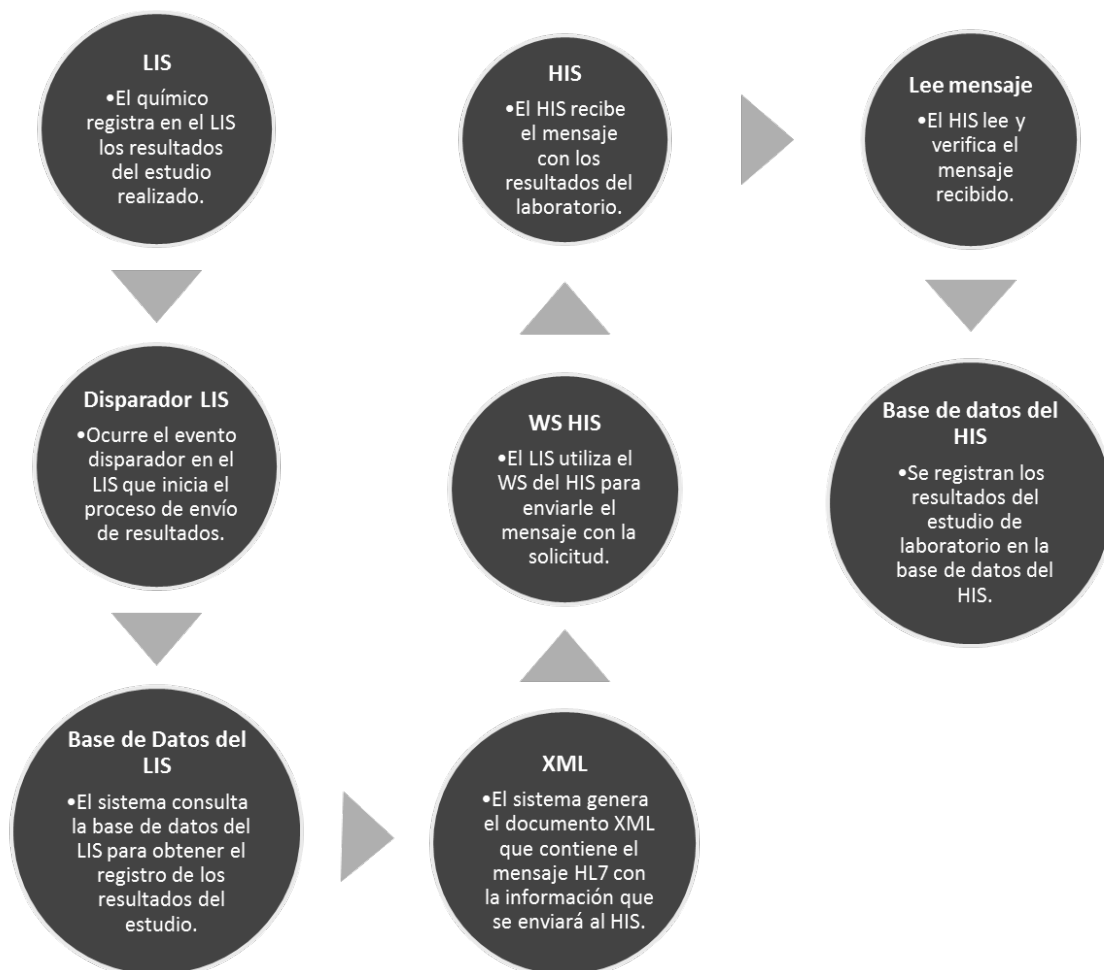


FIGURA 4.3: Flujo de datos de resultados en la implementación, desde el LIS hasta el HIS

ejemplo el sistema de registros académicos de la universidad. Para solucionar el problema se empleó la metodología de implementación para interoperabilidad entre bases de datos heterogéneas. A continuación se describe cómo realizamos cada uno de los pasos:

1. Se especificó que la información que se compartiría serían los resultados de todas las pruebas realizadas a la persona solicitada.
2. El estándar de comunicación elegido fue HL7v3, con los mensajes ADT (para datos de las personas) y ORU (para los resultados de las pruebas).
3. Los campos de la base de datos fueron mapeados a los segmentos de los mensajes. Cada mensaje de resultados (ORU) lleva los datos correspondientes a todas las pruebas realizadas a una persona. No se especifica la información de las tablas y los campos de la base de datos por razones de privacidad.

4. Para la interfaz de intercambio de mensajes se eligió desarrollar un WS para publicar los resultados de las pruebas. El WS recibe como parámetro el identificador de la persona (Clave Única de Registro de Población) y entrega como respuesta un mensaje ORU con los resultados de las pruebas realizadas a la persona especificada.
5. El código para generar los mensajes con los resultados almacenados en la base de datos fue desarrollado con el lenguaje Java. Esta programación extrae la información de la base de datos y crea un documento XML que contiene los datos de los resultados con la estructura del mensaje ORU.
6. La interfaz de intercambio se implementó en el código de Java. En este caso también fue empleada la herramienta JAX-WS para generar los métodos que publican la información en el WS.

Resultados

Se implementó la interoperabilidad de esta base de datos mediante la metodología propuesta, logrando así que los sistemas interesados en consultar los datos tengan una forma de hacerlo. El sistema almacena una base de datos de *SQL Server 2005*, y el manejo que hacen los usuarios es mediante una interfaz gráfica. El trabajo está listo para que los datos puedan ser enviados desde el sistema que contiene esta información hacia otros sistemas sin que se conozcan sus características ni arquitectura, compartiendo los resultados de las pruebas realizadas.

