Instituto de Computación | Facultad de Ingeniería | Universidad de la República

Proyecto de Grado

"*Arquitectura de referencia para ASSE*"

Integrantes

Elisa Aguerre

Yenny Alonso

Iliana Estala

Karina Rapetti

Tutores

Nelson Calero

Ariel Sabiguero

Tribunal

Laura González

Antonio López Arredondo

Flavia Serra

Montevideo, Octubre de 2012

Resumen

El presente trabajo surge como resultado de la realización del proyecto de final de carrera de la Facultad de Ingeniería de la Universidad de la República (UdelaR). El objetivo planteado para este proyecto es recomendar una arquitectura de software para el sistema informático de una institución de salud del estado.

El primer paso realizado fue investigar acerca de los sistemas informáticos utilizados para los sistemas de salud en la región y relevar los objetivos que se plantean desde el gobierno a través del Sistema Nacional Integrado de Salud (SNIS) de Uruguay.

En la siguiente etapa, se realiza un relevamiento de los sistemas informáticos existentes en la institución y la forma en que se relacionan para brindar una recomendación de arquitectura en la que puedan ser integrados.

Finalmente se opta por utilizar una arquitectura existente y bien conocida en la región, que define los módulos que abarcan todos los procesos de un sistema de salud y muestra la interacción entre los mismos. Además de adoptar una arquitectura, se realizan recomendaciones para el uso de estándares de salud en los sistemas existentes y los sistemas nuevos que se deseen agregar al sistema actual.

Glosario

A

ADT – Sigla que identifica los sistemas de Admisión Traslado y Alta
AGESIC - Agencia para el Desarrollo de la Gestión de Gobierno Electrónica y la Sociedad de la Información y del Conocimiento
ANSI – American National Standars Institute
AP – Atención Primaria
AP/SGA – Atención Primaria / Sistema de Gestión Asistencial
API – Application Programming Interface
ASSE – Administración de Servicios de Salud del Estado.
ASTM – American Society for Testing Materials
ATNA – Audit Trail and Node Authentication

C

CDA – Clinical Document Arquitecture
CDR– Repositorio de Datos Clínicos , la sigla proviene del inglés “Clinical Document Repository”
CEN – Comité Europèen de Normalisation
CEPAL – Comisión Económica para América Latina y el Caribe
CPOE – La sigla proviene del inglés “Computer-based Physician Order Entry”

D

DICOM – Digital Imaging and Communication in Medicine
DIT – Directory Information Tree
DN – Distinguished Name
DSML – Directory Services Markup Language
DOM – Document Object Model

E

e-Health – Salud electronica.

F

FEMI – Federación Médica del Interior

H

HC – Historia Clínica
HCE – Historia Clínica Electrónica
HIBA – Hospital Italiano de Buenos Aires
HL7 – Health Level Seven

I

IHE – Integrating the Healthcare Enterprise
ISO – International Standars Organization

L

LDAP – Lightweight Directory Access Protocol
LDIF – Data Interchange Format
LIS – Sistemas de laboratorio de análisis clínicos
LOINC – Logical Observation Identifiers Names and Codes

M

MSP – Ministerio de Salud Pública

O

OID– Object Identifier
OMS – Organización Mundial de la Salud
OPS – Organización Panamericana de la Salud

P

PIX – Patient Identifier Cross-Referencing
PGE – Plataforma de Gobierno Electrónico

R

RAU2 – **Red Académica Avanzada Uruguaya**
RCE – Registro Clínico Electrónico
RDN – Relative Distinguished Name
RRHH – Recursos Humanos
RIM –Reference Information Model
RIS – Sistemas de información en radiología

S

SASL – Simple Authentication and Security Layer
SCT – Subcomités Técnicos
SGA – Sistema de Gestión Asistencial
SIAP – Sistemas de información de anatomía patológica
SIQ – Sistema de Información Quirúrgica
SINADI – Sistema Nacional de Información
SNIS – Sistema Nacional Integrado de Salud
SSH – Secure Shell
SUAT – Servicios de Urgencia, Asistencia y Traslado
SUEIIDISS – Sociedad Uruguaya de Estandarización, Integración e Intercambio de Datos e Información de Servicios de Salud

T

TIC – Tecnologías de la Información y las Comunicaciones

U

UE – Unidad Ejecutora

X

XML – Extensible Markup Language

Tabla de contenidos

Índice de Figuras

Índice de Tablas

1 Introducción

El presente documento surge como resultado de la realización del proyecto de grado para la carrera de Ingeniería en Computación de la Universidad de la República (UdelaR). Dentro de este marco se propone evaluar el estado de los sistemas de una organización de salud de Uruguay para luego plantear una arquitectura que permita integrar los sistemas existentes, mejorar los servicios brindados e incorporar nuevos servicios en el futuro para mejorar la gestión de la institución, así como la atención al paciente.

Al inicio del proyecto la institución contaba con una nueva plataforma de procesamiento informático, que le permitía extender servicios, brindados en Montevideo, a nivel nacional. A raíz de esta necesidad se plantea este proyecto, el cual pretende sacar el máximo provecho de la misma a corto plazo incorporando nuevas aplicaciones, y adaptando las existentes para que sean fácilmente integradas al resto, accediendo a información de los otros sistemas y entregando información propia al resto en tiempo real en forma de servicios.

1.1 Marco de trabajo

El proyecto de grado se realiza en el ámbito de la Administración de Servicios de Salud del Estado (ASSE) donde se concentran distintos hospitales y policlínicas de todo el país. Dentro de la organización se plantea realizar una propuesta de arquitectura para la plataforma informática, que permita integrar las aplicaciones existentes en las distintas dependencias, de forma de optimizar los recursos utilizados y su mantenimiento.

1.2 Público objetivo

El presente documento está destinado al área de tecnología de la información, se exponen conocimientos en el área de arquitectura de software y estándares en la salud, los cuales no son requisito para entender el contenido ya que se introducen presentando una idea global que permite entender el contexto en el cual se los aplica. Para apoyar el entendimiento del contenido se presentan los apéndices, los cuales buscan brindar un detalle de las temáticas tratadas.

1.3 Organización del documento

En el capitulo 1 se presenta la introducción y la organización del documento.

En el capitulo 2 se presenta el contexto de la informática en la salud tanto a nivel nacional como a nivel internacional.

En el capítulo 3 se presenta el relevamiento realizado sobre la situación actual de ASSE en términos de sistemas informáticos.

En el capítulo 4 se detalla el diagnóstico que se obtuvo en base al relevamiento realizado, y se exponen las recomendaciones del equipo.

En el capítulo 5 se detallan los dos prototipos realizados en el alcance de este proyecto, los temas abordados fueron Registro Clínico Electrónico y Seguridad.

En el capítulo 6 se presentan las conclusiones del proyecto, los resultados alcanzados del conjunto de los propuestos inicialmente y se plantean opciones de trabajo futuro para complementar o extender este proyecto.

Luego dAl finalizar el documento, se presentan una serie de apéndices que sirven para profundizar en algunos de los temas expuestos en el desarrollo del presente informe.

Al final del documento se presenta la bibliografía consultada.

El documento cuenta con una serie de apéndices, que permiten al lector, profundizar en algunos de los temas expuestos en el desarrollo del presente trabajo.

- Apéndice A - Contiene detalles técnicos y de implementación del Prototipo de Almacenamiento de Resultados de Análisis Clínicos.
- Apéndice B – Detalla algunas arquitecturas en Sistemas de Salud, las cuales fueron desarrolladas por actores de la salud.
- Apéndice C - Contiene detalles técnicos y de implementación del Prototipo de autenticación mediante LDAP.

2 Informática en salud

En este capítulo se realiza una introducción a la realidad de la informática aplicada a la salud tanto a nivel nacional como internacional. Al final del capítulo se presenta al lector el conjunto de estándares definidos dentro del área de salud, los que son mencionados dentro del capítulo y en el resto del documento.

2.1 Introducción

Según la definición de la Organización Mundial de la Salud (OMS), "*Un sistema de salud es la suma de todas las organizaciones, instituciones y recursos cuyo objetivo principal consiste en mejorar la salud. Un sistema de salud necesita personal, financiación, información, suministros, transportes y comunicaciones, así como una orientación y una dirección generales. Además tiene que proporcionar buenos tratamientos y servicios que respondan a las necesidades de la población y sean justos desde el punto de vista financiero.*" [1]

El avance en el desarrollo de las tecnologías de información y las comunicaciones (TIC) y los avances en la comunicación de datos a través de Internet proveen la plataforma para el acceso universal globalizado tanto a información como a servicios. Este contexto conduce a la identificación de la "salud electrónica" (e-Health) como un área que se caracteriza por utilizar y combinar las TIC para almacenar y compartir datos con objetivos clínicos, administrativos y educacionales. [2]

La atención en salud es una actividad compleja y muy dependiente de la información para la toma de decisiones [2]. Esta información es actualizada y alimentada constantemente por las distintas investigaciones que se llevan a cabo en el mundo entero. El desafío de los sistemas informáticos en salud es capturar y procesar un gran número de datos con diferente nivel de detalle.

Tradicionalmente se ha observado, dentro de las instituciones de salud, una fragmentación del sector de las TIC aplicadas [2]. La utilización de diversas soluciones, en muchos casos incompatibles entre sí, provoca ineficiencias administrativas y en la provisión de servicios, mayor costo de mantenimiento en los sistemas informáticos, vulnerabilidades en la protección de datos, entre otros. Todos estos problemas conducen a un mayor costo de la gestión y un deterioro de la calidad en los servicios brindados al paciente. La introducción de estándares permite la interoperabilidad y la integración de sistemas, mejorando su independencia de los proveedores y disminuyendo los costos de mantenimiento y operación. [2]

El interés por la estandarización en informática toma un impulso especial en la década de los 90`s, a partir de las iniciativas de la Comisión Europea, CEN (Comité Europèen de Normalisation), ANSI (American National Standars Institute) y por ISO (International Standars Organization). [2]

Teniendo en cuenta que la cantidad de personas en transito es cada vez mayor, hace que se incremente la cantidad de situaciones en las que una persona requiere asistencia con respecto a la salud en un país que no es el de origen. Aquí es donde el acceso a la información clínica es cada vez más importante.[3]

Si bien las TIC han realizado enormes mejoras en los sistemas de información de los sistemas de salud, estas mejoras han sido desarrolladas en entornos locales que no siempre son compatibles entre sí. Se ha llegado a la conclusión de que no es el nivel técnico (donde los sistemas informáticos se lleguen a conectar o entender) el más limitante en estos sistemas, sino lo que resulta mas dificil es el hecho de requerir una transmisión semánticamente equivalente y en un contexto que se logre un significado completo, de forma consistente a pesar de las diferencias (de idioma o diferente usos dentro de un mismo idioma) . Por esta razón en los últimos años se ha dado mayor importancia al estudio de como compartir la información clinica (interoperabilidad).[3]

Si el intercambio de datos del que estamos hablando, ademas se dá a nivel internacional, surgen problemas especificos como lo es el marco legal de la protección de datos, la identificacion del paciente, de los productos farmaceuticos, la diferencia de lenguaje entre otros. La disponibilidad y confidencialidad de la información son dos derechos que tienen un mismo nivel de jerarquia. Esto obliga a buscar un equilibrio entre garantizar la mejor asistencia sanitaria posible, mediante el acceso a los profesionales de los datos de salud, garantizando las medidas de seguridad necesarias para que no se produzcan accesos a datos indebidos [3].

2.2 Informática en salud en la región

En la mayoría de los países de la región se ve una fuerte incidencia en el uso de HL7 como estándar para la interoperabilidad de los sistemas de información en salud [4]. A continuación se hace una breve reseña histórica de la incorporación de éste estándar en distintos países de Latinoamérica.

HL7 Internacional

Es una “Organización de Desarrollo de Estándares”, para el área de la salud. Fundada en 1987 sin fines de lucro, acreditada por ANSI, opera a nivel internacional con presencia en más de 57 países. Su misión es lograr una interoperabilidad real entre los distintos sistemas de información en el área de la salud. A tal fin, desarrolla y difunde estándares globales para los dominios: clínico, asistencial, administrativo y logístico. [4]

HL7 LATAM

Es una instancia de coordinación regional con la finalidad de promover y estimular la utilización de estándares de información en Salud en América Latina, fortalecer las instituciones locales de HL7, impulsar la coordinación con otras organizaciones de estándares, y promover la creación de nuevos capítulos en los países de la región. [4]

HL7 Argentina

Se conforma oficialmente en el año 2002. Nació como una organización sin fines de lucro y puso su foco fundamental en la capacitación y difusión del estándar HL7. [5]

HL7 Brasil

En el 2007 un grupo de profesionales y empresas se reunieron con el objetivo de crear un Instituto con el principal fin de formar, capacitar y certificar a las personas en el uso de HL7, y contribuir a la mejora y la integración de los sistemas de información en salud. Uno de los hechos significativos en este sentido fue la publicación de la norma que incluye HL7 y CDA como estándares en Brasil. [5]

HL7 Chile

Fue creada en el 2007 por un grupo de empresas y organizaciones convencidas de la necesidad de estandarizar. En el 2009 iniciaron un proceso de reorganización que finalizó con la formulación de un plan estratégico 2010-2012 que actualmente se encuentran ejecutando. [5]

HL7 Colombia

Desde su origen ha logrado un apoyo importante de la academia y de la industria, al que también se sumaron prestadores y aseguradores. Colombia promueve el uso de la Telesalud y la Historia Clínica Electrónica para el año 2013. [5]

HL7 México

HL7 México fue reconocido como tal en enero del 2003 por HL7 Internacional y tiene los primeros años una participación activa internacional. En el 2006 se definieron los estatutos, cuotas y miembros del Consejo Directivo, además de las bases y políticas con las que cuenta actualmente. En el 2011 se eligió una nueva mesa directiva que tiene como prioridad establecer estándares de interoperabilidad para el Expediente

Clínico Electrónico en México. [5]

HL7 Uruguay

HL7 Uruguay representado por la SUEIIDISS (Sociedad Uruguaya de Estandarización, Integración e Intercambio de Datos e Información de Servicios de Salud) fue fundada en el 2005 [5]. Se conformaron varios Subcomités Técnicos (SCT) a saber: SCT de identificación y SCT de OID, SCT de Seguridad que elaboró un documento sobre seguridad en la transmisión de datos y espera la reglamentación de la Ley de Firma Digital para completar su trabajo. SCT de CDA y SCT de Terminología que está comenzando su actividad.

En el año 2009 se firmó un convenio con AGESIC y MSP que consagra a la SUEIIDISS como observatorio de buenas prácticas en este tema. [5]

HL7 Puerto Rico

Desde Mayo del 2011 se incorpora a HL7 Internacional, conformado por un grupo de profesionales con interés en el desarrollo de estándares en salud e interoperabilidad y que están participando de proyectos en el sector salud de Puerto Rico. [5]

Otros países se encuentran en el proceso de creación de las organizaciones nacionales de HL7, entre ellos se menciona a Perú y Venezuela que se encuentran iniciando y planificando su formación. [4]

2.2.1 Temas destacados en la región [4]

Uno de los temas destacados de la región es el desarrollo de las terminologías clínicas en una historia clínica electrónica (HCE) en el Hospital Italiano de Buenos Aires (HIBA).

La historia clínica (HC) es el repositorio que contiene la información, ordenada cronológicamente, de los episodios clínicos registrados para una persona. Por lo tanto, es un instrumento imprescindible para que el profesional de la salud pueda llevar a cabo su actividad y prestar al paciente la mejor atención posible en cada momento. Su utilidad transciende los fines asistenciales, y puede añadir funciones de planificación, gestión e investigación. En la actualidad, la mayoría de las historias clínicas se almacenan en papel, con las desventajas que esto genera tanto a nivel de consulta, almacenamiento, personal a cargo, deterioro del papel, ilegibilidad de la letra como también seguridad y confidencialidad de los datos. [3]

La HCE, soluciona las carencias descritas arriba y propone beneficios a nivel de la accesibilidad y disponibilidad debido a que mas de una persona a la vez puede acceder a la HCE y a su vez desde distintos lugares físicos, a nivel de visualización en donde dependiendo las necesidades del usuario es el formato con el que se visualiza, a nivel de comunicación permitiendo la misma entre profesionales de forma similar a un correo electrónico o mensajería instantánea vinculado a la salud de un paciente, a nivel de agregación de datos, se permite crear resúmenes según la información almacenada, hacer gestión clínica e investigación clínica y a nivel de la toma de decisiones colaborando con el proceso asistencial brindando soporte a los profesionales proponiendo alternativas posibles.

La HCE en conjunto con la información almacenada y la base de conocimiento acumulado genera salidas tales como recordatorios, alarmas, sugerencias de diagnósticas o terapéuticas por medio, por ejemplo, de la informatización de guías de práctica clínica, con la finalidad de mejorar la calidad asistencial y prevenir errores.

El HIBA y su red asistencial (HIBAred), ha desarrollado en los últimos 15 años un re-diseño de su red de provisión de servicios. Esto implicó diversas estrategias, incluyendo un proyecto de información en salud denominado Itálica, el cual potencia y favorece el logro de los objetivos y metas planteados.

El HIBA optó por desarrollar Servicios Terminológicos en salud, los cuales se iniciaron en el año 2005, partiendo previamente con la implementación de la HCE en 1998. El departamento de Informática en Salud a través del área de Terminología y Documentación Clínica ofrece servicios terminológicos vía Web Services tanto a la red del hospital y también a otras redes asistenciales en Argentina y contando ya con proyectos en la región (Chile y Uruguay).

Otro tema destacado es la experiencia en Uruguay de AGESIC – SUEIIDISS – MSP en la Agenda Digital y la estandarización en información en salud. En el año 2009 se firma un acuerdo marco entre MSP (Ministerio de Salud Pública), AGESIC y SUEIIDISS para generar sinergias entre la investigación y propuesta de la sociedad y la normatización del gobierno.

SUEIIDISS publicó una propuesta de estándar de Identificación de personas y otro de OID que fueron documentos base para el trabajo de AGESIC. En marzo del 2011, se publica la segunda Agenda Digital del país (2011-2015), en la cual se incluye por primera vez el ámbito Salud expresado en el objetivo 14: Redes avanzadas para la salud e historia clínica integrada a nivel nacional.

También se destaca la Estrategia y Plan de acción de la OPS (Organización Panamericana de la Salud) eSalud en setiembre del 2011, en el cual se busca apoyar a los países en la mejora a los servicios de salud y su calidad utilizando tecnologías de la información y comunicación. Algunos de los objetivos planteados en la estrategia son:

- Respaldar y promover la formulación, ejecución y evaluación de políticas públicas de eSalud en la región.
- Mejorar la infraestructura organizacional y tecnológica para la aplicación de políticas de eSalud.
- Identificar un marco legal que facilite el intercambio electrónico de información clínica en el ámbito nacional y regional, que promueva la validez de las acciones de telemedicina y prevea la protección de los datos personales.
- Fomentar la utilización de servicios de vigilancia epidemiológica a través de las tecnologías de la información y comunicación en salud.

2.2.2 Hospital Italiano de Buenos Aires

El Hospital Italiano de Buenos Aires (HIBA) es una de las instituciones pioneras en la región respecto a la utilización de la informática en la salud pues ha venido utilizándola hace más de veinte años. En el año 1998 se comienza el desarrollo de una historia clínica electrónica ambulatoria [6] y para resolver los problemas de interoperabilidad se crea un área de informática médica con el propósito de crear, implementar y mantener la estructura central de vocabularios de la Institución. De esta forma, el hospital, comienza a resolver los aspectos semánticos y sintácticos de la interoperabilidad entre sus sistemas. En el año 2000, el hospital comienza la tarea de desarrollo de una historia clínica única que abarca la atención ambulatoria, la internación general y la internación domiciliaria [6].

2.2.2.1 Componentes del Sistema Informático del HIBA

El sistema de información en Salud del HIBA recibe el nombre de *Italica* y los componentes que lo integran se pueden ver en la Figura 1. *Italica* es un Sistema de Información de Salud (SIS) completamente desarrollado por el Departamento de Informática en Salud del *HIBAred* que incluye todos los sistemas de manejo de información (soporte en papel o electrónico, asistenciales o administrativos y de gestión) y se basa en componentes que dan servicios web. Intenta romper el modelo histórico de los hospitales con sistemas de información para la administración independientes de los utilizados por los profesionales en la asistencia sanitaria. El concepto central es que todo dato debe ser capturado en el sitio primario donde se genera y potencialmente reutilizado por parte de los demás usuarios. Uno de los objetivos del proyecto es garantizar la interoperabilidad y/o portabilidad de la información.

Los sistemas de información clínicos modernos están compuestos por múltiples componentes (o subsistemas) y el desafío central radica en lograr una adecuada articulación de cada uno de ellos para cumplir el objetivo de centrar la información en los pacientes e integrarla longitudinalmente. Dicha articulación se basa en proveer servicios de los componentes entre sí, facilitando y homogeneizando la lógica de los procesos de las aplicaciones, simplificando el mantenimiento de las aplicaciones y, por consiguiente, extendiendo su vida útil. La descripción de los componentes adoptada por la institución se hizo con base en la clasificación funcional y no tecnológica propuesta por el modelo HISA (HealthCare Information System Architecture) del Comité Europeo de Normalización, que es un comité técnico dedicado a temas relacionados con la informática médica (CEN/TC251) (Ceusters y otros, 1997).

Estos componentes no solo están conformados por piezas de software, es decir aplicativos, sino también por recursos humanos y tecnológicos que deben ser considerados como subsistemas en el sistema de información de la organización. Cada uno de ellos tiene funciones específicas, pero se debe tener en cuenta que la clasificación es subjetiva y sus límites pueden ser difusos. Esta clasificación sirvió como guía conceptual para analizar la conformación de los sistemas de información en salud en general y planificar el desarrollo del propio.

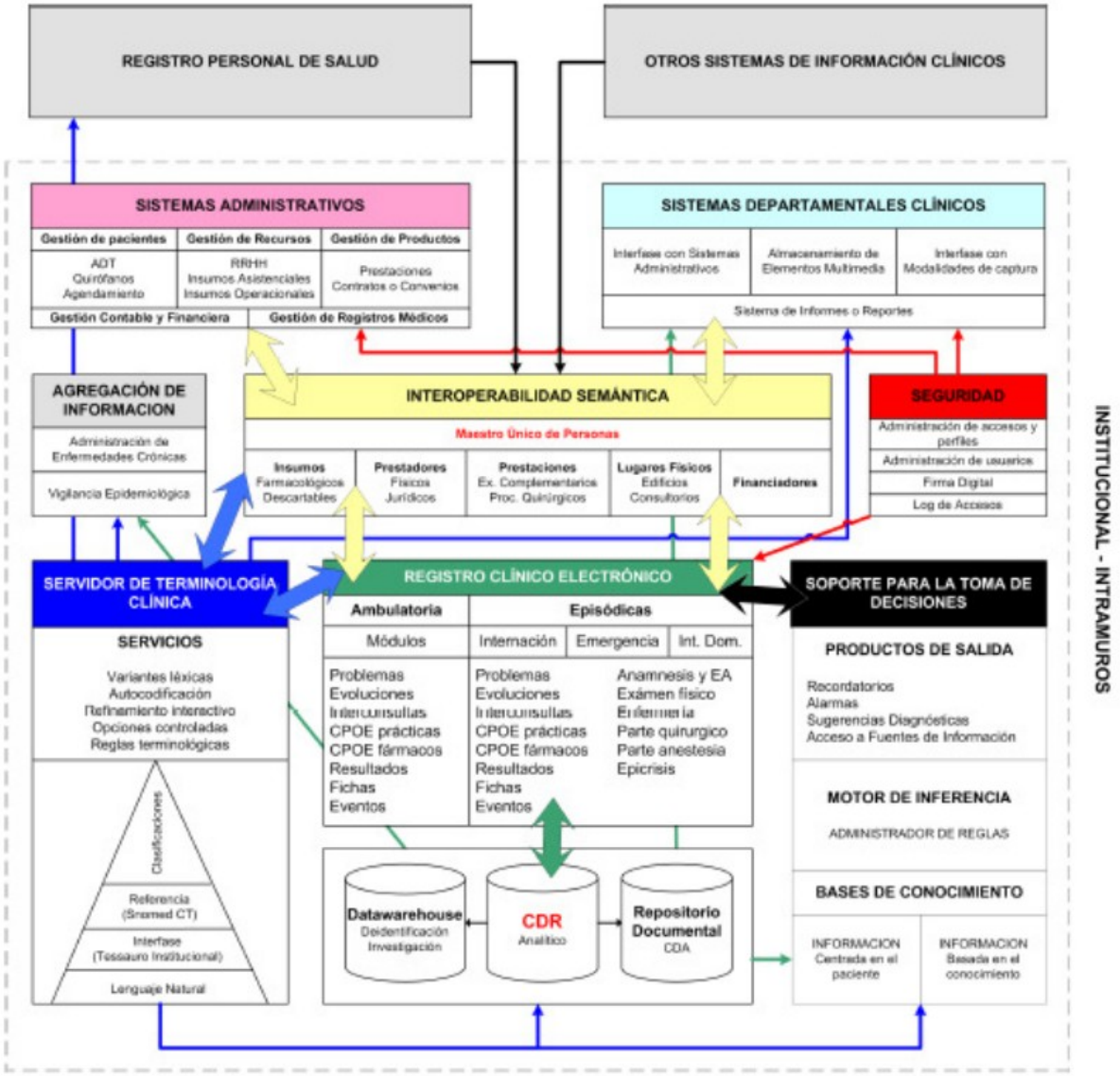

Figura 1: Componentes de Italica. [7]

A continuación se detallan los módulos que forman parte de la arquitectura informática del HIBA, informando sobre el rol que cumple cada uno dentro de la arquitectura presentada. La descripción realizada de cada módulo se toma en base a la documentación provista por el Hospital Italiano [7] y por consultores de la CEPAL que estudiaron el caso del HIBA [8].

Sistemas Administrativos

SISTEMAS ADMINISTRATIVOS		
Gestión de pacientes	**Gestión de Recursos**	**Gestión de Productos**
ADT Quirófanos Agendamiento	RRHH Insumos Asistenciales Insumos Operacionales	Prestaciones Contratos o Convenios
Gestión Contable y Financiera | **Gestión de Registros Médicos**		

Los sistemas administrativos en salud son los que poseen mayor presencia dentro las instituciones sanitarias. Se encargan de la gestión de personal, pacientes, insumos, registros médicos y la gestión contable y financiera. Generalmente son sistemas orientados a la parte contable y al control de stock. A continuación presentamos un resumen de los sistemas que forman parte de este módulo.

- **Gestión de Personas:**
  - Gestión de pacientes, que implica la administración del ingreso, traslado y egreso de los pacientes. Estos sistemas se identifican con la sigla ADT (Admisión, Traslado y Alta)
  - Gestión de camas, que permite llevar el registro actualizado y confiable de las camas libres y ocupadas en cada sala, además de registrar que paciente ocupa cada cama.
  - Gestión de Agendas o Citas, que maneja la programación y reservas de las citas, además de la administración de los pacientes en la sala de espera.
- **Gestión de Insumos:** Estos sistemas administran la logística de los insumos asistenciales, desde la adquisición, ingreso y almacenamiento hasta su distribución. Un ejemplo es el sistema de farmacia que se utiliza para administrar tanto los insumos farmacológicos (compuestos con sustancias farmacológicas) como los no farmacológicos (como por ejemplo material descartable o prótesis). La importancia del sistema de farmacia reside en que aparte de su función de gestión logística de insumos, también es el encargado de dispensar los medicamentos prescritos por los médicos en la Historia Clínica Electrónica.
- **La gestión de productos:** Se deben, en primer lugar, definir los productos de servicios que pretende entregar la institución y ajustarlos según los datos demográficos y clínicos. Los aplicativos que se encargan de administrar las prestaciones o productos de servicio tienen la función de dar el alta a nuevas prestaciones individuales o agrupadas, y en general estos se relacionan con los sistemas que administran los contratos o convenios con los financiadores que darán cobertura a los productos y los contratos o convenios con los prestadores, quedando de esa forma definida las reglas para devengar los honorarios y el pago a prestadores.
- **La gestión de registros médicos:** Las instituciones que tienen registro clínico en papel utilizan estos aplicativos para dar soporte al proceso de apertura y seguimiento de las historias clínicas.
- **La gestión contable y financiera:** Son los aplicativos dedicados a la facturación de los productos entregados que se interrelacionan directamente con el resto de los sistemas de gestión en la capa administrativa. Contiene, entre otros módulos, el de administración del cobro de los productos, liquidación de honorarios y pago a proveedores; contabilidad y balance; presupuesto y planificación financiera.

Sistemas Departamentales Clínicos

SISTEMAS DEPARTAMENTALES CLÍNICOS		
Interfaces con Sistemas Administrativos	Almacenamiento de Elementos Multimedia	Interfaces con Modalidades de Captura
Sistema de Informes o Reportes		

Dentro de esta área deberíamos encontrar los sistemas de informes y reportes sobre datos clínicos. El objetivo primario es dar soporte a los procesos relacionados y aumentar su productividad. En este módulo encontramos los sistemas de laboratorio de análisis clínicos (LIS) y los sistemas de información en radiología (RIS).

Estos sistemas son los que apoyan a las necesidades de información de departamentos clínicos individuales dentro de una organización de salud. Los departamentos clínicos se conocen como servicios auxiliares y en su gran mayoría cuentan con sistemas de información dedicados con un alto nivel de desarrollo y complejidad. Estos servicios son los encargados de llevar a cabo los estudios complementarios solicitados por los médicos.

Cada uno de estos servicios auxiliares tiene su propia forma de plasmar los resultados según las

necesidades particulares de cada especialidad. De este modo, se pueden encontrar desde largos textos narrativos sin estructura alguna hasta tablas con mediciones, gráficos, ondas o imágenes. Ejemplos claros de sistemas departamentales son los Laboratory Information Systems (LIS) y los Radiologic Information Systems (RIS), sistemas dedicados a los laboratorios de análisis clínicos y departamentos de diagnóstico por imágenes, respectivamente.

- **LIS:** Las funcionalidades mas relevantes de un LIS son la identificación del paciente, generalmente por medio de interfaces con los sistemas administrativos; la identificación de la muestra; la adquisición de datos del resultado; el procesamiento de datos y el mantenimiento de registros; la generación de informes y/o reportes; la facturación de la práctica; el control de calidad y la presentación de informes de gestión. Se espera, en el mejor caso, que los LIS interoperen con el repositorio de datos clínicos y transmitan los resultados obtenidos o en su defecto, se definen interfaces particulares encargadas de mostrarlos. Otra alternativa es la de imprimir directamente los reportes obtenidos.
- **RIS:** Se encargan de administrar la lista de pacientes por medio de la agenda de estudios; relacionarse por medio de interfaces con los sistemas centrales de información (HIS) para tomar datos de identificación y facturación; comunicarse con las modalidades de captura digitales como tomógrafos computarizados y resonadores magnéticos o con los sistemas de digitalización de imágenes analógicas a través de estándares como el DICOM, y almacenar las imágenes obtenidas mediante sistemas de almacenamiento y comunicación de imágenes (PACS, por su sigla en inglés Picture Archiving and Communications Systems). Además, están enlazados a aplicativos que permiten generar reportes o informes.

Agregación de Información

AGREGACIÓN DE INFORMACIÓN
Administración de Enfermedades Crónicas
Vigilancia Epidemiológica

La agregación de información abarca áreas de gran interés como ser la vigilancia epidemiológica y el seguimiento de enfermedades crónicas. Diferentes estudios han demostrado que el costo en tratamientos de enfermedades crónicas y atención de enfermedades descubiertas a etapas tardías son muy superiores a los costos de prevención y mantenimiento de enfermedades en un nivel controlado. Los sistemas informáticos deberían ofrecer herramientas a los profesionales de la salud para que puedan hacer un buen seguimiento de los pacientes, un seguimiento proactivo que permita llegar al usuario antes de que tenga que recurrir a asistirse al centro de salud.

El módulo de Agregación de Información, es un componente que administra la información ya no de una persona en particular sino de un grupo de ellas, enroladas según diferentes criterios (por ejemplo, patologías crónicas, neoplasias o enfermedades infecto-contagiosas, entre otras). Para ello es necesario tener información controlada correctamente (repositorio de datos clínicos (CDR) controlado en su terminología) y de calidad sobre los pacientes. Los servicios terminológicos cobran mayor importancia, ya que permiten el análisis poblacional y posibilitan la agregación de los datos acorde a las distintas necesidades.

En el caso de los pacientes con enfermedades crónicas, se busca que el factor de atención de un paciente no sea por su demanda sino que provenga del resultado previo de un análisis que se obtuvo a través del componente de agregación de la información, lo que hace que el sistema sea pro-activo en la atención del paciente en lugar de actuar en forma pasiva esperando que sea el paciente quién solicite atención.

Interoperabilidad Semántica

INTEROPERABILIDAD SEMÁNTICA				
Maestro Único de Personas				
Insumos Farmacológicos Descartables	**Prestadores** Físicos Jurídicos	**Prestaciones** Ex. Complementarios Proc. Quirúrgicos	**Lugares Físicos** Edificios Consultorios	**Financiadores**

La complejidad de los sistemas de salud hace que sea imposible informatizar todas las áreas en un único paso. La informatización del sistema debe ser un proceso continuo con un plan a largo plazo que priorice las necesidades primarias. Para poder realizar desarrollos independientes que luego se interconecten, necesitamos cierto grado de interoperabilidad entre todos los procesos. Como punto de partida deberíamos pensar en contar con un Maestro Único de Personas, cuya responsabilidad será la de identificar unívocamente a cada persona que tenga un rol dentro de los procesos asistenciales. El maestro de personas permite intercambiar información entre distintos procesos asegurando que los datos van a poder relacionarse correctamente. De esta forma los sistemas pueden colaborar entre ellos y asegurar la correctitud de la información intercambiada.

Los sistemas que ya se encuentran en funcionamiento en las instituciones de salud son conocidos como sistemas heredados o "*legacy*". El principal problema que presentan es que para obtener una adecuada interoperabilidad física y semántica, cada uno utiliza distintas plataformas tecnológicas y sus propios identificadores sin diccionarios comunes. Informatizar la capa clínica busca que los actos de cada profesional de la salud y las características de cada paciente estén almacenados en un sistema de información centrado en el paciente; es decir, el acto médico es colocado como eje central de su modelo de información. Para cumplir este objetivo es necesario rediseñar los sistemas heredados hacia una arquitectura integrada en un ambiente interoperable y centrado en el registro médico.

Seguridad

SEGURIDAD
Administración de accesos y perfiles
Administración de usuarios
Firma Digital
Log de Accesos

La información manejada por los sistemas informáticos en salud es altamente sensible y se debe garantizar que únicamente las personas autorizadas tengan acceso a información confidencial. Además, muchos actos médicos llevan consigo una responsabilidad penal del profesional que se encuentra a cargo, por lo tanto es fundamental contar con mecanismos que responsabilicen en forma unívoca y fehaciente al profesional actuante.

Para cumplir con todos estos requerimientos se debe contar con mecanismos de firma digital en los sistemas informáticos, así como procesos de validación y autenticación de usuarios en el manejo de aplicaciones con roles diferenciados y restricciones de acceso.

La seguridad debe ser entendida como un todo y no solamente como una problemática técnica o legal. Este componente se encarga también de administrar los permisos necesarios para el acceso a la información clínica y asegurar que solo los usuarios autorizados puedan acceder a los datos de los pacientes, como así también, el proceso de firma electrónica o digital de los documentos clínicos almacenados en el repositorio mediante estándares de encriptación. La firma digital es una herramienta tecnológica que garantiza la autoría e integridad de los documentos digitales, otorgándoles la misma validez que aquellos firmados en papel.
Servidor de Terminología Clínica

La terminología es uno de los ejes en los que se basa la interoperabilidad. Los sistemas que interoperan deben ser capaces de reconocer los términos semánticamente iguales para procesar la información intercambiada. Un servidor de terminología clínica (STC) agrupa una serie de conceptos y términos relacionados y provee herramientas para acceder a los mismos, de forma que cada sistema puede usar su codificación y contar con un término normalizado dentro del STC para cada una de sus entradas. Contar con este tipo de servidores en la arquitectura de un sistema informático de salud, resuelve la manipulación del gran volumen de información que se maneja en los centros asistenciales. En cada región o país, se utilizan variantes léxicas para referir a términos clínicos. Un sistema informático que pretenda no ser invasivo en la interacción computadora-humano, debería contar con herramientas que permitan al usuario manejar la terminología que le resulte más familiar pero a la vez debe ser capaz de reconocer estas variantes léxicas para poder ayudar a realizar diagnósticos o actualizar al usuario sobre las últimas investigaciones realizadas sobre una determinada patología.

La representación del conocimiento médico es particularmente compleja debido a la utilización de un lenguaje que se caracteriza por depender del contexto; ser muy especializado; utilizar jergas y acrónimos; carecer de definiciones rigurosas y, por último, ser ambiguo. Dadas éstas características es fundamental para que la computadora entienda los datos clínicos del paciente que se utilice la codificación. Codificar es el proceso de organizar, categorizar y dar sentido a los datos. La codificación se debe expresar en varios dominios, como ser lista de problemas, diagnósticos de información internación, fármacos que consumen o exámenes complementarios.

El control de la información y su interpretación por parte de las computadoras permite el intercambio real de datos entre los actores del sistema de salud y sus sistemas de información. Además, posibilita la interacción con bases de conocimiento y sistemas de soporte para la toma de decisiones, así como la realización de análisis epidemiológico, de calidad y la gestión e investigación con información agrupada de mejor forma.

El problema que se presenta al codificar es la pérdida de información, el texto libre siempre es más expresivo y representa el contexto de mejor forma que los códigos de las clasificaciones. Una solución posible a este punto son los vocabularios de interfaz, que son listados de términos predefinidos u ordenados de una determinada manera los cuales permiten a los profesionales interactuar con la computadora a través de textos narrativos predefinidos y codificados.

Representar la información de manera controlada puede lograrse a partir de agrupamientos, clasificaciones, nomenclaturas, terminologías o vocabularios y mediante el lenguaje natural. El HIBA propone la siguiente representación del conocimiento médico.

Agrupaciones	Subconjunto de las calificaciones que agrupa ítems contenidos en las mismas según criterios predefinidos.
Clasificaciones	Sistema ordenado de conceptos pertenecientes a un dominio, con principios de orden implícito o explícito
Nomenclaturas	Es una lista de términos oficialmente aprobada para ser usada en un campo o dominio.
Terminologías	Todas las palabras tienen un significado particular en el campo o dominio especifico
Vocabularios	Todas las palabras usadas en un campo o dominio especifico.
Lenguaje Natural	Todas las palabras actualmente usadas por las personas, tanto médico como pacientes.

El HIBA propone la estructura del lenguaje y su adaptación a los diferentes dominios del conocimiento sanitario a través de una pirámide como se detalla en la Figura 2

Figura 2: Estructura del lenguaje propuesta por el HIBA [8].

Lenguaje natural: Por definición es no controlado y surge en un ámbito determinado del conocimiento. Los médicos se expresan en lenguaje natural.

Vocabulario de interfaz: Es el vocabulario utilizado por los médicos para el registro. Es importante que este vocabulario sea representativo del dominio y la jerga local y contenga todos los términos o sinónimos con los que se conoce una entidad o concepto clínico.

Terminologías de referencia: Es la forma de almacenar información en su máximo nivel de detalle y la representan de manera exacta y entendible para las computadoras, completando el conocimiento de un dominio determinado, incluidas sus entidades e ideas, y sus interrelaciones. Se puede crear un vocabulario de referencia ad hoc o utilizar un vocabulario estándar como puede ser SNOMED CT.

Vocabularios de salida: A través de los mapeos y utilizando el vocabulario de referencia, el contenido registrado libremente es representado en clasificaciones o terminologías que hacen posible su análisis según el nivel de detalle requerido.

Registro Clínico Electrónico

REGISTRO CLÍNICO ELECTRÓNICO			
Ambulatoria	**Episódicas**		
Módulos	Internación	Emergencia	Int. Domiciliaria
Problemas Evoluciones Interconsultas CPOE Prácticas CPOE Fármacos Resultados Fichas Eventos	Problemas Evoluciones Interconsultas CPOE Prácticas CPOE Fármacos Resultados Fichas Eventos	Anamnesis y EA Examen Físico Enfermería Parte Quirúrgico Parte Anestesia Epicrisis	

La historia clínica electrónica (HCE) es el sistema modular en cualquier sistema informático de salud. Allí se encuentran los registros de cada episodio clínico protagonizado por un paciente dado, así como sus enfermedades crónicas y hasta sus antecedentes familiares. Una HCE permite a cualquier médico conocer a fondo a su paciente sin necesidad de hacerle un cuestionario que a veces, por la distancia entre los léxicos hablados por el profesional y el paciente, no conduce a una visión completa y correcta.

El desafío de una HCE es almacenar la mayor cantidad de información a la vez de brindar distintas vistas de esta información de forma eficiente. Debe ser el lugar primario para la carga de toda la información clínica.

Soporte para la toma de decisiones

SOPORTE PARA LA TOMA DE DECISIONES	
PRODUCTOS DE SALIDA Recordatorios Alarmas Sugerencias Diagnósticos Acceso a Fuentes de Información	
MOTOR DE INFERENCIA ADMINISTRACIÓN DE REGLAS	
BASES DE CONOCIMIENTO	
INFORMACIÓN Centrada en el paciente	INFORMACIÓN Basada en el conocimiento

Actualmente se observa en nuestro país, un fenómeno de multi-empleo a nivel de profesionales y técnicos de la salud. Esta dedicación horaria extensiva hace que la actualización de conocimiento mediante asistencia a seminarios o la lectura de revistas especializadas sea relegada. Al mismo tiempo, la era de la información en la que estamos viviendo, ofrece una continua actualización de conocimiento mediante investigaciones realizadas en todas partes del mundo. Un sistema informático en salud debería contar con herramientas que permitan asistir al profesional actuante para ayudarlo en la realización del diagnóstico y prevenir, por ejemplo, errores en la administración de medicamentos, al advertir sobre drogas que no se pueden combinar e incluso advertir sobre las drogas que pueden provocar una reacción alérgica al paciente.

Estos sistemas, denominados CDSS, proveen al médico o paciente información específica procesada de forma inteligente y en el momento oportuno, para garantizar un mejor proceso de atención y optimizar la toma de decisiones para el cuidado de los pacientes. Surgen a raíz de la preocupación constante del equipo de salud para disminuir el error médico mejorar los procesos de salud y garantizar la calidad del cuidado de sus pacientes. Además generan salidas como por ejemplo, alarmas, que permiten al medico alertar posibles errores como pueden ser dosis inadecuadas de un determinado fármaco; recordatorios como prácticas preventivas; sugerencias de diagnósticos; acceso a fuentes de información útiles según el contexto del médico. Los CDSS constan de un motor de inferencia que alberga las reglas médicas y se alimenta de las bases de conocimiento con información del paciente e información propia del dominio (información basada en el conocimiento universal).