Capítulo 5

Conclusiones y trabajos futuros

Los objetivos de este trabajo son: 1) proponer una metodología de implementación para lograr la interoperabilidad entre bases de datos de sistemas heterogéneos, 2) presentar dos casos de estudio de integración de los sistemas de instituciones de salud en México que puedan servir como propuesta para las bases de una guía de implementación de éstos y 3) realizar una investigación exhaustiva sobre la integración de bases de datos, en particular en el sistema de salud de México, conociendo las investigaciones más recientes del tema.

La metodología propuesta fue utilizada para implementar los casos de estudio presentados. Estas implementaciones son de utilidad y tienen impacto en la agilidad de los procesos involucrados en el cuidado de la salud de la población. En el primer caso se logró la interoperabilidad entre un HIS y un LIS, ambos en Salina Cruz, Oaxaca. La comunicación se estableció siguiendo los pasos descritos y se ha integrado a las operaciones diarias de ambos sistemas satisfactoriamente. De esta manera se logró automatizar la comunicación entre el HIS y el LIS sin tener que alterar los sistemas que ya existían antes de esta implementación. Los datos se comparten y tienen interacción los sistemas, logrando hacer más ágil y eficiente la atención del paciente.


En México existen muchas iniciativas para desarrollar un ECE en las diferentes instituciones de salud, cada una con sus recursos y obstáculos particulares. Estos proyectos están en desarrollo, pero también es evidente la necesidad de lograr la interoperabilidad entre todos ellos, y finalmente en todo el sistema de salud. Es necesario generar las guías que orienten a los Prestadores de Servicios de Salud para implementar la

interoperabilidad entre instituciones de salud. Hace falta que los tomadores de decisiones en el área coordinen y aprueben las propuestas. Esta metodología puede utilizarse para la creación de éstas guías. Cabe destacar que siempre es posible mejorar la interoperabilidad pero es muy difícil llegar a una interoperabilidad total.

Además de conseguir el intercambio de información médica entre instituciones, sería deseable que se integre la información de otras áreas, como puede ser el historial crediticio, registro de pago de impuestos, historial académico, etc., de tal manera que se disponga de toda la información que se pueda requerir de una persona en un momento dado. Uno de estos casos es el de los PHR, ya que para que las personas puedan alojar y administrar su PHR, actualmente existen sistemas que proveen este servicio de manera electrónica (principalmente a través de un servidor web). Con la existencia de estos sistemas, es necesario que éstos tengan la capacidad de comunicarse con otros sistemas de salud para que no se conviertan en “islas de información”, como lo menciona Tang en su trabajo [38]. La información sobre medicación, planes de atención, entre otros, podrían ser conectados con los sistemas de diversos proveedores, como pueden ser farmacias, hospitales, instituciones de salud, etc. Para lograr esto se requiere implementar la interoperabilidad del PHR [38], lo cual puede ser realizado mediante la metodología propuesta en este trabajo.

Apéndice A

Evidencia de resultados

Grupo

 Servicios Profesionales S.A. de C.V.

Reporte de Atención al Cliente SIGEMH

Reporte No. SVN 7395

Fecha : 7 de Mayo de 2013	Hora: 07:00
Cliente: Hospital con Especialidades de Salina Cruz, Oaxaca	
Dirección: Camino a San Antonio Monterrey S/N C.P. 70701	
Teléfono (s): 01. 971.28.143.29 Ext. 159	
Objetivo: Implementar la infraestructura para realizar la interface con el sistema de Laboratorio de Análisis Clínicos	
Servicio: <u>Desarrollo:</u> Interface con Laboratorio. <u>Capacitación:</u> Personal Técnico del hospital.	<u>Mantenimiento:</u> Actualización de la solicitud de estudios de laboratorio. <u>Soporte:</u> En sitio para implementación y a distancia para el seguimiento.

Especificación de la Atención Otorgada

Desarrolló de la infraestructura y funcionalidad para la comunicación entre el sistema de administración de estudios de laboratorio de análisis clínicos y el expediente clínico electrónico de SIGEMH, se construyeron los mensajes con el estándar internacional HL7 V.3, utilizando para la identificación de estudios de laboratorio el catálogo internacional LOINC, se creó la bitácora de control del envío de solicitudes de laboratorio y recepción de resultados, de los servicios de Consulta Externa, Hospitalización y Urgencias.

Se actualizó el catálogo de estudios y pruebas de laboratorio, con base a los autorizados para el hospital, se adecuó la pantalla de recepción de resultados para homologarla con la pantalla del sistema de laboratorio; se modificó la funcionalidad de registro de solicitudes de laboratorio para agrupar las pruebas de acuerdo a las áreas que especificó el químico responsable del Laboratorio. Se modificó la impresión de la solicitud de laboratorio, se adecuó la funcionalidad del sistema a los requerimientos:

1. Solicitud de laboratorio por perfil y estudio.
2. Presentación en el expediente clínico electrónico, de los resultados de laboratorio por solicitud.
3. Incorporación de los resultados de bacterias con antibiograma.
4. Control de estatus por solicitud.
5. Impresión de solicitudes acorde a los requerimientos del sistema de laboratorio.

Capacitación al personal técnico encargado de la administración de los sistemas del hospital, en cuanto al proceso e implementaciones de la tecnología aplicada.


Soporte en sitio, una semana durante la implementación del nuevo desarrollo y de manera remota a la fecha en forma permanente.

Al término de la implementación queda de conformidad el personal de Laboratorio de Análisis Clínico Q.B. Salvador Salazar Barrios y del Lic. Juan Carlos Santiago Sánchez.

Madrid 122 Piso 1 Col. del Carmen, Coyoacán, México D.F. CP. 04100
 Teléfonos: 56396397 – 56398633
 www.grupogc.com.mx

FIGURA A.1: Evidencia de resultados en Grupo GC. Hoja 1

Grupo



Servicios Profesionales S.A. de C.V.

Observaciones

Previo a la liberación de los procesos a la instancia de Producción, se llevaron a cabo diversas pruebas y ajustes a los sistemas, siendo validados por el personal de Laboratorio como de Tecnologías de la Información.

Q.B. Salvador Salazar Barrios
Jefe del Laboratorio de Análisis Clínico-
Nombre, Cargo y Firma del Cliente

Lic. Juan Carlos Santiago Sánchez
Jefe de Tecnologías de la Información
Nombre, Cargo y Firma del Cliente

IC. Ulises Dávalos Mendoza
Responsable de PI 7
Nombre, Cargo y Firma del Responsable de GC

Madrid 122 Piso 1 Col. del Carmen, Coyoacán, México D.F. CP. 04100
Teléfonos: 56396397 – 56398633
www.grupogc.com.mx

FIGURA A.2: Evidencia de resultados en Grupo GC. Hoja 2

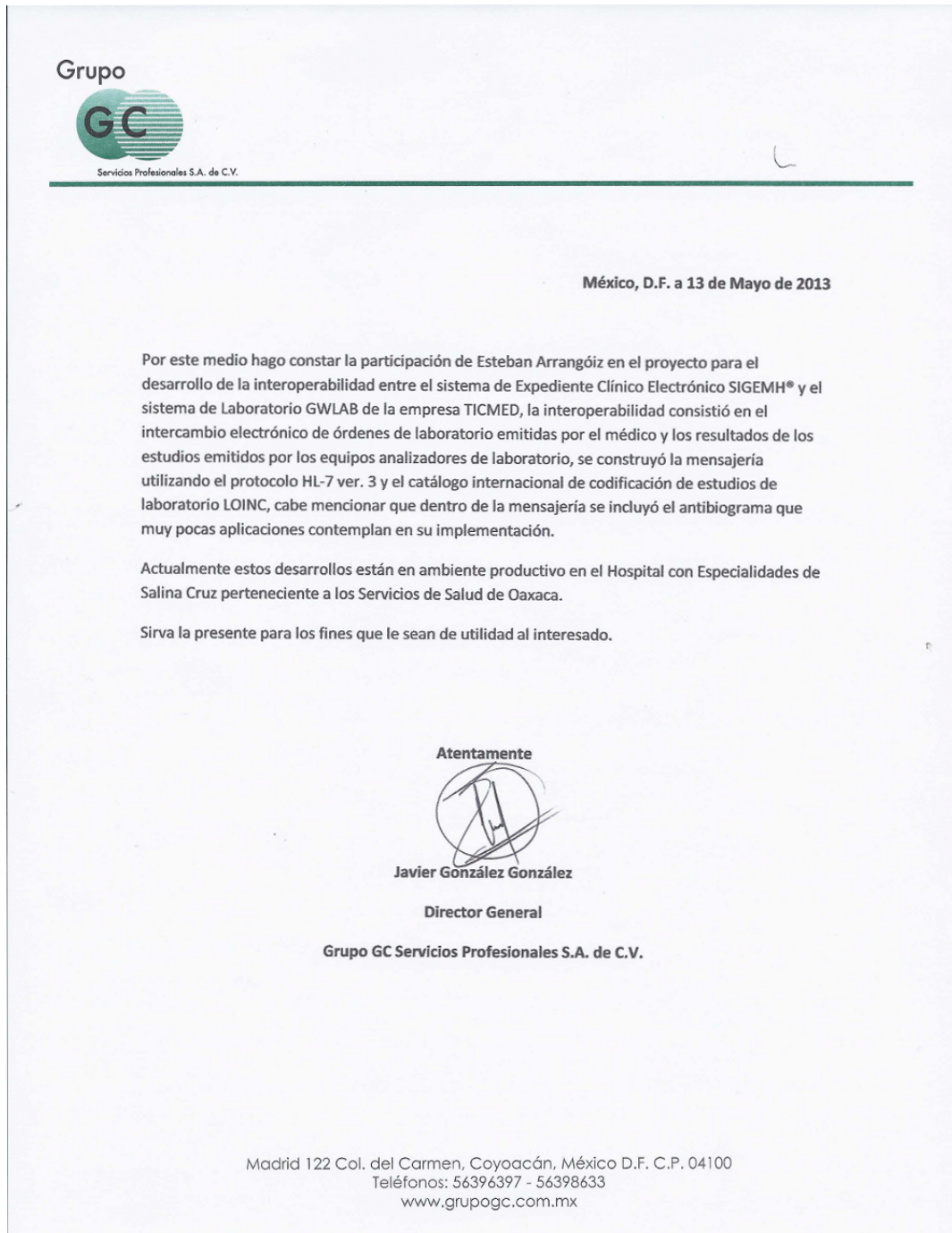


FIGURA A.3: Carta de participación en Grupo GC.