Repositorio de Datos Clínicos (CDR)

El repositorio de datos clínicos, CDR por sus siglas en inglés (Clinical Document Repository), almacena los datos provenientes de laboratorio, farmacia y radiología. Estos datos pueden llegar en formato de imagen o texto, el desafío para el CDR es poder guardar la información en forma normalizada de manera que esté disponible para cualquier otro sistema que desee consultarlo sin importar el formato en que fue generada la información. El CDR posibilita el análisis secundario de la información tanto para investigación como para dar soporte a los aplicativos de DataWarehouse. Junto con los sistemas de gestión administrativa, conforman los tableros de comandos para la toma de decisiones a nivel de planificación estratégica y de gestión clínica de la organización.

Este componente es la clave principal en el proceso de informatización de la capa clínica. El CDR es el sistema que utilizan los actores del sistema de salud para registrar cada acto asistencial. La información almacenada debe tener la correcta asignación de identificadores, estos identificadores son controlados por por el componente de interoperabilidad semántica (por ejemplo, en la asignación de identificadores de

pacientes y médicos actúa el Maestro Único de Personas) y el componente de terminología clínica.

En el CDR también se almacenan los documentos clínicos enviados por el componente de Servicios Departamentales, como por ejemplo los exámenes complementarios y los archivos multimedia. Esta base documental también permite almacenar documentos firmados digitalmente.

Registro Personal de Salud

Este tipo de sistemas funciona como un portal personal de salud donde la información se muestra según la perspectiva y necesidades del paciente. Brinda alarmas y recordatorios relacionados al auto-cuidado del paciente. La información de todos los componentes ya vistos se muestra desde la perspectiva y necesidades del paciente. Por ejemplo, el componente administrativo debe brindar al paciente la posibilidad de ver los turnos otorgados y solicitar la reserva en la agenda de consultas y/o prácticas; el repositorio de datos clínicos deberá seleccionar que datos mostrar al usuario sobre la información de exámenes complementarios o lista de problemas, los sistemas para la toma de decisiones, por su parte, deberán otorgar acceso al paciente a fuentes de información preseleccionados según los problemática, como así también generar recordatorios y alarmas relacionadas con el auto-cuidado que debiera tener el paciente.

2.3 Informática en salud en Uruguay

En Uruguay se han desarrollado algunas iniciativas independientes en lo referente a sistemas informáticos para el área de salud. Cada uno de los actores, tanto a nivel público como privado, desarrollaron estos sistemas según sus necesidades específicas pero sin ningún tipo de coordinación entre ellos. A comienzos de los años 2000, el gobierno intenta coordinar el sistema de salud y así surge el SNIS (Sistema Nacional Integrado de Salud), el cual entra en vigencia el 1º de enero de 2008 [9].

Uno de los principales ejes del SNIS es el cambio del modelo asistencial hacia una orientación que privilegie la atención integral del individuo. Ésto incluye acciones que promuevan la prevención, tratamiento precoz y seguimiento de enfermedades crónicas. Estas actividades se desarrollarán en el marco de la estrategia de Atención Primaria, asegurando la mayor capacidad resolutiva del primer nivel de atención [10]. El SNIS procura distribuir de forma equitativa los fondos destinados al sistema de salud. Con este objetivo, crea un Fondo Nacional de Salud para recibir y administrar los recursos que se destinan al pago de los distintos prestadores de salud [10].A continuación se detallan algunas de las iniciativas privadas y los lineamientos del SNIS.

2.3.1 eSalud en el contexto del SNIS

La eSalud se denomina a la aplicación de las Tecnologías de la Información y las Comunicaciones (TIC) a la salud. Engloba múltiples usos posibles. Sus aplicaciones abarcan muchas, si no todas, las actividades relacionadas con la prevención, diagnóstico, tratamiento y monitoreo, así como a la planificación y control de gestión de los sistemas sanitarios. Bajo el concepto de eSalud, caben aplicaciones tan diversas como la historia clínica electrónica, los distintos tipos de servicios de telemedicina, la vigilancia epidemiológica, los portales de salud, los sistemas de gestión y los programas de educación a distancia en salud. A estos servicios hay que añadir varias necesidades básicas, como la infraestructura tecnológica sobre la que deben funcionar, la interoperabilidad que permite el intercambio de datos entre sistemas y las medidas de seguridad y protección de la información. [3]

En Uruguay respecto a la eSalud y en el contexto del SNIS se pretenden grandes desafíos y comprende diferentes áreas. En el Plan Director de Informática del MSP para el período 2005-2009 se priorizaron algunas líneas estratégicas, dentro de las cuales mencionamos:

- Construir sistemas de información en salud que apoyan las funciones esenciales del MSP y ASSE.
- Avanzar en la definición de estándares de contenido y de interoperabilidad.
- Participar en los proyectos de desarrollo e implantación de sistemas de información con otros

organismos del Estado a través de ámbitos de coordinación permanente.
La definición de estándares de contenido y de interoperabilidad cruza todos los proyectos de informática aplicada a la salud.

La creación de la SUEIIDISS integrándose como capítulo de HL7 Uruguay en el 2005 significó un importante hito en este sentido. El MSP es miembro institucional de esta sociedad y ha trabajado en varios SCT.

De los emprendimientos derivados del Plan Director se destacan el control del embarazo y del niño (SEVEN) que comprende el certificado de nacido vivo electrónico (CNV-e) , el certificado de defunción electrónico (CD-e), el Sistema de Información Perinatal (SIP) y el programa ADUANA (seguimiento del crecimiento y desarrollo del niño hasta los 2 años) [11].

2.3.2 Gobierno electrónico

El gobierno electrónico permite a los ciudadanos relacionarse electrónicamente con las Administración Públicas. Para lograr este objetivo, las administraciones deben relacionarse entre sí con el fin de simplificar los trámites, servicios y procedimientos [12].

Figura 3: Gobierno electrónico.[12]

2.3.2.1 Agenda digital

En los últimos años la mayoría de los países de América Latina definió estrategias y políticas que tienden a poner en práctica políticas públicas en Tecnologías de la Información y las Comunicaciones (TIC), considerándolas como medios para el desarrollo de la sociedad. [13]

Las agendas politicas en TIC son necesarias dado que con ellas se optimizan los procesos productivos y organizativos, generando mayor valor economico y social junto con un efecto positivo en el crecimiento de cada país. Es muy importante plantear estas politicas publicas para que tiendan a reducir la “brecha digital” existente entre América Latina y los países desarrollados de Europa, y promover la cración de sociedades de la información.

En la Tabla 1 se presenta el estado de las políticas públicas para la creación de sociedades de la información en Uruguay a junio de 2009, detallándose su grado de progreso, las características de la agenda política a esa fecha, los documentos previos y el marco institucional concebido para la puesta en práctica de la política digital.

De los veintiún países de Iberoamérica para los cuales se cuenta con esta información, en el 2009, dieciséis se encontraban en el desarrollo de políticas digitales de primera generación y cinco de segunda; Uruguay se encuentra dentro de estos últimos.

País	Característica del documento actual			Antecedentes y estado del proceso		Marco institucional de la estrategia actual		
	Nombre del documento	Período de vigencia	Tipo de Documento	Documento anterior y año de elaboración	Progreso de la política de TIC	Coordinador principal	Conducción estratégica	Conducción operativa
Uruguay	Agenda Digital Uruguay	2008 - 2010	Definitivo	Agenda Digital Uruguay-2007-2008	2ra Generación - Implementación	Agencia para el Desarrollo del Gobierno de Gestión Electrónica y la Sociedad de la Información y del Conocimiento (AGESIC)	Presidencia de la República	Agencia para el Desarrollo del Gobierno de Gestión Electrónica y la Sociedad de la Información y del Conocimiento (AGESIC)

Tabla 1: Estrategia nacional de TIC en Uruguay. Junio, 2009 [13]

El IV Encuentro Nacional de Gobierno Electrónico realizado los días 23 y 24 de noviembre de 2011 culmina con el mensaje: “Construyendo un Gobierno Electrónico entre todos y para todos” [14], en este evento se realizan varios anuncios relacionados con la Sociedad de la Información. Entre ellos se lanzó la Marca “*País Uruguay Digital*” y se comunica la aprobación por decreto de Presidencia de la República de la Agenda Digital 2011-15, desarrollada generando acciones a favor de avances de la Sociedad de la Informacion y el Conocimiento.

La Agenda Digital 2011-2015 [15] es un compromiso entre diferentes instituciones del sector pblico, el privado y la academia a nivel nacional, trabajando con un objetivo en común para llegar al Uruguay Digital. [16] En la Agenda Digital se presentan 15 objetivos para el 2015 planteando un compromiso de “construir una Sociedad centrada en la persona, integradora, y orientada al desarrollo, en que todos puedan crear, consultar, utilizar y compartir la información y el conocimiento, para que las personas, las comunidades y los pueblos puedan emplear plenamente sus posibilidades en la promoción de su desarrollo sostenible y en la mejora de la calidad de vida” [15]. La Agenda sigue las siguientes líneas de estrategia para inspirar y contribuír a su cumplimiento: equidad e inclusión social, participación cuidadana, transformación del estado, impulso a la educación, innovación y generación de conocimiento, integración territorial e inserción internacional.

Dentro de las áreas donde se plantearon metas y se realizarán acciones, se encuentra el área de la salud. En esta área se plantea aplicar Tecnologías de la Información y las Comunicaciones (TIC) para la mejora de la calidad de los servicios médicos. El objetivo es crear una red avanzada para la salud e historia clínica electrónica integrada a nivel nacional. La conectividad de los hospitales y la informatización de las historias clínicas, reducen los costos y mejoran la calidad de la atención recibida por los usuarios. También la telemedicina y la integración a redes regionales de medicina son claves para el avance en este tema.
Las metas dentro de esta área planteadas son las siguientes:

- Crear en el 2012 una red de datos de salud, integrada a la Red Académica Avanzada Uruguaya (RAU2) [17] para el envío y procesamiento de imágenes, dar soporte a la plataforma de historias clínicas y facilitar la colaboración e investigación a nivel nacional y regional.
- Implantar sobre la red un sistema de Telerradiología en al menos 50 centros de salud para el 2014.
- Crear y administrar a partir del 2012 una plataforma de historias clinicas electrónicas, que asegure la disponibilidad de la información con los mecanismos necesarios de seguridad y protección de la privacidad.
- Crear en el período de la Agenda el Banco Nacional de Historias Clínicas Electrónicas cuyo cometido principal será la administración de la plataforma.

2.3.2.2 AGESIC

La Agencia para el desarrollo del Gobierno de Gestión Electrónica y la Sociedad de la Información y del Conocimiento (AGESIC) es un organismo que depende de Presidencia de la República y tiene como objetivo procurar la mejora de los servicios al ciudadano a través de las facilidades brindadas por las tecnologías de la información y las comunicaciones [18].

2.3.2.3 RedUy

El gobierno uruguayo ha creado la RedUy como medio físico para lograr una plataforma de Gobierno Electrónico. Se trata de una red de alta velocidad que permite que cada organismo se conecte, a través de un único enlace, a todos los organismos que forman parte de la red. Esto permite que las instituciones interactúen entre si para intercambiar información con el objetivo de crear una Administración Pública eficiente y centrada en el usuario.

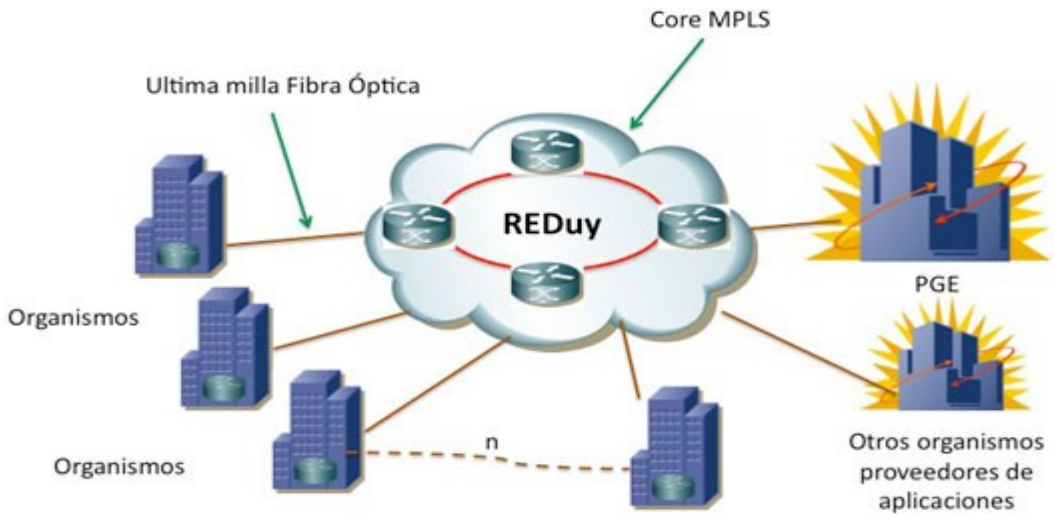

Figura 4: Diagrama de comunicaciones en la RedUy [0]

Como parte de las instituciones del Estado, ASSE tiene el desafío de consumir y prestar servicios dentro de RedUy. Para ello debe cumplir con los requisitos de conexión y publicación de servicios establecidos por la AGESIC, que es quién implementa y administra la red. A continuación, se describen, a grandes rasgos, los requisitos que se deben cumplir para proveer y consumir servicios dentro de RedUy.

Para integrar la RedUy, ASSE debe proveer sus funcionalidades de negocio a través de la tecnología de Web Services. Los mismos pueden ser implementados utilizando distintas tecnologías, pero para mejorar el nivel de interoperabilidad, se requiere que se ajusten a los perfiles Basic Profile [19] y Basic Security Profile [20] definidos por la organización Web Services Interoperability (WS-I).

El mecanismo para consumir un servicio publicado en la Plataforma de Gobierno Electrónico (PGE), es el siguiente:

- ASSE debe solicitar una clave de seguridad al Sistema de Tokens de Seguridad (STS) de la PGE, utilizando el estándar WS-Trust [21].
- En la solicitud es necesario incluir otra clave de seguridad que especifique, entre otros datos, el rol de usuario con el que se quiere acceder al servicio. Esta clave debe estar especificada utilizando el estándar SAML v1.1 o v2.0 y además debe estar firmada electrónicamente por ASSE.
- La comunicación entre las aplicaciones cliente y el STS de la PGE se debe realizar a través de HTTPS.

Para que ASSE pueda comunicarse con la PGE, ya sea para proveer o consumir servicios, es necesario que:

- Esté conectado a la RedUy.
- Los *firewalls* de RedUy estén configurados para habilitar el tráfico de red requerido.
- Se puedan establecer conexiones SSL entre ASSE y la PGE.

2.3.2.4 Servicios publicados y aplicaciones disponibles

Algunos de los servicios y aplicaciones publicados en la RedUy se detallan a continuación: [22]

Descripción de la Aplicación o Servicio	Organismo que publica la aplicación y el servicio
SIIF – Sistema de Información Integrada Financiera	CGN
SLH - Sistema de Liquidación Haberes	CGN
SPA – Sistema de Presentación del Articulado	CGN
Web Service - SIGGA (Sistema para la Gestión del Servicio de Garantía de Alquileres)	CGN
Sistema de Expedientes GEx (GExweb)	Presidencia
Publicación del servidor correspondiente al Sistema Nacional de Siniestros de Tránsito (SNST) para la UNASEV (Unidad Nacional de Seguridad Vial)	Presidencia-UNASEV
Portal informativo para los organismos socios del Proyecto SINARE y aplicación para uso de las oficinas de Empresa en el Día portaldelaempresa.red.uy	Presidencia-SINARE
Sistema de Expediente Electrónico para la Administración Pública	AGESIC
Web Service para la Asignación de la Cédula de Identidad	DNIC-MI
Web Service para Consulta de antecedentes para emisión de pasaporte a uruguayos en el exterior	DNIC-MI
Aplicación SGREC (Sistema de Gestión del Registro de Estado Civil) de la DGREC	MEC
Portal de Dirección Nacional de Aduanas www.aduanas.red.uy	DNA-MEF
Sistema Informático "Lucía", para el control de las operaciones aduaneras de importación, exportación y tránsito de mercaderías en territorio uruguayo, https://www.aduanas.red.uy/lucia/	DNA-MEF
Publicación de LDAP de DINADE (Dir. Nal. de Deportes) para ser consumido por el MTD (Ministerio Turismo y Deportes)	DINADE-MTD
Sistema para Gestión de Afiliados a ASSE	**ASSE**
Web Service para consulta de datos sobre Afiliados a ASSE	**ASSE**
Web Services para Consultas de datos por RUT	DGI
Portal del Correo http://www.correo.red.uy/	CORREO
Servidor ssh para que diferentes organismos puedan subir o bajar archivos hacia/desde el Correo	CORREO
Sistema para peticiones de retiro de envíos online	CORREO

2.3.3 Actores privados

En una visita a SUAT (Servicios de Urgencia, Asistencia y Traslado) pudimos ver algunos de los desarrollos informáticos que la institución ha realizado a lo largo de 15 años. Entre los más destacados está la historia clínica electrónica que se utiliza en todos los consultorios y la utilización de la receta electrónica. La receta electrónica permite que el paciente vaya a cualquier farmacia con acuerdo y retire los medicamentos enviados por el médico. El sistema con el que cuenta el médico al expedir la receta electrónica, permite evitar errores al momento de su creación, ya que el médico elige exactamente el medicamento deseado, su dosificación y formato (gotas, pastillas, pomada, etc). La aplicación utilizada en SUAT, cuenta con un sistema de alarmas y notificaciones para ayudar al médico durante su consulta. Por ejemplo, es capaz de informar acerca de medicamentos incompatibles entre si o de nuevos estudios científicos referentes a las patologías que pueda presentar el paciente. La aplicación utiliza HL7 para almacenar los registros clínicos.

Otro de los actores privados del sistema de salud es la Federación Médica del Interior (FEMI), la cual está integrada por 23 prestadores integrales que son instituciones privadas sin fines de lucro de todos los departamentos de Uruguay actualmente atiende a una población aproximada de 700 mil afiliados. En Montevideo cuenta con el Sanatorio Americano adquirido en 1993 [11]. A fines de marzo del 2008 se firmó un acuerdo entre el Banco Interamericano de Desarrollo (BID) y FEMI en donde se dio inicio formal a un proyecto de informatización que apunta a la gestión clínica. El mismo fue producto de la insuficiente integración de la información de las instituciones de FEMI y de los requerimientos del SNIS. Mediante este proyecto, las instituciones de FEMI contarán con una historia clínica electrónica federal integrada a los sistemas informáticos de cada una de ellas y con un sistema de información gerencial que permitirá el análisis clínico, epidemiológico y contable. Las principales líneas de acción de este proyecto son [11]:

- **Especificación y desarrollo de sistemas.**
  Con respecto a las especificaciones del sistema de HCE Federal, contaron con la ayuda del Hospital Italiano de Buenos Aires en el rediseño de procesos de las áreas clínicas donde la HCE sería implementada (en particular, en sala de emergencias y ambulatorio).
- **Alineación y estándares para la interoperabilidad.**
  Con respecto a este punto podemos decir que avanzaron en los servicios de identificación de personas y de terminología médica, en donde se pretende adaptar a la realidad uruguaya, el servicio utilizado por el Hospital Italiano de Buenos Aires. También existe consenso en el uso de HL7 CDA como estándar de mensajería, recomendado por SUEIIDISS.
- **Gestión del cambio cultural.**
- **Infraestructura.**

2.3.4 Actores públicos

Hasta el año 2005, el Estado uruguayo, a través del Ministerio de Salud Pública (MSP) y la Administración de Servicios de Salud del Estado (ASSE), tenía a su cargo la atención de la población de menores recursos; aquella población con capacidad de pago era delegada al sector privado [23].

ASSE no es el único prestador de salud en el sector público, además existe el Hospital de Clínicas (hospital universitario dependiente de la Facultad de Medicina), Sanidad de las Fuerzas Armadas, Sanidad Policial, Banco de Previsión Social, Banco de Seguros del Estado y los servicios médicos de cada una de las intendencias de cada departamento del Uruguay. También existe el Fondo Nacional de Recursos (FNR) como entidad pública no estatal, que cubre procedimientos altamente especializados a través de un mecanismo de reaseguro universal [23].

A partir del año 2004, con la asunción del primer gobierno de izquierda, se comienza a implementar en el país una “Reforma Sanitaria”. Esta reforma se plantea como metas transformaciones sociales, económicas y políticas y se basa en cambios progresivos y complementarios de los modelos de gestión, financiamiento y atención de la salud. En el año 2007 se crea el Sistema Nacional Integrado de Salud (SNIS) para llevar a cabo estas metas.

En el año 2007, junto con la creación del SNIS, se crea también la Administración de los Servicios de Salud del Estado (ASSE), como un servicio descentralizado que se relaciona con el Poder Ejecutivo a través del Ministerio de Salud Pública (MSP). Entre los cometidos planteados para ASSE, se destacan los siguientes [24]:

- Coordinar con los distintos organismos del Estado que prestan servicios sanitarios de forma de lograr la máxima accesibilidad, calidad y eficiencia evitando duplicaciones y/o superposiciones.
- Formar parte del SNIS y contribuir en su implementación.
- Efectuar y mantener actualizado un diagnóstico sobre el estado de salud de sus usuarios y las condiciones socio-económicas y culturales para detectar y superar las circunstancias que podrían condicionar sus niveles.
- Cubrir y coordinar de forma adecuada el nivel nacional y los niveles departamentales y locales de forma de abarcar las diversas etapas de la atención integral en materia de salud; a la vez de contribuir a la promoción, prevención, diagnóstico precoz y tratamiento oportuno, recuperación y rehabilitación.

A partir de su creación como servicio descentralizado, los usuarios tienen la posibilidad de elegir a ASSE como prestador integral de salud, una opción que hasta este momento no se tenía [23].

La conformación de un sistema mixto público-privado de prestadores de salud que establece el SNIS tiene como objetivo la complementación de servicios prestados entre las instituciones para racionalizar, de forma efectiva y eficaz, la utilización de recursos [23].

Al momento de realizar este informe, ASSE contaba con una Red de Atención Integral a la Salud constituida por 66 Unidades Ejecutoras (UE). Un total de 51 UE se encuentra en el interior del país, de las cuales 18 son Centros Departamentales y 33 Centros Auxiliares. De ellos dependen 228 policlínicas [25].

2.4 Estándares definidos para el área de salud

En esta sección se realiza una breve descripción de algunos de los estándares utilizados en el área de salud, los cuales son citados en distintas partes del documento.

LOINC [26]

La base de datos LOINC proveé un conjunto de nombres y códigos universales para identificar laboratorios y resultados de análisis clínicos. En la actualidad, la mayoría de los laboratorios usa el estándar ASTM 1238 o HL7 para enviar en forma electrónica los resultados clínicos desde el laboratorio hacia los centros de salud. Cada laboratorio codifica las pruebas realizadas según sus códigos internos y los centros de salud poseen los propios. Por lo tanto, al momento de cargar los resultados se deben hacer mapeos entre los códigos recibidos desde el laboratorio y los códigos internos del centro asistencial. Esta problemática es la que resuelve LOINC.

Si al momento de transmitir y compartir información, las partes utilizan códigos LOINC, el problema de conocer los códigos utilizados por cada parte con la que nos comunicamos desaparece. Para cada actor involucrado, la codificación interna es transparente y sólo se deben conocer los códigos del estándar.

El alcance de la codificación LOINC es el de identificar observaciones clínicas o resultados de análisis, no abarca los detalles de cada uno de estos resultados. Estos detalles deberán ser brindados por otros campos del mensaje de intercambio.

Cada código LOINC se corresponde a una prueba unitaria. Por ejemplo, el médico puede solicitar un hemograma completo el cual contiene una lista de análisis individuales. Cada uno de estos análisis tendrá un código LOINC, no así el hemograma completo.

El nombre LOINC completo de un test o una observación clínica contiene cinco o seis partes principales como ser: el nombre del componente (ej: glucosa), la propiedad observada (ej:concentración, masa, volumen, etc), el tiempo de la medición, el tipo de muestra (ej: sangre), la escala de la medición (cualitativa vs cuantitativa) y el método de medición. Por lo tanto, el nombre se puede ver estructurado de la siguiente manera:

<Analyte/component>:<kind of property of observation or measurement>:<time aspect>:
<system (sample)>:<scale>:<method>

Ejemplos de nombres completos pueden verse en el siguiente cuadro:

Sodium:SCnc:Pt:Ser/Plas:Qn
Sodium:SCnc:Pt:Urine:Qn
Sodium:SRat:24H:Urine:Qn
Creatinine renal clearance:VRat:24H:Ur+Ser/Plas:Qn
Glucose^2H post 100 g glucose PO:MCnc:Pt:Ser/Plas:Qn

Tabla 2: Ejemplo de codificación LOINC

HL7 [27]

Health Level Seven (HL7) es una organización acreditada por American National Standars Institute (ANSI) para desarrollar estándares, y actualmente, es la autoridad global en estándares para la interoperabilidad de tecnologías de sistemas de salud. HL7 proveé un framework y estándares para el intercambio, integración y recuperación de información electrónica producida por los servicios de salud.

HL7 CDA [27]

El estándar HL7 Clinical Document Arquitecture (CDA) es un paso muy importante para lograr la interoperabilidad entre sistemas de salud. El CDA es un estándar aprobado por la International Organization for Standarization (ISO) que proveé un modelo de intercambio para documentos clínicos. Este modelo es un primer paso que acerca a las instituciones de salud a la elaboración de un registro médico electrónico.

Los documentos clínicos son la parte central de la historia clínica de un individuo a lo largo de su vida, por lo tanto, la utilización de esta información crítica debe ser independiente de las aplicaciones donde fue creada. El tiempo de vida de esta información es mayor, en algunos casos, al tiempo de vida del software donde se generó. Un documento CDA presenta como características ser persistente, autenticado, proveé un contexto, es íntegro y legible para los humanos y las aplicaciones informáticas.

CDA Release 2 hace uso del XML como base de su estructura, de un vocabulario codificado y del HL7 Reference Information Model (RIM). El RIM es un modelo estático de los sistemas de salud y la información generada por ellos, tal como se ve en el ámbito de desarrollo de los estándares HL7. Este modelo fue construido en colaboración entre el grupo de trabajo de HL7 y todos sus afiliados a nivel mundial. Se trata de un modelo de objetos que representa la información clínica e identifica el ciclo de vida de los eventos relacionados a un mensaje o a un grupo de mensajes relacionados entre sí. El RIM muestra explícitamente las conexiones existentes entre los campos de información contenidos en un mensaje HL7.

ASTM [28]

ASTM Internacional es reconocida como líder a nivel mundial en el desarrollo y la entrega de las normas internacionales de consenso voluntario.

Esta sociedad se formó en 1898 por los químicos e ingenieros de la Pennsylvania Railroad. En el momento de su creación, la organización era conocida como American Society for Testing and Materials (ASTM). Charles B. Dudley, Ph.D.(un químico), fue la fuerza impulsora detrás de la formación de la sociedad. Fue en el 2001 que la sociedad llegó a ser conocida como ASTM International.

ASTM E 1394 91 [28]

Esta norma relaciona la estructura de los mensajes que son transmitidos entre un equipo clínico y una computadora. El protocolo ASTM estandariza y organiza la información clínica a transmitir, estructurando los mensajes en campos (delimitados por el carácter '|') y subcampos (delimitados por el carácter '^').

DICOM

Digital Imaging and Communication in Medicine (DICOM) es un estándar para el intercambio de imágenes médicas que permite manipular, almacenar, imprimir y transmitir imágenes [29]. El estándar incluye la definición de un archivo y un protocolo de comunicación de red, que utiliza TCP/IP para la comunicación entre sistemas [30].

CIE 10 [31]

Es el acrónimo de la Clasificación internacional de enfermedades, décima versión correspondiente a la versión en español de la ICD (International Statistical Classification of Diseases and Related Health Problems). Determina la clasificación y codificación de las enfermedades y una amplia variedad de signos, síntomas, hallazgos anormales, denuncias, circunstancias sociales y causas externas de daños y/o enfermedad.

Fue publicada por la Organización Mundial de la Salud y se utiliza a nivel internacional para fines estadísticos relacionados con morbilidad y mortalidad, los sistemas de reintegro y soportes de decisión automática en medicina. Fue diseñado para promover la comparación internacional de la recolección, procesamiento, clasificación y presentación de estas estadísticas.

La lista CIE-10 tiene su origen en la "Lista de causas de muerte", cuya primera edición se realizó por el Instituto Internacional de Estadística en 1893. La OMS se hizo cargo de la misma en 1948, en la sexta edición, también la primera en incluir causas de morbilidad. A la fecha, la lista en vigor es la décima, y la OMS sigue trabajando en ella.

Estructura básica de la Cie10 Revisión [31]

La CIE (Clasificación Internacional de Enfermedades) es un sistema de clasificación de ejes variables cuyo esquema debe servir a todos los propósitos prácticos y epidemiológicos. Este patrón puede ser identificado en los capítulos de la C.I.E. y hasta el momento es considerado como la estructura más útil que cualquiera de las alternativas que se han probado.

Utiliza un código alfanumérico, con una letra en la 1° posición y números en la 2°,3°, y 4° posición; el cuarto carácter sigue a un punto decimal, los códigos posibles van por lo tanto de A00.0 a Z99.9.

IUPAC [32]

La Unión Internacional de Química Pura y Aplicada (International Union of Pure and Applied Chemistry), IUPAC, desarrolla estándares para la denominación de compuestos químicos. También se encuentra involucrada en el desarrollo de protocolos para procedimientos analíticos y clínicos, y normas que establecen la calidad y el diseño de laboratorios de investigación.

La IUPAC se fundó en 1919 por químicos tanto de la industria como de las universidades, quienes reconocieron la necesidad de establecer estándares globales en la simbología y protocolos operacionales de la química.

EUCLIDES [3]

EUCLIDES (European Clinical Laboratory Information Data Exchange Standard) es un estándar abierto europeo que sirve para la interoperabilidad de diferentes sistemas de información de laboratorios clínicos.

3 Relevamiento de la situación de ASSE

En esta sección se describe el contexto general en el cual fue presentado este proyecto, y los principales objetivos y resultados esperados para ASSE. El principal método de relevamiento utilizado fue la realización de diferentes entrevistas con personal informático o con usuarios. En algunos casos se pudo obtener documentación orientada al usuario final del sistema, pero en ningún caso se pudo obtener documentación técnica de los sistemas relevados (relevamiento de requerimientos, diseño y arquitectura, deployment, etc).

3.1 Introducción

ASSE es un organismo público descentralizado que tiene a su cargo la atención integral de salud de aproximadamente el 36% de la población del país. Para el cumplimiento de su misión[1] cuenta con una red de atención a escala nacional, integrada por unidades asistenciales de diferente nivel de complejidad y es la organización que tiene la mayor red de cobertura en todo el país.[11]

Como metas a cumplir, se requiere avanzar en el desarrollo del sistema de información hacia un registro nuevo de usuarios (Producto: Padrón de usuarios). Además se requiere Integrar los Sistemas de Información existentes (Producto: Sistemas integrados) [25].

3.2 Sistemas informáticos relevados

ASSE cuenta con tres sistemas principales, el AP/SGA (Atención Primaria / Sistema de Gestión Asistencial), el SIQ (Sistema de Información Quirúrgica) y Escritorio Clínico.

AP/SGA

Este sistema se comenzó a desarrollar en el año 2003 con el objetivo de generar una trazabilidad de datos del paciente en consulta o internación. Busca seguir el proceso desde que el paciente solicita una atención, es atendido por un médico que registra las indicaciones dadas y finaliza con la gestión de estas indicaciones que pueden ser solicitudes de exámenes o medicamentos.

El sistema de Atención Primaria(AP) forma parte del Sistema de Gestión Asistencial (SGA) y se encarga de la gestión de la atención primaria realizada por ASSE en todo sus centros asistenciales. El desarrollo del sistema AP está a cargo de una empresa externa a ASSE.

Según FEMI, el primer nivel de atención "*es el conjunto de recursos y procedimientos tecnológicos, organizados para resolver las necesidades básicas y las demandas más frecuentes en la atención de la salud de una población dada. Constituye la puerta de entrada y el primer contacto de la población con el sistema de salud*". Algunas de las actividades que se realizan en este primer nivel de asistencia son: la educación sobre los principales problemas de salud y la forma de prevención y lucha asociados a los mismos, la asistencia materno-infantil con inclusión de la planificación de la familia, el suministro de medicamentos esenciales, la inmunización contra las principales enfermedades infecciosas, entre otras [33].

El AP/SGA está basado en un parte diario el cual es realizado por el médico, en papel, mientras atiende al paciente. Posteriormente este documento se eleva a un administrativo quien lo ingresa al sistema.

Actualmente se encuentra instalado en su totalidad en Montevideo y Canelones, sin embargo el módulo "Aduana" es utilizado en todo el país. Interactúa con el sistema *Padrones* para validar y controlar los datos del paciente.

El relevamiento realizado del AP/SGA es a nivel funcional pues no se mantuvieron reuniones con los

1 Ser el prestador público de referencia, basado en la Atención Primaria, con equidad, eficiencia y calidad, y con capacidad para responder a las necesidades de su población usuaria, en un marco de políticas de equidad social.

desarrolladores de este sistema y sólo se cuenta con un documento para el usuario final que especifica la forma de interactuar con el sistema. Se detallan a continuación algunas de sus funcionalidades [34]:

- Gestionar los datos personales del usuario.
- Historial de consultas realizadas, pendientes de realizar o canceladas por el usuario.
- Administrar consultas médicas de los usuarios (alta, baja, modificación).
- Registrar exámenes solicitados a partir de la consulta médica. Las cuales pueden ser referenciados a otros centros.
- Administrar ingresos a emergencias, llevar un control de los pacientes ingresados a emergencia que no han sido atendidos.
- Administrar las agendas de los funcionarios.
- Controlar el registro de horas de los funcionarios.
- Auditar datos de los usuarios, de movimientos de farmacia, consultas.
- Administrar las internaciones (alta, baja, modificación).
- Administrar las camas ocupadas de los hospitales.
- Administrar el stock de medicamentos (alta, baja).
- Administrar la reposicionar de medicamentos
- Registrar datos de pacientes crónicos (como ser droga, dosis, médico tratante).
- Autorizar, preparar y expedir pedidos de insumos médicos.
- Consultar el estado de los insumos en tránsito y de los insumos entregados.
- Enviar medicamentos hacia Farmacia central.
- Registrar la captación y realización del seguimiento del recién nacido, lactante (menor de un año de edad)
- Registrar la madre adolescente (hasta los 19 años de edad).
- Realizar las siguientes funcionalidades en el marco del modulo laboratorio,
  - Extraer Muestras a examinar
  - Ejecución de exámenes
  - Recibir muestras en laboratorio
  - Captar resultados de exámenes.
  - Consultar exámenes realizados.

ASSE pretende adaptar este sistema en una arquitectura en capas capaz de publicar servicios únicos en forma eficiente en tiempo real, así como también incorporar datos relevantes del resto de los sistemas.

Actualmente cuenta con una nueva plataforma de procesamiento informático, que le permite extender servicios de Montevideo a nivel nacional. Se pretende sacar el máximo provecho de la misma a corto plazo incorporando nuevas aplicaciones, y adaptando las existentes para que sean fácilmente integradas al resto, accediendo a información de los otros sistemas y entregando información propia al resto en tiempo real en forma de servicios.

El AP/SGA es un sistema monolítico y como tal tiene la principal desventaja de la complejidad para agregar nuevas funcionalidades, además de ser poco escalable. Se conoce también que maneja información redundante (Por ejemplo, con el Sistema Padrón), teniendo así el problema del cruzamiento de información erróneo entre sistemas. La Figura 5 muestra los distintos módulos que conforman el AP/SGA.

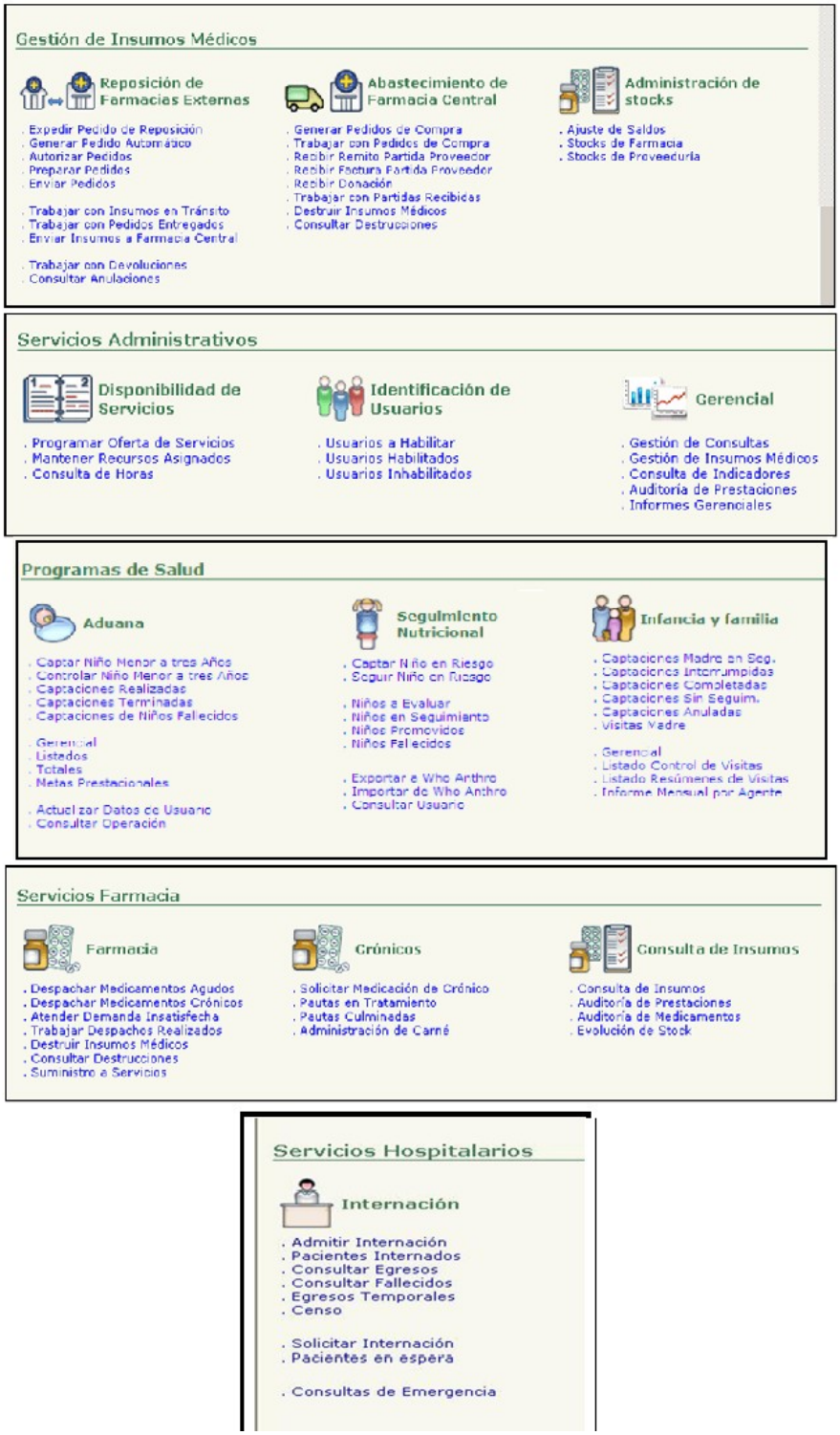

Figura 5: Módulos del sistema AP/SGA. [34]

SIQ

El Sistema de Información Quirúrgica (SIQ), es un sistema web que fue desarrollado dentro del Hospital Maciel de Montevideo y se integró, posteriormente, a la estructura informática de ASSE. Esta integración posibilitó que el sistema comenzara a ser utilizado por todas las instituciones pertenecientes a ASSE dentro del país.

Al momento de ingresar al sistema, se identifica la institución (unidad ejecutora) a través de su dirección en la red (dirección IP). Cada unidad ejecutora tiene un código definido por ASSE, estos códigos son cargados en tablas del sistema SIQ, no se toman de una tabla o servicio centralizado.

Para realizar el control de usuarios y roles en las distintas unidades ejecutoras, se utiliza un LDAP gestionado por la Administración de Correos del Uruguay. Cada usuario se identifica con su cédula de identidad y tiene una única contraseña, pero en cada unidad puede tener permisos diferentes. La contraseña de cada usuario es mantenida por el sistema y se almacena encriptada en una base de datos. Los usuarios y contraseñas son utilizados únicamente por el sistema SIQ.

El sistema cuenta con dos módulos: el de Descripción Operatoria y de Demanda Quirúrgica. La descripción de las características más relevantes de cada módulo se realiza a continuación.

Descripción Operatoria:

Este módulo es utilizado por los cirujanos para registrar los datos de una cirugía una vez finalizada. Este registro genera un documento clínico con los datos de la cirugía, el equipo que intervino y los resultados obtenidos. La descripción operatoria utiliza códigos LOINC, CIE-10 y SAQ-ASSE. El código CIE-10 se utiliza para la clasificación de enfermedades (clasificación internacional) y la codificación SAQ-ASSE para clasificar los procedimientos, estos códigos se clasifican para que resulte más fácil para el usuario encontrar el código deseado al completar el parte. Para registrar el paciente, el módulo busca la cédula de identidad ingresada en un sistema de ASSE de padrón de usuarios.

Además de los reportes de interés para el módulo existen dos informes que se relacionan con entidades externas. El primero de ellos es una planilla electrónica que se genera con los datos de las cirugías y una puntuación de las mismas. Esta planilla se envía al departamento de Gestión Financiera para que lo procese y luego las cirugías sean abonadas como parte del sueldo de los profesionales involucrados. El segundo de los reportes es el que se prepara para el Sistema Nacional de Información (SINADI) del Ministerio de Salud Pública (MSP). El informe consiste en una planilla electrónica con varias pestañas donde una de ellas corresponde a información quirúrgica. La información contenida en el archivo está definida por el ministerio y no corresponde a ningún estándar. En ambos casos, el módulo de Descripción Operatoria genera las planillas electrónicas para que luego sean enviadas a los destinatarios correspondientes, no existe ninguna interacción entre sistemas informáticos.

El módulo cuenta con la capacidad de generar un documento CDA con esta información. El CDA generado fue realizado tomando como modelo la especificación utilizada en Colombia, los códigos utilizados fueron establecidos por el equipo que desarrolló el módulo y su generación fue una iniciativa propia de este equipo. Actualmente no se trabaja con este CDA porque no existe ningún lugar definido para su almacenamiento ni ninguna entidad con la que se pueda interactuar a través de este documento.

El documento clínico generado por el módulo no cuenta con firma electrónica del cirujano y tiene como autor al usuario logueado al momento de su creación. Los usuarios sólo pueden ser médicos de la institución. Una vez que el documento ha sido creado, se imprime para que sea firmado por el cirujano. Esta copia en papel firmada es la que va a la historia clínica del paciente.

En la Figura 6 se muestra una descripción operatoria a modo de ejemplo, tomada del sistema Descripción Operatoria.

Figura 6: Ejemplo de descripción operatoria.

Gestión de Demanda Quirúrgica

Este módulo permite centralizar a nivel nacional toda la demanda de usuarios de ASSE en espera para ser operados, permite coordinar fecha de cirugía, teniendo un menú quirúrgico por área quirúrgica (block y sala) para cada UE (Unidad Ejecutora). Teniendo de ésta forma gestionada la agenda de operaciones.

En la Figura 7 se detalla el flujo del sistema, en el cual se puede ver que la coordinación de una cirugía se completa por dos motivos: se cancela o se genera la nota quirúrgica en el modulo de Descripción Operatoria visto anteriormente.

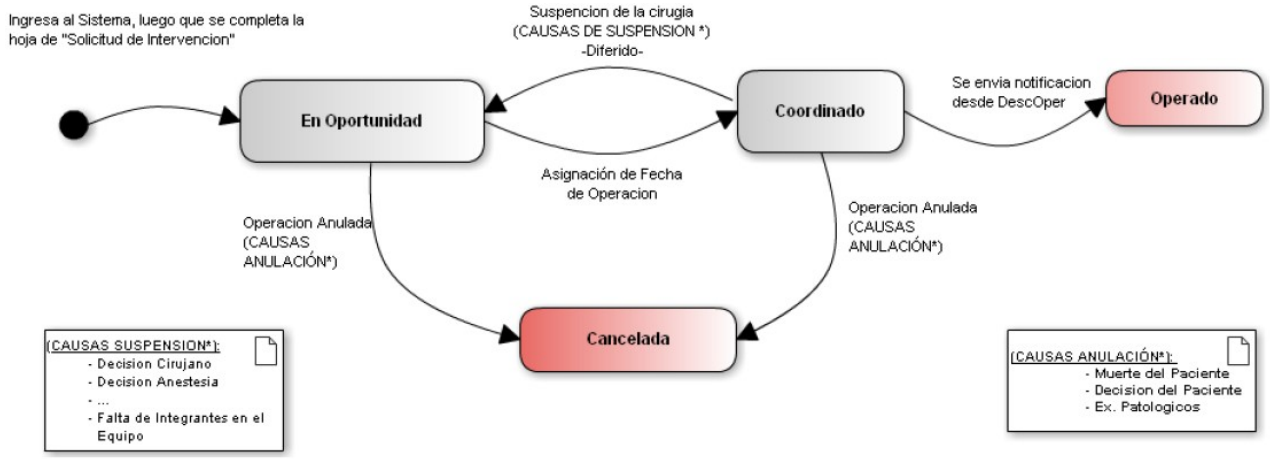

Figura 7: Diagrama de Estados en módulo Descripción Operatoria.

La coordinación de una cirugía parte desde una hoja de solicitud que es firmada por el cirujano. Luego el paciente debe firmar el consentimiento y estos dos documentos son los que van a la historia clínica. En ese momento se puede ingresar al sistema la hoja de coordinación y comenzar el proceso de coordinación. Existen distintas hojas de solicitud, según la institución que las maneja o el tipo de cirugía a realizar, por lo que el módulo de Demanda Quirúrgica cuenta con todo el conjunto de campos existentes, entre los que se eligen cuales completar y cuales dejar en blanco.

A modo de ejemplo, la Figura 8 muestra una funcionalidad del módulo Gestión de Demanda Quirúrgica. En la parte superior se tiene el listado de pacientes según el filtro deseado, y en la parte inferior se puede ver la agenda de cirugías.

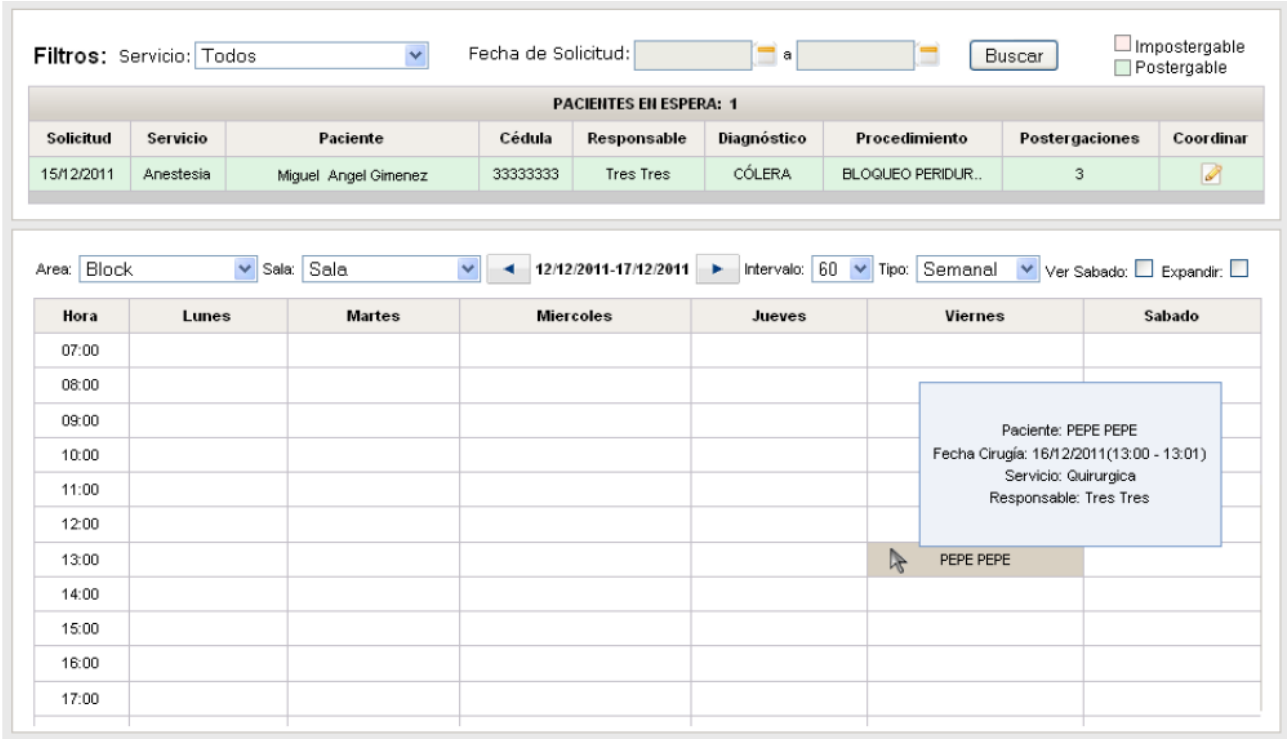

Figura 8: Gestión de Demanda Quirúrgica.

Escritorio Clínico

El relevamiento de este sistema de ASSE se hizo a través de los datos aportados por una medica, que actúa como usuario del sistema y en base a la información publicada del proyecto que da origen a la creación del Escritorio Clínico. Si bien se hicieron gestiones para dialogar u obtener documentación de parte de los profesionales encargados del desarrollo del sistema, las mismas no prosperaron. Por lo tanto los detalles técnicos del sistema no están presentes en este relevamiento.

En mayo del año 2011 se realiza la presentación del proyecto SIEMBRA (Sistema Informático de Escritorio Médico basado en la Red Asistencial), conocido internamente en ASSE como “Escritorio Clínico”.

Se trata de un emprendimiento estratégico para la atención en el sistema público de salud, que permite a los médicos del primer nivel de atención el acceso a la totalidad de la información sobre el estado de salud de sus pacientes mediante la Historia Medica Clínica Electrónica.
ANTEL apoya el proyecto brindando conectividad 3G, por otra parte ASSE realiza el enlace de los puestos de atención en todo el país a través de computadoras portátiles que permitirán el acceso a las historias clínicas digitalizadas de cada paciente [35].

Se termino de implantar en el año 2011 en el interior del país. Los departamentos que cuentan con el sistema son Artigas, Salto Paysandú, Río Negro, Flores, San José, Rivera ,Tacuarembó, Cerro Largo, Rocha y Lavalleja. Actualmente se utiliza en el primer nivel de atención dentro de las siguientes especialidades: medicina general, medicina familiar, pediatría y adolescencia.

El objetivo del sistema es registrar la asistencia y evolución de sus pacientes, coordinar estudios, resultados de exámenes, internaciones, controles y hasta las futuras consultas, centralizando toda la información de salud y convirtiéndose en médico de referencia [36].

3.3 Visión general del sistema informático

Como parte de las entrevistas realizadas para el relevamiento de los sistemas informáticos de ASSE, se pudo dialogar con médica de la institución que cumple el rol de usuario de los sistemas informáticos desarrollados y además es uno de los principales actores del sistema sanitario en su rol de médico. De este diálogo se capturan las principales sensaciones y percepciones en cuanto al sistema informático. Estas percepciones son fundamentales, no sólo porque es la devolución del usuario final del sistema, sino que de éstas depende, en gran medida, la colaboración de los médicos en la correcta utilización de los sistemas y la cantidad y calidad de información volcada en el desarrollo y evolución de los mismos.

La profesional de la salud entrevistada, nos hizo notar los siguientes puntos, en etapas tempranas del relevamiento y durante la evolución de los sistemas existe una notoria diferencia de visiones entre el equipo técnico y el equipo médico, la cual acompañada de poca interacción dificulta el entendimiento de conceptos básicos y complejos que los médicos necesitan transmitir al equipo técnico repercutiendo en desarrollos que no reflejan la realidad planteada, también hacía notar que un tema critico y en el cual hay que seguir trabajando es en el tema de la seguridad y confidencialidad de los datos del paciente. Por último nos mencionó que al ir creciendo los sistemas, por la incorporación de diferentes solicitudes de servicios, se fue perdiendo el foco en el objetivo principal del sistema informático, creando así soluciones que atacan las nuevas funcionalidades o servicios solicitados perdiendo de foco y sin tener en cuenta el sistema global y la interacción entre las distintas áreas de la salud.

De la situación planteada podemos concluir que se deben de conformar equipos mixtos (técnicos y médicos), y estos se deben de mantenerse durante el tiempo, se necesita de una etapa fuerte de análisis detallado el cual vaya acompañado de una documentación la cual permita ser tomada como medio de entendimiento entre los profesionales médicos y técnicos, como así también la documentación permita registrar el conocimiento transmitido por parte de los médicos y técnicos, esta documentación debe ser bien gestionada y actualizada para que aporte valor al equipo. Al ser proyectos de gran alcance es fundamental una buena gestión que tenga claros los lineamientos y mantenga a los equipos alineados y motivados, para lograr los objetivos es fundamental una buena comunicación entre las partes afectadas.

La visión de los médicos es fundamental, ya que un mal entendimiento genera confusión y rechazo al usuario final que utiliza el sistema pues no lo entiende y o considera que pierde el tiempo. Se debe de generar conciencia en los usuarios finales del sistema, haciéndolos participe del cambio del sistema y comunicándole los beneficios del mismo. Esto disminuye el rechazo que pueda llegar a tener la implantación del producto.

4 Diagnóstico y Recomendaciones

En esta sección se realiza un diagnóstico del escenario actual de ASSE, en base al relevamiento descrito en la sección anterior. El análisis se desprende de los datos relevados y la experiencia obtenida al intentar realizar este relevamiento.

Un vez presentado este diagnóstico, y tomándolo como base, se presenta la arquitectura de referencia sugerida para la institución.

4.1 Diagnóstico

La investigación realizada sobre el estado del arte de la informática en salud en la región y en Uruguay, muestra que existen varios estándares definidos para el área, la cual involucra una gran cantidad de actores y procesos. Uruguay fortaleció la participación del gobierno en el área de salud, fomentando la creación del Sistema Integrado de Salud y definiendo una agenda digital y un conjunto de estándares a seguir. En ese ámbito se creó la SUEIIDIS, organismo que representa a HL7 en Uruguay. Por lo tanto las instituciones tienen un camino trazado hacia el uso de estándares que permitan la interoperabilidad en el futuro.

El uso de estándares no garantiza la interoperabilidad entre sistemas sino que aporta un lenguaje común para la comunicación entre ellos. Para lograr interoperabilidad se debe tener además de un lenguaje común, una arquitectura que permita construir las interfaces para interactuar con entidades externas y permita interpretar los códigos externos que identifican procesos, personas, insumos, etc. El reconocimiento de estos códigos externos permite el posterior mapeo hacia los códigos internos que maneja la institución, lo cual permite comprender la semántica de los datos intercambiados y la posterior alimentación de los sistemas internos con estos datos obtenidos de entidades externas. Intercambiar información sólo resulta útil si podemos interpretarla.

A nivel de arquitecturas existen algunas iniciativas para estandarizar y proponer arquitecturas para el área de salud como pueden ser OpenEHR o proyectos de IHE (ver sección de Apéndices para más información), pero estas iniciativas atacan un conjunto limitado de los módulos y sistemas existentes. Una arquitectura que presenta una vista funcional completa de todos los procesos involucrados en el área de salud es la presentada por el Hospital Italiano de Buenos Aires. La misma ha sido desarrollada durante varios años y es conocida y tomada como ejemplo en la región. En Uruguay, FEMI ha optado por utilizar esta arquitectura como modelo para la estructura de su sistema informático. La ventaja de esta arquitectura es que presenta una vista funcional y separada en módulos, que representa los variados procesos existentes en el área de salud y muestra la interacción entre ellos.

A través del relevamiento de la arquitectura informática de ASSE, se constata que existen diversas iniciativas para informatizar varias áreas del sistema de salud pero cada una de ellas surge como un proyecto independiente, que pasa a formar parte de la estructura informática de ASSE pero no tiene ninguna integración con los sistemas existentes. Cada uno de los sistemas funciona como una isla, puesto que cada uno tiene sus propias bases de datos y maneja la información de forma aislada. Si bien, hay códigos a nivel de datos establecidos por la institución, como ser los de cada unidad ejecutora, cada sistema toma estos códigos y los ingresa en su propia base de datos. Existe interoperabilidad por parte del SIQ y el Padrón de Usuarios para tomar los datos de los pacientes, pero no existe, por ejemplo, una sistema de donde tomar los datos del personal médico. Además cada sistema tiene su propio sistema de gestión de usuarios, privilegios y roles dentro de las aplicaciones. Ésto genera que una misma persona deba tener un usuario y contraseña para acceder a cada aplicación que utiliza. Tener módulos de forma independiente no permite generar cruzamiento de datos de forma integral impidiendo generar reportes y estadísticas necesarios para una buena administración y gestión.

Si bien desde el año 2003 se está informatizando el área de la salud en ASSE, se detecta que éstos sistemas informáticos no se han integrado en su totalidad en los procesos de trabajo cotidianos, ya que aún existen registros que son realizados en papel.

4.2 Arquitectura de referencia

Para comenzar con la reingeniería del sistema informático en ASSE se debe comenzar por tomar una arquitectura modelo para adaptar los sistemas existentes y generar los nuevos en base a la misma. En base al diagnóstico realizado en ASSE, debemos tener en cuenta los siguientes puntos al momento de tomar esta decisión:

- El desarrollo aislado de los sistemas existentes pone de manifiesto la falta de coordinación y de un objetivo trazado y conocido que acompañe los desarrollos informáticos. Al existir grupos de trabajo descoordinados dentro de ASSE y grupos de trabajo externos, se debe dar a conocer la arquitectura elegida mostrando los beneficios que aporta y no como una imposición. En este sentido se debe procurar elegir una arquitectura fácil de entender y dar participación a los diferentes grupos de desarrollo para que modifiquen los sistemas informáticos existentes de forma de adaptarse a la arquitectura elegida. De esta forma se asegura que todos se sientan partícipes e involucrados en un proyecto común que intenta integrar los sistemas actuales y de ninguna manera desecharlos.
- La arquitectura tomada como modelo de referencia deberá ser capaz de permitir el alcance de los objetivos planteados por la institución y facilitar la realización de los objetivos trazados por el SNIS. Dentro de este punto, se destaca la importancia de contar con sistemas que permitan el fácil acceso a la información para trazar estrategias que permitan mejorar la atención en el primer nivel de asistencia, ayude a realizar campañas de prevención de enfermedades y optimice el uso de recursos, tanto humanos como materiales, en los distintos niveles de atención.
- La arquitectura debe ser lo suficientemente modularizada para permitir la incorporación paulatina de los sistemas informáticos existentes y deberá contar con suficiente documentación para garantizar que los nuevos sistemas informáticos sean desarrollados en base a la misma.
- Sería interesante considerar que la arquitectura de referencia facilite, a futuro, el intercambio de información de ASSE con los actores privados del sistema de salud uruguayo y con los principales actores de la región.

En base a los puntos mencionados anteriormente consideramos que la arquitectura del HIBA es un buen modelo a seguir para ASSE. Entre las principales ventajas se puede mencionar que refleja un caso exitoso de informatización del sistema de salud, alimentado por la experiencia de muchos años que ha permitido ir haciendo los ajustes y correcciones necesarios. La arquitectura del HIBA es conocida en Uruguay y en la región y ha sido tomada como modelo por FEMI, uno de los principales actores privados del país. Además, esta arquitectura se presenta modularizada y con una vista funcional, que permite identificar fácilmente a qué módulo pertenece cada uno de los procesos existentes y la relación con los otros componentes del sistema. La vista funcional permite la fácil comprensión de la arquitectura y los módulos delimitan las funcionalidades y responsabilidades de cada sistema.

4.3 Mapeo de sistemas actuales en arquitectura de referencia

Una vez elegida la arquitectura de referencia, el siguiente paso es ubicar a los principales sistemas relevados dentro de ASSE en la arquitectura de referencia. Con este mapeo se podrán ver los posibles puntos de contacto entre ellos y la información y/o procesos redundantes de la arquitectura actual. Para mostrar este mapeo se utiliza como base la Figura 1 que muestra la arquitectura del HIBA, para mostrar los módulos de esta arquitectura, que afecta cada sistema relevado.

AP/SGA

En la Figura 9 se muestran las funcionalidades del SGA categorizadas según su pertenencia a los módulos de la arquitectura de referencia.

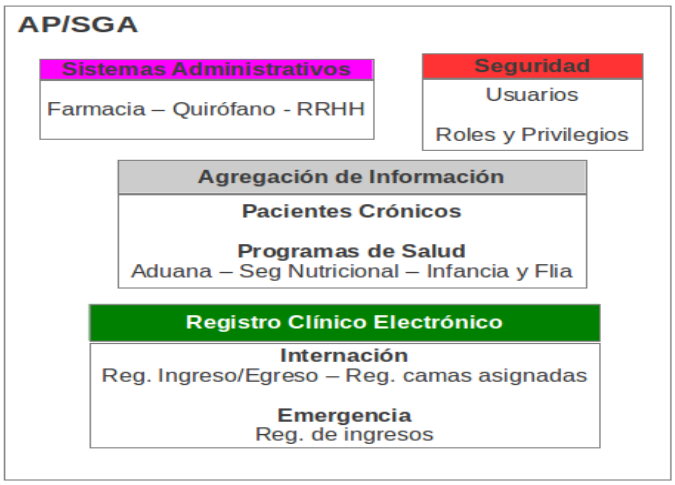

Figura 9: Mapeo del sistema AP/SGA a la arq. del HIBA.

Sistemas Administrativos
Como parte de los sistemas administrativos, el SGA cuenta con la gestión de Recursos Humanos, Insumos Médicos, Insumos de Farmacia, Quirófano y Atención de Pacientes. La gestión financiera no se realiza dentro de la aplicación ya que los datos de movimientos y stock son exportados para que sean manejados por un sistema externo denominado WinMesa.

Agregación de Información
Dentro de este módulo podemos ubicar a los Programas de Salud como Aduana, Seguimiento Nutricional e Infancia y Familia. Además, el SGA cuenta con un módulo de seguimiento de pacientes crónicos que también podemos incluir en este módulo.

Seguridad
El SGA cuenta con manejo de usuarios y administración de accesos a la aplicación. Estos usuarios y perfiles son de uso exclusivo de la aplicación, no toma ni comparte datos con ninguna aplicación externa.

Registro Clínico Electrónico
A nivel de internación, la aplicación lleva el registro de los pacientes internados, las camas asignadas y los egresos de los pacientes. Pero no lleva un registro de los exámenes realizados durante el período de internación.
A nivel de emergencia, lleva un registro de los pacientes ingresados pero tampoco cuenta con el registro de los tratamientos efectuados durante la atención.

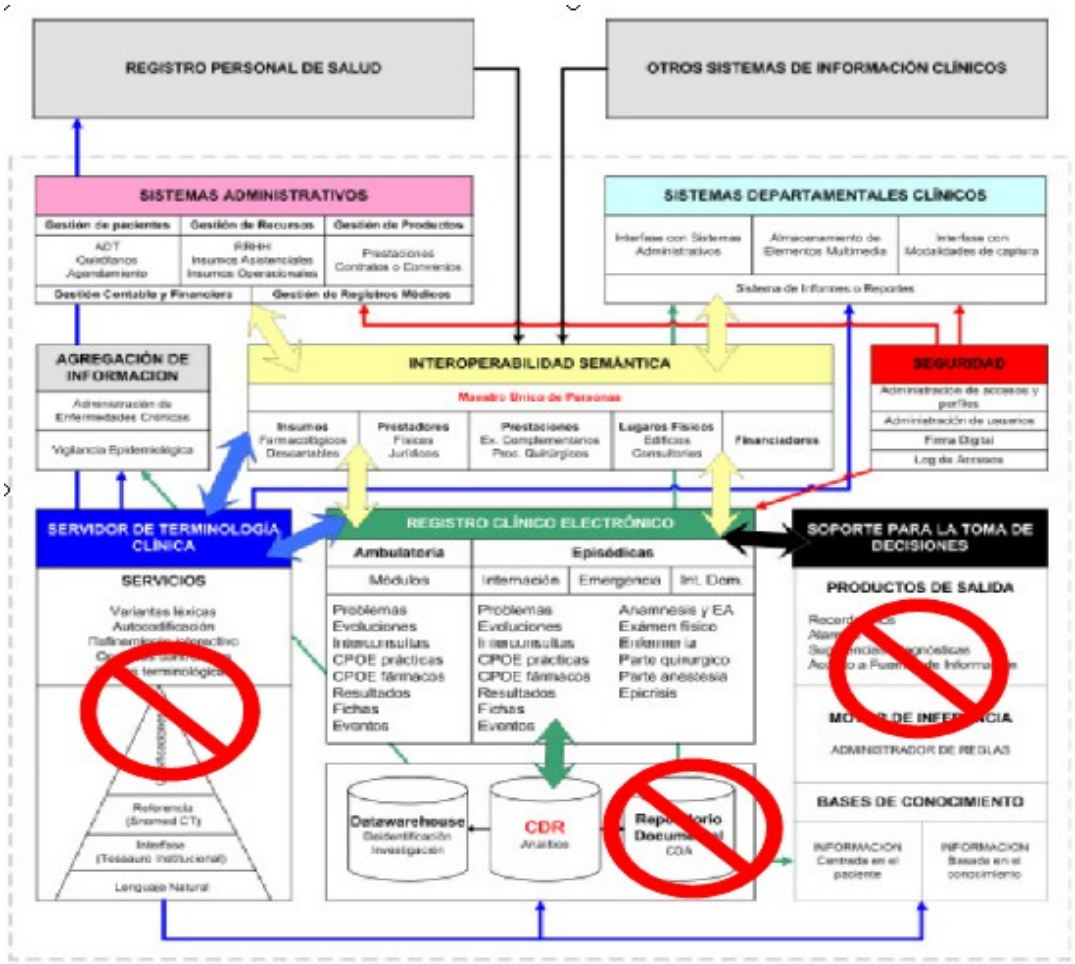

Figura 10: Vista general del AP/SGA en la arquitectura de HIBA.

SIQ

La Figura 11 muestra las funcionalidades del Sistema de Información Quirúrgica (SIQ) categorizadas según los módulos de la arquitectura de referencia.

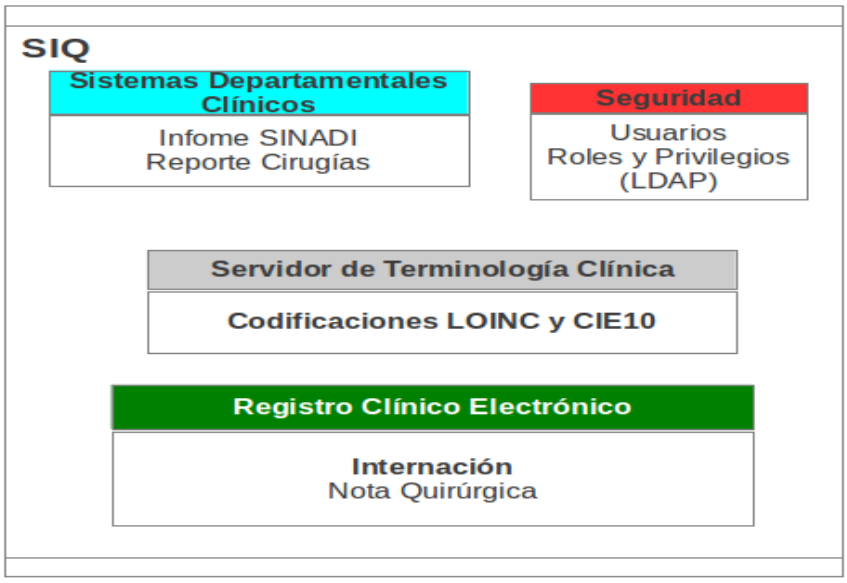

Figura 11: Mapeo del sistema SIQ a la arquitectura del HIBA.

Sistemas departamentales clínicos
El SIQ exporta dos informes a entidades externas, uno con la información de las cirugías realizadas en un período dado de tiempo y el otro con los datos que solicita el Ministerio de Salud Pública para el departamento quirúrgico.

Seguridad
El SIQ tiene un esquema de usuarios y privilegios para controlar el acceso a la aplicación. Este esquema de usuarios y permisos es utilizado únicamente por esta aplicación.

Servidor de terminología clínica
La aplicación cuenta con un registro de códigos pertenecientes a los estándares LOINC y CIE10 para que el usuario utilice momento de generar la información.

Registro clínico electrónico
La aplicación guarda la información de las intervenciones quirúrgicas. El SIQ tiene la capacidad de exportar este documento clínico a un documento CDA, pero esta funcionalidad no es utilizada.

Interoperabilidad Semántica
Para los dos módulos que forman parte del SIQ, se utiliza una codificación única de las entidades que comparte, como por ejemplo los códigos CIE-10. El SIQ además comparte la identificación de las personas, ya que consume la información desde un sistema externo denominado Padrones.

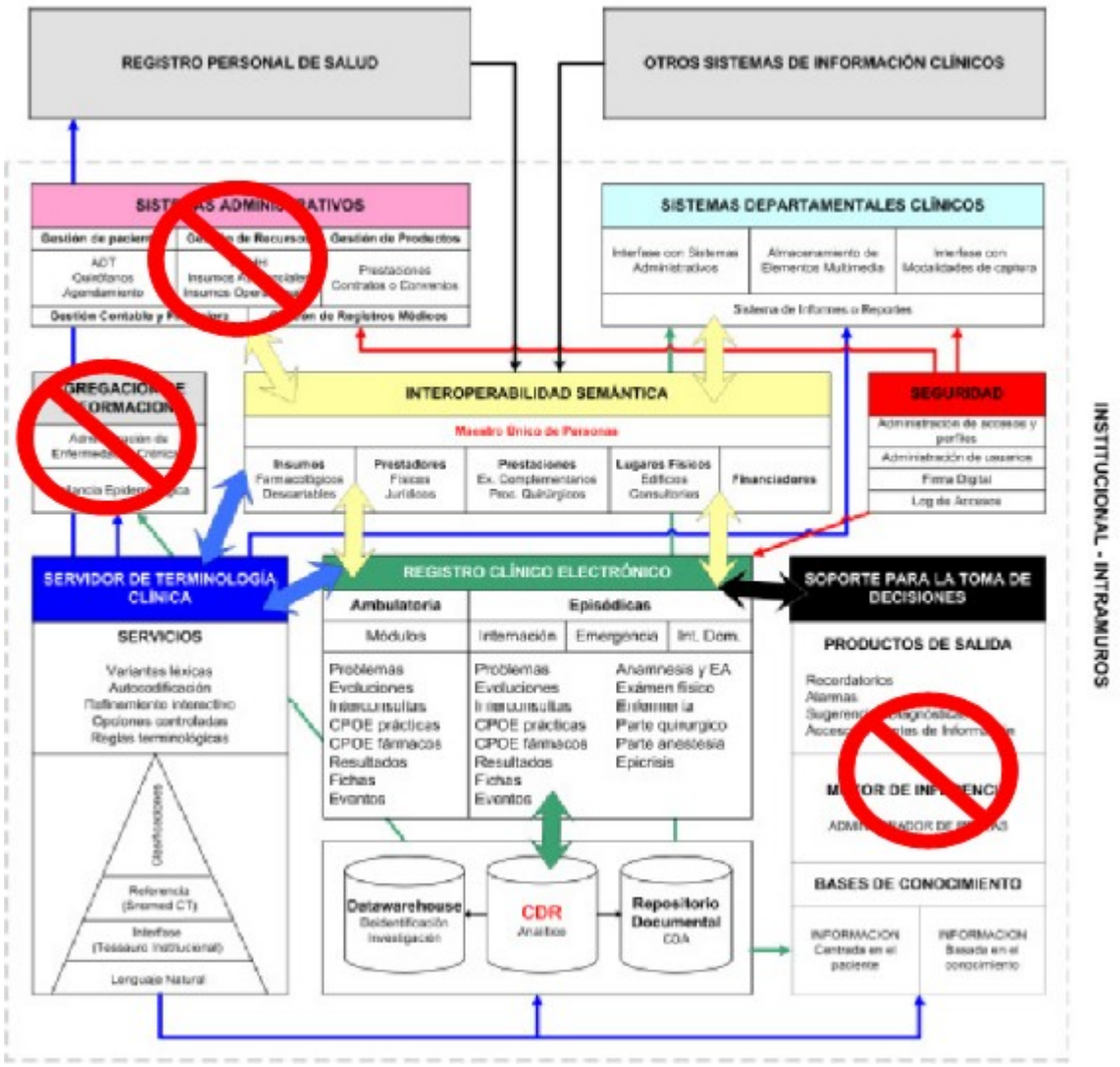

Figura 12: Vista gral del SIQ en la arquitectura del HIBA.

4.4 Recomendaciones

Una vez realizado el diagnóstico del estado actual de los sistemas informáticos de ASSE y la presentación de una arquitectura de referencia para comenzar a implementar un Sistema Integrado de Salud dentro de la institución, se toman las principales debilidades detectadas y se recomiendas estrategias a seguir para avanzar en la implementación de la arquitectura planteada. Para facilitar la visualización al usuario, las debilidades y recomendaciones se presentan en la Tabla 3 .

Área / Sistema	Situación Actual	Meta a alcanzar	Módulo afectado en arquitectura de referencia.
Seguridad	Cada aplicación define su propio sistema de usuarios y control de accesos. Una misma persona puede tener varios usuarios y passwords si utiliza distintas aplicaciones.	Centralizar la gestión de usuarios y el acceso a las aplicaciones.	Seguridad
Sistema Quirúrgico	El sistema de agenda de cirugías es manejado por la aplicación SIQ.	Centralizar la agenda de procedimientos clínicos, así como también la agenda de exámenes de laboratorio y radiología.	Sistemas Administrativos
Sistema Quirúrgico	Se cuenta con la capacidad de generar el parte quirúrgico en un documento CDA, pero no hay donde almacenar ese dato.	Almacenar datos del parte quirúrgico en un CDR centralizado y almacenar el CDA generado en un repositorio documental centralizado.	CDR – Repositorio Documental
Codificación de datos generales	Cada aplicación maneja su propia codificación. Existen algunos códigos utilizados en común, como por ejemplo, el de las unidades ejecutoras o la identificación de usuarios.	• Crear un sistema de codificación único que pueda ser consultado por cada aplicación. • Trabajar con la SUEIIDISS para generar los OID's que correspondan a ASSE y mapearlos con estos códigos únicos para permitir la interoperabilidad con sistemas externos en HL7.	Interoperabilidad Semántica
Codificación de datos clínicos	Cada aplicación define sus códigos y además los resultados obtenidos de entidades externas, por ejemplo, laboratorios definen los suyos.	• Definir la codificación a utilizar para datos clínicos y centralizar esta información para que esté disponible para todas las aplicaciones. • Definir variantes léxicas para cada término clínicos de forma de ayudar al usuario en aplicaciones como el Escritorio Clínico.	Servidor de Terminología Clínica
Laboratorio	Las instituciones nucleadas en ASSE trabajan con distintos laboratorios para realizar los análisis clínicos. La información proveniente de estos laboratorios es heterogénea.	Normalizar y almacenar los resultados de análisis clínicos de forma independiente a la fuente de la que provienen.	CDR – Repositorio Documental

Table 3: Recomendaciones para ASSE en base a la arquitectura de referencia.

Al finalizar la redacción de este documento, la Sociedad Española de Informática de la Salud y la Comisión Económica para América Latina y el Caribe, publicaron un documento titulado "*Manual de Salud Electrónica para directivos de servicios y sistemas de salud*" [3]. Dentro de esta publicación se encuentran muchos conceptos alineados al diagnóstico y las recomendaciones realizadas, por lo tanto, se decide incluir un resumen de las principales ideas expuestas que, a nuestro entender, refuerzan la elección de la arquitectura recomendada.

Interoperabilidad

Dentro de las instituciones sanitarias existen múltiples sistemas de información que generan una gran cantidad de información que permite la atención médica oportuna y de calidad para el paciente, así como también permite la gestión en todos los niveles de la organización. Es frecuente encontrar instituciones que poseen información fragmentada en sistemas de información independientes que forman islas. Debido a esta situación, aún cuando la información existe y está disponible, no se tiene accesos a ella o este acceso es parcial, lo cual puede derivar en una decisión médica incorrecta que puede llegar, en un caso extremo, a poner en riesgo la vida del paciente.

Cuando abandonamos la visión particular de la realidad de las instituciones sanitarias y nos movemos hacia la visualización global del sistema de salud, en su realidad multi-institucional, el problema de acceso a la información mencionado, se traslada y genera problemas a la hora de coordinar políticas y ejecutar proyectos para la mejora global de la salud de la población en general. Las entidades reguladores y el gobierno, tienen una visibilidad parcial, y en muchos casos tardía, de la realidad de las instituciones lo que complica la evaluación de las políticas sanitarias planteadas y el impacto de las mismas.

La forma de abandonar este paradigma basado en islas de información, es lograr la interoperabilidad entre las instituciones. La interoperabilidad se debe entender como la capacidad de los sistemas de información computarizados y las aplicaciones de software, de comunicarse para intercambiar datos y la capacidad de utilizar esta información intercambiada. Existen distintos niveles de interoperabilidad. Como base para conseguirla, se debe tener ***interoperabilidad sintáctica*** que establece la sintaxis para representar los datos intercambiados. Una vez lograda la interoperabilidad sintáctica es de interés alcanzar el nivel de ***interoperabilidad semántica*** que refiere a la capacidad de reconocer e interpretar la información intercambiada. Para lograr la interoperabilidad, en cualquiera de sus niveles, es imprescindible el uso de estándares, los cuales rigen los protocoles de comunicación e intercambio de datos.

Para lograr la interoperabilidad semántica, se deben normalizar los códigos utilizados para identificar personas, instituciones, hospitales, consultorios, médicos, etc. Además se deben normalizar los procedimientos quirúrgicos, fármacos, estudios radiológicos, análisis clínicos, etc. Existen muchos estándares desarrollados que normalizan algunos de estos códigos como ser el código LOINC para normalizar exámenes de laboratorio, CIE10 para clasificar enfermedades, etc. Existen otros estándares que establecen las reglas para el intercambio de información como imágenes digitales (DICOM) o de actos administrativos, contables o datos clínicos (HL7). Y también existen modelos de referencia para historias clínicas (OpenEHR) e incluso arquitecturas que proveen diferentes perfiles con el uso de estándares para distintas áreas del sistema sanitario (IHE).

Historia Clínica Electrónica (HCE)

Durante el proceso de atención sanitaria, independientemente de quién la realice y dónde se preste dicho servicio, se genera información que suele ser almacenada en un repositorio denominado historia clínica. Con frecuencia, dicha denominación se utiliza de forma intercambiable con diferentes términos tales como ficha clínica, registro médico, expediente clínico, expediente médico o prontuario médico. Todos estos términos son utilizados para referirse al conjunto de documentos que contienen los datos, valoraciones e

informaciones de cualquier índole sobre la situación y la evolución clínica de un paciente a lo largo del proceso asistencial. La historia clínica está constituida por documentos, tanto escritos como gráficos, que hacen referencia a los episodios de salud y enfermedad de esa persona y a la actividad sanitaria que se genera con motivo de esos episodios.

El soporte tradicional de las historias clínicas fue siempre el papel. Dicho formato de almacenamiento posee algunos problemas que se solucionan con la incorporación de la HCE, a continuación se detallan los mismos.

- *Disponibilidad y accesibilidad*, debido a la falta de integración entre los diferentes niveles de atención (ambulatorio, emergencias, internación, seguimiento domiciliario, cuidados crónicos), lo que genera que las historias clínicas estén disponibles en un solo lugar a la vez, disminuyendo su accesibilidad y aumentando su fragmentación y duplicación.
- *Formato*, debido a que son poco estructuradas, es común que sean muy personales y con falta de organización y uniformidad. Se deterioran con el tiempo y consumen mucho espacio físico y recursos para su almacenamiento y manipulación.
- *Contenido*, los documentos manuscritos son frecuentemente ilegibles e incompletos y la información puede ser alterada. Pacientes crónicamente enfermos pueden acumular cantidades inmanejables de papel a través del tiempo y la recuperación de la información es una tarea manual muy costosa

La HCE, se puede definir como la colección longitudinal de información electrónica sobre la salud de las personas donde la información sobre salud es definida como información pertinente a la salud de un individuo, o la información de los cuidados de salud provistos a un individuo, por medio de cualquier miembro del equipo de salud. Tiene la posibilidad de dar acceso electrónico inmediato a la información de salud personal o poblacional solo a los usuarios autorizados. Provee las bases de conocimiento y sistemas de soporte para la toma de decisiones que mejoren la calidad, seguridad y eficiencia de la atención de los pacientes. Tiene el objetivo primordial de dar soporte para la eficiencia de los procesos de cuidados de salud

La HCE está pensada como la interfaz que utilizan los miembros del equipo de salud para registrar su quehacer asistencial. Debe ser el lugar primario para la carga de toda la información clínica. Está compuesta por diferentes interfaces de carga respetando las necesidades de registro del ámbito ambulatorio (registro longitudinal que almacena contactos) y el resto de los ámbitos de atención que poseen una estructura episódica de atención (períodos de tiempo con inicio y finalización clara). La columna vertebral de ambos tipos de registro es la lista de problemas que actúa como integrador de la carga mórbida del paciente. El resto de los módulos contienen los aspectos básicos del registro en las notas de evolución, interconsultas, prescripciones de fármacos y exámenes complementarios, visualización de resultados y un ingreso estructurado por especialidades y patologías. En el registro episódico se agregan módulos de carga especiales como los partes anestésicos, quirúrgicos, de enfermería y otros.

Las barreras que afronta la implantación de la HCE, están dados en diferentes niveles,

- *Financieros*, debido a altos costos asociados a la inversión inicial, altos costos de mantenimiento, incertidumbre sobre el retorno.
- *Falta de infraestructura informática adecuada (*hardware, software y comunicaciones), insuficientes habilidades informáticas de los médicos o auxiliares, problemas asociados con la interoperabilidad e interconexión con otros sistemas también constituyen una barrera importante.
- *Tiempo*, pues la selección, adquisición e implementación del sistema consume mucho tiempo .
- *Legales*, aspectos relacionados con la privacidad y seguridad de la información.
- *Sociales*, incertidumbre sobre las empresas comercializadoras de los productos de HCE.
- *Manejo del cambio*, inadecuada transición en la cultura organizacional al migrar hacia la HCE, falta de incentivos, participación y de liderazgo.

Los requisitos para la implementación de una HCE se detallan a continuación;

- *La identificación unívoca de individuos,* tanto a nivel local como nacional, la mayor dificultad para integrar la información clínica de una persona reside en el mecanismo de identificación unívoca de esta. El foco del problema ya no está en la obtención de un identificador universal, sino en la identificación que contemplen tanto el proceso de acreditación de identidad como la correlación de múltiples padrones de individuos y una permanente auditoria que asegure la calidad de los datos en el maestro único.
- *Integración con otros sistemas*, se requiere de la interoperabilidad. La HCE requiere información de otros sistemas (de la institución o fuera de ella), por lo que es necesario desarrollarla teniendo en cuenta la posibilidad de intercambio electrónico de datos entre ellos. Esto puede lograrse mediante la creación de interfaces dedicadas para cada caso
- *Estándares*, aquellos orientados al intercambio de datos y mensajería electrónica, de terminología, de documentos, conceptuales, de aplicaciones y, por último, de arquitectura.
- *Adecuada representación de la información clínica ,e*l texto narrativo es una gran cantidad de información contextual necesaria para la comunicación con los miembros del equipo medico y asegurar un correcto proceso diagnóstico y terapéutico, la información descripta en texto narrativo puede ser ambigua. Una solución para disminuir la ambigüedad es obligar el ingreso estructurado de información, la codificación primaria y el ingreso estructurado no son siempre bien recibidos por los profesionales.
- *Disponer de servicios terminológicos centralizados,* posibilita el logro de un adecuado equilibrio entre la libertad de los textos narrativos y los beneficios del ingreso estructurado de datos tanto a nivel institucional como inter-institucional.
- *Aspectos relacionados con la usabilidad,.* el diseño de las interacciones humano-computadora correlaciona directamente con la aceptación y el uso de las HCE por parte de sus usuarios.
- *Aspectos legales,* se busca que el soporte electrónico tenga la misma validez legal que el tradicional en papel**.**
- *Seguridad, privacidad y confidencialidad, e*s necesario tomar las medidas necesarias para asegurar una adecuada división de entornos de desarrollo, testeo y producción de sistemas, otorgamiento de perfiles de usuario y accesos, así como el registro sistemático del quehacer de los usuarios en el sistema que posibilita su trazabilidad, algo imposible de lograr en los tradicionales registros en papel. De ser posible, debe lograrse la implementación de firma electrónica/ digital (mediante estándares de encriptación asimétrica por llaves públicas y privadas) de los documentos contenidos en las HCE
- *Manejo del cambio,* La resistencia al cambio que presentan los miembros del equipo de salud es una constante en todos los procesos de informatización del registro clínico en las instituciones, un factor de éxitos en la implantación de una HCE, es conformar un equipo multidisciplinario para la definición de los alcances, definir un diseño o elegir una HCE.
- *Manejo de la transición,* la transición es el período comprendido desde que se deja la historia clínica en papel y se comienza a usar la electrónica. Se debe de minimizar la inconsistencia que puede darse entre la información de los registros médicos en papel con los electrónicos, en caso de darse impacta directamente en el accionar diario del equipo médico.
- *Pérdida de productividad, a*l menos al inicio de las implementaciones es de esperar una sensación de pérdida de productividad por parte de los profesionales. Ya que impacta en el tiempo de documentación. Luego de la meseta de estabilización y acostumbramiento de los cambios, se nota la ventaja de tener un acceso a información clínica centralizada de los diferentes niveles de atención lo cual permite tomar decisiones mas acertada.

Sistema de información del hospital (HIS)

El HIS (acrónimo de Hospital Information System, sistema de información hospitalario), es un sistema integrado de información diseñado para gestionar todos los aspectos clínicos, administrativos y financieros de un hospital. Además, permite obtener estadísticas generales de pacientes, datos epidemiológicos, de salud laboral y salud pública, entre otros.