Apéndice B

Ejemplo de un mensaje (XML) de solicitud de laboratorio

```
<?xml version="1.0" encoding="UTF-8"?>
<solicitudLab>
  <paciente>
    <origenPaciente>3</origenPaciente>
    <fechaAtencion>2013-04-09T16:57:42.798-05:00</fechaAtencion>
    <cveIdee>63651</cveIdee>
    <cveTurno>22032472</cveTurno>
    <apellidoPaterno>FLORES</apellidoPaterno>
    <apellidoMaterno>MARTINEZ</apellidoMaterno>
    <nombre>GII DE MARIBEL</nombre>
    <sexo>F</sexo>
    <fechaNacimiento>2013-02-04T00:00:00.000-06:00</fechaNacimiento>
  </paciente>
  <medico>
    <cveEspecialidad>9</cveEspecialidad>
    <apellidoPaterno>PEREZ</apellidoPaterno>
    <apellidoMaterno>JIMENEZ</apellidoMaterno>
    <nombre>DR. LUIS</nombre>
    <cedula></cedula>
    <cveMatricula>16</cveMatricula>
  </medico>
  <capturista>
    <apellidoPaterno>PEREZ</apellidoPaterno>
    <apellidoMaterno>JIMENEZ</apellidoMaterno>
    <nombre>DR. LUIS</nombre>
  </capturista>
</solicitudLab>
```

```
<cveMatricula>16</cveMatricula>
</capturista>
<lugarAtencion>
  <cveLugarAtencion>OC SSA 01987 3</cveLugarAtencion>
  <cveTipoLugarAtencion>0</cveTipoLugarAtencion>
  <cveUnidadMedica>OC SSA 01987 3</cveUnidadMedica>
</lugarAtencion>
<solicitud>
  <cveTipoSolicitud>Ordinaria</cveTipoSolicitud>
  <stpOrden>2013-04-08T16:57:42.798-05:00</stpOrden>
  <cveCie10>P369</cveCie10>
  <refComplementoDiagnostico>PREMATURO DE 35 SDG PEQUENIO PARA LA EDAD
    GESTACIONA</refComplementoDiagnostico>
  <numFolioSolicitud>220130730</numFolioSolicitud>
  <cveEstudio>1751-7</cveEstudio>
  <cvePrueba>1751-7</cvePrueba>
  <stpEstimadaResultado>2013-04-08T16:57:45.042-05:00</stpEstimadaResultado>
  <tipoProcesamiento>0</tipoProcesamiento>
</solicitud>
<control>
  <numFolioOrden>154</numFolioOrden>
  <stpTransaccion>2013-04-08T16:57:42.798-05:00</stpTransaccion>
  <sisistemaSolicitante>SIGEMH</sisistemaSolicitante>
  <cveTipoServicio></cveTipoServicio>
  <idAplicacion>HL7_CORE</idAplicacion>
  <cveRFC>RFCGrupoGC</cveRFC>
  <cveTipoMensaje>1</cveTipoMensaje>
  <numContrato></numContrato>
</control>
</solicitudLab>
```

Apéndice C

Ejemplo de un mensaje (XML) de resultados de laboratorio

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<resultadoLab>
  <resultadoSolicitud>
    <stpFechaAtencion>2013-04-10T00:00:00</stpFechaAtencion>
    <cveIdee>62179</cveIdee>
    <numFolioOrden>138</numFolioOrden>
  </resultadoSolicitud>
  <jefeServicio>
    <cveMatricula>CVE_MATRICULA</cveMatricula>
    <nombre>REF_NOMBRE</nombre>
    <cedula>REF_CEDULA_QUIMICO</cedula>
  </jefeServicio>
  <estudio>
    <cveEstudio>REF_OBSERVACIONES</cveEstudio>
    <stpValidacionResultado>2013-04-10T00:00:00</stpValidacionResultado>
    <prueba>
      <cvePrueba>3084-1</cvePrueba>
      <indToma></indToma>
      <refInterReferencia>I</refInterReferencia>
      <numValor>10.0</numValor>
      <cveSerieEquipo>BS200</cveSerieEquipo>
      <unidadMedida>mg/dL</unidadMedida>
      <interpretacion>2.0| 3.0</interpretacion>
    </prueba>
  </estudio>
  <quimico>
```

```
    <nombre>r.aquino</nombre>
  </quimico>
</estudio>
<control>
  <cveTipoServicio>CVE_TIPOSERVICIO</cveTipoServicio>
  <cveRFC>CVE_RFC</cveRFC>
  <numAplicacion>NUM_APLICACION</numAplicacion>
  <cveTipoMensaje></cveTipoMensaje>
  <numContrato>NUM_CONTRATO</numContrato>
  <cveUnidadMedica>CVE_PRESUPUESTAL</cveUnidadMedica>
  <cveStatusSolicitud>1</cveStatusSolicitud>
  <error>
    <cveError>E_ES</cveError>
    <tagError>handlingCode</tagError>
    <descError>handlingCodeExposedMaterial no encontrado (NUM_MAX NUM_MIN)<
      /descError>
    <comentario></comentario>
  </error>
  <stpTransaccion>2000-01-01T12:12:12</stpTransaccion>
</control>
</resultadoLab>
```

Apéndice D

Código de Java que genera los mensajes de solicitud de estudios de laboratorio