Los HIS, inicialmente en 1950, surgieron para colaborar con la gestión administrativa de los pacientes, o sea registrar actividades de los pacientes como así también registrar el stock de productos, registrar finanzas y contables entre otros. La tendencia de los últimos veinte año a sido la de desarrollar sistemas descentralizados especializados en la resolución de problemas concretos que recogen la información requerida y ponen a disposición del resto de los sistemas los datos mas relevantes.

El HIS puede estar compuesto por uno o varios componentes de software y una gran variedad de subsistemas de especialidades como el RIS/PACS (Sistema de información de radiología), SIL (Sistema de información de laboratorios), sistemas de información para anatomía patológica, entre otros.

Uno de los subsistemas presentes en el HIS es el denominado Estación Clínica el cual resume la información clínica del paciente que ha sido transmitida desde los subsistemas de las distintas especialidades médicas del hospital junto con aquella generada por el propio sistema principal del HIS.
En la estación clínica se recogen los datos que constituyen la historia clínica electrónica y es una de las herramientas principales utilizadas por los profesionales sanitarios en la atención de pacientes.

El HIS, debe de participar en todo el proceso del paciente. El paciente solicita una cita, se requiere tener una identificación del paciente en el sistema según el maestro de usuarios, solicitar la cita teniendo en cuenta el catalogo de prestaciones del hospital, configurar las agendas teniendo en cuenta prioridades o características propias de cada hospital, el día de la cita el paciente se debe de identificar correctamente obteniendo la admisión, posteriormente se realiza la cita, en el momento que el paciente es atendido el médico puede solicitar una o varias prestaciones, los resultados pueden ser consultados en la estación clínica de acuerdo a la información que contiene la historia clínica del paciente El médico puede volver a ver al paciente para explicarle el diagnóstico y el tratamiento a seguir.

La configuración que se haga de los procesos descritos anteriormente, junto con el resto de los subsistemas del HIS, debe reflejar la planificación estratégica que la gerencia quiere llevar a cabo en el hospital. Para adecuar las políticas del hospital a los objetivos estratégicos se dispone del HIS.

Sistemas de información del laboratorio clínico (SIL)

Los sistemas de información del laboratorio clínico (SIL) han evolucionado desde el simple registro de petición e impresión de los informes de resultados, a la gestión de todas las fases del proceso del laboratorio (aspectos pre-analíticos, analíticos y post-analíticos) y se integran con el resto de los sistemas informáticos tanto de gestión como clínico. Ha cambiado incluso el perfil del usuario de estos sistemas, que inicialmente se trataba de personal administrativo.

En la actualidad estos sistemas se han vuelto críticos, pues en la mayor parte de los casos no existen alternativas manuales al SIL pues no se cuenta con la suficiente capacidad operativa, impactando directamente en la atención al paciente.

En cuanto a aspectos legales respecta, la normativa legal de seguridad de la información dictamina que cualquier acción sobre los datos debe quedar registrada a los efectos de poder delimitar responsables, acciones de mejora y poder establecer indicadores de calidad.

Algunos problemas que actualmente enfrentan estos sistemas son:

- Identificación del paciente
- Catálogos de pruebas y transferibilidad de los datos: Existen varios estándares, entre ellos LOINC, IUPAC (International Union of Pure and Applied Chemistry), SNOMED, EUCLIDES (European Clinical Laboratory Information Data Exchange Standard) .

La IHE promueve 6 perfiles en cuanto a la integración de los datos para el caso de los laboratorios, los cuales están basados en el estándar HL7. Los mismos son:

- Laboratory Testing Workflow (LTW)
- Laboratory Device Automation (LDA)
- Laboratory Point of Care Testing (LPOCT)
- Laboratory Code Set Distribution (LCSD)
- Laboratory Specimen Barcode Labeling (LBL)
- Sharing Laboratory Reports (XD-LAB)

Se espera que una progresiva utilización de estas recomendaciones permita una interoperabilidad efectiva.

5 Aplicación práctica

Una vez realizado el relevamiento de los sistemas informáticos en ASSE y la posterior decisión de tomar la arquitectura del HIBA como modelo de referencia, se decide tomar algunos de los módulos de esta arquitectura para mostrar la factibilidad de incorporarlos dentro del sistema informático de ASSE. Como primer paso hacia la adopción de la arquitectura modelo, se decide implementar dos prototipos: uno para el Registro Clínico Electrónico de los datos de laboratorio y otro para el manejo centralizado de acceso de usuarios como parte del módulo de Seguridad.

Las soluciones planteadas cumplen con determinados requisitos propuestos por ASSE, entre ellos se destacan:

- Cumplir con la necesidad de código abierto y multi-plataforma.
- Capaces de interactuar con los sistemas existentes dentro de la institución.
- Brindar una interfaz con estándares en salud aceptados y conocidos a nivel nacional e internacional.

Los prototipos son presentados, en esta sección, de forma descriptiva y con poca profundidad técnica para que el lector sea capaz de conocer las principales características y objetivos planteados al momento de su realización. Los detalles de su implementación se pueden ver en la sección de Apéndices de este documento.

5.1 Registro Clínico Electrónico

Para obtener una historia clínica electrónica es fundamental poder contar, no sólo con la lista de análisis solicitados a un paciente, sino también con los resultados de los mismos. Estos resultados están en manos de los laboratorios que realizan los análisis y se debe encontrar una forma de almacenarlos en un registro documental de ASSE. Es así que se plantea la necesidad de implementar un repositorio documental centralizado y de alimentarlo con los resultados clínicos que manejan distintas aplicaciones en los diferentes laboratorios. Esto implica la comunicación con estos laboratorios y el almacenamiento de la información en un formato común.

La mayoría de los laboratorios de análisis clínicos con los que trabajan las distintas instituciones que conforman ASSE, son entidades externas. Por lo tanto se requiere contar con una forma de interoperar con estas entidades para obtener los resultados de los análisis y almacenarlos dentro de un registro clínico electrónico perteneciente a ASSE. El mayor porcentaje de los laboratorios externos, trabajan con un software de la empresa Izasa llamado Modulab. Este software cuenta con distintas versiones, la más utilizada es su versión Win.

Modulab Win tiene la capacidad de realizar la agenda de análisis para un paciente y el informe de resultados a través del uso del sistema de archivos. De esta manera, se pueden agregar solicitudes a un archivo que es leído por Modulab para luego grabarlos en una base de Citaciones. Cuando el paciente se presenta en el laboratorio, se recupera la citación de la base y se realizan los análisis solicitados. A medida que se van obteniendo los resultados de las distintas pruebas realizadas, un técnico valida los resultados y Modulab los graba en un archivo configurado para estos fines. La interacción aquí detallada se puede ver en la Figura 13.

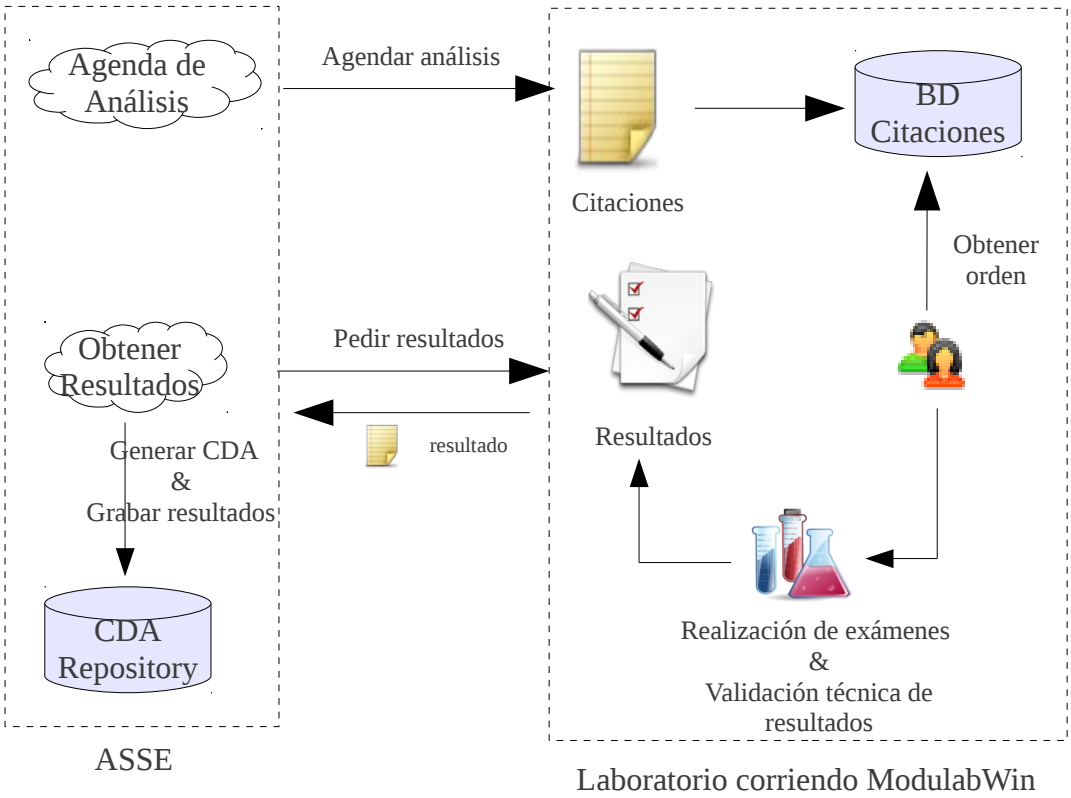

Figura 13: Interacción entre ASSE y un laboratorio que utiliza el software Modulab Win.

Para mostrar la factibilidad del prototipo de Registro Clínico Electrónico, se decide tomar los datos del software Modulab, dado que representa a la mayoría de las instituciones con las que se debe dialogar. Una vez que se obtienen los resultados se crea un documento HL7 CDA. Para el prototipo se toma como ejemplo un esquema de CDA bautizad en Colombia, el mismo que utiliza el sistema SIQ. La información obtenida de cada resultado se mapea a los campos del CDA que luego es almacenado en una base de datos. Junto con el CDA se almacena una clave para asegurar que no haya modificaciones posteriores. Esta clave se genera a partir de los datos contenidos en el CDA y luego aplicando una función criptográfica. Si los datos del CDA son alterados, la clave no coincidirá con la almacenada, por lo tanto se sabrá que hubo una alteración.

5.1.1 Sistemas relacionados al Registro Clínico Electrónico (RCE)

En la Figura 14 podemos ver la cantidad de sistemas que interactúan con el registro clínico electrónico de una institución. Entre ellos destacamos aquellos involucrados con el block quirúrgico, el stock de farmacia y los sistemas de diagnósticos.

La estandarización en los sistemas de salud, ha permitido que al día de hoy el equipamiento médico cuente con la tecnología necesaria para brindar los resultados de ecografías, tomografías, o electrocardiogramas, por ejemplo, en forma digital y estandarizada. Contar con sistemas que manejan esta información y puedan volcarla en el registro electrónico de pacientes, contribuye a que interoperen, no sólo los procesos informáticos, sino también el equipamiento de las instituciones.

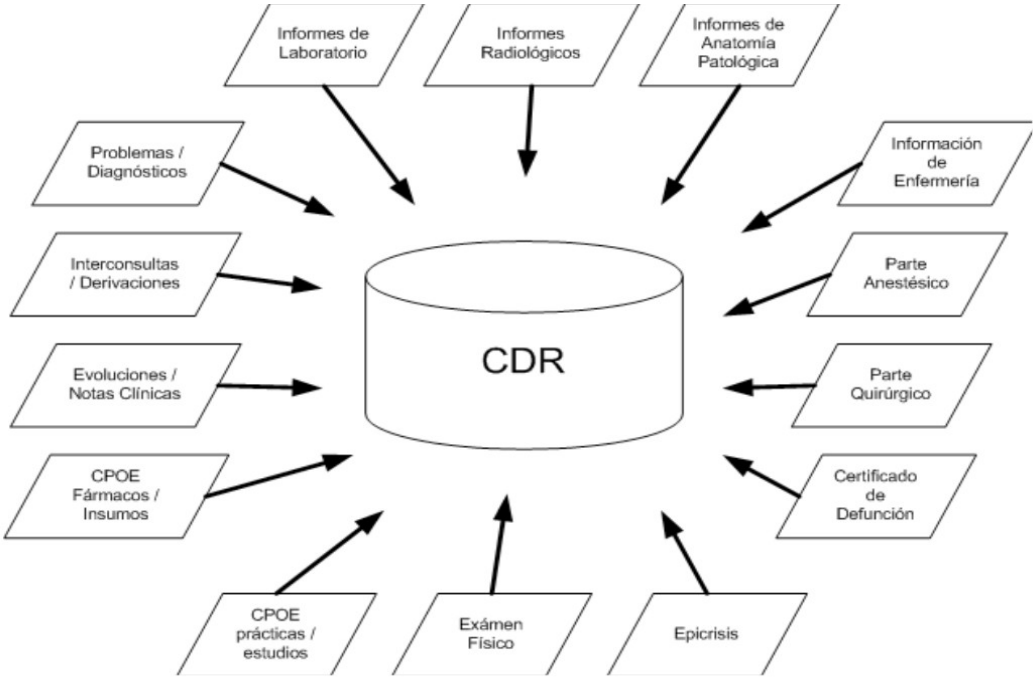

Figura 14: Sistemas que interactúan con el Registro Clínico Electrónico (RCE)

Teniendo en cuenta la importancia del RCE y sus múltiples interacciones con otros módulos, se centra el diseño del prototipo en buscar la interoperabilidad y por este motivo se elige adoptar el estándar CDA (Clinical Document Architecture) de HL7 (Health Level Seven) que es utilizado para registrar y documentar los distintos episodios clínicos. Este estándar está estructurado en un XML el cual cuenta con datos estructurados y codificados y también texto libre para que sea fácilmente legible a la vista humana.

5.1.2 Diseño de la solución

Si bien el prototipo se comunica con el software Modulab Win, se hace hincapié en realizar un diseño que permita comunicarse con cualquier software de laboratorio a futuro. Esto permitirá a ASSE tener una base de datos clínicos completa que permita contar con una HCE que refleje el paso del paciente por cualquier centro de la red asistencial.

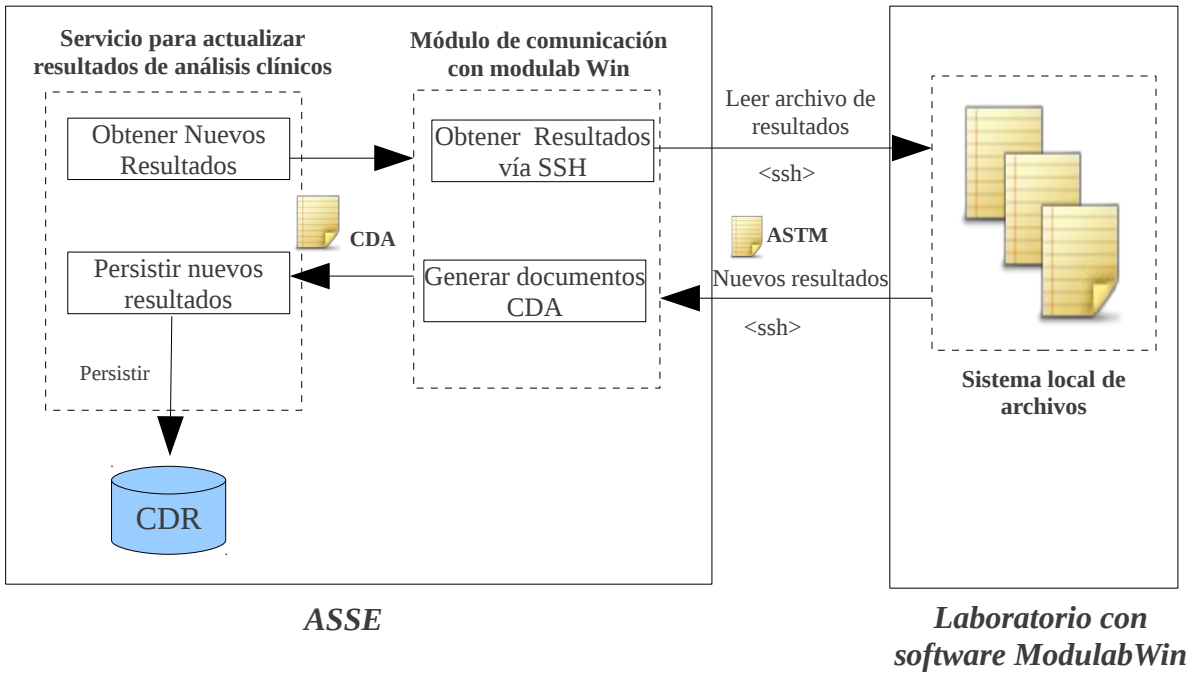

Figura 15: Actualización de resultados de análisis clínicos

La Figura 15 muestra la interacción entre el servicio que actualiza los resultados clínicos desde el laboratorio hacia el repositorio de datos clínicos (CDR en la figura). El proceso contiene dos componentes, el servicio que actualiza los resultados y un módulo de comunicación con el laboratorio. El servicio es quién provee la interfaz hacia el exterior, por lo tanto es el punto de entrada al proceso. El módulo de comunicación se encarga de resolver la comunicación con el software del laboratorio, obtener los resultados y generar el documento CDA a partir de los datos obtenidos. Tendremos uno de estos módulos por cada software de laboratorio distinto con el cual se deba interactuar y el servicio se comunicará con estos módulos a través de una interfaz establecida para abstraerse del tipo de software que maneja el laboratorio con el cual se está comunicando.

5.2 Autenticación centralizada

La autenticación hace referencia al proceso por el cual un usuario de una red adquiere el derecho a usar una identidad dentro de la misma. Existen en la actualidad varios mecanismos para autenticar a un usuario tales como, el uso de una clave, el uso de un token, mecanismos biométricos como la huella digital o alguna característica biológica del individuo que lo identifique, así como una combinación de los anteriores.

Al relevar la autenticación en ASSE, se determina que la autenticación se realiza a través de claves y que no se realiza de forma centralizada, sino por el contrario cada aplicativo maneja su propia gestión de usuarios y permisos, a su vez el acceso a la red se hace de forma local en cada equipo. Los problemas que acarrea ésta forma de administrar a los usuarios no es menor, ya que de cara al usuario éste debe manejar distintas claves según el sistema en donde se autentifique y a su vez depende del equipo al que se conecte es la clave que debe utilizar, por lo general el usuario tiende a simplificar las claves para facilitar su labor, permitiendo que terceros puedan descifrar las mismas. De cara al administrador de los usuarios su trabajo tiende a ser mayor, ya que hay varios sistemas que se tienen que actualizar y mantener, además de tener replicados los datos del usuario lo cual puede ocasionar inconsistencias en los mismos.

El objetivo del prototipo es proponer una solución la cual permita el manejo de la autenticación de forma centralizada para acceder a la red de ASSE. Esto implica que se tenga un único repositorio central, el cual contenga los usuarios y las claves de los mismos. Para llevar adelante lo expuesto se propone instalar en un servidor Linux un controlador de dominio con Samba y OpenLDAP, el cual permita autenticar a los usuarios de la red de ASSE. Como trabajo futuro resta integrar al directorio central la autenticación de los distintos sistemas presentes en ASSE, respecto a éste punto se propone un posible diseño del directorio en la sección 5.2.3.

5.2.1 Descripción de LDAP

LDAP (Lightweight Directory Acces Protocol), en español Protocolo Ligero de Acceso a Directorios, es un protocolo a nivel de capa de aplicación, del tipo cliente-servidor el cual permite acceder a un servicio de directorio para buscar información en un entorno de red [37]
Las operaciones de borrado y actualización en el directorio LDAP se hacen de forma lenta, con lo que se puede decir que LDAP es un tipo de base de datos pero no relacional. No está diseñado para implementar los complicados esquemas de transacciones que las bases de datos utilizan para actualizar grandes volúmenes de datos. Las actualizaciones en un directorio LDAP son usualmente cambios sencillos, sin embargo LDAP si está diseñado para soportar de forma eficiente las lecturas de datos.

El servicio de directorio es un término ambiguo, que se utiliza para referirse tanto a la información que contiene el directorio, al conjunto de hardware/software que gestiona dicha información, como también a las aplicaciones cliente/servidor que utilizan la información. En resumen se puede decir que el servicio de directorio es un conjunto de componentes que trabajan en forma conjunta para prestar un servicio. Existen ejemplos de directorios clásicos usualmente usados por las personas como pueden ser la guía telefónica, o la revista del cable TV, éstos directorio son estático, y el medio de difusión es el papel, a diferencia de los directorios electrónicos los cuales permiten cambios dinámicos respecto al contenido, haciendo más confiable la información brindada, como también mas seguros permitiendo ingresar controles para acceder a la información.

Un directorio es un conjunto de objetos con atributos organizados en una manera lógica y jerárquica. [37]. La estructura jerárquica del directorio LDAP se denomina DIT (Directory Information Tree) o árbol de información del directorio, como su nombre lo define es una estructura de árbol, donde las ramas pueden ser contenedores de información, o simplemente ser entradas del árbol.
Los directorios proporcionan una respuesta rápida a operaciones de búsqueda o consulta. En general tienen una gran capacidad de replica de información para aumentar la disponibilidad y fiabilidad de los datos y así también reducir el tiempo de respuesta.

Históricamente LDAP surgió como estándar de los directorio de servicios, la versión original fue desarrollada por la universidad de Michigan. La primer versión surgió en 1995 versión LDAPv2, en 1997 se publicó los RFC (Request For Comments) para la versión LDAPv3. La versión 3 incluye características como el de listas de acceso (ACL) y replicación de directorios [38].

5.2.2 Por qué LDAP

Para el manejo de la autenticación centralizada se seleccionó utilizar LDAP, abajo detallamos las ventajas que presenta [39] :

- Funciona directamente sobre redes TCP/IP y SSL/TLS.
- LDAP es un estándar, facilitando la integración.
- Posee lectura de registros rápidos, pues está construido para optimizar las lecturas.
- Tiene la capacidad de replicar el contenido del LDAP, permitiendo dar alta disponibilidad y reparto de carga de forma sencilla y económica.
- Dispone de un modelo de nombres globales, asegurando que todas las entradas sean únicas.
- Almacenamiento de la información de forma jerárquica. Favoreciendo a la asignación de permisos ya que se puede hacer en base a la estructura del directorio, a su vez permite aprovechar mejor el ancho de banda, ya que no es obligatorio enviar toda la estructura de directorio ante un pedido, la respuesta puede retornar parte de la estructura del directorio.
- Solución escalable. Facilita la integración de nuevos usuarios, grupos y aplicaciones al LDAP.
- Los servidores LDAP no presentan dificultad para instalarlos y mantenerlos.

La mayor ventaja de LDAP es que se puede consolidar información para toda una organización dentro de un repositorio central. Por ejemplo, en vez de administrar listas de usuarios para cada grupo dentro de una organización, puede usar LDAP como directorio central, accesible desde cualquier parte de la red.

El protocolo LDAP es implementado en la actualidad por varias empresas a continuación se detalla algunas implementaciones que han apostado en LDAP [40] :

- *Active Directory*. Desarrollado por Microsoft (desde Windows 2000) como almacén centralizado de información de uno de sus dominios de administración. El servicio Active Directory proporciona la capacidad de establecer un único inicio de sesión y un repositorio central de información para toda su infraestructura, lo que simplifica ampliamente la administración de usuarios y equipos, proporcionando además la obtención de un acceso mejorado a los recursos en red. Es un servicio de directorio, en el cual se pueden resolver nombres de URLs o de determinados recursos. Active Directory es seguro, distribuido, particionado y replicado. Está diseñado para funcionar perfectamente en una instalación de cualquier tamaño, desde sólo un servidor con algunos cientos de objetos, hasta múltiples servidores y millones de objetos [41].

- *OpenLDAP.* Desarollado de forma libre, es un protocolo independiente de la plataforma. Soporta la versión 3 de LDAP, la cual introduce mejoras a nivel de seguridad permitiendo soporte para la capa de autenticación y seguridad (SASL), soporte para la capa de conexión segura (SSL) y la seguridad de la capa de transporte (TLS), características que permiten proteger los datos que circulan por la red de los curiosos. Además OpenLDAP soporta los próximos protocolos de Internet Ip versión 6 y se puede comunicar dentro de un sistema usando comunicación interproceso (IPC) mejorando esto la seguridad al eliminar la necesidad de comunicarse a través de la red [42].

- *eDirectory.* Desarrollado por Novell, utilizado para manejar el acceso a recursos en diferentes servidores y computadoras de una red. Como ventaja se destaca que puede correr en diversas plataformas, por lo que puede adaptarse fácilmente a entornos que utilicen más de un sistema operativo. Es un servicio de directorio seguro, escalable y de alto rendimiento. Puede almacenar y gestionar millones de objetos tales como usuarios, aplicaciones, dispositivos de red y datos. NDS eDirectory soporta, de forma nativa, el directorio estándar LDAP versión 3. usa autenticación con mecanismos biométricos, smart cards, tockens y se integra con Kerberos [43].

- *Iplanet o Sun ONE Directory Server.* Desarrollado por AOL (América Online) y comercializado en conjunto con Sun Microsystems. Es un poderoso y escalable servicio de directorio distribuido basado en el estándar LDAP. El software Sun ONE Directory Server es parte de la Sun Open Net Environment. Es la piedra angular para la construcción de un repositorio de datos centralizado y distribuido que puede ser usado en la intranet, sobre la extranet con los socios comerciales o sobre internet pública para llegar a los clientes [44].

- *Apache DS.* Disponible bajo la licencia de Apache Software provee una solución de directorio escrita completamente en Java éste contiene un servidor de directorio basado en LDAP versión 3. Permite que se use con cualquier servidor LDAP, pero es diseñado particularmente para trabajar con Apache DS. Tiene disponible un conjunto de herramientas como ser un explorador LDAP que funciona en cualquier servidor LDAP, un editor de Schema que funciona en Apache DS y OpenLDAP el cual permite modificar los tipos de atributos y clases de objetos, un editor de archivos LDIF, un editor ACI (Acces control information). Apache DS basado en eclipse y es multi-plataforma [45].

- *Open DS*. Es un proyecto de código abierto de la comunidad y basado en los estándares LDAPv3 y DSMLv2, implementa un servicio de directorio, maneja la replicación con varios maestros de replicación a través de las instancias del servidor del directorio, posee fácil configuración e implementación [46].

5.2.3 Descripción y Diseño

El objetivo del prototipo propuesto se divide en tres metas, la primera es la investigación e implementación de un servidor Controlador de Dominio con Samba y OpenLDAP en un sistema operativo openSUSE 11.2, el controlador de dominio permite administrar de forma centralizada las cuentas de usuario y grupos para sistemas Unix, GNU/Linux y también para cuentas de usuario, grupos, y computadoras para sistemas Windows almacenando la información requerida para la autenticación en el directorio LDAP instalado en el servidor OpenLDAP.

La estructura del DIT instalado en el prototipo es la que se detalla en la Figura 16 el cual contiene tres contenedores:

- *Usuarios*, que almacena a las cuentas de usuarios de los sistemas Linux y Windows.
- *Computadoras*, almacena las cuentas de computadoras, denominadas Trusted Machine Accounts las cuales son cunetas de computadoras en donde la contraseña es compartida con el controlador de dominio, y éste la utiliza para verificar que realmente es el equipo correcto y no robo la identidad a ningún otro por mas que tengan el mismo nombre NetBIOS que es una direcciones de 16 bytes que se utilizan para identificar un recurso de NetBIOS en la red. Los nombres NetBIOS son nombres únicos (exclusivos).
- *Grupos*, que contiene los grupos del sistema tanto para sistema operativo Windows como Linux

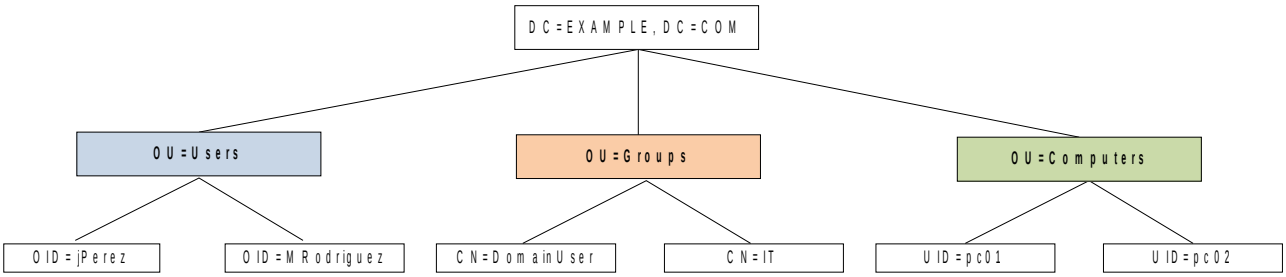

Figura 16: DIT instalado en el prototipo

La segunda meta del prototipo es la de proponer una estructura del DIT que permita centralizar la autenticación de las distintas aplicaciones existentes en ASSE. Del relevamiento realizado se detectaron los siguientes puntos que se deben de reflejar en el diseño del directorio LDAP:

- Los usuarios de una institución se pueden loguear dentro de una dependencia con determinados permisos y en otra dependencia con otros permisos distintos.
- Se pretende que un usuario de una institución pueda acceder a esa institución y no a otra.
- Existen usuarios temporarios, que acceden por un período de tiempo y luego deben de darse de baja.
- Se pretende tener un único usuario para loguearse a la red así como a las distintas aplicaciones.
- Se pretende definir un directorio LDAP para integrar ASSE a la redUY.

El diseño planteado para el directorio LDAP se detalla en la Figura 17.

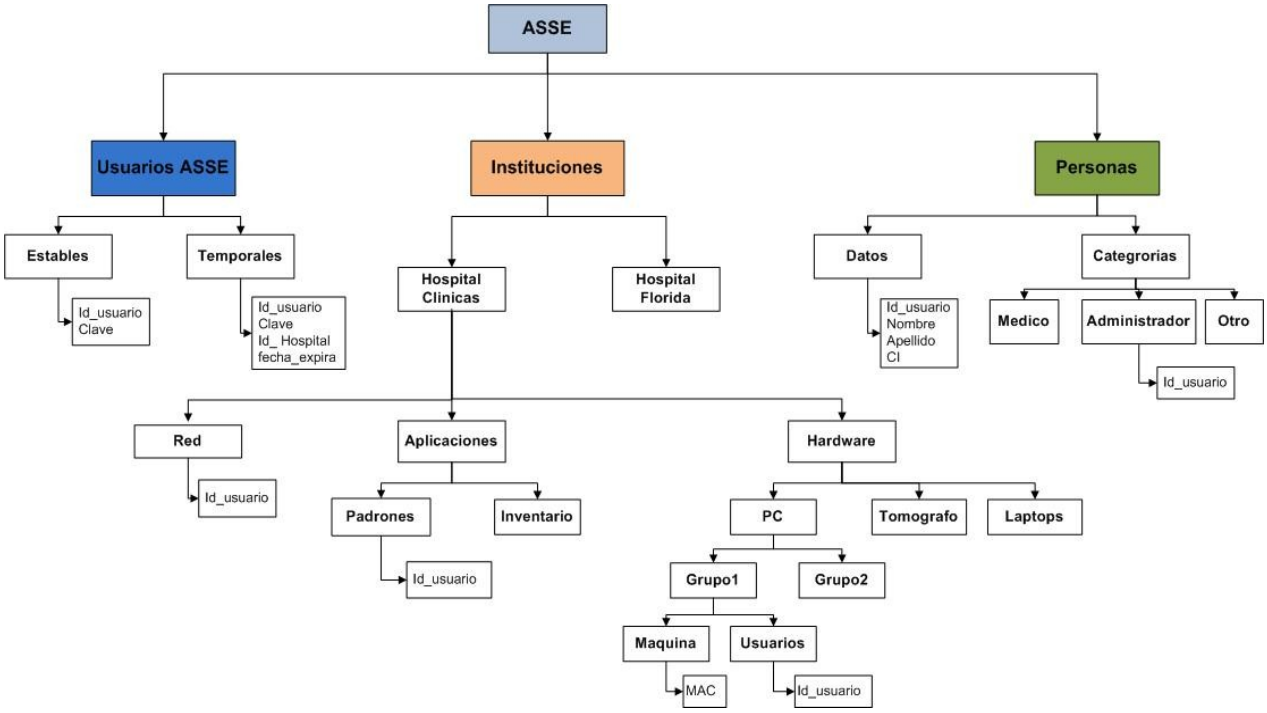

Figura 17: Diseño propuesto para el LDAP de ASSE.

La estructura cuenta básicamente de tres ramas.:

- **Usuarios ASSE:** Se corresponde al registro de usuarios que pertenecen de forma fija a la institución, y usuarios temporarios, los cuales son dados de alta por un período de tiempo. En ambos casos se almacenarán el identificador y la clave.
- **Instituciones:** Se corresponden con las instituciones de ASSE, y cada una de ellas se clasifica en 3 categorías (Red, Aplicaciones y Hardware). La categoría RED, contiene los identificadores de usuarios que pertenecen a la institución de la que deriva. La categoría Aplicaciones busca identificar las distintas aplicaciones que se ejecutan en la institución de la que deriva. La categoría Hardware identifica el hardware de la institución de la que deriva.
- **Personas:** Se corresponden con las personas que trabajan en ASSE, y dicha rama almacena los datos personales, así como el organigrama de ASSE. Ésta rama del árbol se diseña para representar la estructura de privilegios que se requieren para integrar ASSE a la redUY. En la interacción con AGESIC los entes se comunican con el nodo central, y es éste nodo quien deriva la comunicación y resuelve con quien corresponda la solicitud. Estos servicios serán publicados en la plataforma de AGESIC. Para que se pueda concretar el consumo del servicio web es necesario que cada ente del estado que va a consumir los servicios de la AGESIC tenga definido una estructura de privilegios. En el caso de ASSE si requiere consumir servicios de la DGREC, dicho organismo va a tener que contar previamente con el árbol de estructura de permisos de ASSE, para que pueda validar una vez que se quiera consumir servicios del mismo. Una recomendación del equipo de AGESIC fue tener centralizado en un directorio LDAP la estructura de privilegios el cual permita autenticar a los usuarios y esté publicado en un webServices. AGESIC recomienda que una buena fuente de información para poder armar la estructura del directorio LDAP, sea basada en el SGH (Sistema de Gestión Humana). Es por éste motivo que en el diseño planteado, se tiene la rama Personas, dicha rama intenta reflejar la estructura que AGESIC necesita para validar a los usuarios.

Con el diseño detallado anteriormente se logra identificar los usuarios por institución, pudiendo restringir el acceso de un usuario a la institución a la que pertenecen. A su vez se logra que un usuario en un institutos tenga determinados permisos y en otro instituto otros permisos. Ambos puntos fueron requeridos por la institución.

La tercer meta del prototipo es la de investigar y realizar la replica del contenido del servidor LDAP instalado en ASSE en otro servidor secundario o esclavo, el cual permite brindar alta disponibilidad de los datos, tener los datos distribuidos en distintos puntos geográficos mejorando el tiempo de acceso e impidiendo tener inconsistencia de datos ya que los mismos se encuentran sincronizados con el sistema central.

Los detalles del prototipo de autenticación centralizada se ubican en la sección de Apéndices.

6 Conclusiones y trabajo futuro

En este documento se ha planteado una recomendación de arquitectura de software para el sistema informático de ASSE. Se mostraron diferentes herramientas, estándares en salud y un ejemplo real de como aplicar los mismos. Evidentemente el documento queda escaso teniendo en cuenta la amplitud del tema y contemplando los avances tecnológicos actuales.

En este capítulo se presentan las conclusiones obtenidas al término de éste proyecto y se plantean recomendaciones y temas como posibles trabajos futuros.

6.1 Conclusiones

Al inicio de este proyecto, nos encontramos con la dificultad de conocer la realidad y el funcionamiento de los sistemas de salud; la terminología médica, los estándares de comunicación en salud, la diversidad de actores involucrados y las complejas interacciones entre ellos. Los sistemas de salud comparten características similares con la mayoría de las organizaciones de servicio y producción, pero revelan características específicas relacionadas con la complejidad y la diversidad de los procesos de atención de salud, sumado a las diferentes formas con que los profesionales realizan las tareas clínicas.

Dentro de ASSE pudimos ver que existen sistemas informáticos para el soporte de estos procesos pero estos desarrollos se realizaron en forma independiente y sin el debido involucramiento del personal médico y técnico de la institución. Estos desarrollos independientes se deben a que antes de la creación de ASSE, las instituciones pertenecían al sector público pero no había ningún organismo central que tuviera un sector de informática para marcar las pautas en el desarrollo de software, cada institución realizaba la compra o el desarrollo de sistemas según sus necesidades y sus especificaciones. Al existir software desarrollado por empresas contratadas y desarrollos propios de ASSE, creemos que es de vital importancia que se comiencen a definir en la institución las pautas que deben tener en común todos los sistemas para permitir la interoperabilidad, cooperación y el manejo adecuado de los datos, sin redundancia. En este sentido la definición y adopción de una arquitectura de software es el primer paso para comenzar a modificar los sistemas existentes y para exigir ciertos requerimientos en los futuros desarrollos, tanto si se realizan dentro o fuera de ASSE. La interoperabilidad interna entre los sistemas y el uso de estándares utilizados en la región e incluidos en la agenda digital del país, permitirá que ASSE pueda interactuar fácilmente con entidades externas para brindar un mejor servicio y más recursos al paciente.

El proyecto de grado se planteó con el objetivo de recomendar una arquitectura de servicios que contemplara los sistemas actuales de ASSE, definiera una nueva forma de interacción y convivencia entre ellos y planteara recomendaciones para los desarrollos futuros. La principal aplicación práctica de esta propuesta de arquitectura era la refactorización del sistema AP/SGA, uno de los sistemas más grande de ASSE y con una estructura monolítica y cerrada. Al comenzar a trabajar para llegar a los objetivos planteados, notamos que la definición de una arquitectura para un sistema de salud no puede realizarse sin tomar en cuenta el contexto en el que está inmerso el sistema y los desafíos planteados para su funcionamiento. El desarrollo del proyecto permitió documentar los sistemas existentes y las debilidades de la institución y culminó con la recomendación de una arquitectura de referencia ampliamente conocida en la región y en Uruguay. Además de recomendar una arquitectura, se brindaron recomendaciones para integrar los sistemas actuales a la misma. Por todo lo expuesto, consideramos que los objetivos propuestos al inicio del proyecto fueron parcialmente cumplidos, tomando en cuenta la dificultad de relevar la situación actual en ASSE y cuales son los requisitos a futuro para la institución.

6.2 Trabajo futuro

En esta sección se presentan los posibles temas y trabajos a futuro que evaluamos de interés, cuya aplicación y/o investigación podrían resultar de importancia a pesar de no haber sido incluidos en el alcance de este proyecto.

- **Normalización de los registros de la HCE**: La información generada por los distintos sistemas debe normalizarse para que pueda ser compartida entre los sistemas. El SIQ ya cuenta con la capacidad de generar un CDA que puede ser almacenado junto con los resultados de análisis clínicos obtenidos de los laboratorios externos. Este repositorio debe ser utilizado por el Escritorio Clínico, que es quién maneja la historia clínica del usuario, para contar con toda la información generada a través de los distintos procesos de la institución y no únicamente la generada en los consultorios.
- **Repositorio central de usuarios:** Del relevamiento se obtuvo que cada aplicativo de forma individual lleva la administración de los usuarios y roles internamente, replicando datos y dificultando la gestión de los mismos este tipo de diseño es poco escalable. Como mejora se propuso gestionar de forma central en el directorio LDAP los usuarios. Existen dos puntos en los cuales se deben de profundizar:
  - **Integrar al directorio LDAP propuesto la autenticación de los aplicativos en ASSE:** La gestión de los usuarios y claves de los aplicativos debieran de autenticase en el directorio LDAP propuesto, para lograr éste punto se debe de anexar al directorio LDAP las ramas adecuadas, un diseño posible a implementar puede ser el propuesto en el prototipo, el cual cumple con las necesidades que ASSE requiere. Cabe mencionar que se deberá analizar cada aplicativo de forma puntual para replicar el mismo esquema de seguridad en el directorio LDAP.
  - **Migrar los usuarios de los aplicativos actuales al directorio LDAP.** Una vez configurado el directorio LDAP, el próximo paso es el de migrar los usuarios al DIT. Una vez realizada la migración las aplicaciones podrán comenzar a utilizar el LDAP administrado por ASSE para controlar la seguridad en el acceso a las aplicaciones. Logrando de ésta forma tener un maestro de usuarios único.
- **Implementación de firma digital**: Al contar con un repositorio documental de datos clínicos a través del uso del estándar CDA, se debe agregar la firma digital a todos los documentos para que tengan la misma validez que los generados en papel. Para ésto es necesario definir quién será el encargado de distribuir estas firmas digitales y cuál será el mecanismo utilizado para su utilización

El documento publicado por la CEPAL [3] al término de la confección de este informe, podría ser un buen punto de referencia para la planificación del trabajo futuro pues aporta una visión global del sistema de salud, indicando pautas y recomendaciones.

A continuación se presenta un resumen de las principales áreas a atacar en trabajos futuros en base a este documento.

Un punto fundamental en los sistemas de salud es la planificación y control de la gestión de los mismos. Un sistema de salud, visto como cualquier organización, posee un área directiva que traza las metas y objetivos para un período dado, y en base a estas directrices asigna los recursos necesarios basado en el presupuesto otorgado para dicho período. Esta labor se conoce como Gestión y abarca, la planificación de objetivos y estrategias, la organización para la asignación de recursos y el control del cumplimiento de las metas trazadas.

Un elemento clave para la gestión es la disponibilidad de información certera y en tiempo real para conocer posibles desviaciones en el cumplimiento de objetivos y realizar una corrección oportuna. Además, al final de cada período, la información recabada durante el mismo permite conocer los resultados obtenidos en la aplicación de estrategias trazadas y permite conocer el escenario de partida para planear el próximo período.

Los sistemas de e-Health permiten dejar disponible la información generada en las distintas áreas del sistema sanitario, lo cual es de vital importancia para los directivos de las instituciones en la toma de decisiones y planificación de estrategias

Una parte importante en la toda la administración digital es la gestión de informes, la gestión de imágenes y control de téctnicas de laboratorio, todo esto está definido en el termino Patología Digital o Telepatología. Los servicios de anatomía patológica son fundamentales para la detección precoz de enfermedades y la atención sanitaria.

El objetivo principal de los sistemas de información de anatomía patológica (SIAP), es gestionar eficientemente imágenes y datos para la generación de informes para ser incorporados a la historia clínica del paciente. Algunas de las funcionalidades propias del SIAP son, identificación y gestión de pacientes, registro y gestión de muestras, elaboración y circuito de informes, archivo de muestras, sistemas de trazabilidad y control de flujo, planificación y distribución del trabajo , y control de calidad de procesos técnicos y de diagnósticos.

Las soluciones integradas de patología digital con la historia clínica electrónica y el almacenamiento centralizado de imágenes mejorarán la cooperación entre clínicos y patólogo y aumentarán la seguridad del paciente y la calidad de los servicios de salud. Actualmente permiten una gran flexibilidad en cuanto a la emisión de informes de anatomía patológica respecta, pues es posible generar diferentes tipos y formatos de informes para diversos usuarios con distintos niveles de comprensión.

Controlar costos crecientes, optimizar procesos y reasignar recursos son retos permanentes de cualquier sistema de salud. Tomando en cuenta que el medicamento es la tecnología sanitaria más utilizada y que un importante porcentaje del gasto sanitario corresponde a esta área, utilizarlos de manera racional es uno de los objetivos más importantes. Llamaremos entonces Gestión electrónica de la farmacoterapia a toda la administración de fármacos por medio de la tecnología informática.

En el documento se plantea de forma simplificada una propuesta de dicha incorporación a las TIC para la gestión de la farmacoterapia con una visión global, independiente e integradora. Para ello hay que tomar en consideración los siguientes elementos de la cadena del medicamento: adquisición, prescripción, dispensación, administración, facturación y explotación de la información para la toma de decisiones asistenciales, de gestión y económicas.

Contar con una base de datos de medicamentos (fundamental para que funcionen adecuadamente la prescripción, la dispensación, las adquisiciones de medicamentos y facturación) y un buen sistema de información son elementos básicos de la gestión de la farmacoterapia. Los sistemas de información de farmacia conforman una parte indispensable para el seguimiento de la prestación farmacéutica, para la toma de decisiones de gestión y para el desarrollo de políticas de uso racional de los medicamentos. La reseta informatizada y electrónica constituyen un elemento esencial para la mejora de la legibilidad y prevención de errores en el proceso que lleva desde la prescripción, dispensación y utilización de los medicamentos en los sistemas de salud.

Bibliografía

[1] Sistema de Salud según OMS: *http://www.who.int/features/qa/28/es/index.html* - Último acceso<04/04/2012>

[2] Tendencias y temas emergentes en salud.: *http://hinfo.humaninfo.ro/gsdl/healthtechdocs/documents/s16588s/s16588s.pdf* - Último acceso<04/04/2012>

[3] Manual de Salud Electrónica para directivos de servicios y sistemas de salud.: *http://www.eclac.org/cgi-bin/getProd.asp?xml=/prensa/noticias/comunicados/3/46103/P46103.xml&xsl=/prensa/tpl/p6f.xsl&base=/tpl/top-bottom.xsl* - Último acceso<04/04/2012>

[4] HL7 Latam: *http://hl7latam.org/index.php/acerca-de-hl7-latam* - Último acceso<04/04/2012>

[5] Hl7 Latam - Argentina: *(http://www.hl7latam.org/HL7LATAMNews/N1* - Último acceso<04/04/2012>

[6] Hospital Italiano de Buenos Aires: *http://www.hospitalitaliano.org.ar/infomed/index.php?contenido=ver_seccion.php&id_seccion=56* - Último acceso<04/04/2012>

[7] Componentes de Italica: *http://www.slideshare.net/HIBACampusVirtual/sistemas-de-informacin-en-salud* - Último acceso<04/04/2012>

[8] "Incorporación de tecnologías de la informacióny de las comunicaciones en elHospital Italiano de Buenos Aires" - Fernán González Bernaldo de Quirós, Daniel Luna, Analía Baum, Fernando Plazzotta,Carlos Otero, Sonia Benítez - Enero, 2012

[9] Presidencia de la República: *http://archivo.presidencia.gub.uy/_web/noticias/2007/12/2007120703.htm* - Último acceso<04/04/2012>

[10] Sistema Nacional Integrado de Salud: *http://www.smu.org.uy/sindicales/documentos/snis/snis.pdf* - Último acceso<04/04/2012>

[11] eclac: *http://www.eclac.cl/publicaciones/xml/5/41825/di-salud-electrinica-LAC.pdf* - Último acceso<04/04/2012>

[12] Gobierno Electrónico: *http://www.agesic.gub.uy/innovaportal/v/163/1/agesic/gobierno_electronico_.html* - Último acceso<04/04/2012>

[13] Agenda Digital en América Latina: *http://www.eclac.org/ddpe/publicaciones/xml/1/39181/W314Esp.pdf* - Último acceso<04/04/2012>

[14] IV Encuentro Nacional de Gobierno Electrónico: *http://www.agesic.gub.uy/innovaportal/v/1804/1/agesic/ente_todos_y_para_todos:_iv_encuentro_nacional_de_ge.html* - Último acceso<04/04/2012>

[15] Agenda Digital 2011-2015: *http://www.agesic.gub.uy/innovaportal/file/1443/1/agenda_digital_2011_2015.pdf* - Último acceso<04/04/2012>

[16] Uruguay Digital: *http://www.agesic.gub.uy/innovaportal/v/1803/1/agesic/por_un_uruguay_digital:_anuncios_de_gobierno_electronico.html?menuderecho=13* - Último acceso<04/04/2012>

[17] RAU2: *http://www.rau.edu.uy/redavanzada/rau2* - Último acceso<04/04/2012>

[18] AGESIC: *http://www.agesic.gub.uy/innovaportal/v/19/1/agesic/que_es_agesic.html* - Último

acceso<04/04/2012>

[0] : *http://www.agesic.gub.uy/innovaportal/v/759/1/agesic/reduy.html* - Último acceso<04/04/2012>

[19] Basic Profile: *http://www.ws-i.org/Profiles/BasicProfile-1.1.html* - Último acceso<04/04/2012>

[20] Basic Security Profile: *http://www.ws-i.org/Profiles/BasicSecurityProfile-1.0.html* - Último acceso<04/04/2012>

[21] WS-Trust 1.3: *http://docs.oasis-open.org/ws-sx/ws-trust/200512/ws-trust-1.3-os.html* - Último acceso<04/04/2012>

[22] Servicios ofrecidos por RedUy: *http://www.agesic.gub.uy/innovaportal/v/759/1/agesic/REDuy.html* - Último acceso<04/04/2012>

[23] "El Sistema Nacional Integrado de Salud en Uruguay y los desafíos para la Atención Primaria" - Ana Sollazzo, Rosario Berterretche - Febrero, 2011

[24] Ley 18.161 - Creación de ASSE: *http://www0.parlamento.gub.uy/leyes/AccesoTextoLey.asp?Ley=18161&Anchor=* - Último acceso<04/04/2012>

[25] Contexto ASSE: *Contexto ASSE* - Último acceso<04/04/2012>

[26] Página oficial de LOINC: *http://loinc.org/* - Último acceso<04/04/2012>

[27] Página oficial de HL7: *www.hl7.org* - Último acceso<04/04/2012>

[28] Página oficial de ASTM: *www.astm.org* - Último acceso<04/04/2012>

[29] Página oficial de DICOM: *http://medical.nema.org/* - Último acceso<04/04/2012>

[30] DICOM - Wikipedia: *http://es.wikipedia.org/wiki/DICOM#Ejemplo_de_im.C3.A1genes_DICOM* - Último acceso<04/04/2012>

[31] Página oficial de CIE10: *http://www.cie10.org* - Último acceso<04/04/2012>

[32] Página principal de IUPAC: *http://www.iupac.org/* - Último acceso<04/04/2012>

[33] "Atención Primaria en Salud" - -

[34] Documentación de Usuario: Sistema de Gestión de Salud para la Red de Salud Pública del Uruguay, Junio, 2007

[35] Proyecto Siembra: *http://www.elacontecer.com.uy/11659-antel-y-asse-lanzan-hoy-proyecto-siembra.html* - Último acceso<04/04/2012>

[36] Proyecto Siembra-ANTEL: *http://www.antel.com.uy/wps/wcm/connect/e8008d004702da2da039a896c21f9a18/Asse-Siembra.pdf?MOD=AJPERES&CACHEID=e8008d004702da2da039a896c21f9a18* - Último acceso<04/04/2012>

[37] Definición LDAP: *http://es.wikipedia.org/wiki/LDAP* - Último acceso<04/04/2012>

[38] Conceptos LDAP: *http://www.goa.es/docs/curso_openldap.pdf* - Último acceso<04/04/2012>

[39] Caracteristicas LDAP: *http://www.goa.es/docs/curso_openldap.pdf* - Último acceso<04/04/2012>

[40] Ventajas LDAP: *http://www.rediris.es/jt/jt2000/trans/jt2000-2-3* - Último acceso<04/04/2012>

[41] Active Directory: *http://www.microsoft.com/latam/technet/productos/windows/windowsserver2003/adsrv.mspx* - Último acceso<04/04/2012>

[42] OpenLDAP: *http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/ch-ldap.html* - Último acceso<04/04/2012>

[43] e-directory: *http://www.novell.com/es-es/documentation/ndsedir/docui/] - http://www.novell.com/products/edirectory/* - Último acceso<04/04/2012>

[44] Sun ONE Directory Server: *http://docs.oracle.com/cd/E19850-01/816-6697-10/816-6697-10.pdf* - Último acceso<04/04/2012>

[45] Apache DS: *http://directory.apache.org/studio/* - Último acceso<04/04/2012>

[46] Opend DS: *https://www.opends.org/wiki/page/DirectoryOverview* - Último acceso<04/04/2012>

APÉNDICE A - Prototipo de Almacenamiento de Resultados de Análisis Clínicos

En esta sección se brinda el detalle del diseño e implementación del prototipo para la persistencia de resultados de análisis clínicos en un documento CDA.

1.1 Introducción

Las distintas instituciones que pertenecen a ASSE, utilizan los servicios brindados por diferentes laboratorios de análisis clínicos. Estos laboratorios cuentan con software propio para realizar el manejo de los análisis solicitados y sus respectivos resultados, producto de que estas instituciones originalmente eran consideradas como unidades independientes, sin ningún tipo de comunicación.

Según datos relevados por ASSE, el 80% de los laboratorios externos utilizan el software Modulab provisto por la empresa Izasa. Este software cuenta con distintas versiones, la más utilizada es la versión Win.

Los resultados de análisis clínicos son una parte esencial de la historia clínica electrónica (HCE) de un paciente. El objetivo principal de cualquier sistema que gestione una HCE en ASSE debe ser que la misma pueda ser compartida entre otras instituciones debido a la gran heterogeneidad y dispersión geográfica de las instituciones que conforman la administración.

El estándar en salud HL7 cuenta con un estándar de arquitectura de documentos clínicos llamado CDA que fue especialmente diseñado para registrar actos médicos como internaciones, análisis clínicos, etc. Con el objetivo de facilitar el intercambio de información.

1.2 Objetivo del prototipo

El objetivo del prototipo presentado es mostrar la interacción con distintos laboratorios para obtener resultados de análisis clínicos digitalizados y su posterior transformación en un registro clínico almacenado en formato CDA para ser persistido en ASSE. La elección de un CDA para persistir los resultados, se debió a que es un estándar y además resulta fácil de firmar digitalmente, permitiendo así la integridad y auditoria de cada documento producido.

Para mostrar la factibilidad del proyecto, el prototipo mostrará la interacción con el software Modulab en su versión Win.

1.3 Relevamiento

El módulo de Laboratorio es uno de los nuevos módulos que se están desarrollando en ASSE. La interacción con los laboratorios es una oportunidad para almacenar los resultados de análisis clínicos en forma digital y comenzar a trabajar en la historia clínica electrónica. El trabajar con un estándar para guardar estos resultados permite tener la posibilidad de interoperar entre otros módulos de ASSE o con sistemas externos.

Se eligió utilizar el software de Modulab en su versión Win para llevar a cabo las primeras pruebas de interacción porque es el software con el que trabajan la mayoría de los laboratorios relacionados a ASSE. El software cuenta con la posibilidad de utilizar el sistema de archivos local del laboratorio para leer solicitudes de análisis de un archivo y para dejar los resultados de los análisis que se van realizando en otro.

En la actualidad, existe un webservice que permite solicitar análisis clínicos, escribiendo un mensaje de petición para Modulab en un archivo. La máquina donde corre Modulab tiene instalado el software Cygwin, que emula un ambiente Linux. Este ambiente permite instalar un servidor ssh para realizar la comunicación segura entre las máquinas cliente (ASSE) y el host (Modulab).

1.3.1 Marco teórico

En esta sección se presentarán los conceptos teóricos que el lector debe tener para entender la solución planteada. Se presentan los conceptos básicos de las distintas herramientas utilizadas y los principales conceptos necesarios para entender el funcionamiento del software desarrollado así como del software con el cual se interactúa.

1.3.1.1 XML

Extensible Markup Language (XML) es un estándar creado por World Wide Web Consortium (W3C) [1], que especifica un conjunto de reglas para generar un documento de texto que pueda ser leído por una máquina. El estándar enfatiza en la simplicidad, usabilidad y generalidad a través de Internet. La estructura de un archivo XML se define a través de dos conceptos: marcado y contenido. Una marca es una cadena de caracteres encerrada entre los símbolos '<' y '>' o entre '&' y ';'. Todas las demás cadenas que no cumplen esta regla se leen como contenido. Con estas dos estructuras se definen los elementos de un XML:

- **Tag (Etiqueta):** es una marca que comienza con el caracter '<' y finaliza con '>'. Los tags se dividen en tres tipos: ***"tag de comienzo"***, por ejemplo <style>, ***"tag de finalización"***, por ejemplo </style> y ***"tag sin elemento"***, por ejemplo <line-break/>
- **Element (Elemento):** Los caracteres incluídos dentro un tag de inicio y un tag de finalización, corresponden al concepto lógico de Elemento dentro del XML. Un elemento puede contener otros elementos los cuales son considerados elementos hijos. Así se forma una estructura jerárquica.
- **Attribute (Atributo):** Un atributo es un par nombre/valor que se encuentra dentro de un tag de inicio o un tag sin elemento.

1.3.1.2 Estándar de documentación CDA

El estándar HL7 Clinical Document Architecture (CDA) fue creado para homogeneizar la utilización de los documentos clínicos generados por un paciente a lo largo de su vida. Algunos ejemplos de documentos clínicos son: análisis de sangre, tomografías, rayos-X, altas hospitalarias o internaciones, etc. El estándar CDA define los documentos clínicos con las siguientes propiedades:

- Persistencia y Administración
- Firmados digitalmente para su autenticación
- Contexto
- Totalidad
- Lectura humana

Debido al uso del Reference Information Model (RIM) de HL7, la utilización de XML y la codificación del vocabulario utilizado, el estándar hace posible que los documentos clínicos sean fácilmente accesibles tanto por software (ya que son fácilmente parseados) como por los actores humanos que requieran la información almacenada [2].

1.3.1.3 Object Identifier (OID)

Un Identificador de Objeto (OID) es una representación numérica universal que permite identificar cualquier tipo de objetos. Cada OID se define por una estructura jerárquica establecida por la Organización Internacional de Estandarización (ISO) que garantiza la unicidad de los mismos. Cada OID contiene una raíz que identifica el país donde se define. En Uruguay esa raíz es: **2.16.858** [3]

1.3.1.4 Protocolo SSH [4]

El protocolo Secure Shell (SSH) es un protocolo de comunicaciones que utiliza la arquitectura cliente/servidor para conectar dos máquinas(hosts) remotas. La principal característica es que los mensajes intercambiados por el protocolo son encriptados, lo cual permite una comunicación segura punto a punto.

1.3.1.4.1 Métodos de autenticación

Todo cliente que desee comunicarse con un servidor SSH debe autenticarse. El protocolo define distintos métodos para realizar esta autenticación, la configuración que se haga en el servidor establecerá los métodos utilizados.

Autenticación basada en usuario/contraseña

Es el método de autenticación clásico. El servidor guarda en uno de sus archivos de configuración, los usuarios y contraseñas habilitados para conectarse. El cliente envía sus datos al momento de la conexión y el servidor los compara con los datos almacenados.

Autenticación basada en host/usuario

Para utilizar este método, el servidor SSH mantiene un archivo con los usuarios que pueden conectarse desde cada máquina. Al momento de la conexión se chequea que el usuario está autorizado a conectarse desde la máquina en la que se encuentra.

Autenticación mediante claves

Este método se realiza a través del uso de pares de clave pública/privada. El cliente debe generar un par de claves pública/privada y luego compartir su clave pública con el servidor. De esta manera, el servidor autentica al cliente al momento de la conexión utilizando su clave pública, sin requerir de usuarios y contraseñas. La Tabla 1 muestra los pasos que se deben seguir para utilizar este método de autenticación.

1. El cliente debe generar un par de claves utilizando el comando: ***ssh-keygen -t <algoritmo>*** donde el algoritmo elegido puede ser: rsa o dsa (ssh versión 2) ó rsa1 (ssh versión1).
2. Al ejecutar el comando anterior se generan dos archivos. ***id_<alg>*** contendrá la clave privada que sólo debe ser conocida por el cliente y el archivo ***id_<alg>.pub*** que contendrá la clave pública.
3. Para compartir la clave pública con el servidor ssh, se debe agregar la clave pública en el archivo ***authorized_keys2*** del servidor.
4. Una vez realizados estos pasos, el cliente se puede comunicar con el servidor ssh elegido utilizando el intercambio de claves en lugar de utilizar usuario y contraseña.

Tabla 1: Pasos a seguir para utilizar intercambio de claves como método de autenticación.

1.3.1.5 JDOM [5]

El Document Object Model (DOM) es una iniciativa de World Wide Web Consortium (W3C) que presenta una interfaz que permite manipular y representar de forma estándar documentos HTML y XML. A través del DOM, los programas pueden acceder y modificar el contenido de este tipo de documentos. La interfaz provista por DOM es independiente de la plataforma y lenguaje de programación utilizados.

JDOM es una biblioteca de código abierto para la manipulación de datos XML utilizando el lenguaje Java. Aunque es similar a DOM, su principal diferencia es que se encuentra optimizado para el lenguaje Java y toma ventaja de las características del lenguaje como ser el uso de colecciones, herencia de clases, etc. Por lo tanto, JDOM permite la manipulación del contenido y la estructura de archivos XML o HTML a través de una interfaz simple, al igual que lo hace DOM, pero con construcciones propias del lenguaje Java.

1.4 Modulab Win

En esta sección se presentarán las características del software Modulab Win. Este software se encarga de la gestión de laboratorios de análisis clínicos y se encuentra instalado en un gran porcentaje de laboratorios con los cuales trabaja ASSE.

1.4.1 Características

Modulab Win trabaja con mensajes ASTM (registros con campos separados por un carácter), tanto para agendar peticiones de estudios como para la comunicación de sus resultados. El estándar ASTM especifica las convenciones para estructurar el contenido del mensaje y para representar los datos dentro de esas estructuras. En acuerdo con ASSE, el software lee las peticiones desde un archivo y guarda los resultados en otro.

Un análisis clínico puede contener un conjunto de pruebas simples para realizar. Por ejemplo, un hemograma completo involucra la realización de varias pruebas de sangre. A medida que el técnico del laboratorio obtiene el resultado de alguna de estas pruebas, lo ingresa al sistema y el nuevo resultado se agrega al análisis clínico original y se graba en el archivo de salida. Esto significa que si un estudio contiene tres pruebas simples, en el archivo de resultados podemos tener tres mensajes ASTM con resultados parciales, mostrando la actualización de la información a medida que se va adquiriendo.

1.4.2 Mensajes utilizados por Modulab

Los resultados de los análisis clínicos recibidos por ASSE desde los laboratorios llegan en formato ASTM como un conjunto de segmentos de campos, donde cada campo viene separado por un carácter fijo y conocido ('|'). Estos resultados cuentan con la siguiente estructura:

Segmento	*Cantidad*
Header Record (H)	1
Patient Information Record (P)	1
Test Order Record (O)	1..N
Result Record (R)	1..N
Terminator Record (L)	1

Se detalla a continuación el formato de cada segmento:

Header Record

Siempre debe estar presente y es el primer segmento en la transmisión.

Nombre del campo ASTM	*Descripción*
Record Type	**H**
Delimiters Definition	Always the standard "|\^&"
Message Control ID	Ignored
Access Password	Ignored
Sender Name or ID	Ignored when received. Modulab Gold always send "Moduwin"
Sender Street Address	Ignored
Reserved Field	Ignored
Sender Phone Number	Ignored
Characteristics of Sender	Ignored
Receiver ID	Ignored when received. Modulab Gold always send the text "Host LIS"
Comment or Special Ins	Ignored
Processing ID	Always assumed "P" when received. Modulab always send "P"
Version Number	Ignored when received. Modulab always send the text "1.01"
Message Date and Time	In standard format YYYYMMDDHHMMSS

Patient Information Record

Define los atributos del paciente al cual se le realizará el análisis clínico.

Nombre del campo ASTM	*Descripción*
Record Type	**P**
Sequence Number	Sequence number of the patient transmitted
Practice Assigned Patient ID	Modulab History Number (12 characters)
Laboratory Assigned Patient ID	Ignored
Patient ID No. 3	Ignored
Patient Name	Modulab Name (30 characters)

Nombre del campo ASTM	*Descripción*
Mother's Maiden Name	Ignored
Birthdate	Two fields. First field is birthdate and must be in format YYYYMMDD, second field is the age. Only will be transmited the second field
Patient Sex	M for male, F for female or U for unknown
Patient Race	Ignored
Patient Address	Three fields. Modulab Adress, City and State
Reserved Field	Ignored
Patient Telephone	Modulab Phone (12 characters)
Attending Physician ID	Modulab Doctor (16 characters)
Special Field 1	Modulab Comment (25 characters)
Special Field 2	Two fields. Modulab Fee Code (2 characters) and Discount (6 characters)
Patient Height	Ignored
Patient Weight	Ignored
Patient Diagnosis	Modulab Diagnostics (8 characters)
Patient Medications	Ignored
Patient Diet	Ignored
Practice Field No. 1	Ignored
Practice Field No. 2	Ignored
Admission and Discharge Dates	Modulab Date and Time of Registration
Admission Status	Ignored
Location	Modulab Location (8 characters)
Nature of Alt.Diag.Code & Class	Ignored
Alternative Diagnostic Code and Classification	Ignored
Patient Religion	Ignored
Marital Status	Ignored
Isolation Status	Ignored
Language	Ignored
Hospital Service	Three fields. Modulab Origin (8 characters), Area (2 characters) and Type (2 characters)
Hospital Institution	Ignored
Dosage Category	Ignored

Test Order Record

Define los atributos de un pedido de orden particular.

Nombre del campo ASTM	*Descripción*
Record Type	**O**
Sequence Number	Sequence number of the test order transmitted.
Specimen ID	Modulab Number (8 characters)
Instrument Specimen ID	Ignored
Universal Test ID	Four fields. Identifier is ignored Name is Modulab Description Type is Modulab Test Host Code Local Code: Test is Modulab Test Code
Priority	Ignored

Nombre del campo ASTM	*Descripción*
Requested Date and Time	Must be in format (YYYYMMDDHHMMSS)
Collection Date and Time	Must be in format (YYYYMMDDHHMMSS)
Collection End Time	Ignored
Collection Volume	Ignored
Collector ID	Ignored
Action Code	Ignored
Danger Code	Ignored
Relevant Clinical Info	Ignored
Date/Time Specimen Received	Ignored
Specimen Descriptor	Ignored
Ordering Physician	Ignored
Physician's Phone Number	Ignored
User field Number1	Ignored
User field Number2	Ignored
Laboratory field No 1	Ignored
Laboratory field No 2	Ignored
Date/Time	Ignored
Instrument Charge to Computer	Ignored
Instrument Section ID	Ignored
Report Types	Ignored
Reserved Field	Ignored
Location or ward of speciment	Ignored
Nosocomial infection flag	Ignored
Specimen Service	Ignored
Specimen Institution	Ignored

Result Record

Define los atributos del resultado del análisis clínico.

Nombre del campo ASTM	*Descripción*
Record Type	**R**
Sequence Number	Sequence number of the result record transmitted
Universal Test ID	Four fields. Identifier is ignored Name is Modulab Description Type is Modulab Test Host Code Local Code: Test is Modulab Test Code
Data Measurement	Modulab Result (45 characters)
Units	Modulab Units (10 characters)
Reference Ranges	Modulab Reference Range (10 characters)
Result Abnormal Flags	Defines the normalcy status of the result. Values shall be: L for below low normal. H for above high normal. LL for below panic normal HH for above panic normal.
Nature of Abnormality testing	Ignored
Result Status	^Status (V = Validated, I = Printed)
Date of Change	Ignored

Nombre del campo ASTM	*Descripción*
Operator Identification	^User
Date/Time Test Started	Ignored
Date/Time Test Completed	Ignored
Instrument Identification	Ignored

Message Terminator Record

Este es el último segmento del mensaje. Un Header Record debe ser transmitido después de este segmento significando el comienzo de un nuevo mensaje.

Nombre del campo ASTM	*Descripción*
Record Type	**L**
Sequence Number	Sequence number of the result record transmitted
Terminator Code	Ignored

Una vez presentada la estructura y formato de los resultados de análisis clínicos enviados por Modulab, pasaremos a presentar el mapeo realizado entre los resultados obtenidos y el documento CDA producido.

1.5 Diseño de la solución

En esta sección se muestran las decisiones de diseño realizadas para implementar el prototipo. Se presentan los archivos de configuración utilizados y el diseño de clases. Además se muestra el mapeo de campos realizados entre el mensaje ASTM y los campos de la estructura del CDA. Este mapeo lo implementa la clase ASTM_CDAParser que es la encargada de generar el contenido del CDA para la interacción del software con los laboratorios que utilizan el software ModulabWin.

1.5.1 Diagrama de clases

Se presenta a continuación el diagrama de clases de la solución y se realiza una breve descripción de cada una de las clases involucradas en el diseño.

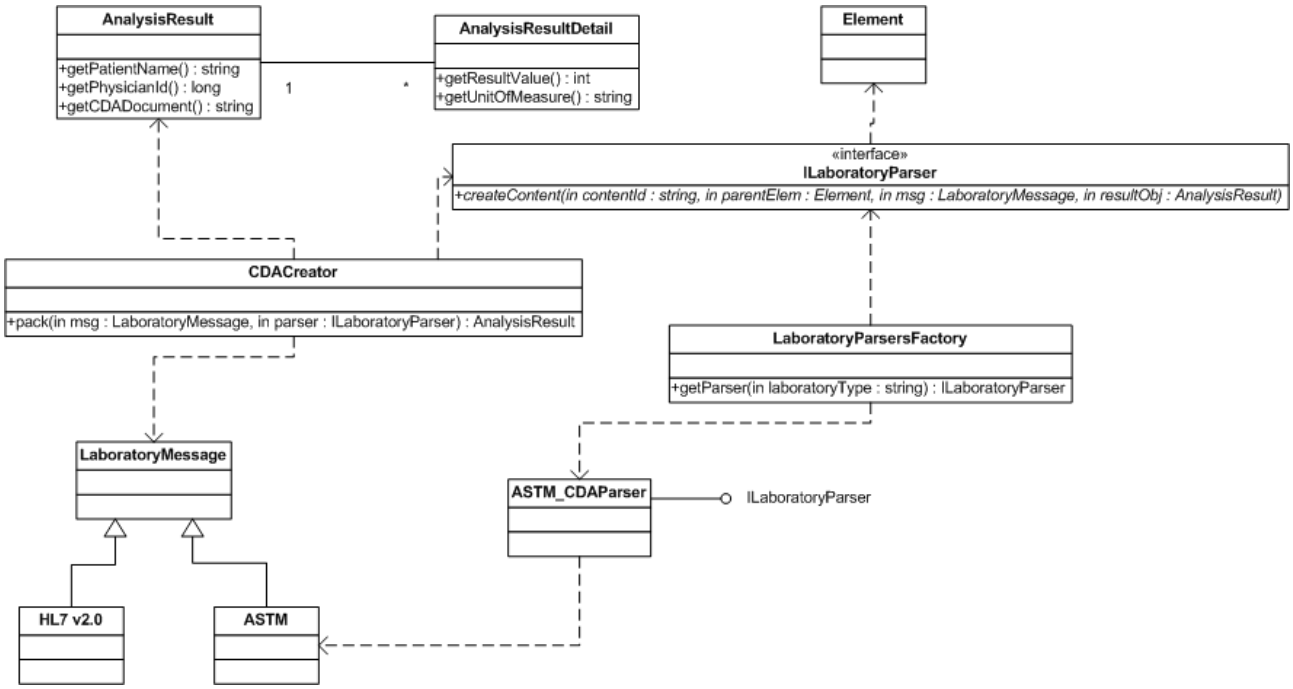

Figura 1: Diagrama de clases.

1.5.1.1 AnalysisResult & AnalysisResultDetail

Para evitar problemas de redundancia de datos, lo recomendable es que los datos de cada resultado se recuperen del documento CDA, aprovechando las nuevas capacidades de los motores de base de datos para realizar consultas en documentos XML. Dado que aún no se ha definido esta plataforma en el ambiente de ASSE, se crean las clases AnalysisResult y AnalysisResultDetail para recuperar el CDA y los campos más relevantes del mismo. La clase AnalysisResult cuenta con campos como PatientId, PhysicianId que se graban como campos separados del CDA en la base de datos. La clase AnalysisResultDetail almacena los resultados de cada prueba incluida en el análisis clínico.

La clase AnalysisResult será la responsable de generar un hash criptográfico para validar que los campos grabados individualmente se correspondan con el CDA almacenado. Esta key permite validar que los campos que están fuera del CDA no sean actualizados luego de generar el CDA. De esta forma se mantiene la propiedad que el CDA es el documento certificado que contiene la información.

1.5.1.2 LaboratoryMessage

Esta clase tiene la responsabilidad de abstraer cada uno de los mensajes obtenidos desde los laboratorios con los cuales se interactúa. Cada implementación y definición concreta de un mensaje utilizado por un laboratorio dado debe definirse como clase derivada de LaboratoryMessage. En el caso del prototipo creado, el mensaje ASTM recibido desde el software Modulab deriva de esta clase.

1.5.1.3 ILaboratoryParser

Esta interfaz define el método createContent que será invocado para crear cada rama del XML definido por el CDA. Las clases que implementen esta interfaz serán las encargadas de generar el contenido de la estructura del CDA. Los parámetros que recibe el método son:

- **contentId**: String que define la rama que se desea construir del XML.
- **parentElem**: Es una instancia de la clase jdom::Element que representa el nodo padre de la rama que se está construyendo.
- **msg**: Es una instancia de LaboratoryMessage que representa el mensaje obtenido desde el laboratorio, que contiene los datos de los resultados de análisis clínicos.
- **resultObj**: Es una instancia de la clase AnalysisResult a la cual se le agregarán los campos como patientId, physicianId, etc y al terminar de construir el CDA se grabará en esta instancia.

1.5.1.4 LaboratoryParsersFactory

El sistema está diseñado para permitir la inclusión de distintos parsers que se encarguen de convertir los mensajes obtenidos desde distintos laboratorios. Para desacoplar el conocimiento de los distintos parsers existentes en el sistema, se agrega la clase LaboratoryParsersFactory. La responsabilidad de esta clase es conocer los distintos parsers existentes en el sistema y ofrecerlos como instancias que implementan la interfaz IlaboratoryParser.

1.5.1.5 CDACreator

Esta clase tiene la responsabilidad de conocer los contenidos que se deben generar para crear el documento CDA e instanciar la clase jdom donde se cargarán estos contenidos. El método que define es el pack que se encarga de generar un CDA y crear una instancia de AnalysisResult que será devuelto al método que lo invoca. Los parámetros de entrada que define el método pack se detallan a continuación:

- **msg:** Instancia de LaboratoryMessage que contiene el mensaje obtenido de un laboratorio.
- **parser:** Instancia que implementa la interfaz IlaboratoryParser a la cual se le solicitará que genere el contenido del CDA.

1.5.1.6 ASTM_CDAParser

Esta clase define el parser que se encarga de crear el contenido del CDA a partir del mensaje ASTM definido por el software Modulab Win. La clase implementa la interfaz IlaboratoryParser y es creada por una instancia de LaboratoryParsersFactory. En el método createContent que implementa toma la instancia de LaboratoryMessage pasada y realiza un casting a la clase ASTM para poder acceder a los contenidos del mensaje recibido desde el laboratorio con este formato.

1.5.2 Mapeo de Campos

A continuación se especifica la estructura que debe tener el documento CDA para almacenar los resultados de los análisis clínicos en formato ASTM, y su correspondiente mapeo. Existen campos para los cuales no está especificado ningún mapeo, esto se debe a que sólo están relacionados con el formato estándar de un documento CDA.

Campo	*Descripción*	*Mapeo con ASTM*
ClinicalDocument	Elemento raíz del CDA	
typeId	root/extension	CDAConfigHandler.ResultCodeSystem
classCode	Default: DOCLIN	

Campo	*Descripción*	*Mapeo con ASTM*
moodCode	Intención del documento Default: EVN	
Id	Identificador único del CDA	
Code	Tipo de doc. Ej "Resultado del Laboratorio"	CDAConfigHandler.CDAType
Title	Título del CDA	CDAConfigHandler.ResultsTitle
effectiveTime	Fecha de creación del documento original del acto clínico asociado al CDA (yyyyMMddHHmmss)	MessageDateAndTime
confidentialityCode	Normal, alta, muy alta	
Code	N,V,R	
Code System	Sistema de codificación (LOINC)	CDAConfigHandler.ConfidentialityCode
recordTarget	Agrupa información del paciente	
typeCode	Default: RCT	
contextControlCode	Default: OP	
patientRole	Rol del paciente	
classCode	Default: PAT	
Id	Identificación del paciente Ej Id de la historia clínica	PracticeAssignedPatientID
Patient	Agrupa información del paciente	
classCode	Default: PSN	
determinerCode	Default: INSTANCE	
Id(*DEPRECATED*)	Identificación del paciente Ej CI.	LaboratoryAssignedPatientID
Name	Nombre del paciente	
Given	Nombres	PatientName
Family	Apellidos	PatientName
administrativeGenderCode	Género	PatientSex
birthTime	YYYYMMDD	BirthDate
providerOrganization	Agrupa información del laboratorio	
classCode	Default: ORG	
determinerCode	Default: INSTANCE	
Id(root, extension)	OID	
Author	Agrupa información del autor del CDA (Persona o dispositivo)	
TypeCode	Default: AUT	
contextControlCode	Default: OP	
Time	Hora en la que se produjo el resultado	MessageDateAndTime
assignedAuthor	Información del autor	

Campo	*Descripción*	*Mapeo con ASTM*
classCode	Default: ASSIGNED	
Id(root, extension)	Identificación del autor. El laboratorio debe tener su catálogo de máquinas de análisis con OID	
Custodian	Agrupa la información de la entidad encargada de almacenar y administrar el CDA.	
typeCode	Default: CST	
assignedCustodian	Entidad encargada de almacenar y administrar el CDA.	
classCode	Default: ASSIGNED	
representedCustodianOrganization	Entidad encargada de almacenar y administrar el CDA.	
classCode	Default: ORG	
determinerCode	Default: INSTANCE	
Id(root, extension)	Id de la entidad	
Participant	Agrupa información del médico que solicitó la orden.	
typeCode	REF(Referrer, médico que remitió al laboratorio para que le practiquen el examen).	
contextControlCode	Default: OP	
associatedEntity	Médico que solicitó la orden.	AttendingPhysicianID
classCode	Código del médico (OID).	
associatedPerson	Corresponde al médico que pidió la orden, es opcional porque se supone que si un paciente es ambulatorio y se hace un examen, no necesariamente la orden fue dada por un médico. Poner los campos de la estructura según se defina.	AttendingPhysicianID
Component	Agrupa la información de uno de los exámenes de la orden.	
typeCode	Default: COMP	
contextConductionInd	Default: trae	
bodyChoice	Encabezado de la sección de resultados.	
structuredBody	Inicio del cuerpo.	
classCode	Default: DOCBODY	
moodCode	Default: EVN	
Component		
typeCode	Default: COMP	
contextConductionInd	Default: trae	

Campo	*Descripción*	*Mapeo con ASTM*
Section	Agrupa la información de los resultados de un examen y el área clínica en la que se realiza el mismo.	
classCode	Default: DOCSECT	
moodCode	Default: EVN	
Title	Nombre del área clínica.	
typeCode	Default: COMP	
contextConductionInd	Default: trae	
Section	Agrupa información del examen o procedimiento.	
classCode	Default: DOCSECT	
moodCode	Default: EVN	
entry	Dentro va cada examen	
typeCode	Default: COMP	
contextConductionInd	Default: trae	
clinicalStatement	Resultado del análisis.	
Observation	Resultado del examen.	
classCode	Default: OBS	
moodCode	Default: EVN	
Code	Código y nombre de la variable basado en un sistema de codificación (LOINC).	UniversalTestID
Value	Valor del resultado. Tipo PQ incluyendo unidad de medida o texto.	Data Measurement/Units
Text	Comentario del resultado	CommentText

1.5.3 Archivos de configuración

Al comienzo de la ejecución la aplicación implementada lee algunos parámetros de configuración de un archivo XML. La descripción de cada una de las secciones que contiene dicho XML se presenta a continuación.

- **CDAProperties:** Esta sección contiene un conjunto de propiedades que permiten construir el CDA. Estas propiedades corresponden a OID's y constantes que se utilizan para campos generales como el sexo de un paciente, el título que llevará el resultado en el CDA, etc.
- Laboratories: Esta sección contiene un conjunto de elementos del tipo Laboratory. Estos elementos contienen la información de cada uno de los laboratorios con los cuales se comunica la aplicación. La información más relevantes es la forma de conexión al laboratorio, en la cual se especifica la dirección IP de la máquina, el método utilizado para la autenticación, etc.

```
<?xml version="1.0" encoding="UTF-8" ?>
<ASSELaboratoryConfig>
	<CDAProperties>
		<property name = "ResultCodeSystem" value = "2.16.840.1.113883.2.10.1.1.3" />
		<property name = "MaleGender" value = "M|2.16.840.1.113883.5.1" />
		<property name = "FemaleGender" value = "F|2.16.840.1.113883.5.2" />
		<property name = "ConfidentialityCode" value = "26436-6|2.16.840.1.113883.5.25" />
		<property name = "CDAType" value = "F|2.16.840.1.113883.6.1" />
		<property name = "NameSeparator" value = " " />
		<property name = "ResultsTitle" value = "Results from MODULAB" />
	</CDAProperties>
	<Laboratories>
		<Laboratory name= "Modulab-Test" code = "001">
			<sshProperties>
				<host>192.168.1.101</host>
				<port>22</port>
				<logFile></logFile>
				<knowHostsFile>known_hosts</knowHostsFile>
				<user>pepe</user>
				<!--<Authentication method="RSA">
					<rsaKeyFile>id_rsa.pub</rsaKeyFile>
				</Authentication> -->
				<Authentication method="PASSWORD">
					<password>pepe123</password>
				</Authentication>
			</sshProperties>
		</Laboratory>
	</Laboratories>
</ASSELaboratoryConfig>
```

Tabla 2: Archivo de configuración para prototipo.

1.6 Ejecución del prototipo

En esta sección se muestra una ejecución típica del prototipo implementado. Se muestra el flujo seguido en la interacción con el software instalado en un laboratorio de referencia. Para guiar al lector se agregan capturas de pantalla en cada paso.

1.6.1 Agendar un análisis clínico

Para agendar un análisis, la aplicación debe conectarse al servidor ssh del laboratorio donde se desea realizar el estudio. Una vez realizada la conexión, se debe enviar un archivo ASTM con la lista de análisis requeridos. El mensaje ASTM se agrega a un archivo de la máquina host (laboratorio) el cual es leído por Modulab. Cada solicitud de análisis se agrega a una base de datos de citaciones.

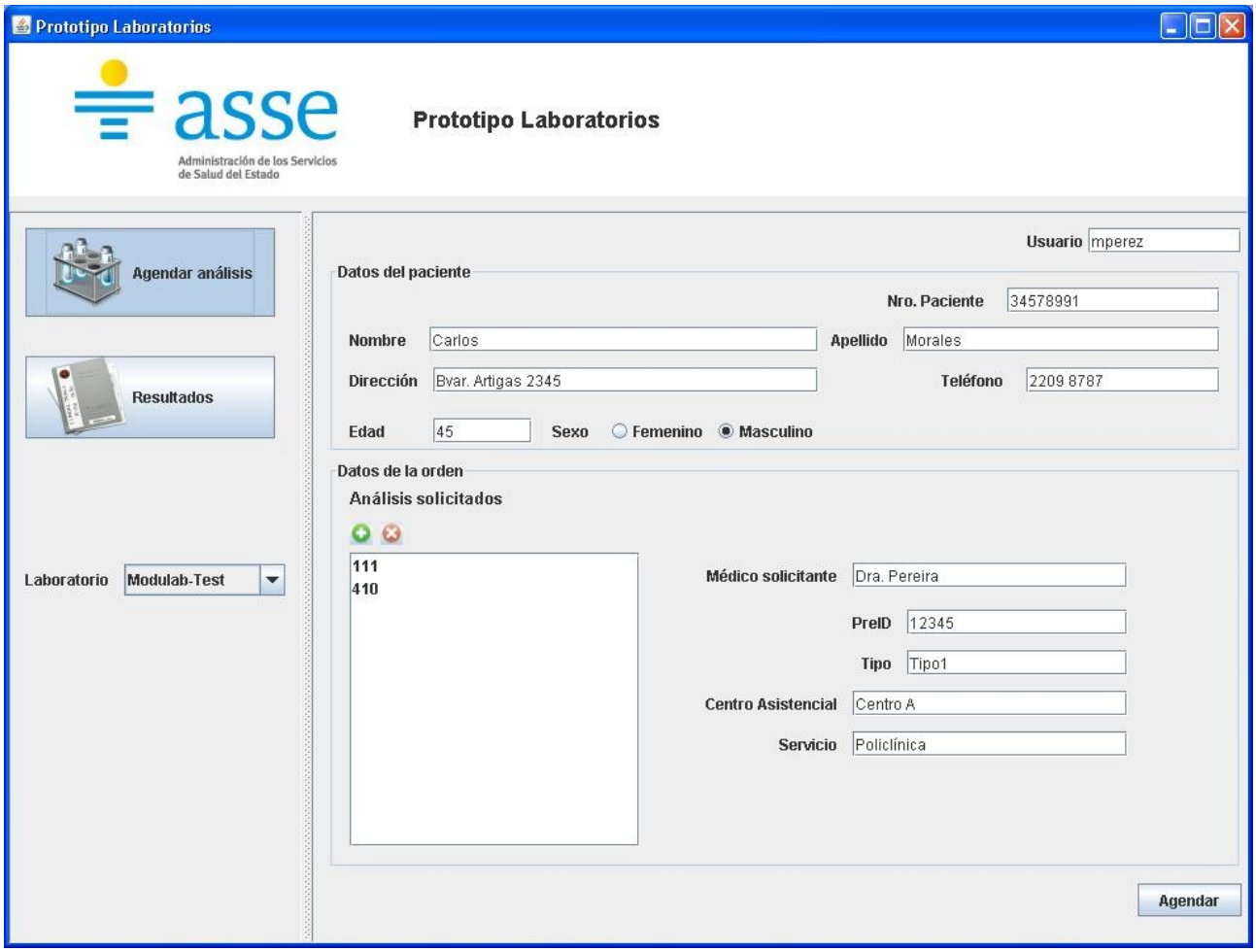

Figura 2: Agenda de análisis clínicos.

Cuando el paciente se presenta en el laboratorio, se recupera su orden de análisis de la base de citaciones y se le hacen las pruebas solicitadas. A medida que se obtienen los resultados de las muestras, un técnico valida estos resultados e inmediatamente se crea un archivo de resultados. Este comportamiento puede generar varias versiones intermedias de resultados. Por ejemplo, supongamos que llega un paciente y se realiza las pruebas A, B y C . Los resultados de A y C están disponibles al siguiente día donde un técnico los valida. En ese momento se crea un mensaje ASTM con los resultados de A y C. Al siguiente día, el técnico valida los resultados de la prueba B y nuevamente se genera un mensaje ASTM con los resultados de A, B y C. Entonces, se generaron dos versiones para tener los resultados de todos los tests solicitados.