```
public static String generaXmlSolicitud(List<SolicitudLabSigemhVo>
    listaSolicitudesLaboratorioVo) {
    //metodo que genera el codigo XML de la lista de solicitudes recibida en la
    lista listaSolicitudesLaboratorioVo
    StringWriter stringWriter = new StringWriter(); //utilizado para escribir
    los datos en una cadena de texto

    try {
        SolicitudLabSigemhVo solicitudLabSigemhVo =
            listaSolicitudesLaboratorioVo.get(0);
        ObjectFactory factory = new ObjectFactory(); //crea un nuevo objeto
            de la clase ObjectFactory para almacenar los datos que iran en
            el XML
        SolicitudLab solicitudLab = factory.createSolicitudLab(); //crea el
            elemento SolicitudLab
        Paciente paciente = factory.createPaciente(); //crea el elemento
            Paciente
        if( solicitudLabSigemhVo.getPacienteOrigenPaciente() .toUpperCase()
            .trim().equals("CONSULTA EXTERNA") )
            paciente.setOrigenPaciente("1"); //si el paciente proviene
            de Consulta externa, entonces se asigna un 1 al campo
            OrigenPaciente
    }
}
```

```
else if( solicitudLabSigemhVo.getPacienteOrigenPaciente() .
    toUpperCase().trim().equals("HOSPITALIZACION") )
    paciente.setOrigenPaciente("3"); //si el paciente proviene
    de Hospitalizacion, entonces se asigna un 3 al campo
    OrigenPaciente
else
    paciente.setOrigenPaciente("0"); //en cualquier otro caso se
    asigna el valor 0
GregorianCalendar fechaAtencion = new GregorianCalendar(); //se
    utiliza este objeto para crear la fecha
fechaAtencion.setTime( new Date( solicitudLabSigemhVo.
    getPacienteFechaAtencion().getTime() ) ); //se le asigna la
    fecha
XMLGregorianCalendar xmlGregorianCalendar = DatatypeFactory.
    newInstance().newXMLGregorianCalendar(fechaAtencion); //se
    transforma al formato gregoriano
paciente.setFechaAtencion(xmlGregorianCalendar); //se asigna el
    valor de la fecha al campo FechaAtencion
paciente.setCveIdee(solicitudLabSigemhVo.getPacienteCveIdee().
    toString()); //se asigna la clave del paciente
paciente.setApellidoPaterno(solicitudLabSigemhVo.
    getPacienteApellidoPaterno().trim()); //se asigna el apellido
    paterno
paciente.setApellidoMaterno(solicitudLabSigemhVo.
    getPacienteApellidoMaterno().trim()); //se asigna el apellido
    materno
paciente.setNombre(solicitudLabSigemhVo.getPacienteNombre().trim())
    ; //se asigna el o los nombres
paciente.setSexo(solicitudLabSigemhVo.getPacienteSexo().trim()); //
    se asigna el sexo del paciente
GregorianCalendar fechaNacimiento = new GregorianCalendar();
fechaNacimiento.setTime(new Date(solicitudLabSigemhVo.
    getPacienteFNacimiento().getTime()));
xmlGregorianCalendar = DatatypeFactory.newInstance().
    newXMLGregorianCalendar(fechaNacimiento);
paciente.setFechaNacimiento(xmlGregorianCalendar);
solicitudLab.setPaciente(paciente); //se asigna el elemento
    Paciente a la solicitudLab creada anteriormente
Medico medico = factory.createMedico(); //se crea el elemento
    Medico
medico.setCveEspecialidad(solicitudLabSigemhVo.
    getMedicoCveEspecialidad()); //se asigna la clave de su
    especialidad
```

```
medico.setApellidoPaterno(solicitudLabSigemhVo.  
    getMedicoApellidoPaterno().trim()); //se asigna su apellido  
    paterno  
medico.setApellidoMaterno(solicitudLabSigemhVo.  
    getMedicoApellidoMaterno().trim()); //se asigna su apellido  
    materno  
medico.setNombre(solicitudLabSigemhVo.getMedicoNombre().trim()); //  
    se asigna su(s) nombre(s)  
medico.setCedula(solicitudLabSigemhVo.getMedicoCedula().trim()); //  
    se asigna su cedula profesional  
medico.setCveMatricula(solicitudLabSigemhVo.getMedicoCveMatricula()  
    .toString().trim()); //se asigna su numero de trabajador  
solicitudLab.setMedico(medico); //se asigna el elemento Medico a la  
    solicitudLab creada anteriormente  
Capturista capturista = factory.createCapturista(); //se crea el  
    elemento Capturista  
capturista.setApellidoPaterno(solicitudLabSigemhVo.  
    getCapturistaApellidoPaterno().trim()); //se asigna su apellido  
    paterno  
capturista.setApellidoMaterno(solicitudLabSigemhVo.  
    getCapturistaApellidoMaterno().trim()); //se asigna su apellido  
    materno  
capturista.setNombre(solicitudLabSigemhVo.getCapturistaNombre().  
    trim()); //se asigna su(s) nombre(s)  
capturista.setCveMatricula(solicitudLabSigemhVo.  
    getCapturistaCveMatricula().toString().trim()); //se asigna su  
    numero de trabajador  
solicitudLab.setCapturista(capturista); //se asigna el elemento  
    Capturista a la solicitudLab creada anteriormente  
LugarAtencion lugarAtencion = factory.createLugarAtencion(); //se  
    crea el elemento lugarAtencion  
lugarAtencion.setCveLugarAtencion(solicitudLabSigemhVo.  
    getCveLugarAtencion().toString()); //se asigna la clave del  
    lugar  
lugarAtencion.setCveTipoLugarAtencion(solicitudLabSigemhVo.  
    getCveTipoLugarAtencion().toString()); //se asigna la clave del  
    tipo de lugar  
lugarAtencion.setCveUnidadMedica(solicitudLabSigemhVo.  
    getCveUnidadMedica().toString()); //se asigna la clave  
    presupuestal de la unidad medica  
solicitudLab.setLugarAtencion(lugarAtencion); //se asigna el  
    elemento LugarAtencion a la solicitudLab creada anteriormente  
for (SolicitudLabSigemhVo solicitudActualVo :  
    listaSolicitudesLaboratorioVo) {
```



```

Solicitud elementoSolicitud = factory.createSolicitud();
if( solicitudActualVo.getSolicitudCveTipoSolicitud() )
    elementoSolicitud.setCveTipoSolicitud("1");
else
    elementoSolicitud.setCveTipoSolicitud("0");
GregorianCalendar gregCal = new GregorianCalendar();
gregCal.setTime(new Date(solicitudActualVo.
    getSolicitudStpOrden().getTime()));
XMLGregorianCalendar xmlGrogCalStpOrden = DatatypeFactory.
    newInstance().newXMLGregorianCalendar(gregCal);
elementoSolicitud.setStpOrden(xmlGrogCalStpOrden);
elementoSolicitud.setCveCie10(solicitudActualVo.
    getSolicitudCveCie10().trim());
elementoSolicitud.setRefComplementoDiagnostico(
    solicitudActualVo.getSolicitudRefComplementoDiagnostico(
    ));
elementoSolicitud.setNumFolioSolicitud(solicitudActualVo.
    getSolicitudNumFolioSolicitud().toString());
elementoSolicitud.setCveEstudio(solicitudActualVo.
    getSolicitudCveEstudio());
elementoSolicitud.getCvePrueba().add(solicitudActualVo.
    getSolicitudCvePrueba());
if (solicitudActualVo.getSolicitudStpEstimadaResultado() !=
    null)
    gregCal.setTime(new Date(solicitudActualVo.
        getSolicitudStpEstimadaResultado().getTime()));
else
    gregCal.setTime(new Date());
XMLGregorianCalendar xmlGrogCalStpEstimadaResultado =
    DatatypeFactory.newInstance().newXMLGregorianCalendar(
    gregCal);
elementoSolicitud.setStpEstimadaResultado(
    xmlGrogCalStpEstimadaResultado);
elementoSolicitud.setTipoProcesamiento("0");
elementoSolicitud.setPerfil("SIN_PERFIL");
solicitudLab.getSolicitud().add(elementoSolicitud);
}
Control control = factory.createControl();
GregorianCalendar fechaTransaccion = new GregorianCalendar();
fechaTransaccion.setTime(new Date(solicitudLabSigemhVo.
    getControlStpTransaccion().getTime()));
XMLGregorianCalendar xmlGrogerianCalendarStpTransaccion =
    DatatypeFactory.newInstance().newXMLGregorianCalendar(
    fechaTransaccion);

```