1.6.2 Generación de resultados en Modulab

Una vez que Modulab recibe la petición para realizar un análisis clínico, la almacena en una base de datos de Peticiones. Cuando el paciente se presenta en el laboratorio, se recupera la orden de esta base de datos y se le realizan los estudios correspondientes. A medida que cada resultado está pronto, un técnico del laboratorio lo valida y en este momento se genera un mensaje ASTM con los resultados obtenidos hasta el momento.

Cabe recordar que una orden de análisis puede contener muchos tests simples para realizar, por lo tanto, a medida que el técnico valida los resultados de cada test, se va generando un mensaje ASTM de resultados que contiene el total de resultados obtenidos. Al momento de validar el último test se genera un archivo ASTM que cuenta con el total de resultados.

Debido a este comportamiento incremental de la información obtenida, la aplicación va generando CDAs con los resultados intermedios. Estos documentos contienen un número de versión. Al presentar los resultados en la pantalla se toma la última versión pues es la más actualizada y contiene la información completa de los resultados obtenidos hasta el momento.

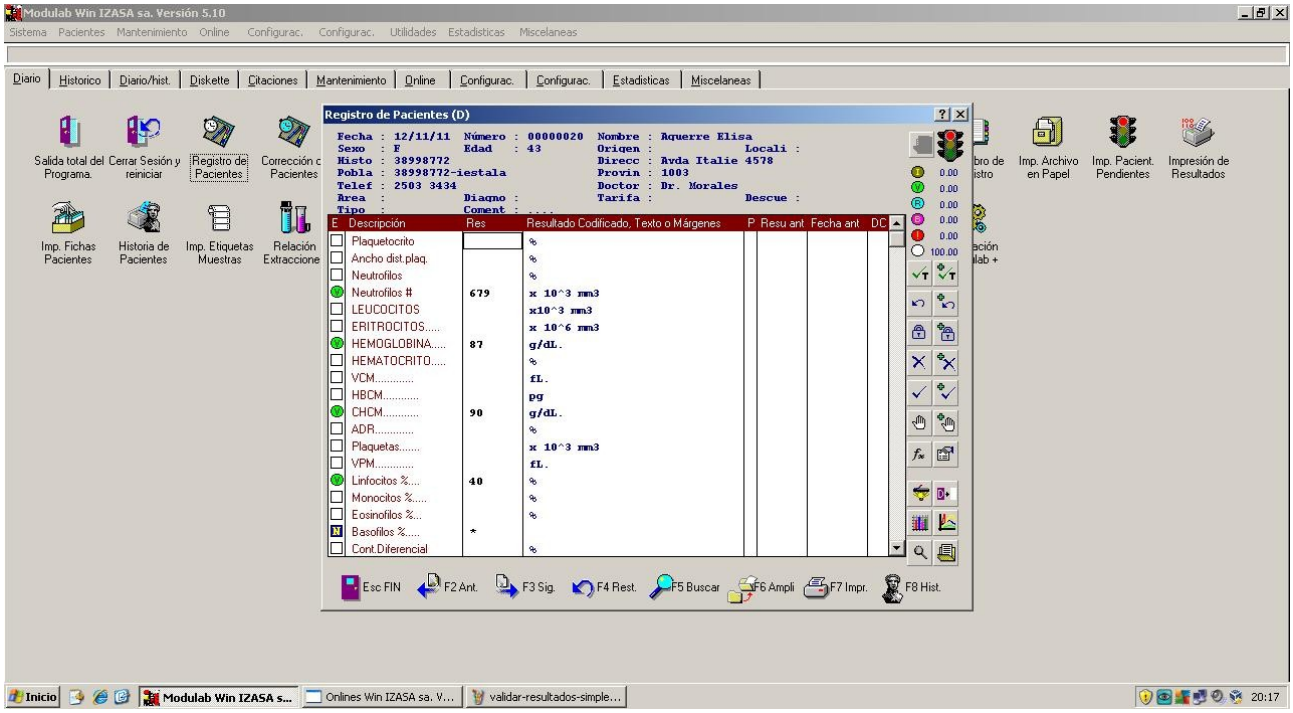

Figura 3: Validación de resultados en Modulab.

1.6.3 Vista de Resultados

La aplicación levanta un demonio (proceso que se ejecuta en segundo plano) para ir actualizando los nuevos resultados de análisis. Para ello, se conecta al servidor ssh para luego acceder al archivo donde Modulab va guardando los archivos ASTM con los nuevos resultados. Renombra este archivo, toma cada archivo ASTM y lo convierte a un documento CDA que luego es persistido en la base de datos. Cuando Modulab va a guardar un nuevo resultado crea un nuevo archivo ya que el anterior fue renombrado.

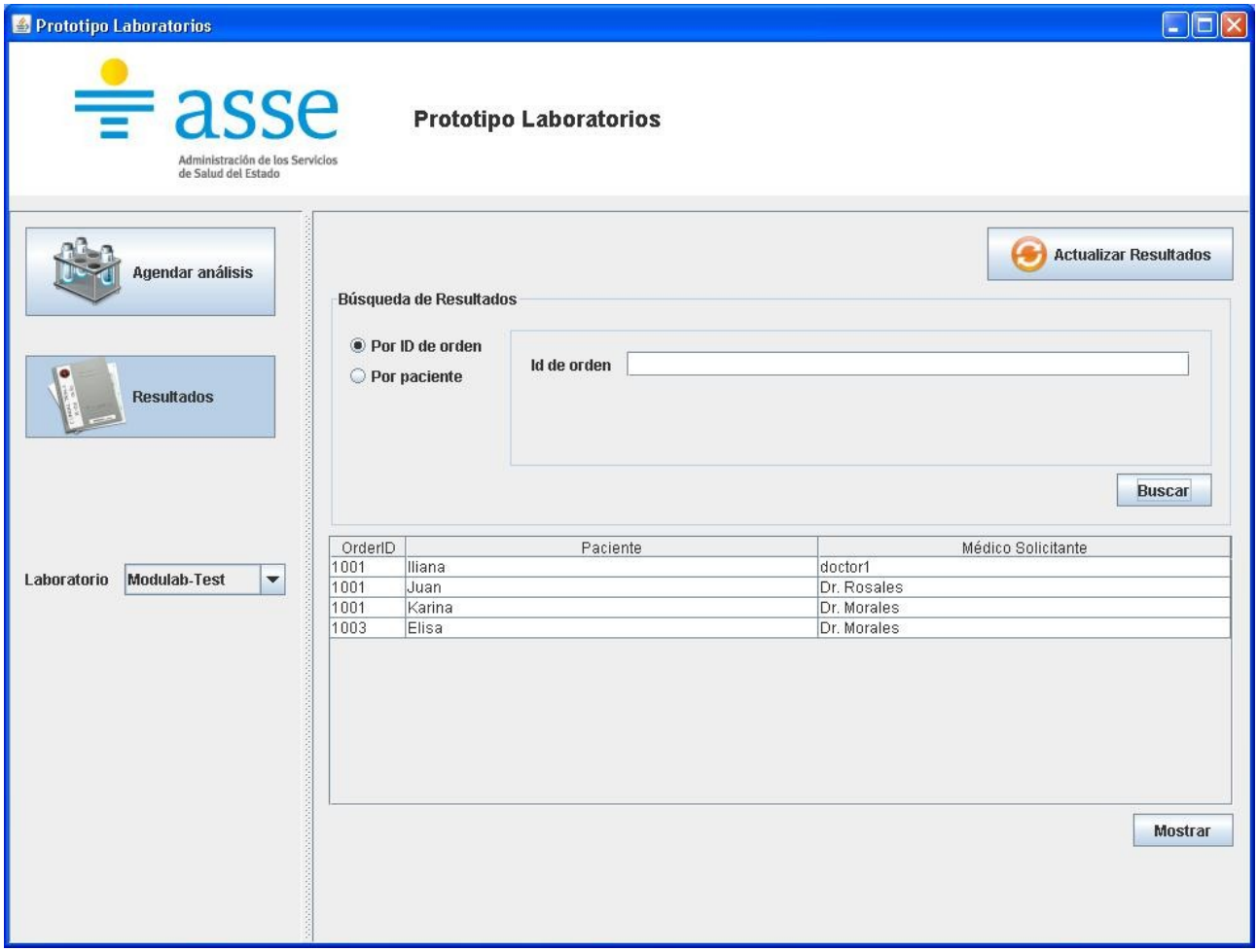

Figura 4: Resultados obtenidos.

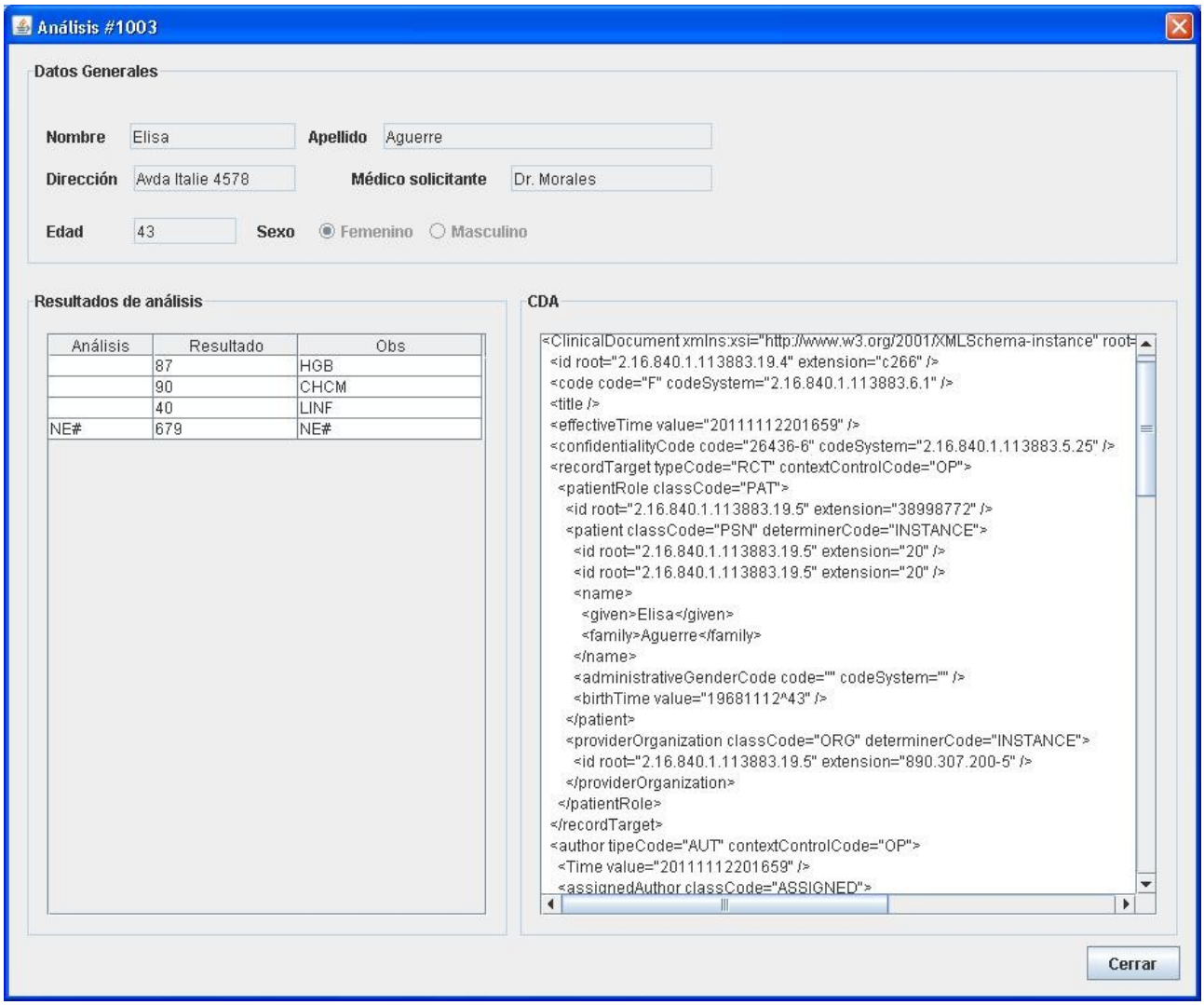

Figura 5: Detalle de resultados mostrando CDA.

1.7 Pruebas realizadas

El CDA construido cumple con la especificación del esquema xsd pero semánticamente presenta carencias como resultado del ingreso de muchos valores ficticios relativos a campos relacionados con OID's (Object Identifier), y códigos LOINC (Logical Observation Identifiers Names and Codes).

A continuación se muestra a modo de ejemplo un mensaje ASTM resultado de un análisis clínico enviado por Modulab, y el documento CDA producido en base al mismo:

Mensaje ASTM:

H|\^&|||ModuWin|||||Host LIS||P|1.0|20111107021054
P|1|12345678|18||PÚrez Juan||19631107^48|M||Avda. Flores 3452^12345678-mgonzalez^1001||2209 8765|Dr. Rosales|....|^|||||||....|20111107020939|R|||||||^^|||
O|1|18||^COLESTEROLEMIA^111^COL^||20111107020939|20111107020939||||A||||||||||||O|||
R|1|^COLESTEROLEMIA^111^COL^^^^B|70|mg/dL||||^V||^000|||
L|1|

Tabla 3: Mensaje ASTM de ejemplo.

CDA producido:

```
<?xml version="1.0" encoding="UTF-8"?>
<ClinicalDocument xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" root="2.16.840.1.113883.1.3"
extension="POCD_HD000040">
  <id root="2.16.840.1.113883.19.4" extension="c266" />
  <code code="F" codeSystem="2.16.840.1.113883.6.1" />
  <effectiveTime value="201111 7 21054" />
  <confidentialityCode code="26436-6" codeSystem="2.16.840.1.113883.5.25" />
  <recordTarget typeCode="RCT" contextControlCode="OP">
    <patientRole classCode="PAT">
      <id root="2.16.840.1.113883.19.5" extension="12345678" />
      <patient classCode="PSN" determinerCode="INSTANCE">
        <id root="2.16.840.1.113883.19.5" extension="18" />
        <id root="2.16.840.1.113883.19.5" extension="18" />
        <name>
          <given>Juan</given>
          <family>PÚrez</family>
        </name>
        <administrativeGenderCode code="" codeSystem="" />
        <birthTime value="19631107^48" />
      </patient>
      <providerOrganization classCode="ORG" determinerCode="INSTANCE">
        <id root="2.16.840.1.113883.19.5" extension="890.307.200-5" />
      </providerOrganization>
    </patientRole>
  </recordTarget>
  <author tipeCode="AUT" contextControlCode="OP">
    <Time value="201111 7 21054" />
    <assignedAuthor classCode="ASSIGNED">
      <id root="2.16.840.1.113883.19.5" extension="KP00017" />
    </assignedAuthor>
  </author>
  <custodian typeCode="CTS">
    <assignedCustodian classCode="ASSIGNED">
      <representedCustodianOrganization classCode="ORG" determinerCode="INSTANCE">
        <id root="2.16.840.1.113883.19.5" extension="890.307.200-5" />
      </representedCustodianOrganization>
    </assignedCustodian>
  </custodian>
  <participant typeCode="REF" contextControlCode="OP">
    <associatedEntity classCode="ASSIGNED">
      <id root="2.16.840.1.113883.19.5" extension="Dr. Rosales" />
      <associatedPerson>
        <name>
          <given>Dr. Rosales</given>
          <family>Dr. Rosales</family>
        </name>
      </associatedPerson>
    </associatedEntity>
  </participant>
  <component typeCode="COMP" contextConductionInd="trae">
    <structuredBody classCode="DOCBODY" moodCode="EVN">
      <component>
        <section classCode="DOCSECT" moodCode="EVN">
          <title>Results from MODULAB</title>
          <component>
            <section classCode="DOCSECT" moodCode="EVN">
              <entry typeCode="COMP" contextConductionInd="trae">
                <observation classCode="OBS" moodCode="EVN">
                  <code code="111" codeSystem="2.16.840.1.113883.2.10.1.1.3" displayName="COL" />
                  <value unit="mg/dL" value="70" xsi:type="PQ" />
                </observation>
              </entry>
            </section>
          </component>
        </section>
      </component>
    </structuredBody>
  </component>
</ClinicalDocument>
```

Tabla 4: CDA resultante del ejemplo.

1.8 Consideraciones realizadas por el proveedor de Modulab

1.8.1 Códigos Host

El campo UniversalTestID del registro O (Test Order Record) tiene el formato **Id^Name^Type^LocalCode** donde el sub-campo Type corresponde al código del análisis dado por el Host, que es la máquina con la que se comunica el software. Para el caso de estudio, el host corresponde a la máquina que se comunica con Modulab desde ASSE. Este código es aquel que le da el host al análisis detallado, cada máquina del laboratorio tiene la capacidad de configurar un mapeo entre los códigos de Modulab y los códigos del host. Cada vez que se envía un resultado, el sub-campo Type contendrá el valor de este mapeo. Para el host, que es ASSE, este código debería corresponderse con el código LOINC del análisis. De esta manera, al obtener un resultado se podrá colocar en el cda generado el código estandarizado del análisis de forma de no manejar códigos locales sino manejar un código reconocido internacionalmente, el cual permite compartir información que sea comprensible por cualquier software externo.

El mapeo realizado por el prototipo considera que este código de host será el código LOINC del análisis, por lo que graba en el cda el sub-campo Type contenido en el campo UniversalTestID del mensaje ASTM de Modulab.

1.8.2 Número de historia clínica

Para Modulab el valor del campo PracticeAssignedPatientID que contiene el registro P (Patient Identifying Record) es único y corresponde al identificador de historia clínica del paciente. Con este identificador el software es capaz de generar reportes con la información, por ejemplo, de los análisis realizados por un paciente en el laboratorio o con la información de cuántas muestras mostraron valores fueras de rango para un determinado análisis.

ASSE no cuenta con un módulo maestro de identificación (PKI) definido para identificar a sus pacientes, por lo tanto, el prototipo envía la cédula de identidad como identificador único de la historia clínica.

1.9 Trabajo futuro

El punto más interesante para una futura discusión es el que concierne a la forma de firmar digitalmente los resultados almacenados en formato CDA. La Firma digital es un código informático que permite determinar la autenticidad de un documento electrónico y su integridad.

Ya que tenemos dos softwares interactuando para generar el CDA, es de suponer que la firma digital debería corresponder al software Modulab. Ante la pregunta de si tiene validez esta firma, podemos ver que únicamente valida que los datos fueron manipulados por Modulab y no por algún proceso extraño pero no nos ofrece garantías respecto a la validez de los datos ingresados. En caso de error en el valor de una prueba ¿quién es responsable? la persona encargada de validar los resultados en el laboratorio siempre puede argumentar, por ejemplo, que no fue el valor que ingresó, que el software tuvo alguna falla y cambió la información e incluso puede decir que no fue él quién ingresó la información errónea.

Para validar este tipo de información es importante que el software de laboratorio cuente con la firma digital de cada uno de los resultados, esta firma debe pertenecer a la persona que valida que los resultados se han ingresado correctamente. De esta forma, nos aseguramos que los resultados manejados por el software que interactúa con nosotros, están avalados por una persona.

Luego, para validar la comunicación entre los dos softwares, tendríamos que contar con un mecanismo que nos permita validar que el dato recogido y la firma digital que acompaña su validación no fueron modificados en ninguna parte del ciclo de interacción entre los procesos. Se puede pensar que este mecanismo podría ser un cifrado del par (valor, firma) con un algoritmo y claves acordadas entre los procesos informáticos. De esta forma el destinatario del documento electrónico tendrá la certeza de que el documento es auténtico e íntegro.

Otro punto interesante a destacar es el concerniente a la persistencia de datos clínicos. Los documentos CDA están estructurados en un XML, por lo tanto al grabarse en la base se está guardando una gran cantidad de información en un campo de texto. Para buscar el resultado de un análisis clínico para un paciente dado en una fecha determinada, se debe buscar dentro del XML la información necesaria para ubicar el registro y luego buscar la información de valor para visualizarla o imprimirla. Este proceso es muy costoso e inviable, por lo cual se presentan dos posibles soluciones.

Figura 6: CDR y esquema relacional con información redundante.

La primera es contar con una base de datos relacional con la información más relevante de cada resultado obtenido además del cda correspondiente. De manera que al crear el cda, se persiste el documento en su formato XML y también se graban determinados campos (identificador del paciente, fecha, resultados obtenidos, etc) en otra tabla o base de datos. El principal problema que presenta esta solución es el mecanismo para garantizar que la información contenida en el cda es exactamente igual al conjunto de campos almacenados en el repositorio relacional. La redundancia de datos no controlada puede llevar a inconsistencias y a generar un conflicto con el documento cda, que a nivel de interoperabilidad debería ser el documento compartido por los diferentes procesos.

Figura 7: CDR como único repositorio de datos clínicos.

La segunda solución apuesta a utilizar las bases de datos XML que han surgido en el mercado. Estas bases de datos cuentan con un lenguaje de consulta especialmente diseñado para obtener información a partir de un XML sin pérdida de performance.

El prototipo realizado para el módulo de RCE hace uso de la primera solución, duplicando la información con los campos de consulta más importantes para grabarlos en forma separada del documento cda almacenado.

Bibliografía

[1] Página oficial de W3C : http://www.w3.org/DOM/ - Último acceso <04/04/12>

[2] Página oficial de HL7 : www.hl7.org - Último acceso <04/04/12>

[3] Agesic-Definición OID : http://www.agesic.gub.uy/innovaportal/file/236/1/oidenuruguay.pdf - Último acceso <04/04/12>

[4] SSH : http://tools.ietf.org/html/rfc4252 - Último acceso <04/04/12>

[5] "Wikipedia XML- JDOM " , mayo- 2011

APÉNDICE B – Arquitecturas en Sistemas de Salud

En este apéndice se muestran arquitecturas desarrolladas por actores del sistema de salud.

1.1 IHE

Integrating the Healthcare Enterprise (IHE) es una iniciativa de profesionales de la salud y empresas proveedoras cuyo objetivo es mejorar la comunicación entre los sistemas de información que se utilizan en la atención al paciente. IHE define “Perfiles de Integración” que utilizan estándares ya existentes para la integración de sistemas de manera de lograr una interoperabilidad efectiva y un flujo de trabajo eficiente[1].

Cada Perfil de Integración IHE describe una necesidad clínica de integración de sistemas y una posible solución a la misma. Define también los componentes funcionales, a los que llamaremos Actores IHE, y especifica con el mayor grado de detalle posible, las transacciones que cada Actor deberá llevar a cabo. Cada una de estas transacciones están siempre basadas en estándares como Digital Imaging and Communication in Medicine (DICOM) y Health Level 7 (HL7).

1.1.1 PIX

Uno de los perfiles de integración centrales de la infraestructura IHE, y que presenta el mayor interés para nuestro trabajo, es el PIX (Patient Identifier Cross-Referencing) . El perfil PIX define los actores y transacciones necesarios para mantener un registro maestro de identificadores de pacientes y proporcionar esta información a otras aplicaciones . Las transacciones realizadas por los actores se definen en base a mensajes HL7.

El registro maestro de identificación de pacientes es uno de los módulos centrales dentro de la arquitectura para sistemas de salud presentada en este documento. Este módulo asegura que cada paciente contará con una identificación única que será usada por todos los componentes de software lo que asegura el correcto cruzamiento de la información entre ellos.

El perfil de integración PIX soporta referencias cruzadas de identificadores de pacientes desde distintos dominios, lo cual permite la interoperabilidad entre distintos procesos. El flujo de proceso para este perfil se muestra en la Figura 1.

Un Dominio Identificador de Pacientes (Patient Identifier Domain) es definido como una única aplicación o como un conjunto de aplicaciones interconectadas que comparten un esquema de identificación común y una autoridad expedidora de identificadores de pacientes. El esquema de identificación común está compuesto por un identificador y un proceso de asignación de cada identificador a un paciente.

El dominio de PIX incorpora las siguiente hipótesis acerca del acuerdo que deben cumplir los Dominios de Indentificadores de pacientes:

- Se ha acordado un conjunto de políticas que describe como se hará la referencia cruzada de la identificación de cada paciente entre los dominios participantes.
- Se ha acordado un conjunto de procesos para administrar las políticas citadas en el ítem anterior.
- Se ha acordado una autoridad administradora para manejar los citados procesos de administración y las políticas.

La principal característica del perfil PIX es que impone un conjunto mínimo de restricciones a los dominios participantes y centraliza la mayor parte de las restricciones en el actor identificado como Administrador del Dominio PIX (PIX Domain Manager).

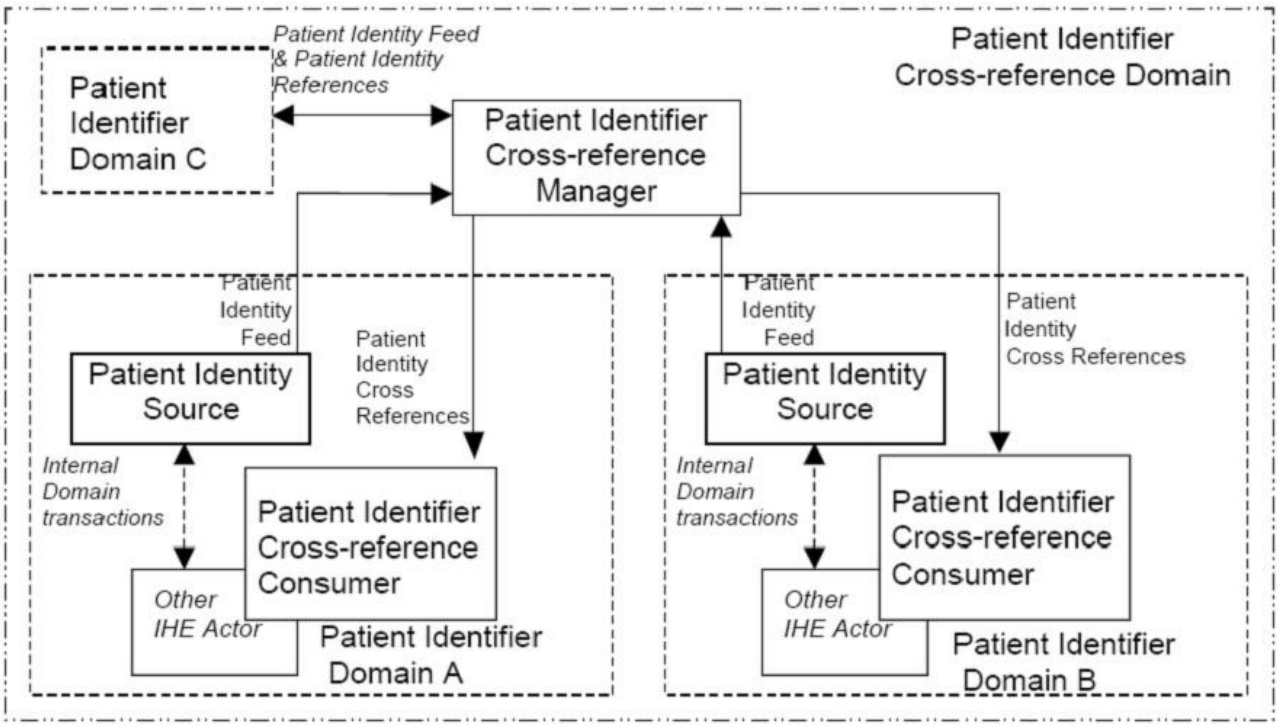

Figura 1: Flujo de proceso para el perfil de integración PIX.

1.1.2 Ejemplo de aplicación del perfil PIX

IHE cuenta con un proyecto llamado OpenPIXPDQ que permite obtener datos patronímicos de un paciente de forma segura. El proyecto utiliza los perfiles de integración PDQ y ATNA, además del perfil PIX. El perfil PDQ permite consultar los datos patronímicos de un paciente y el perfil ATNA garantiza la seguridad entre las transacciones involucradas.

1.1.2.1 PDQ

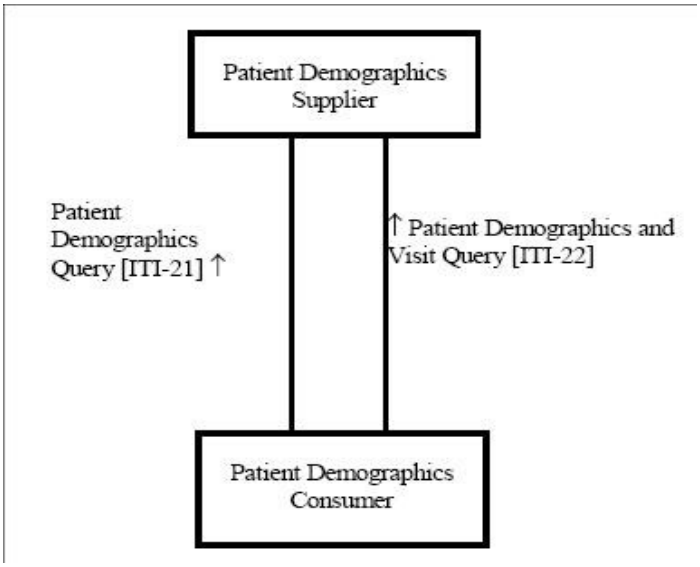

Figura 2: Interacción entre actores del perfil PDQ

Este perfil de integración defido por la IHE cuenta con dos actores: un servidor central (Provider) que proveé los datos patronímicos de un paciente dado y un cliente que consume el servicio ofrecido por el servidor (Consumer). La interacción entre ambos actores se muestra en la Figura 2.

1.1.2.2 ATNA

El perfil de integración ATNA (Audit Trail and Node Authentication) establece medidas de seguridad que, junto con los procedimientos y políticas de seguridad, proveen la confidencialidad de la información del paciente, la integridad de los datos y el seguimiento de la cuentas del usuarios. Este perfil contribuye al control de acceso ya que únicamente permite la comunicación entre nodos de un mismo dominio y además cada nodo limita el acceso de los usuarios en base a las reglas de autenticación local y las políticas de control de acceso definidas. Los requerimientos del perfil a los nodos que forman parte de la red segura se detallan a continuación:

- **Autenticación de usuarios:** el perfil requiere que cada nodo realice la autenticación local del usuario pero deja libre la elección de la tecnología utilizada en cada uno de ellos. Cada nodo podría tener un mecanismo distinto de autenticación.
- **Autenticación de la conexión:** se requiere el uso de certificados en forma bidireccional para autenticar la conexión desde y hacia cada nodo seguro.
- **Trazas de auditoría:** el seguimiento de la actividad de cada usuario se realiza a través del Audit Trail que debe permitir realizar las actividades de auditoría interna o externa para evaluar el cumplimiento de las políticas de seguridad de los dominios seguros, para detectar anomalías en el funcionamiento de los procesos y facilitar la detección temprana de la creación, modificación o eliminación no permitida de la información de salud protegida (PHI).

1.1.2.3 OpenPIXPDQ

Este proyecto reúne los perfiles anteriormente vistos para obtener los datos patronímicos de un paciente de forma segura. Para ello define cuatro transacciones, las que realiza con transacciones de ATNA para garantizar la seguridad en los procesos. A continuación se presentan las cuatro transacciones definidas por el proyecto.

- **PIX Feed:** Esta transacción comunica la información del paciente, incluyendo su modificación y actualización.
- **PIX Query:** Esta transacción involucra una petición de un consumidor PIX de una lista de identificadores de pacientes.
- **Notificación de Actualización de PIX:** Esta transacción involucra al Administrador PIX que proveé una notificación cuando se actualizan asociaciones de Referencias cruzadas de identificadores de pacientes, a los consumidores PIX registrados para recibir este evento.
- **PDQ Query:** Esta transacción involucra a un consumidor PDQ que solicita información acerca de pacientes cuyos datos patronímicos coinciden con los enviados en el mensaje de consulta.

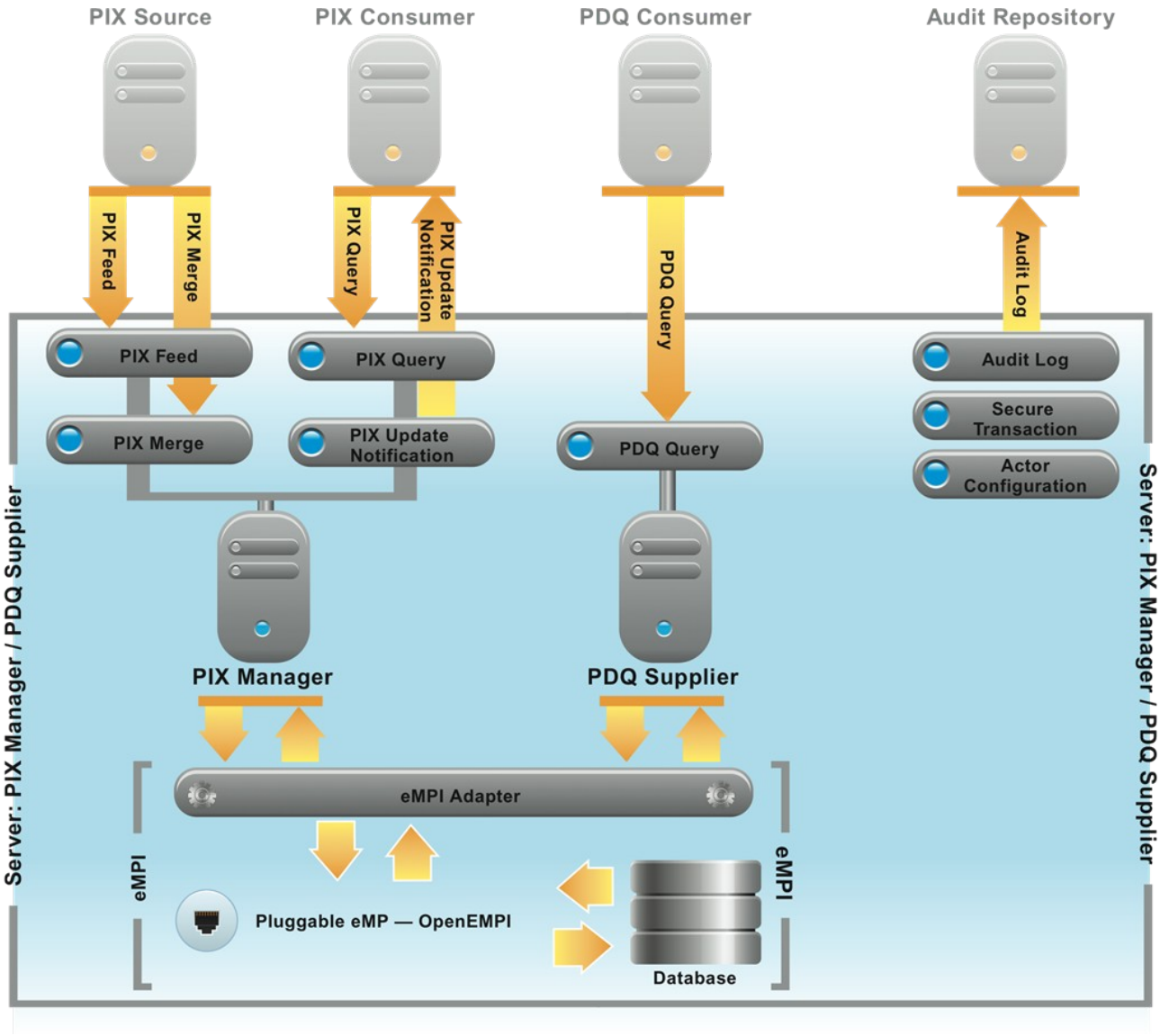

Figura 3: Arquitectura OpenPIXPDQ

1.2 OpenEHR

Es un estándar abierto que intenta proporcionarle a las TIC (tecnologías de la información) un soporte eficiente en cuanto a la administración y almacenamiento de información sanitaria. El principal desafío es el de representar la semántica del sector [2].

Para lograr esto se requiere de un marco de trabajo orientado en la gestión del conocimiento, el cual incluye ontologías, terminología y semántica para una plataforma de salud computacional en salud mantenible, adaptable y centrado en los registros electrónicos de salud del paciente.

Se trata de crear modelos clínicos de contenido y procesos reutilizables de alta calidad conocidos como arquetipos, junto con interfaces formales de terminología. La Junta de Revisión Clínica de OpenEHR ha propuesto 3 grandes áreas de desarrollo:

1. Un conjunto internacional de los arquetipos que proporcionan un modelo estándar compartido de datos clínicos importantes y requerimientos estándar para la terminología.

2. Un proceso acordado y formal para la revisión, publicación y mantenimiento continuo de estos arquetipos

3. Un portal web para habilitar este proceso y el uso efectivo de estos arquetipos en aplicaciones clínicas y programas nacionales de salud en línea.

Bibliografía

[1] Página oficial de Integrating the Healthcare Enterprise (IHE) : http://www.ihe.net/ - Último acceso <04/04/12>

[2] OpenEHR : 9.2 http://www.openehr.org/home.html - Último acceso <04/04/12>

APÉNDICE C – Prototipo Autenticación centralizada con LDAP

Se plantea implantar en ASSE, un controlador de dominio que permita autenticar a los usuarios de red en ASSE, utilizando un servicio de directorio LDAP el cual contenga de forma centralizada a los usuarios.

En ésta apéndice se hará una introducción en los conceptos necesarios para comprender el desarrollo del prototipo, se describirán los pasos seguidos para instalar el controlador de dominio con Samba y OpenLDAP en un sistema operativo openSUSE 11.2, se expondrá como realizar la replica de los datos del servidor LDAP, así como también se expondrán las pruebas realizadas.

Como resultado de la investigación del protocolo LDAP se creó el APÉNDICE LDAP, el cual detalla el protocolo.

1.1 Propósito

Actualmente ASSE cuenta con varios repositorios de usuarios, cada repositorio hace referencia a usuarios y contraseñas de un aplicativo en particular. De forma similar la administración de las cuentas de usuarios que acceden a la red de ASSE, se realiza de forma local en cada equipo. Estos dos puntos dieron el inicio de proponer la administración centralizada de usuarios utilizando el protocolo LDAP.

Al tener una autenticación local, provoca que un usuario deba tener una cuenta en cada ordenador al que quiera acceder, además de que un cambio en la personalización de la cuenta del usuario se debe de repetir manualmente en cada maquina en donde se haya creado el usuario, ésta forma de trabajo es poco escalable he impide manejar de forma centralizada y consistente la autenticación.

Las ventajas de un manejo centralizado de las cuentas de usuarios es que se tiene toda la información en un único lugar haciendo mas fácil su administración, un cambio en una cuenta se replica a todo el dominio de forma automática y a su vez es una solución mas escalable ya que permite crear y administrar redes de gran tamaño.

Los beneficios de implementar un controlador de dominio basado en Samba y OpenLDAP se podrían resumir en los siguientes:

- Tener un control del inicio de sesión (autenticación/autorización) a sistemas GNU/Linux y Windows con controles de acceso basado en cuentas de usuarios y grupos centralizados en el servidor controlador de dominio.
- Permite centralizar la administración de cuentas de usuario y grupos Unix/Linux, Windows y otras aplicaciones con soporte a LDAP.
- Centralizar la autenticación y los controles de acceso por usuarios y grupos para recursos compartidos en clientes o servidores GNU/Linux y/o Windows.
- Brinda la posibilidad de que otros servicios de red pueden realizar su autenticación/autorización de forma centralizada, algunos de estos servicios podrían ser los detallados abajo (este punto está fuera del alcance del prototipo):
  - Servidores de correo SMTP, POP3, IMAP usando aplicaciones como Postfix, Sendmail, Courier IMAP ó Dovecot.
  - Servidores Proxy HTTP como Squid.
  - Servidores Web como Apache.
  - Servidores DNS y DHCP como ISC bind e ISC dhcpd.
  - Servidores FTP como Pure-ftpd.
  - Servidores VPN como OpenVPN.

- Servidores de mensajería instantánea basados en Jabber como: jabberd ú Openfire.
- Aplicaciones Web como sistemas manejadores de contenido (CMS) con soporte de autenticación vía LDAP como: mediawiki, drupal, ezpublish, moodle entre otros.

1.2 Introducción

Cuando se habla de autenticación se refiere al proceso que se da inicio una vez que un usuario intenta iniciar sesión en un dominio o intenta tener acceso a los recursos de la red y se comprueban las credenciales del usuario consultando una autoridad determinada. Por lo general, ésto significa que un usuario tiene que proporcionar un nombre de usuario y una contraseña.

Existen hoy en día varios protocolos de autenticación disponibles para utilizar como se detalla en la Table 1 en la cual se expone la descripción de algunos de éstos protocolos.

Protocolo	*Descripción*
Kerberos	Es un protocolo de autenticación en una red de computadoras que permite a dos computadores en una red insegura demostrar su identidad mutuamente de manera segura. Sus diseñadores se concentraron primeramente en un modelo de cliente-servidor, y brinda autenticación mutua es decir tanto cliente como servidor verifican la identidad uno del otro. Los mensajes de autenticación están protegidos para evitar ataques de terceros. Kerberos se basa en criptografía de clave simétrica y requiere un tercero de confianza. Además, existen extensiones del protocolo para poder utilizar criptografía de clave asimétrica. Kerberos fue desarrollado por el Instituto Tecnológico de Massachusetts (MIT) [1]
SSL/TLS	Secure Sockets Layer (SSL en español capa de conexión segura) y su sucesor Transport Layer Security (TLS en españolseguridad de la capa de transporte) SSL proporciona autenticación y privacidad de la información entre extremos de la red comunmente Internet. Yse ejecuta entre la capa de aplicación y la capa de transporte. En SSL xisten básciamente 3 fases basicas: Fase 1- Negociar el algoritmo de comunicación entre las partes. Negocian el algoritmo criptográfico a utilizar. Puede ser criptografía de clave pública (como RSA, DSA o Fortezza), cifrado simétrico (como IDEA, AES, DES, RC2, RC4), o funciones de Hash (como MD5). Fase 2- Intercambiar claves públicas y autenticación basada en certificados digitales. Fase 3- Cifrado del tráfico basado en cifrado simétrico. [2]
NTLM	Las contraseñas están basadas en el conjunto de caracteres Unicode, se reconocen las minúsculas y las mayúsculas, la longitud aumenta hasta 128 caracteres. El sistema no almacena la contraseña sino que almacena una representación de la misma mediante el NTLM OWF. OWF se combina con el algoritmo MD4 Hash, que implementa una función de cifrado denominada digest (resumen)-un hash de 16 bytes- de una cadena de texto de longitud variable, que en este caso es la contraseña del usuario. NTLM es el proveedor predeterminado de autenticación e Windows NT, Windows 2000 y Windows Server 2003 mientras no sean miembros de un dominio de Active Directory.[3]
Autenticación implicita	Transmite credenciales por la red como un hash MD5 o autenticación implícita del mensaje. Se aplica a Windows 7, Windows Server 2008, Windows Server 2008 R2, Windows Vista. La autenticación de texto implícita utiliza un controlador de dominio de Windows para autenticar a los usuarios que soliciten acceso al contenido de su servidor web.[4]
Autenticación de Passport	Es un servicio de autenticación de usuarios que ofrece inicio de sesión único. Permite la autenticación centralizadao proporcionada por Microsoft que ofrece a los sitios Web suscritos servicios de perfil básico y un inicio de sesión único. Passport supone un beneficio para los usuarios porque no necesitan conectarse a nuevos recursos o sitios de acceso limitado. Para que un sitio sea compatible con la autorización y la autenticación mediante Passport, se debe utilizar este proveedor. Passport es un servicio de autenticación basado en cookies.[5]

Protocolo	*Descripción*
Ipsec	Protocolos que actúan en la capa de red, la capa 3 del modelo OSI. Una ventaja importante de IPsec frente a SSL y otros métodos que operan en capas superiores, es que para que una aplicación pueda usar IPsec no hay que hacer ningún cambio, mientras que para usar SSL y otros protocolos de niveles superiores, las aplicaciones tienen que modificar su código.IPsec es una parte obligatoria de IPv6, y su uso es opcional con IPv4.[6]
PAP	Protocolo simple de autenticación para autenticar un usuario contra un servidor de acceso remoto o contra un proveedor de servicios de internet. PAP es un subprotocolo usado por la autenticación del protocolo PPP (Point to Point Protocol), validando a un usuario que accede a ciertos recursos. PAP transmite contraseñas o passwords en ASCII sin cifrar, por lo que se considera inseguro. PAP se usa como último recurso cuando el servidor de acceso remoto no soporta un protocolo de autenticación más fuerte.
CHAP	Protocolo de autenticación por desafío mutuo (CHAP) es un método de autenticación muy utilizado en el que se envía una representación de la contraseña del usuario, no la propia contraseña. Con CHAP, el servidor de acceso remoto envía un desafío al cliente de acceso remoto. El cliente de acceso remoto utiliza un algoritmo hash (también denominado función hash) para calcular un resultado hash de Message Digest-5 (MD5) basado en el desafío y un resultado hash calculado con la contraseña del usuario. El cliente de acceso remoto envía el resultado hash MD5 al servidor de acceso remoto. El servidor de acceso remoto, que también tiene acceso al resultado hash de la contraseña del usuario, realiza el mismo cálculo con el algoritmo hash y compara el resultado con el que envió el cliente. Si los resultados coinciden, las credenciales del cliente de acceso remoto se consideran auténticas. El algoritmo hash proporciona cifrado unidireccional, lo que significa que es sencillo calcular el resultado hash para un bloque de datos, pero resulta matemáticamente imposible determinar el bloque de datos original a partir del resultado hash.[7]
RADIUS	RADIUS (Remote Authentication Dial-In User Server). Es un protocolo de autenticación y autorización para aplicaciones de acceso a la red o movilidad IP. Utiliza el puerto 1812 UDP para establecer sus conexiones.Comúnmente usado por el estándar de seguridad 802.1x [8]

Table 1: Protocolos de Autenticación

Actualmente ASSE maneja la administración de usuarios de forma local, a través de la autenticación brindada por el sistema operativo, por tanto se maneja según la administración de usuarios que brinda Unix/Linux y Windows.

Los usuarios en Unix/Linux [9]se identifican por un número único de usuario, User ID (UID). Y pertenecen a un grupo principal de usuario, identificado también por un número único de grupo, Group ID (GID). El usuario puede pertenecer a más grupos además del principal.

Se puede indicar de forma general que se identifican 3 tipos de usuarios:

- El usuario root con UID valor 0, es la cuenta con privilegios sobre todo el sistema.
- Los usuarios especiales, que son las cuentas que pueden heredar algunos permisos del root, ejemplos de éstas cuentas son mail, operator, sync.
- Los usuarios normales, los cuales cuentan de un directorio de trabajo, ubicado generalmente en /home y solo tienen privilegios completos sobre dicho directorio.

Existen cuatro archivos de configuración, que permiten la autenticación en Linux/Unix, estos son passwd, shadow, group y login.defs. A continuación se menciona cada archivo y su función.

En el archivo passwd, se ubican todas las cuentas de usuarios definidas en el archivo de en el sistema, ubicado en el directorio etc/passwd.. Este archivo es de texto tipo ASCII, se crea al momento de la instalación con el usuario root y las cuentas especiales, más las cuentas de usuarios normales que se hayan indicado al momento de la instalación. En la figura Figura 1 se muestra el formato del archivo de configuración.

```
root:x:0:0:root:/root:/bin/bash
sergio:x:501:500:Sergio González:/home/sergio:/bin/bash
```

Figura 1: Formato del archivo passwd

El significado de cada campo delimitado por ":" es el detallado en la Table 2

Campo	*Descripción*
Campo 1	el nombre del usuario, identificador de inicio de sesión (login). Tiene que ser único.
Campo 2	La 'x' indica la contraseña encriptada del usuario, además también indica que se está haciendo uso del archivo /etc/shadow, si no se hace uso de este archivo, este campo se vería algo así como: 'ghy675gjuXCc12r5gt78uuu6R'.
Campo 3	Número de identificación del usuario (UID). Tiene que ser único. 0 para root, generalmente las cuentas o usuarios especiales se numeran del 1 al 100 y las de usuario normal del 101 en delante, en las distribuciones mas recientes esta numeración comienza a partir del 500.
Campo 4	Numeración de identificación del grupo (GID). El que aparece es el número de grupo principal del usuario, pero puede pertenecer a otros, esto se configura en /etc/groups.
Campo 5	Comentarios o el nombre completo del usuario.
Campo 6	Directorio de trabajo (Home) donde se sitúa al usuario después del inicio de sesión.
Campo 7	Shell que va a utilizar el usuario de forma predeterminada.

Table 2: Campos del archivo passwd

El archivo shadow se puede considerar como una extensión del archivo passwd, pues contiene las contraseñas encriptadas de los usuarios definidos en dicho archivo y solo puede ser leído por el root. Además almacena otros campos de control. de las contraseñas. Éste archivo se encuentra ubicado en /etc/shadow y se definió para mejorar la seguridad, ya que originalmente las contraseñas encriptadas se almacenaban en el archivo passwd y éste archivo tiene definido acceso de lectura por todos los usuarios, facilitando ésto a que se puedan descifrar las contraseñas débiles.

Por cada usuario definido en el sistema se tiene una linea en el archivo shadow de la forma que se muestra en la Figura 2.

```
root:ghy675gjuXCc12r5gt78uuu6R:10568:0:99999:7:7:-1::
sergio:rfgf886DG778sDFFDRRu78asd:10568:0:-1:9:-1:-1::
```

Figura 2: Formato del archivo Shadow

El significado de cada campo delimitado por ":" se detalla en la Table 3.

Campo	*Descripción*
Campo 1	Nombre de la cuenta del usuario.
Campo 2	Contraseña cifrada o encriptada, un '*' indica cuenta de 'nologin'.
Campo 3	Días transcurridos desde el 1/ene/1970 hasta la fecha en que la contraseña fue cambiada por última vez.
Campo 4	Número de días que deben transcurrir hasta que la contraseña se pueda volver a cambiar.
Campo 5	Número de días tras los cuales hay que cambiar la contraseña. (-1 significa nunca). A partir de este dato se obtiene la fecha de expiración de la contraseña.
Campo 6	Número de días antes de la expiración de la contraseña en que se le avisará al usuario al inicio de la sesión.
Campo 7	Días después de la expiración en que la contraseña se inhabilitara, si es que no se cambio.
Campo 8	Fecha de caducidad de la cuenta. Se expresa en días transcurridos desde el 1/Enero/1970 (epoch).
Campo 9	Reservado.

Table 3: Campos del archivo Shadow

El archivo group almacena la relación entre usuario y grupos al que pertenecen, se tiene una línea por cada usuario, el archivo se encuentra ubicado en la ruta /etc/group en la Figura 3 se observa el formato del archivo.

```
root:x:0:root
ana:x:501:
sergio:x:502:ventas,supervisores,produccion
cristina:x:503:ventas,sergio
```

Figura 3: Formato del archivo Group

El detalle de cada campo del archivo group delimitado por ":" se detalla en la Table 4.

Campo	*Descripción*
Campo 1	Indica el usuario.
Campo 2	'x' indica la contraseña del grupo, que no existe, si hubiera se mostraría un 'hash' encriptado.
Campo 3	Group ID (GID) o identificación del grupo.
Campo 4	Es opcional e indica la lista de grupos a los que pertenece el usuario.

Table 4: Campos del archivo Group

Siguiendo con el cuarto archivo de configuración mencionaremos al archivo login.defs, el cual establece variables que controlan aspectos en la creación de los usuarios y los campos que figuran en el archivo shadow que se utilizan por defecto. Las variables que define se utilizan cuando se crea un usuario a través del comando useradd y cuando se modifican con el comando usermod. Ejemplos de éstas variables son PASS_MIN_LEN que indica el número mínimo de caracteres de la contraseña , CREATE_HOME que indica si el comando useradd debe crear el directorio home por defecto.

La autenticación en el sistema operativo Windows [10]se hace manejando sesiones, y en cada sesión se solicita un usuario y contraseña. Inicialmente cuando se instala el sistema operativo crea por defecto la cuenta administrador y una cuenta invitado. Posteriormente el usuario puede agregar nuevas cuentas. Windows posee la herramienta administrador de usuarios el cual permite crear, borrar y modificar cuentas según los permisos otorgados.

Las cuentas de usuarios cuando se crean de forma local es decir en un computador particular se almacenan automáticamente en la memoria de acceso secuencial (SAM) del equipo, ubicada en \WINDOWS\system32\config\, lo que ocasiona que solamente éste usuario pueda iniciar sesión en dicho computador, en caso de crear la cuenta de usuario en un controlador de dominio, el usuario es almacenado automáticamente en la SAM del controlador de dominio principal (PDC) y luego se sincroniza con el resto del dominio, permitiendo que con la cuenta creada se pueda iniciar sesión en cualquier computador del dominio.

Al igual que Linux, se tiene un directorio particular del usuario en la cual el usuario puede almacenar sus archivos. Además se maneja el concepto de Perfil, el cual se crea al iniciar sesión. El perfil configura el entorno de trabajo del usuario como también conexiones a red y/o a impresoras entre otras configuraciones.
El perfil del usuario se puede configurar para restringir el acceso a ciertos elementos en el escritorio u ocultar algunas herramientas del sistema. Los perfiles se almacenan en C:\Winnt\Profiles.
Existe la posibilidad de tener perfiles móviles, estos permiten tener un mismo entorno de trabajo independiente del computador al cual se conecte el usuario dentro de la red. Los perfiles móviles se guardan en un servidor de la red. Existen dos formas de configurar el perfil móvil, de forma obligatorio o personalizado. La forma obligatorio permite que el administrador sea quien decide que opciones se le habilita en el entorno de trabajo al usuario, el cual no puede modificar, es decir que si el usuario cambia alguna configuración del entorno de trabajo una vez que termina la sesión éstas se pierden, la otra forma de configurar el perfil móvil es de forma personal, en donde en este caso si se le permite al usuario cambiar el entorno de trabajo y los cambios perduran a lo largo de las sesiones.

La administración de los usuarios se puede hacer a través de grupos. Esto significa que se puede definir grupos de usuarios con el mismo tipo de permisos, organizando a estos en categorías.
Un grupo es un conjunto de cuentas de usuarios, una vez que un usuario se agrega al grupo obtiene todos los permisos y derechos de ese grupo. Existen grupos predefinidos y se pueden dar de alta según las necesidades requeridas.

En ASSE como ya se mencionó la autenticación se hace de forma local.. Como una mejora a éste escenario se instaló un controlador de dominio en un servidor Linux, el cual permita administrar la gestión de usuarios de forma central a través de un directorio LDAP, el cual también centralice la autenciación de usuarios y grupos de redes Windows, para lograr éste ultimo punto se requiere de Samba, el cual implementa el protocolo SMB y permite que un ordenador con Linux pueda explorar una red Windows, acceder a sus recursos, compartir los propios y autenticarse.

Un controlador de dominio es una entidad administrativa, que fija reglas de seguridad y autenticación comunes. Para regular un dominio, se precisa al menos de un equipo que sea el controlador principal, la fuente primera donde se almacenan las reglas del dominio, y donde serán consultadas esas reglas en última instancia denominado un controlador primario de dominio (PDC).

Para que la autenticación de los usuarios se realice teniendo en cuenta el directorio LDAP, es necesario instalar un Plugin para la biblioteca de sistema NSS (Name Service Switch).
La biblioteca NSS se encarga de realizar el mapeo entre el nombre de usuario y el identificador que le asigna el sistema Unix a los usuarios y a los grupos denominado respectivamente UID (User ID) y GID (Group ID).
Por defecto la biblioteca NSS consulta los usuarios y grupos en los archivos almacenados en /etc/password, /etc/shadow y /etc/group. Para evitar que consulte estos archivos y tome la información del directorio LDAP es necesario instalar el plugin nss-ldap

También se requiere la instalación del modulo PAM (Pluggable Authentication Modules) [11] no es un modelo de autenticación en sí, sino que se trata de un mecanismo que proporciona una interfaz entre las aplicaciones de usuario y diferentes métodos de autenticación, el cual permite al administrador del sistema elegir el modo en que los programas autenticarán a los usuarios, con PAM es posible intercambiar los métodos de autenticación ya sea mediante /etc/passwd hasta dispositivos hardware como lectores de huella digital, pasando por servidores LDAP o sistemas de gestión de bases de datos sin necesidad de cambiar ninguna linea de código por tanto evitando recompilar.
PAM va más allá todavía, permitiendo al administrador del sistema construir políticas diferentes de autenticación para cada servicio. En el prototipo se utilizó para que utilice le método de autenticación del servidor LDAP.

1.3 Implementación

Para cumplir con los objetivos planteados de autenticar los usuarios de forma centralizada , se siguieron los siguientes pasos [12]:

1. Instalar y configurar servidor LDAP
2. Configurar el cliente LDAP
3. Configurar Samba
4. Configurar herramientas smbldap
5. Configurar resoluciones de identidades con NSS-LDAP
6. Configurar autenticación con PAM-LDAP
7. Migración al LDAP de los usuarios y grupos Linux
8. Integrar cliente Linux al dominio Samba
9. Integrar clientes Windows al dominio Samba

El detalle técnico de cada paso y los archivos de configuración se muestran a continuación.

1.3.1 Instalación y configuración del servidor LDAP

El servidor LDAP instalado es OpenLDAP. Versión 2.4.24, la IP del servidor LDAP es 10.202.152.14, el sistema operativo del servidor es OpenSUSE 11.2 (i586). La versión de OpenLDAP instalada es 2.4.17-5.3 y la versión de Samba instalada es 3.4.3-3.3.1

Las sentencias para operar el servidor LDAP se detallan a continuación:

- Comando para levantar el servidor LDAP: service ldap start
- Comando para bajar el servidor LDAP: service ldap stop
- Comando para revisar sintácticamente el archivo de configuración: slaptest -v -u

Una vez instalado el OpenLDAP, es necesario configurar el servidor LDAP. El archivo de configuración que se requiere modificar es:

- /etc/openldap/slapd.conf

La estructura del archivo slapd.conf tiene cuatro secciones básicamente que se deben de configurar. Se detallan en la figura Figura 4.

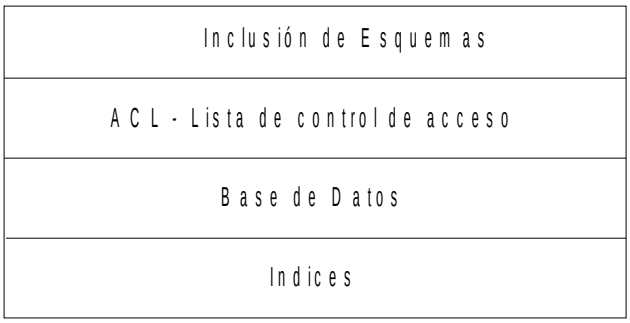

Figura 4: Secciones del archivo slapd.conf

Los parámetros que debemos adaptar son los detallados a continuación.

Identificador	Descripción
database	Tipo de base de datos.
suffix	Especifica la parte del árbol de directorios LDAP de la que se va a ocupar él servidor.
rootdn	Determina quién dispone de derechos de administración para este servidor. No es necesario que el usuario indicado en esta sección posea una entrada LDAP o que exista como usuario “normal”.
rootpw	Contraseña del administrador.
directory	Directorio en el que están almacenados los directorios de la base de datos en el servidor.
index objectClass eq	Permite que se cree un índice con las clases de objetos.

Se debe ubicar el la siguiente sección del archivo:

```
database bdb
suffix "dc=<MY-DOMAIN>,dc=<COM>"
rootdn "cn=Manager,dc=<MY-DOMAIN>,dc=<COM>"
rootpw secret
directory /var/lib/ldap
```

Para nuestro prototipo asignamos lo siguiente:.

```
database        bdb
suffix          "dc=example,dc=com"
rootdn          "cn=admin,dc=example,dc=com"
rootpw          secret
directory       /var/lib/ldap/prueba
```

Otro punto que se debe agregar es la inclusión del esquema samba3.schema, el cual permite almacenar atributos para cuentas de usuario y dominios Samba/Windows. Este esquema no viene presente como esquema por defecto en la instalación de OpenLDAP. Un esquema define los tipos de objetos que podemos manejar en el DIT, los tipos de atributos que se pueden utilizar como también las reglas de sintaxis de cada atributo.

Para agregar el esquema se debe copiar el esquema al directorio /etc/openldap/schema/ y se debe de agregar la siguiente sentencia al archivo slapd.conf.

```
include         /etc/openldap/schema/samba3.schema
```

El archivo /etc/openldap/schema/samba3.schema debe tener los permisos siguientes:

```
rw-r--r-- 1 root  root
```

Se agregaron indices sobre algunos atributos para tener una búsqueda mas eficiente. Los mismos se detallan con la palabra index al inicio de la sentencia en la sección de indices (esto es opcional).

El archivo slapd.conf debe tener los siguientes permisos:

```
rw-r----- 1 root ldap.
```

En la Table 5 se expone como debiera quedar configurado el archivo.

```
# See slapd.conf(5) for details on configuration options.

include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include         /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/nis.schema
include         /etc/openldap/schema/yast.schema
include         /etc/openldap/schema/samba3.schema

  ########################################################################
#########
# Define global ACLs to disable default read access.

pidfile         /var/run/slapd/slapd.pid
argsfile        /var/run/slapd/slapd.args
loglevel        stats
logfile         /var/log/ldap/ldap.log

access to dn.base=""
        by * read

access to dn.base="cn=Subschema"
        by * read

access to attrs=userPassword,userPKCS12,sambaLMPassword,sambaNTPassword
        by self write
        by * auth

access to attrs=shadowLastChange
        by self write
        by * read

access to *
        by * read
  ########################################################################
#########
# BDB database definitions

database        hdb
suffix          "dc=example,dc=com"
rootdn          "cn=admin,dc=example,dc=com"
rootpw          ldapadmin

# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd and slap tools.

directory       /var/lib/ldap/prueba
  ########################################################################
#########
# Indices to maintain
index   objectClass,uidNumber,gidNumber                 eq
index   cn,sn,uid,displayName pres,sub,eq
index   mail,givenname          eq,subinitial
index   sambaSID,sambaPrimaryGroupSID,sambaDomainName           eq

  ########################################################################
#########
# Replicas
moduleload syncprov
overlay syncprov
syncprov-checkpoint 100 10
Syncprov-sessionlog 100
```

Table 5: Archivo de configuración del servidor OpenLdap

1.3.2 Configurar el cliente LDAP

EL cliente LDAP permite utilizar herramientas para accionar con el directorio, como pueden ser ldapsearch que permite buscar una entrada, o ldapadd que permite agregar una entrada en el directorio. Estas herramientas necesitan que se configure el archivo /etc/ldap/ldap.conf en dicho archivo se configura la siguiente información:

- URI - Dirección del servidor o servidores LDAP predeterminados
- BASE - Sufijo de la base de búsqueda ó Base DN
- SIZELIMIT - Tamaño máximo de las búsquedas
- TIMELIMIT - Limite de tiempo para las consultas
- BINDDN - DN de la cuenta con la que se efectuarán las operaciones

En la Table 6 se expone el archivo de configuración del cliente LDAP.

```
#
# LDAP Defaults
#

# See ldap.conf(5) for details
# This file should be world readable but not world writable.

TLS_REQCERT allow
BASE   dc=example,dc=com
URI    ldap://127.0.0.1
BINDDN          cn=admin,dc=example,dc=com
```

Table 6: Archivo de configuración para el cliente LDAP /etc/ldap/ldap.conf

1.3.3 Instalar y configurar Samba

Es necesario instalar y configurar Samba para que actúe como controlador de dominio primario (PDC) Windows NT el cual tenga centralizado en el directorio LDAP la base de datos SAM (security account manager) la cual tiene información de las cuentas de usuarios. Los parámetros generales de configuración son:

- Identificación de red: Dominio y Nombre Servidor.
- Interfaces y direcciones de red a las que está conectado el servidor
- Configuración de parámetros para el controlador de dominio
- Información del servidor LDAP y como utilizarlo
- Definición de script para automatizar la administración de las cuentas Samba/ldap.
- Directorios compartidos especiales como homes, netlogon y profiles.

Se instalaron los paquetes *samba y samba-cliente 3.4.3-3.3.1*. Además de éstos, se requirió instalar los siguientes paquetes:

- clamav-0.97-5.2.2.i586.rpm
- dhcp-server-3.1.2p1-4.10.1.i586.rpm
- clamav-db-0.96.5-0.2.1.i586.rpm
- samba-pdb-3.0.12-5
- perl-Convert-ASN1-0.18-69
- samba-doc-3.0.12-5
- samba-winbind-3.4.3-3.2.1.i586.rpm
- perl-ldap-0.29-137

- smbldap-tools-0.9.5-22.2.src.rpm

En la Table 7 se detallan sentencias útiles en el desarrollo del prototipo junto a su descripción.

Sentencia	Descripción
/sbin/service smb start	Permite levantar el servicio Samba.
/sbin/service smb stop	Pemite bajar el servicio Samba.
testparm	Corrobora la correctitud del archivo de configuración de Samba.
net get localsid	Permite obtener el SID del PC.
smbldap-populate	Permite poblar el directorio LDAP con información del controlador de dominio, el script se ubica en el directorio /etc/smbldap-tools.
smbldap-migrate-groups	Permite migrar los grupos del archivo /etc/gruopal directorio LDAP.
smbldap-migrate-accounts	Permite migrar los usuarios de los archivos /etc/passwd y /etc/shadow al directorio LDAP.
pdbedit	Corrobora la correctitud de la configuración de Samba y Ldap. Opción ***-Lv cuenta usuario***. Detalla los atributos de la cuenta. Opción **-L**. Detalla el mapeo de los usuarios Samba con el UID del usuario Unix.
net groupmap list	Verificar el mapeo de los grupos Samba con los grupos Unix.

Table 7: Sentencias utilizadas para la instalción y configuración Samba

En la instalación de Samba por defecto se importa los usuarios que existen en el archivo /etc/passwd. A la base de datos de la cuenta samba. Es por esto que previamente se debe eliminar el contenido de /var/lib/samba/ con extensión tdb y dat.

El archivo de configuración de samba se divide en secciones y se encuentra ubicado en /etc/samba/smb.conf y posee 4 sección detalladas en la Table 8.

Sección	Descripción
[global]	Especifica parámetros globales que afectan el comportamiento del servidor Samba.
[home]	Es un recurso compartido especial que es usado para compartir el directorio $HOME de cada usuario, por ejemplo, si se tiene un usuario Unix/Windows de nombre **jperez**, éste usuario tiene un directorio $HOME en la ruta /home/jperez, con este recurso compartido, cada vez que el usuario jperez inicia sesión por Samba, tendrá disponible un recurso compartido en el servidor con el nombre "jperez" y por medio de él podrá entrar a sus archivos privados en el servidor.
[netlogon]	Es usado cuando Samba actuá como un Controlador de Dominio, y la finalidad de éste recurso compartido es almacenar los scripts de inicio (*logon scripts*), estos scripts son ejecutados cada vez que un usuario inicia sesión en el dominio.
[profiles]	Es usado cuando cuando Samba actuá como Controlador de Dominio, el propósito de este recurso compartido es almacenar los Perfiles de Usuarios.

Table 8: Secciones del archivo de configuración de Samba /etc/samba/smb.conf

El archivo de configuración del servidor Samba se encuentra ubicado en la ruta /etc/samba/smb.conf

a continuación se exponen las cuatro secciones del archivo smb.conf resultante.

```
[global]
        workgroup =EXAMPLE
        netbios name = PDC
        server string = Servidor PDC
        enable privileges = Yes
        map to guest = Bad User
        username map = /etc/samba/smbusers

        # Manejo de LOGS
        syslog = 0
        log level = 0
        max log size = 50
        log file = /var/log/samba/%m.log

        # Manejo de RED
        # Configuración del CONTRALADOR DOMINIO
        time server = Yes
        wins support = Yes
        os level = 33
        domain logons = Yes
        preferred master = Yes
        logon path = \\%L\Profiles\%U
        logon home = \\%L\%U
        logon drive = H:
        logon script = scripts\logon.bat
        case sensitive = No
        utmp = Yes

        # Comunicacion SAMBA-LDAP
        passdb backend = ldapsam: ldap://127.0.0.1/
        #passdb backend = tdbsam

        ldap admin dn = cn=admin,dc=example,dc=com
        ldap suffix = dc=example,dc=com
        ldap group suffix = ou=Groups
        ldap user suffix = ou=Users
        ldap machine suffix = ou=Computers
        ldap idmap suffix= ou=Idmap
        idmap backend= ldap:ldap://127.0.0.1
        idmap uid = 10000-20000
        Idmap gid = 10000-20000
        ldap ssl = No
        ldap passwd sync = Yes
        add machine script = /usr/sbin/smbldap-useradd -w "%u"
        add user script = /usr/sbin/smbldap-useradd -m "%u"
        add group script = /usr/sbin/smbldap-groupadd -p "%g"
        ldap delete dn = No
        delete user script = /usr/sbin/smbldap-userdel "%u"
        delete group script = /usr/sbin/smbldap-groupdel "%g"
        add user to group script = /usr/sbin/smbldap-groupmod -m "%u" "%g"
        delete user from group script = /usr/sbin/smbldap-groupmod -x "%u" "%g"
        set primary group script = /usr/sbin/smbldap-usermod -g "%g" "%u"
        dos charset = 850
        Unix charset = ISO8859-1
```

Table 9: Sección [Global] del archivo smb.conf

```
[homes]
        comment = Home Directories

        valid users = %U
        read only = No
        create mask = 0664
        directory mask = 0775
        browseable = No

[netlogon]
        # Es usado cuando Samba actua como Controlador Dominio, la idea es que
        # almacene los scripts de inicio (logon scripts) ejecutados al inicio de  # una sesion de usuario

        path = /home/samba/netlogon/
        browseable = No
        read only = Yes

[profiles]
        # Es usando cuando Samba actua como Controlador Dominio, la idea es que  #    almacene    los
perfiles de los usuarios

        path = /home/samba/profiles
        read only = No
        create mask = 0600
        directory mask = 0700
        browseable = No
        guest ok =Yes
        profile acls =Yes
        csc policy = disable
        force user = %U
        valid users = %u @"Domain Admins"

        # Ejemplo: el usuario jmedina ProfilePath= \\PDC\profiles\jmedina el
        # perfil se almacena en /home/smaba/profiles/jmedina

[publico]
        # Espacio publico de docuemtnos. Todo usuario del dominio puede acceder

        comment= Directorio de datos publicos
        path = /home/samba/publico
        read only = Yes
        guest ok = Yes
```

Table 10: Sección [homes],[netlogon], [profile] y [publico] del archivo smb.conf

En la sección netlogón, se define el directrorio en donde se ubican los scripts de inico (*logon scripts*), estos scripts son ejecutados cada vez que un usuario inicia sesión en el dominio. Por ésto se debe crear la carpeta netlogon dentro del directorio /home/samba/. Se ejecuta el siguiente comando mkdir -p --mode 755 /home/samba/netlogon. Posteriormente dentro de dicho directorio se crea el script logon.bat cuyo contenido perimite sincronizar la hora del pc con la del servidor y mapear un recurso a la unidad h, el script se muestra en Figura 5.

```
net time \\PDC /set /yes
net use h: \\PDC\homes
```

Figura 5: Script ejecutado al inicio de sesión logon.bat

Posteriormente se debe de cambiar el formato del archivo DOS a CR/LF. Para realizar ésta acción se debe ejecutar la siguient sentencia: sed -i 's/$/\r/' /home/samba/netlogon/logon.bat

Se agregó un recurso publico el cual es accedido por todos los usuarios el cual contiene imágenes,

documentos y programas. Para ésto se debe de crear el directorio publico ejecutando la sentencia:
mkdir -p /home/samba/publico/{docs,programas,imágenes}

Para detectar que la configuración es correcta se debe ejecutar la sentencia testparm desde la consola. Si no muestra error es correcta la configuración.

Para que Samba pueda acceder al directorio para agregar o modificar una cuenta de usuario en el directorio LDAP, debe saber la contraseña del usuario ldap admin dn especificada en el archivo smb.conf. Y a su vez dicha cuenta debe de tener los permisos para leer y escribir entradas en el directorio LDAP.
Para ésto lo que se hizo fue ejecutar el comando smbpasswd -W dicho comando solicita que se ingrese el usuario y contraseña. La contraseña se guarda en el directorio /var/lib/samba/secrets.tdb y permite que no sea solicitada en cada operación en el directorio. Los permisos de ese directorio debe ser -rw------- 1 root root.

1.3.4 Configurar herramientas smbldap

Las herramientas smbldap que provee el paquete *smbldap-tools*, son un conjunto de scripts que permiten la manipulación de usuarios y grupos almacenados en un directorio LDAP, herramientas destinadas a sistemas con Samba-LDAP y PAM/nss_ldap.

Adicionalmente, contiene algunos scripts para facilitar la migración de servidores PDC Windows NT 4.0 a servidores PDC Samba-LDAP. Estas son: smbldap-populate, smbldap-migrate-groups y smbldap-migrate-accounts.

Las herramientas smbldap-tools se encuentran en el directorio /etc/smbldap-tools y contiene dos archivo de configuración:

- /etc/smbldap-tools/smbldap.conf – Archivo que contiene directivas de configuración tanto para la creación y modificación de cuentas Unix y Samba.
- /etc/smbldap-tools/smbldap_bind.conf – Archivo que contiene el usuario y contraseña con el que nos conectaremos al servidor LDAP para efectuar las operaciones de administración de usuarios y grupos Unix/Samba.

Se le asignó los siguientes permisos a los archivos de configuración (smbldap_bind.conf y smbldap.conf) -rw-r----- 1 root ldap.

A continuación se detallan los archivos de configuración.

```
##############################
# Credential Configuration #
##############################

slaveDN="cn=admin,dc=example,dc=com"
slavePw="ldapadmin"
masterDN="cn=admin,dc=example,dc=com"
masterPw="ldapadmin"
```

Table 11: Archivo de configuración smbldap-populate /etc/smbldap-tools/smbldap_bind.conf

```
                        #######                    General                    Configuration
##############################################################################
###
SID="S-1-5-21-3190323790-1237239473-3967027434"

######## Domain name the Samba server is in charged.
sambaDomain="EXAMPLE"

                        #######                    LDAP                    Configuration
##############################################################################
###
######## Slave LDAP server
slaveLDAP="127.0.0.1"

######## Slave LDAP port
slavePort="389"

######## Master LDAP server: needed for write operations
masterLDAP="127.0.0.1"

######## Master LDAP port
masterPort="389"

######## Use TLS for LDAP
ldapTLS="0"

######## Use SSL for LDAP
ldapSSL="0"

######## How to verify the server's certificate (none, optional or require)
verify="require"

######## CA certificate
cafile="/etc/smbldap-tools/ca.pem"

######## certificate to use to connect to the ldap server
clientcert="/etc/smbldap-tools/smbldap-tools.iallanis.info.pem"

######## key certificate to use to connect to the ldap server
clientkey="/etc/smbldap-tools/smbldap-tools.iallanis.info.key"

######## LDAP Suffix
suffix="dc=example,dc=com"

######## Where are stored Users
usersdn="ou=Users,${suffix}"

######## Where are stored Computers
computersdn="ou=Computers,${suffix}"

######## Where are stored Groups
groupsdn="ou=Groups,${suffix}"

######## Where are stored Idmap entries (used if samba is a domain member server)
idmapdn="ou=Idmap,${suffix}"

######## Where to store next uidNumber and gidNumber available for new users and groups
sambaUnixIdPooldn="sambaDomainName=${sambaDomain},${suffix}"

######## Default scope Used
scope="sub"

######## Unix password encryption (CRYPT, MD5, SMD5, SSHA, SHA, CLEARTEXT)
```

```
hash_encrypt="SSHA"

####### if hash_encrypt is set to CRYPT, you may set a salt format.
crypt_salt_format="%s"

# Unix Accounts Configuration
##################################################
# Login defs
userLoginShell="/bin/false"

# Home directory
userHome="/home/%U"

# Default mode used for user homeDirectory
userHomeDirectoryMode="700"

# Gecos
userGecos="System User"

# Default User (POSIX and Samba) GID
defaultUserGid="513"

# Default Computer (Samba) GID
defaultComputerGid="515"

# Skel dir
skeletonDir="/etc/skel"

# Default password validation
defaultMaxPasswordAge="99999"

# SAMBA Configuration  ##################################################
# The UNC path to home drives location (%U username substitution)
userSmbHome="\\PDC\%U"

# The UNC path to profiles locations (%U username substitution)
userProfile="\\PDC\profiles\%U"

# The default Home Drive Letter mapping
userHomeDrive="H:"

# The default user netlogon script name (%U username substitution)
userScript="logon.bat"

# Domain appended to the users "mail"-attribute
mailDomain="example.com"

# SMBLDAP-TOOLS Configuration (default are ok for a RedHat)
##############################
# Allows not to use smbpasswd (if with_smbpasswd="0" in smbldap.conf) but
with_smbpasswd="0"
smbpasswd="/usr/bin/smbpasswd"

# Allows not to use slappasswd (if with_slappasswd="0" in smbldap.conf)
# but prefer Crypt:: libraries
with_slappasswd="0"
slappasswd="/usr/sbin/slappasswd"

# comment out the following line to get rid of the default banner
# no_banner="1"
```

Table 12: Table 14: Archivo de configuración smbldap-populate /etc/smbldap-tools/smbldap.conf

Luego de instalar y configurar las herramientas smbldap, se pobló el directorio LDAP, con la información para el dominio NT: EXAMPLE. Para esto se utiliza el comando smbldap-populate.

El directorio Ldap se pobló con la siguiente información:

Directorio Ldap	Contenido
Base DN	dc=example,dc=com
Contenedor para las cuentas Unix/Samba	ou=Users
En el contenedor para los Usuarios	se crearán por defecto dos usuarios: **root** (*Usuario Unix/Samba* - Administrador Unix y Administrador de Dominio Samba) y ***nobody*** *(Usuario Unix/Samba -Cuenta anónima para entornos Unix y Dominio Samba)*
Contenedor para los Grupos Unix/Samba	ou=Groups. Contenedor para almacenar Grupos de sistema para sistemas Unix y Windows (o para cualquier otro sistema LDAP-aware).
En el contenedor para los grupos	Se crearán por defecto los Grupos globales, predeterminados de un dominio Samba: ***Domain Admins***, Grupo Global para los Administradores del Dominio NT: EXAMPLE. ***Domain,*** Users, Grupo Global para los Usuarios de Dominio NT: EXAMPLE ***Domain Computers*.** Grupo Global para las Cuentas de Computadoras del Dominio NT: EXAMPLE Además se crean 5 grupos locales: ***Account Operators***, ***Administrators***, ***Backup Operators***, ***Print Operators***, ***Replicators***
Contenedor para las cuentas de Computadoras Windows	ou=Computers. Contenedor para las cuentas de Computadoras (Trusted Machine Accounts) para sistemas Windows
Contenedor para los mapeos de Cuentas Unix a Cuentas Samba/Windows (SID)	ou=Idmap

Table 13: Detalle del contenido LDAP luego de ejecutar smbldap-populate

Existe un error ya conocido al ejecutar el comando smbldap-populate, el cual cual asigna a los grupos locales el valor 5 al atributo *sambaGroupType* pero debiera se 4, para modificar éste parámetro se debe de realizar un archivo ldif y correrlo para actualizar el DIT. El comando que permite realizar la modificación es el comando ldapmodify. En la Table 17 se muestra el contenido del archivo ldif y se almacenó en la ruta /tmp/samba-builtin-changetype.ldif

```
dn: cn=Account Operators,ou=Groups,dc=example,dc=com
changetype: modify
replace: sambaGroupType
sambaGroupType: 4

dn: cn=Administrators,ou=Groups,dc=example,dc=com
changetype: modify
replace: sambaGroupType
sambaGroupType: 4

dn: cn=Backup Operators,ou=Groups,dc=example,dc=com
changetype: modify
replace: sambaGroupType
sambaGroupType: 4

dn: cn=Print Operators,ou=Groups,dc=example,dc=com
changetype: modify
replace: sambaGroupType
sambaGroupType: 4
```

Table 14: Archivo LDIF que permite realizar el cambio del atributo sambaGroupType

Para efectuar el cambio, se debe ejecutar el siguiente comando: ldapmodify -h localhost -x -D "cn=admin,dc=example,dc=com" -W -f /tmp/samba-builtin-changetype.ldif

Para realizar una comprobación de la correctitud del archivo de configuración de Samba y Ldap se debe ejecutar el comando pdbedit.

Para realizar una comprobación de la correctitud del mapeo entre grupos Samba y grupos Unix se puede ejecutar el comando: net groupmap list. El cual muestra el mapeo de los grupos Windows/Samba a grupos Unix a través del SID (*Security Identifier)* del grupo Windows/Samba al GID del grupo Unix.

1.3.5 Configurar resoluciones de identidades con NSS-LDAP

Por defecto la biblioteca de sistema NSS realiza la consulta de usuarios, grupos y shadow usando archivos locales, es decir, /etc/passwd, /etc/group y /etc/shadow respectivamente, como se está configurando que se use el directorio LDAP para almacenar la información de los usuarios y grupos será necesario indicarle a la biblioteca NSS que debe usar el directorio LDAP para obtener información sobre dichas identidades, es aquí donde entra el paquete nss_ldap, el paquete nss_ldap es un plugin para la biblioteca de sistema NSS para poder realizar la resolución de identidades usando como fuente de origen un directorio LDAP.
Se instaló el siguiente paquete: nss_ldap-265-4.2.i586.

La configuración de NSS-LDAP, se encuentra en el archivo /etc/ldap.conf. Se muestra en la Table 15 el contenido del archivo. Se debe de constatar que tenga los siguientes permisos: -rw-r--r-- 1 root root

```
# Información del servidor LDAP
uri ldap://localhost/
ldap_versión 3
scope sub

# Base de busqueda
base dc=example,dc=com

# Contenedores para cuentas de usuario, grupos y computadoras
nss_base_passwd         ou=Users,dc=example,dc=com
nss_base_passwd         ou=Computers,dc=example,dc=com
nss_base_shadow         ou=Users,dc=example,dc=com
nss_base_group          ou=Groups,dc=example,dc=com

# Tipo binding
bind_policy soft
        nss_initgroups_ignoreusers          daemon,bin,sys,sync,games,man,lp,mail,news,uucp,proxy,www-
data,backup,list,irc,gnats,nobody,libuuid,dhcp,syslog,klog,sshd,ntp,snmp,openldap

# Filtros PAM
pam_login_attribute uid
pam_member_attribute memberuid
pam_filter objectclass=posixAccount

# Cambio contraseÃ±ia passwd(1)
pam_password exop
```

Table 15: Archivo de configuración de NSS-LDAP, ubicado en /etc/ldap.conf

Es necesario configurar la resolución de entidades vía LDAP en el archivo de configuración de la biblioteca NSS: /etc/nsswitch.conf y modificar las entradas para las entidades: passwd, group y shadow. El archivo de configuración se muestra en la Table 16.

```
#
# /etc/nsswitch.conf

shadow: compat ldap
passwd: compat ldap
group:  compat ldap

hosts:          files wins mdns4_minimal [NOTFOUND=return] dns mdns4
networks:       files dns

services:       files ldap
protocols:      files
rpc:            files
ethers:         files
netmasks:       files
netgroup:       files ldap
publickey:      files

bootparams:     files
automount:      files nis
aliases:        files ldap
passwd_compat:          ldap
group_compat:ldap
```

Table 16: Configuración de la biblioteca NSS ubicado en /etc/nsswitch.conf

1.3.6 Configurar autenticación con PAM-LDAP

PAM [11] como ya se mencionó permite que el administrador pueda fijar el método de autenticación sin necesidad de recompilar, permite también construir políticas de autenticación diferente para cada servicio y a su vez su alcance queda delimitado por cuatro grupos de gestión estos son account, authentication, password y sesion. En la Table 17 se detalla cada grupo.

Control	Descripción
account	En este grupo se engloban tareas que no están relacionadas directamente con la autenticación. Algunos ejemplos son permitir/denegar el acceso en función de la hora, los recursos disponibles o, incluso, la localización. Ofrece verificación de cuentas de usuario. Por ejemplo, se encarga de determinar si el usuario tiene o no acceso al servicio, si su contraseña ha caducado, etc.
authtentication	Tareas encaminadas a comprobar que, efectivamente, el usuario es realmente quien dice ser. A menudo, cuando se habla de PAM, solo se tiene en cuenta ésta tarea, ignorando las demás. Estas tareas ofrecen incluso un sistema de credenciales que permiten al usuario ganar ciertos privilegios (establecidos por administrador).
password	Se encarga de mantener actualizado el elemento de autenticación asociado a cada usuario como puede ser su contraseña. Acciones como comprobar la fortaleza de una clave son típicas de este grupo.
session	En este grupo se engloban tareas que se deben llevar a cabo antes de iniciarse el servicio y después de que este finalice. Es especialmente útil para mantener registros de acceso o hacer accesible el directorio home del usuario

Table 17: Grupos de alcance de PAM

La configuración de PAM, se puede lograr de dos formas distintas, la primera es modificando el archivo de configuración /etc/pam.conf (en éste caso dentro del archivo se debe especificar en cada directiva a que servicio hace referencia) y la segunda forma es agregar por cada servicio un archivo de configuración distinto en el directorio /etc/pam.d/. De ambas formas la sintaxis del archivo es la misma salvo que con la segunda forma no se especifica el nombre del servicio ya que es parte del nombre del archivo.

El archivo de configuración de PAM asumiendo la segunda forma de configuración se construye con cuatro campos, a continuación se detalla el significado de cada campo, si optamos por la primera forma de configuración es lo mismo pero se agrega al principio el nombre de la aplicación/servicio.

Tipo Modulo	**Control** (Indica a PAM como actuar)	**Ruta**	**Argumentos** (Opcional)

El campo “Tipo Modulo” indica a que grupo de administración está asociada la regla. Sus posibles valores el área al que se destina esta regla. Puede adoptar uno de estos valores: auth, account, session o password. Los módulos correspondientes a una misma área forman una pila.

El campo “Control” indica que hacer cuando falla el control aplicado, sus posible valores son:

Control	Descripción
required	Indica que es necesario que el módulo tenga éxito para que la pila también lo tenga. Si se produce un fallo, no se notifica hasta que se procesa el resto de la pila.
rquisite	Es como el required, pero devuelve el control a la aplicación enseguida de fallar.
sufficient	El éxito en este módulo, si no se ha producido un fallo en los procesados anteriormente en la pila, es “suficiente”. Llegados a este punto, el procesamiento se detiene (ignorando incluso posibles required posteriores). Un fallo no siempre resulta definitivo para la pila.
optional	Por lo general, PAM ignora los módulos marcados con este indicador. Su valor será tenido en cuenta sólo en caso de que no se haya llegado a ningún valor concreto de éxito o fracaso —por ejemplo, PAM IGNORE—.