```
        control.setStpTransaccion(xmlGrogerianCalendarStpTransaccion);
        control.setSistemaSolicitante("SIGEMH");
        control.setCveTipoServicio("");
        control.setIdAplicacion("HL7_CORE");
        control.setCveRFC("RFCGrupoGC");
        control.setCveTipoMensaje("1");
        control.setNumContrato("");
        control.setNumFolioOrden(solicitudLabSigemhVo.
            getControlNumFolioOrden());
        solicitudLab.setControl(control);
        JAXBContext jc = JAXBContext.newInstance(ObjectFactory.class);
        Marshaller m = jc.createMarshaller();
        m.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT, Boolean.TRUE);
        m.marshal(solicitudLab, stringWriter);
    } catch (JAXBException e) {
        e.printStackTrace();
    } catch (DatatypeConfigurationException e) {
        e.printStackTrace();
    }
    return stringWriter.toString();
}
```


Abreviaciones

CDA	C línical D ocument A rchitecture
DICOM	D igital I maging and C ommunications in M edicine
ECE	E xpediente C línico E lectrónico
GCS	G lobal C onceptual S chema
HIS	H ospital I nformation S ystem
ISO	I nternational S tandards O rganization
LIS	L aboratory I nformation S ystem
LOINC	L ogical O bservation I dentifiers N ames and C odes
OSI	O pen S ystems I nterconnection
PACS	P icture A rchiving and C ommunications S ystem
PHR	P ersonal H ealth R ecord (Historia de Salud Personal)
RIS	R adiology I nformation S ystem
SNOMED	S ystematized N omenclature O f M EDicine
TIC	T ecnologías de la I nformación y la C omunicación
W3C	W orld W ide W eb C onsortium
WS	W eb S ervice

Bibliografía

- [1] Mark Evered and Serge Bögeholz. A case study in access control requirements for a health information system. In *Proceedings of the Second Workshop on Australasian Information Security, Data Mining and Web Intelligence, and Software Internationalisation - Volume 32*, ACSW Frontiers '04, pages 53–61, Darlinghurst, Australia, Australia, 2004. Australian Computer Society, Inc. URL <http://dl.acm.org/citation.cfm?id=976440.976447>. Consultado el 7 de abril de 2014.
- [2] Andrea Zisman and Jeff Kramer. Towards Interoperability in Heterogeneous Database Systems. Technical Report 11, Department of Computing, Imperial College of Science, Technology and Medicine, 1995. URL <http://www.doc.ic.ac.uk/research/technicalreports/1995/DTR95-11.pdf>. Consultado el 7 de abril de 2014.
- [3] Sarah Humpage. Benefits and Costs of Electronic Medical Records: The Experience of Mexico's Social Security Institute. *Inter-American Development Bank. Department of Research and Chief Economist. Technical Notes*, (IDB-TN-122):1–38, 2010. URL http://www.iadb.org/en/research-and-data/access-document,3181.html?pub_id=35219845. Consultado el 7 de abril de 2014.
- [4] Héctor Vázquez Leal, Raúl Martínez Campos, Carlos Blázquez Domínguez, and Roberto Castañeda Sheissa. Un expediente clínico electrónico universal para México: características, retos y beneficios. *Revista Médica de la Universidad Veracruzana*, 11(1):44–53, 2011. URL http://www.uv.mx/rm/num_anteriores/

- [revmedica_vol11_num1/articulos/expediente.pdf](#). Consultado el 7 de abril de 2014.
- [5] Judith Mariscal Avilés, Ramón Gil-García, and Fernando Ramírez-Hernández. e-Salud en México: antecedentes, objetivos, logros y retos. *Redalyc*, 15(34):65–94, 2012. URL <http://www.redalyc.org/pdf/676/67623463015.pdf>. Consultado el 7 de abril de 2014.
- [6] Mihaela Vida, Lăcrămioara Stoicu-Tivadar, Bernd Blobel, and Elena Bernad. Interoperability Evaluation Case Study: An Obstetrics-Gynecology Department and Related Information Systems. *Studies in health technology and informatics*, 186:177 – 181, 2013. URL <http://ebooks.iospress.nl/publication/32788>. Consultado el 7 de abril de 2014.
- [7] Mustafa Yuksel and Asuman Dogac. Interoperability of Medical Device Information and the Clinical Applications: An HL7 RMIM based on the ISO/IEEE 11073 DIM. *IEEE Transactions on Information Technology in Biomedicine*, 15(4):557–566, July 2011.
- [8] Xuhong Liu, Yunmei Shi, Yabin Xu, Yingai Tian, and Fuheng Liu. Heterogeneous Database Integration of EPR System Based on OGSA-DAI. In Wu Zhang, Zhangxin Chen, CraigC. Douglas, and Weiqin Tong, editors, *High Performance Computing and Applications*, volume 5938 of *Lecture Notes in Computer Science*, pages 257–263. Springer Berlin Heidelberg, 2010. ISBN 978-3-642-11841-8. doi: 10.1007/978-3-642-11842-5_35. URL http://dx.doi.org/10.1007/978-3-642-11842-5_35. Consultado el 7 de abril de 2014.
- [9] Cristhian Cuenca, Juan Gomez, and Sebastian Scotti. Design an Architecture that Allows the Interoperability between Information Systems, Software or Medical Device Adopting the Standard HL7 V2.x. *IEEE Colombian Communications Conference (COLCOM)*, mayo .
- [10] Maqbool Hussain, Muhammad Afzal, Farooq Ahmad, Naeem Khalid, and Arshad Ali. Healthcare Applications Interoperability through Implementation of

- HL7 Web Service Basic Profile. *Sixth International Conference on Information Technology: New Generations (ITNG'09)*.
- [11] William Edward Hammond. Solving the Interoperability Dilemma. *Paper Kills: Transforming Health and Healthcare with Information Technology*, pages 31–46, 2007.
- [12] Amanda Ryan, Peter Eklund, and Brett Esler. Toward the interoperability of HL7 v3 and SNOMED CT: a case study modeling mobile clinical treatment. *IOS Press*, 129(1):626 – 630, 2007. URL <http://ebooks.iospress.nl/publication/11051>. Consultado el 7 de abril de 2014.
- [13] Tamer Özsu and Patrick Valduriez. “Chapter 4 Database Integration.” *Principles of Distributed Database Systems*. Springer, third edition, 2011.
- [14] Tamer Özsu and Patrick Valduriez. *Principles of Distributed Database Systems*. Springer, third edition, 2011.
- [15] IEEE. Glosario de estándares. URL http://www.ieee.org/education_careers/education/standards/standards_glossary.html. Consultado el 7 de abril de 2014.
- [16] David Brailer. Interoperability: the key to the future health care system. *Health Affairs*, (2005):W5–19–W5–21, 2005. URL <http://content.healthaffairs.org/content/early/2005/01/19/hlthaff.w5.19.citation>. Consultado el 7 de abril de 2014.
- [17] Rita Kukafka and William Yasnoff. Public health informatics. *J Biomed Inform*, 40(4):365–369, 2007. URL <http://www.sciencedirect.com/science/article/pii/S1532046407000640>. Consultado el 7 de abril de 2014.
- [18] Leah Hoffmann. Implementing Electronic Medical Records. *Communications of the ACM*, 52(11):18–20, 2009. URL <http://cacm.acm.org/magazines/2009/11/48440-implementing-electronic-medical-records/fulltext>. Consultado el 7 de abril de 2014.

- [19] Magdi Kamel and Nabil Kamel. Federated database management system: Requirements, issues and solutions. *Computer Communications*, 15(4):270–278, 1992. URL [http://dx.doi.org/10.1016/0140-3664\(92\)90110-Z](http://dx.doi.org/10.1016/0140-3664(92)90110-Z). Consultado el 7 de abril de 2014.
- [20] Health Level Seven International. Página web de HL7 International. URL <http://www.hl7.org>. Consultado el 7 de abril de 2014.
- [21] HL7 México. Página web del Capítulo HL7 México. URL <http://hl7.org.mx/>. Consultado el 7 de abril de 2014.
- [22] Secretaría de Salud. Norma Oficial Mexicana NOM-024-SSA3-2012. *Diario Oficial de la Federación*, 2012. URL http://dof.gob.mx/nota_detalle.php?codigo=5280847&fecha=30/11/2012. Consultado el 7 de abril de 2014.
- [23] Glossario de términos de HL7. URL http://www.hl7.org/search/viewSearchResult.cfm?search_id=323429&source=suggest&search_result_url=%2Fdocumentcenter%2Fpublic%2Fcalendarofevents%2FFirstTime%2FGlossary%2520of%2520terms%2Epdf. Consultado el 7 de abril de 2014.
- [24] George Beeler, James Case, Jane Curry, Ann Hueber, Lloyd McKenzie, and Gunther Schadow. HL7 Reference Information Model. *Ann Arbor, MI: HL72003*, 2003. URL <http://www.hl7.org/library/data-model/RIM/C30204/rim.htm>. Consultado el 7 de abril de 2014.
- [25] HL7 International. HL7 Reference Information Model. URL <http://www.hl7.org/implement/standards/rim.cfm>. Consultado el 7 de abril de 2014.
- [26] Regenstrief Institute. Logical Observation Identifiers Names and Codes. URL <http://loinc.org/>. Consultado el 7 de abril de 2014.
- [27] Martin Dugas, Sylvia Thun, Thomas Frankewitsch, and Kai Heitmann. LOINC(R) Codes for Hospital Information System documents: a case study. *J Am Med Inform Assoc*, 16(4):400–403, 2009. URL <http://jamia.bmj.com/content/16/3/400>. Consultado el 7 de abril de 2014.

- [28] NEMA. Página web de DICOM. URL <http://dicom.nema.org/>. Consultado el 6 de abril de 2014.
- [29] Tim Bray, Jean Paoli, Michael Sperberg-McQueen, Eve Maler, and François Yergeau. Extensible Markup Language (XML) 1.0 (Fifth Edition), 2013. URL <http://www.w3.org/TR/xml/>. Consultado el 7 de abril de 2014.
- [30] ORACLE. Java Architecture for XML Binding: Binding Compiler (xjc). URL http://docs.oracle.com/cd/E17802_01/webservices/webservices/docs/1.6/jaxb/xjc.html. Consultado el 3 de junio de 2014.
- [31] Oracle. Java SE Development Kit 7 Documentation. URL <http://docs.oracle.com/javase/7/docs/>. Consultado el 3 de junio de 2014.
- [32] Lawrence Weed. *Medical records, medical education, and patient care: the problem-oriented record as a basic tool*. Press of Case Western Reserve Univ., 1971. URL <http://books.google.com.mx/books?id=unNrAAAAMAAJ>. Consultado el 7 de abril de 2014.
- [33] Eréndira Salgado Ledesma. Lo que no se dice del expediente clínico electrónico. Su impacto en los datos sensibles del paciente. *Revista del Colegio de Profesionistas Posgraduados En Bioética*, 22(3):123–140, 2012. URL <http://www.comexbio.org.mx/comexbio/Etbio/Year2Num3/SALGADO%20LEDESMA%20ERENDIRA.pdf>. Consultado el 7 de abril de 2014.
- [34] Randall D. Cebul, Thomas E. Love, Anil K. Jain, and Christopher J. Hebert. Electronic health records and quality of diabetes care. *New England Journal of Medicine*, 365(9):825–833, 2011. doi: 10.1056/NEJMsa1102519. URL <http://www.nejm.org/doi/full/10.1056/NEJMsa1102519>. PMID: 21879900.
- [35] Nancy Gertrudiz. e-Salud: El caso de México. *Latin Am J Telehealth, Belo Horizonte*, 2(2):127–167, 2010. URL <http://cetes.medicina.ufmg.br/revista/index.php/rlat/article/download/71/109>. Consultado el 7 de abril de 2014.
- [36] Health Level Seven International. HL7 Announces Plans for Freely Available IP and Future Membership Model. URL <http://www.hl7.org/documentcenter/>

-
- [public/pressreleases/HL7_PRESS_20130304.pdf](#). Consultado el 7 de abril de 2014.
- [37] Nils Agne Nordbotten. XML and Web Services Security Standards. *Communications Surveys & Tutorials, IEEE*, 11(3):4–21, 2009. URL <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5208730>. Consultado el 7 de abril de 2014.
- [38] Paul Tang, Joan Ash, David Bates, Marc Overhage, and Daniel Sands. Personal health records: definitions, benefits, and strategies for overcoming barriers to adoption. *JAMIA*, 13(2):121–126, 2006. URL <http://jamia.bmj.com/content/13/2/121.long>. Consultado el 7 de abril de 2014.