Table 18: Valores posibles del campo Control del archivo de configuración /etc/pam.d/servicio

EL campo “Ruta” indica el path del modulo PAM asociado a la regla. Los módulos se pueden encontrar en /lib/security. Si la ruta comienza con ”/” indica que es absoluta.

El campo “Argumentos” Se trata de argumentos que pueden ser pasados al módulo para su operación. Generalmente, los argumentos son específicos para cada módulo y deberían estar documentados. Si se pasara un argumento no válido, el módulo lo ignoraría, aunque debería usar syslog para informar del error.

Los archivos de configuración que se modificaron y su contenido se muestran a continuación. Tener en cuenta que el módulo de autenticación pam_ldap debe ser llamado por el nombre pam_ldap.so en los archivos de configuración, el módulo pam_ldap.so utiliza los parámetros de configuración definidos en el archivo /etc/ldap.conf para conectarse y consultar el servidor LDAP en donde buscará la información de cuentas UNIX.

```
#%PAM-1.0

auth   sufficient      pam_unix.so likeauth nullok
auth   sufficient      pam_ldap.so use_first_pass
auth    required        pam_deny.so
```

Table 19: Archivo de configuración /etc/pam.d/common-auth

```
#%PAM-1.0

account      required        pam_unix.so
account      sufficient      pam_ldap.so
```

Table 20: Archivo de configuración /etc/pam.d/common-account

```
#%PAM-1.0
#
session      required        pam_unix.so
session      optional        pam_ldap.so
```

Table 21: Archivo de configuración /etc/pam.d/common-session

```
#%PAM-1.0

password        required        pam_cracklib.so difok=2 minlen=8 dcredit=2
ocredit=2 retry=3

password        sufficient      pam_unix.so nullok use_authtok md5 shadow
password        sufficient      pam_ldap.so use_authtok
password        required        pam_deny.so
```

Table 22: Archivo de configuración /etc/pam.d/common-password

1.3.7 Migración al LDAP de los usuarios y grupos Linux

En éste capitulo explicaremos como migrar las cuentas locales de sistema (/etc/passwd) y los grupos locales (/etc/group) a nuestro directorio LDAP.
El script para migrar cuentas Unix se encuentra ubicado en: /usr/sbin/smbldap-migrate-unix-groups
El script para migrar grupos Unix se encuentra ubicado en:/usr/sbin/smbldap-migrate-unix-accounts

Para realizar la migración de usuarios (/etc/passwd y /etc/shadow) ejecutar:
smbldap-migrate-accounts -a -P “ruta archivo passwd“ -S “ruta archivo shadow”

Para realizar la migración de grupos (/etc/group) ejecutar:
smbldap-migrate-groups -a -G “ruta archivo group”

1.3.8 Agregar un equipo Linux al domino

Una vez que se instaló el servidor LDAP, Samba y se definió el controlador de dominio, se requiere integrar PC`s al dominio (tanto PC con sistema operativo linux como windows).

En éste se capitulo se detalla la integración al dominio de una PC con sistema operativo linux. Las PC a integrar tiene su autenticación independiente usando sus propios archivo /etc/passwd, /etc/shadow y /etc/group principalmente. Lo que se pretende es que el equipo sea un miembro del dominio, y participe en la autenticación centralizada en nuestro directorio LDAP y a su vez que se pueda comunicar con los otros hosts presentes al dominio.

Básicamente los pasos seguidos son:

1. Instalar y configurar en la PC el paquete libnss-ldap. Como se indicó en el apartado 10.3.5 permite la resolución de cuentas de usuario y grupos Unix mediante el directorio LDAP . Los archivos de configuración modificados son:
   - /etc/ldap.conf
   - /etc/nsswitch.conf

2. Instalar y configurar en la PC, el paquete libpam-ldap. Como se indicó en el apartado 10.3.6 permite elegir de forma transparente el esquemas de autenticación a utilizar .
   Los archivos de configuración modificados son:
   - /etc/pam.d/common-auth
   - /etc/pam.d/common-account
   - /etc/pam.d/common-session
   - /etc/pam.d/common-password

3. Instalar Samba en la PC, para lograr autenticar usuarios y grupos de dominio Samba. En la PC. Los archivos de configuración modificados son:
    - /etc/samba/smb.conf

Los datos de contexto a tener en cuenta son los siguientes:
El servidor LDAP tiene la ip: 10.202.152.14. El pc que se quiere integrar al dominio tiene la ip: 10.202.152.15 y su sistema operativo es: OpenSUSE 11.3 (i586).

A continuación se lista el contenido de los archivos de configuración modificados de la PC Linux que se quiere integrar al dominio:

```
[global]
  workgroup =EXAMPLE
  netbios name = jmlap
  server string = jmlap en EXAMPLE
  security = DOMAIN
  username map = /etc/samba/smbusers

  # Lo agregue yo para no usar ssl***************
  ldap ssl = NO

#========== Configuraciones de Red ============================

  socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
  interfaces = eth0 lo
  bind interfaces only = Yes
  smb ports = 139 445

 # hosts allow = 127.0.0.1
  hosts deny = 0.0.0.0

 # remote announce = 127.0.0.1
 # wins server = 10.202.152.14
  name resolve order = wins hosts lmhosts bcast

#=========  Opciones para registro de eventos (Logging)======

  log level = 1
  syslog = 0
  log file = /var/log/samba/%m.log
  max log size = 50
  utmp = Yes

#=========  Opciones para la codificaciÃ³==================

  Dos charset = 850
  Unix charset = ISO8859-1
  display charset = ISO8859-1

#========= Configuraciones para LDAP ======================

  passdb backend = ldapsam:ldap://10.202.152.14
  #passdb backend = ldapsam
  ldap admin dn = cn=admin,dc=example,dc=com
  ldap suffix = dc=example,dc=com
  ldap group suffix = ou=Groups
  ldap user suffix = ou=Users
  ldap machine suffix = ou=Computers
  ldap idmap suffix = ou=Idmap
  idmap backend = ldap:ldap://10.202.152.14
  idmap uid = 10000-20000
  idmap gid = 10000-20000
  winbind trusted domains only = Yes
```

Table 23: Archivo /etc/samba/smb.conf del equipo Linux que se quiere integrar al domino

```
#Config file for libnss-ldap and libpam-ldap

uri ldap://10.202.152.14/
ldap_version 3
scope sub

base dc=example,dc=com

pam_filter objectclass=posixAccount
pam_login_attribute uid
pam_member_attribute memberuid

pam_password exop

nss_base_passwd         ou=Users,dc=example,dc=com
nss_base_passwd         ou=Computers,dc=example,dc=com
nss_base_shadow         ou=Users,dc=example,dc=com
nss_base_group          ou=Groups,dc=example,dc=com

bind_policy soft
                                                                nss_initgroups_ignoreusers
backup,bin,daemon,dhcp,games,gnats,irc,klog,libuuid,list,lp,mail,man,news,openldap,proxy,root,sshd,sy
nc,sys,syslog,uucp,www-data
```

Table 24: Archivo /etc/ldap.conf del equipo Linux que se quiere integrar al domino

```
passwd:         compat ldap
group:          compat ldap
shadow:         compat ldap

hosts:          wins files mdns4_minimal [NOTFOUND=return] dns mdns4
```

Table 25: Archivo /etc/nsswitch.conf del equipo Linux que se quiere integrar al domino

```
auth        sufficient      pam_unix.so likeauth nullok
auth        sufficient      pam_ldap.so use_first_pass
auth            optional        pam_mount.so     use_first_pass
auth        required        pam_deny.so
```

Table 26: Archivo /etc/pam.d/common-auth del equipo Linux que se quiere integrar al domino

```
account         required        pam_unix2.so
```

Table 27: Archivo /etc/pam.d/common-account del equipo Linux que se quiere integrar al domino

```
session     required      pam_unix.so
session     optional      pam_ldap.so
session     optional      pam_mount.so
```

Table 28: Archivo /etc/pam.d/common-session del equipo Linux que se quiere integrar al domino

```
password        required        pam_cracklib.so difok=2 minlen=8 dcredit=2 ocredit=2 retry=3
password        sufficient      pam_unix.so nullok use_authtok md5 shadow
password        sufficient      pam_ldap.so use_authtok
password        required        pam_deny.so
```

Table 29: Archivo /etc/pam.d/common-password del equipo Linux que se quiere integrar al domino

1.3.8.1 Paquetes instalados

pam_ldap-184-149.1.i586
nss_ldap-265-4.2.i586
samba-3.5.4-4.1.i586
samba-client-3.5.4-4.1.i586

1.3.9 Resumen archivos de configuración

En la Tabla 30 se muestra en resumen, el listado de archivos que se deben modificar para lograr implementar un controlador de dominio con Samba y OpenLDAP.

Nombre del archivo	Ubicación
slapd.conf	/etc/openldap/slapd.conf
ldap.conf	/etc/openldap/ldap.conf y /etc/ldap.conf (para nss)
smb.conf	/etc/samba/smb.conf
smbldap.conf	/etc/smbldap-tools
smbldap_bind.conf	/etc/smbldap-tools
nsswitch.conf	/etc/nsswitch.conf
nss-ldapd.conf	/etc/nss-ldapd.conf
common-auth	/etc/pam.d/common-auth
common-session	/etc/pam.d/common-session
common-account	/etc/pam.d/common-account
common-password	/etc/pam.d/common-password

Tabla 30: Archivos a configurar para implementar dominio Samba con OpenLDAP.

1.4 Replicas LDAP

Se plantea la necesidad en ASSE de distribuir los datos del directorio LDAP en distintos puntos geográficos, para logra de ésta forma alta disponibilidad, acceso local con menor tiempo, logrando de ésta forma tener la información sincronizada con el sistema central o maestro evitando tener duplicados los datos.

Los servidores LDAP pueden replicar parte de sus datos como alguna rama del árbol LDAP, lo que permite seleccionar parte de los datos para enviar a diferentes sistemas remotos, incrementando la seguridad y evitando duplicación de datos, todo esto fácilmente configurable.

1.4.1 Arquitecturas de replicas

Existen dos configuraciones posibles para realizar las replicas del directorio LDAP, Maestro-Esclavo y Multi-Maestro.[13].

1.4.1.1 Maestro-Esclavo

En ésta configuración se utiliza un solo servidor maestro, el cual es el único que soporta operaciones de escritura (inserción y modificación) de los datos, por tanto es el que contiene los cambios más recientes de la información. Por otro lado se ejecutan actualizaciones que realizan la replicas o copias de datos a uno o más servidores designados como esclavos.

Los servidores esclavos operar con copia de datos del maestro y solo soportan operaciones de lectura. Los usuarios acceden a los servidores esclavos para leer datos, en caso de solicitar una escritura (inserción y modificación) sólo pueden hacerlo accediendo al servidor maestro. Por otro lado las replicas proporcionan la funcionalidad de copia de seguridad (respaldo o backup).

La configuración Maestro-esclavo tiene dos defectos evidentes:

- Un único punto de fallo. Por existir sólo un servidor maestro.
- Mas de un servidor para el acceso según la operación a realizar. Si los clientes tienen la necesidad actualizar datos, entonces tendrán que acceder a un servidor esclavo para el acceso de lectura y a otro servidor maestro para realizar actualizaciones o inserciones.

1.4.1.2 Multi-Maestro

En ésta configuración de uno o más servidores maestros las operaciones de escrituras se pueden realizar en cualquier maestro, las cuales se propagan a los maestros correspondientes y luego a los esclavos.
Esta funcionalidad fue introducida a partir de la versión 2.4.

La configuración Multi-Maestro tiene las siguientes desventajas:

- Problemas por actualizaciones sobre un mismo datos. Al tener varios servidores maestros, las actualizaciones se pueden dar al mismo tiempo sobre el mismo datos en varios servidores. Provocando inconsistencias..
- Problemas en la eliminación. Si un usuario agrega el dato “x” y otro usuario lo elimina puede figurar que existe.

1.4.2 Formas de sincronizar al Esclavo o Consumidor

Una vez que se realizó una escritura en un servidor Maestro estos datos deben replicarse en el o los servidores Esclavos. La configuración de ésta acción se indica en el esclavo. Existen dos formas posibles para realizar la sincronización, RefreshOnly y RefreshAndPersist. [14] [15] [16]

1.4.2.1 RefreshOnly

En ésta forma de replicación el consumidor inicia una conexión con el proveedor posteriormente se da lugar a la sincronización del directorio una vez realizada la conexión se cierra. Periódicamente, el consumidor o esclavo vuelve a conectarse con el proveedor o maestro y se vuelve a sincronizar. RefreshOnly funciona en modo de ráfaga y el tiempo de ciclo de replicación es el tiempo entre re-conexiones.

Al finalizar una sesión de sincronización el proveedor envía una cookie de sincronización (SyncCookie). Esta cookie contiene un número de secuencia de cambio, básicamente una marca de tiempo que indica el último cambio enviado a éste consumidor y que puede ser visto como un cambio o punto de sincronización (checkpoint). Cuando el consumidor inicia una sesión o conexión, se envía al proveedor la ultima cookie o checkpoint suministrada por el proveedor, de esta forma el proveedor determina a partir de que checkpoint debe enviar los datos.

1.4.2.2 RefreshAndPersist

En ésta forma de replicación el consumidor inicia una conexión con el proveedor, se da lugar la sincronización del directorio inmediatamente y al final de éste proceso, la conexión no se cierra es decir se mantiene abierta (persiste). Los cambios posteriores en el proveedor son propagados de inmediato a los consumidores.

El proveedor envía periódicamente una cookie de sincronización (SyncCookie). Esta cookie contiene un número de secuencia de cambio (contextCSN), marca de tiempo que indica el último cambio enviado al consumidor y que puede ser visto como un checkpoint o punto de sincronización.

1.4.3 Respaldo y Recuperación de datos

Desgraciadamente OpenLDAP no se integra muy bien con las aplicaciones comerciales de backup. Por suerte el sistema de réplica nos permite tener varios servidores los cuales son accedidos desde los clientes (aplicaciones) de tal forma que si falla uno de los servidores siempre podemos tener otro servidor (esclavo) con todos los datos actualizados.[17]

Existen dos formas posibles de realizar un backup de un directorio OpenLDAP, en caliente o en frío como se detalla a continuación.

1.4.3.1 Respaldo en “caliente”

Para llevar adelante el respaldo se exporta todo el directorio LDAP, a un archivo con formato LDIF, posteriormente se debe importar en el directorio LDAP destino. Para realizar éstas dos acciones se ejecutan dos comandos slapcat y slapadd, que se detalla en la Table 31.

Comando	Descripción
slapcat -v -l <nombre_archivo respaldo>	Lee la información de la base de datos en orden secuencial y genera la salida correspondiente en formato LDIF incluyendo atributos de usuario y operacionales. El programa slapcat solo respaldara las entradas que se leyeron en el momento de la ejecución, si en el momento de que slapcat esta ejecutándose se realiza una operación de escritura sobre alguna entrada dicho cambio no será incluido en el respaldo.
slapadd -v -l <nombre_archivo respaldo>	Restaurar el directorio a partir del archivo LDIF

Table 31: Comandos para Respaldar y restaurar el directorio LDAP

Tener en cuenta que éste método puede ser muy lento cuando el servidor tiene miles de entradas. Por eso es más práctico el siguiente método.

1.4.3.2 Backup en "frio"

Este es el método de backup más utilizado. Se basa en programar un script para que ejecute a una determinada hora. Los comandos que se deben detallar en el script son los siguientes;

- Detener el servidor de LDAP (comando service ldap stop).
- Hacer una copia de la base de datos y la configuración (comando slapcat).
- Comprimir la copia y dejarla en un punto accesible por los programas de backup
- Iniciar el servidor LDAP y comprobar que todo funciona correctamente (comando service ldap start).

Para realizar la recuperación de los datos respaldados en un servidor destino se debe de:

- Tener instalado la misma versión de OpenLDAP con la que se hizo el respaldo.
- Copiar la base de datos y el archivo de configuración del respaldo realizado.
- Iniciar el servidor LDAP (comando service ldap start).

1.4.4 Configuración de las Replicas en ASSE

Como arquitectura de replica se optó por Maestro-Esclavo. Dicha arquitectura evita los problemas de concurrencias e inconsistencia de datos que se puedan dar en la arquitectura Multi-Maestro. Para recuperarse del defecto que tiene la arquitectura maestro-esclavo de un único punto de falla, se recomienda tener una política de respaldo la cual permita sobrellevar cualquier inconveniente en el servidor Maestro. Respecto al otro inconveniente planteado de tener un único servidor para las escrituras (maestro) y otros para la lectura (esclavos), y teniendo en cuenta el número de escrituras (alta, baja, modificación de usuarios y contraseñas) no consideramos un inconveniente el tener un único servidor maestro que realice estas operaciones.

Para la política de respaldo se recomienda realizar respaldos totales diarios (totalidad del contenido del directorio LDAP) al finalizar el día laboral, el tipo de respaldo que proponemos usar es "En frío". Las operaciones de escritura (nuevos usuarios o actualizaciones de contraseñas) en el master se realizan en horario laboral, viendo así viable bajar el servidor LDAP una vez al día fuera del horario administrativo. Los cambios que se realicen posterior al último respaldo diario full no podrán ser recuperados. Para solucionar esto se podrían realizar respaldos incrementales durante el día en caliente. Éstos respaldos incrementales aplicarían, si el número de escrituras diarias así lo requieran.

En la configuración propuesta los distintos centros podrán conectarse a su servidor local (esclavo) para obtener datos confiables. Si se estuvieran haciendo cambios en los datos, éstos se realizarían en el servidor maestro, el cual posterior al cambio envía la actualización correspondiente a los servidores locales (esclavos) para mantenerlos en sincronía.

Se destaca el beneficio que brinda ésta configuración, la cual permite que los datos estén siempre disponibles en un menor tiempo. Todas las consultas LDAP (la mayoría de lectura) realizadas por el usuario se realizan al servidor local, que es sustancialmente más rápido y siempre (o con mayor probabilidad) visible.

Para llevar adelante la distribución de datos se configuran los servidores descentralizados como esclavos del servidor maestro el cual se encuentra instalado en ASSE, la replica de datos hacia los esclavos se hace de forma automática a través del servicio syncrepl previamente configurado con las necesidades de ASSE.

En el servidor central (master, maestro o proveedor) se permiten realizar todas las escrituras de datos, estos datos serán replicados automáticamente a todos los servidores esclavos (también llamados replicas o consumidores). Se sugiere utilizar el método de sincronización RefreshOnly, debido a que las escrituras no se dan continuamente no vemos la necesidad de mantener la conexión abierta desaprovechando el ancho de banda.

Los servidores esclavos tendrán toda la información del maestro evitando así tener problemas de acceso a datos si en algún momento se interrumpe la comunicación entre el servidor central y estos sub-sistemas descentralizados. Estos sistemas distribuidos geográficamente tomaran los datos del servidor LDAP local, teniendo los datos siempre actualizados del servidor maestro.

1.4.4.1 Archivos de Configuración

Para que se lleven adelante las replicas se utiliza Syncrepl (LDAP Sync Replication Engine – Motor de replica sincronizada para LDAP). Este motor se utiliza a partir de la versión 2.4 de OpenLDAP.

1.4.4.1.1 Configuración Maestro

En el maestro se agregan las siguientes lineas en el archivo slapd.conf

```
moduleload syncprov
overlay syncprov
syncprov-checkpoint 100 10
syncprov-sessionlog 100
```

En la Table 32 se expone las directivas utilizadas en la configuración del maestro..

Directivas	Descripción
moduleload syncprov	Indica que cargue el modulo syncrprov.
overlay syncprov	Indica que el DIT será el maestro.
syncprov-checkpoint <ops> <minutes>	Indica las condiciones para generar un checkpoint. En nuestra configuración se crea un checkpoint después de haber tenido 100 operaciones de escritura satisfactorias o si pasaron 10 minutos del último checkpoint [18].
syncprov-sessionlog <size>	Especifica el número de operaciones que son grabadas en el log, en el log de sesión se almacena todas las operaciones salvo los adds [19].

Table 32: Directivas de la configuración del archivo slapd.conf del servidor LDAP Maestro

1.4.4.1.2 Configuración Esclavo

Antes de iniciar el servicio LDAP esclavo se debe eliminar todos los datos que éste contenga y luego de realizar los cambios de configuración que se detallan a continuación, reiniciar el servicio de LDAP.

En el servidor esclavo se agregan las siguientes lineas al archivo slapd.conf

```
syncrepl rid=123
        provider=ldap://10.202.152.14:389
        type=refreshOnly
        interval= 00:00:10:00
        retry="120 20 300 +"
        searchbase="dc=example,dc=com"
        schemachecking=off
        bindmethod=simple
        binddn="cn=admin,dc=example,dc=com"
        credentials=ldapadmin
```

En laTable 33 se exponene los parametros de la directiva syncrepl.

Parametros de syncrepl	Descripción
rid (Replica Identifier)	Número de tres dígitos identifica de manera unica a cada uno de los esclavos que van contra un mismo proveedor (master). El master utiliza este dato para identificar al esclavo, y realizar un seguimiento de los servidores que estan conectados con él.
provider	Contiene la url LDAP del maestro.
type	Modo de replicación utilizado por el esclavo para conectar con el maestro (valores válidos: refreshOnly - modo polling que saca periodicamente las actualizaciones del proveedor, y refreshAndPersist – modo push que solicita al proveedor para inpulsar cambios). Si se utiliza el parámetro refreshAndPersist el parámetro interval será ignorado y los datos seran transmitidos inmediatamente despues de detectar un cambio en el maestro. Si usamos refreshOnly los cambios se replicaran al servidor esclavo en los intervalos definidos en la configuración (parámetro interval).
interval	Solo aplica si el type es refreshOnly indica el tiempo que tardará el servidor esclavo entre cada proceso de sincronización, el esclavo se conectará al servidor y verificará si hay actualizaciones, luego se desconecta, esperarando el tiempo indicado en interval para comprobar nuevamente. La sintaxis del parámetro intervalo es: dd:hh:mm:ss, en nuestro caso se propone acualizarse cada 00:00:10:00 (10 minutos).
retry	se utiliza en caso que el servidor maestro no esté disponible al momento que se tiene que dar lugar la sincronización (debido a un fallo en el servidor o demasiada carga en la red), En la configuración propuesta se agregarón dos elementos adicinoales <intervalo> <intento> el 1ero. indica el timpo en segundos que espera para intentar re-conectarse, y el 2do. Indica que cantidad de veces realiza dicha acción. Si se desea que se intente indefinidamente se puede agregar otro par de elementos adicionales a los dos expuestos <intervalo> <+>, resulando en <intervalo> <intento> <intervalo> <+>
searchbase	Directivas para indicar desde donde (estructuralmente hablando) vamos a sincronizar, nosotros vamos a sincronizar todo el arbol por eso searchbase coincide con el RootDN.
scope	Es igual a sub (la busqueda es en profundidad), el valor del filtro es filter=(objectclass=*) y el attrs="*,+", ser recuperan todos los objetos y todos los atributos.
binddn y **credentials**	El usuario y el password con el que se conecta al SLAPD master. El uid=admin tiene que estar creado en el maestro.

Table 33: Parametros de la directiva syncrepl en el servidor Esclavo

1.5 Pruebas Controlador Dominio con Sama y OpenLdap

Respecto al controlador de dominio con Samba y OpenLdap, la primer prueba que se realizó es la de chequear el acceso al directorio LDAP y realizar una busqueda en el DIT para obtener una respuesta del servidor. Se ejecuta la sentencia que se detalla abajo (en el password solicitado se ingresa *ldapadmin*).

ldapsearch -x -D "cn=admin,dc=example,dc=com" -b "dc=example,dc=com" -W

El resultado de ésta busqueda es todo el contenido del DIT, la respuesta del servidor es en formato LDIF y se expone a continuación, tener en cuenta que por motivos del tamaño del archivo se truncó el resultado a la primeras líneas del archivo. Para poder ver completo el archivo se debiera replicar la sentencia en una consola en el servidor LDAP. Ver Figura 6.

```
# extended LDIF
#
# LDAPv3
# base <dc=example,dc=com> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# example.com
dn: dc=example,dc=com
objectClass: dcObject
objectClass: organization
o: example
dc: example

# Users, example.com
dn: ou=Users,dc=example,dc=com
objectClass: top
objectClass: organizationalUnit
ou: Users

# Groups, example.com
dn: ou=Groups,dc=example,dc=com
objectClass: top
objectClass: organizationalUnit
ou: Groups

# Computers, example.com
dn: ou=Computers,dc=example,dc=com
objectClass: top
objectClass: organizationalUnit
ou: Computers

# Idmap, example.com
dn: ou=Idmap,dc=example,dc=com
objectClass: top
objectClass: organizationalUnit
ou: Idmap
```

```
# root, Users, example.com
dn: uid=root,ou=Users,dc=example,dc=com
cn: root
sn: root
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: sambaSamAccount
objectClass: posixAccount
objectClass: shadowAccount
gidNumber: 0
uid: root
uidNumber: 0
sambaLogonTime: 0
sambaLogoffTime: 2147483647
sambaKickoffTime: 2147483647
sambaPwdCanChange: 0
sambaPrimaryGroupSID: S-1-5-21-3190323790-1237239473-3967027434-512
sambaSID: S-1-5-21-3190323790-1237239473-3967027434-500
gecos: Netbios Domain Administrator
sambaLMPassword: 1C033863603E740EF0D412BD764FFE81
sambaAcctFlags: [U]
sambaNTPassword: 85DC2CE0A9C3638ECB17B54CF1113443
sambaPwdLastSet: 1314208561
sambaPwdMustChange: 9954122161
userPassword:: e1NTSEF9cmIvbTl4SEhGTklFSWNqNFNiOEdlVkpEMW1rMll6QXc=
shadowLastChange: 15210
shadowMax: 99999
homeDirectory: /root
loginShell: /bin/bash
```

Figura 6: LDIF devuelto por el servidor LDAP, con la estructura del DIT

Continuando con las pruebas se chequeó que al crear un nuevo usuario, no sea ingresado en los archivos de autenticación que utiliza un sistema Unix/Linux (/etc/password, /etc/shadow y /etc/group) y que los usuarios estén dados de alta en el directorio LDAP.

Para crear usuarios en un sistema con LDAP no se usan mas las sentencias *useradd* para crear un usuario y *passwd* para asignarle una contraseña. En su lugar se debe de ejecutar el comando *smbldap-useradd* el cual es provisto por las herramientas smbldap instaladas en el prototipo.

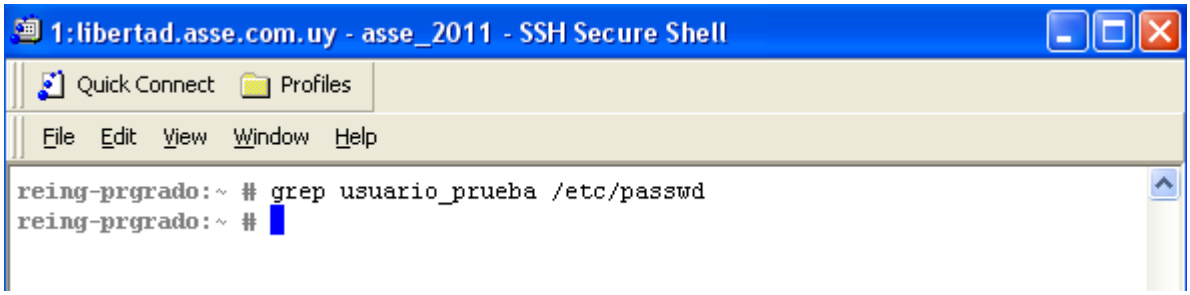

Figura 7: Probar inexistencia del usuario "usuario_prueba" en passwd

Se crea el usuario de nombre “usuario_prueba” con contraseña “prueba”. Antes de esto se evalúa que no exista registro del usuario a crear denominado “usuario_prueba” para ésto ejecutamos el comando grep sobre el archivo passwd. VerFigura 7.

Para chequear que no existe el usuario en nuestro directorio LDAP, ejecutamos la siguiente búsqueda:

ldapsearch -x -D "cn=admin,dc=example,dc=com" -b "uid=usuario_prueba, ou=Users,dc=example, dc=com" -W

El servidor al ejecutar la búsqueda detalla el resultado de la existencia del usuario en la rama Users del DIT o indica que no encontró el objeto. VerFigura 8.

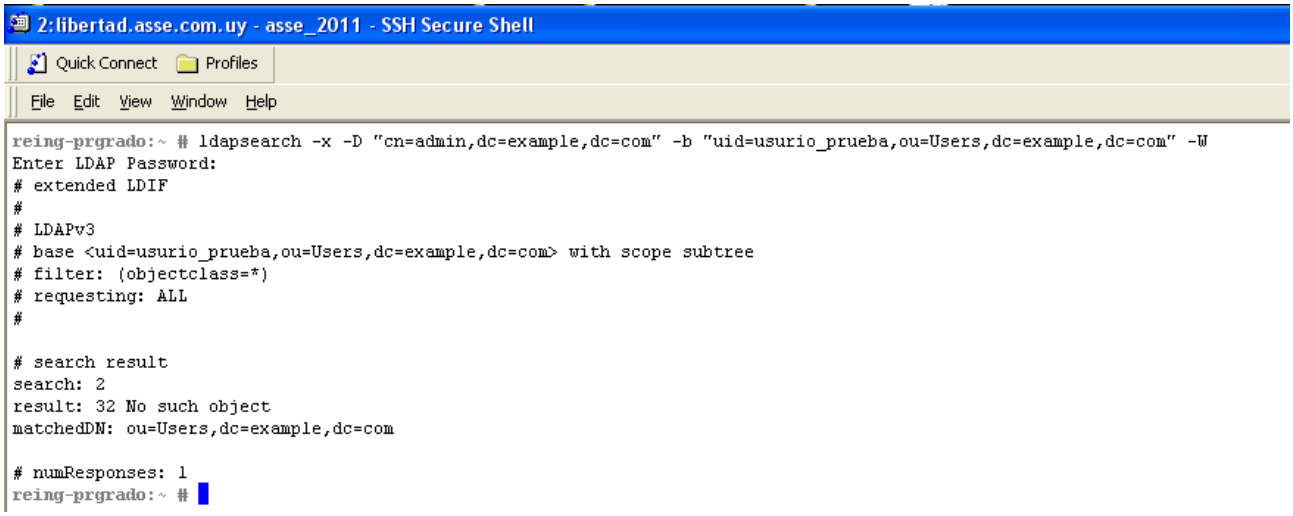

Figura 8: Probar inexistencia del usuario "usuario_prueba" en el DIT

Como se muestra en la imagen el objeto no fue encontrado, indicando como resultado “No such object”.

A continuación se creó el usuario con la sentencia siguiente. El password ingresado es *prueba*. **smbldap-useradd -m -s /bin/bash -P usuario_prueba.** Ver Figura 9.

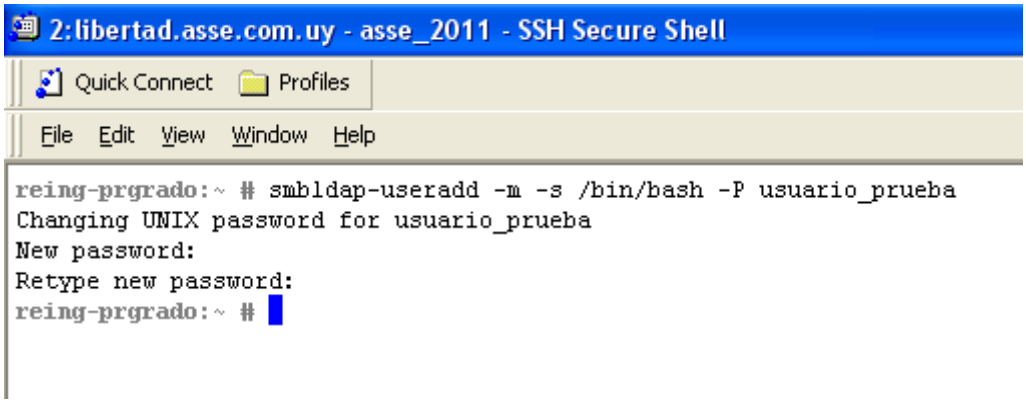

Figura 9: Comando para crear el usuario de nombre "usuario_prueba"

Como se puede ver en las siguiente pantalla el usuario no existe registro del usuario creado en el archivo passwd. Ver Figura 10.

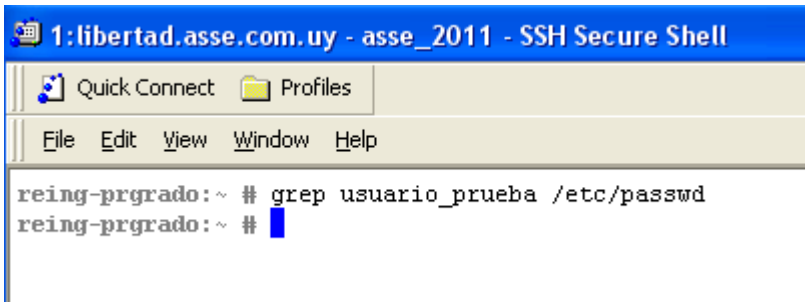

Figura 10: Probar inexistencia del usuario "usuario_prueba" en el archivo passwd

Se consulta el directorio LDAP con el comando ldapsearch y se comprueba que el usuario se ingresó correctamente en el DIT. Ver Figura 11.

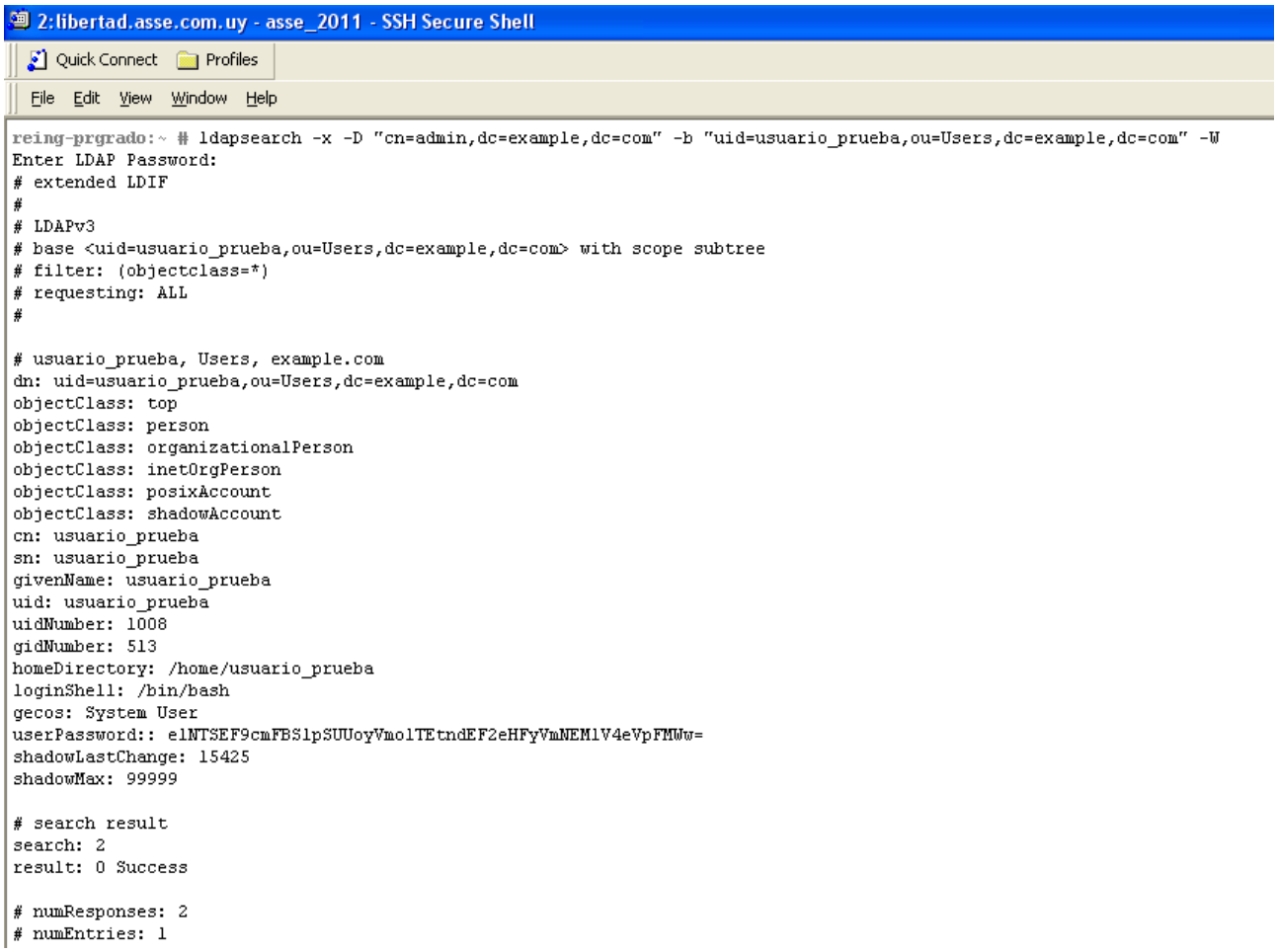

Figura 11: Probar existencia del usuario "usuario_prueba" en el directorio LDAP

A continuación se prueba la resolución de identidades con el comando id. Ver Figura 12.

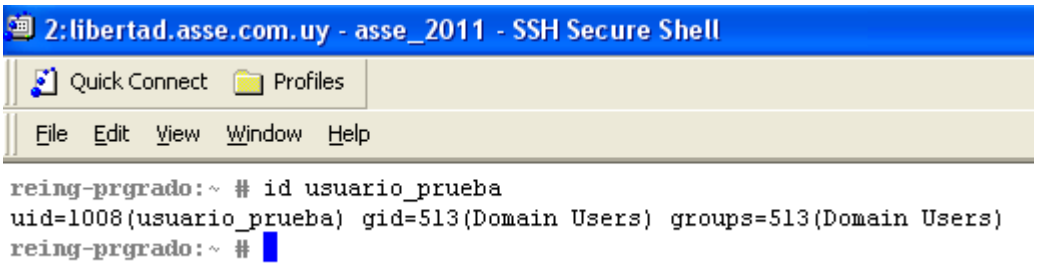

Figura 12: Prueba de resolución de identidades

Una vez dado de alta el usuario y comprobado su correcta configuración se prueba realizar un logueo con dicho usuario en una sesión SSH.

Se puede ver que el usuario trabaja en el directorio home establecido en la configuración del servidor Samba y que está resolviendo correctamente las identidades UID y GID con los del LDAP. Ver Figura 13

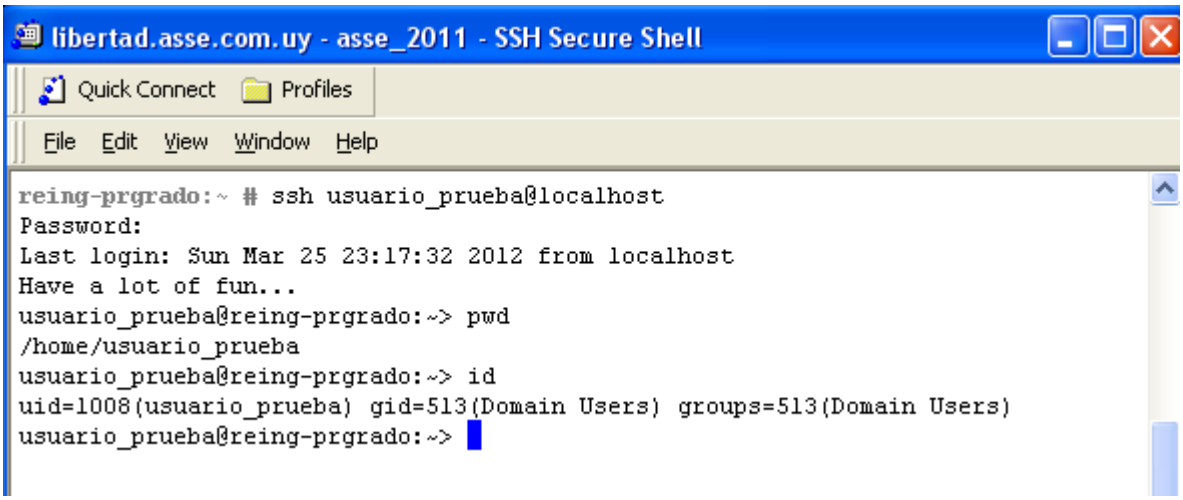

Figura 13: Prueba de logueo usando ssh y el usuario "usuario_prueba"

1.6 Pruebas Réplicas LDAP

Respecto a la replicación del contenido del directorio LDAP, se optó por una arquitectura de un solo maestro y varios esclavos. Se probó como método de sincronización del esclavo las dos formas que existen refreshOnly y refreshAndPersist.

Se expone el detalle de la configuración y las pruebas realizadas según el método de sincronización.

1.6.1 Configuración del Maestro

La configuración del maestro es igual para ambos modos de sincronización, se detalla en la Figura 14.

1.6.2 Configuración del Esclavo RefreshAndPersist

La configuración del esclavo para el modo de sincronización refreshAndPersist es el detallado en la Figura 15.

A modo de ejemplo, se ejecuta una inserción en el servidor master del "usuario_replica", verificando que dicho cambio se replica en el servidor esclavo.

Se valida que no existe el usuario a crear "usuario_replica" en el servidor esclavo, utilizando el comando ldapsearch. Ver Figura 16.

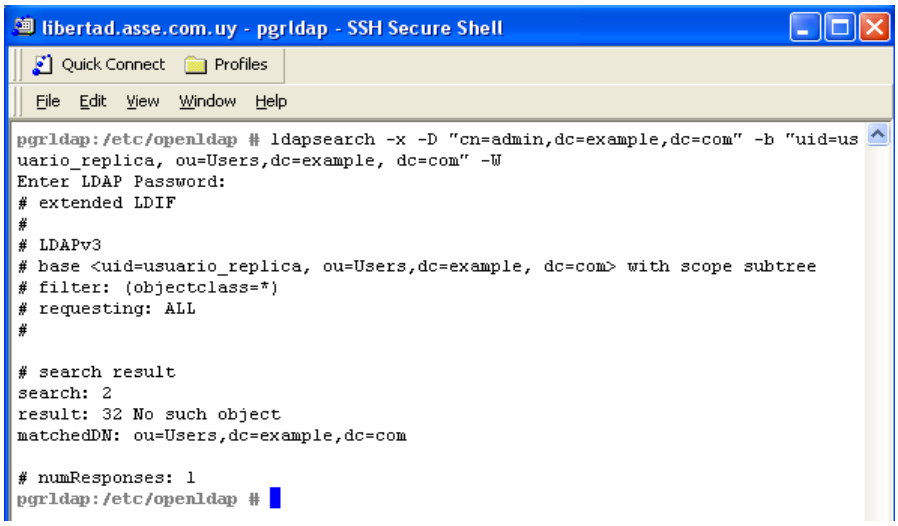

Figura 16: Prueba de la inexistencia del usuario "usuario_replica" en el servidor Esclavo

Se inserta el usuario “usuario_replica”, en el servidor maestro, utilizando el comando smbldap-useradd. Ver Figura 17.

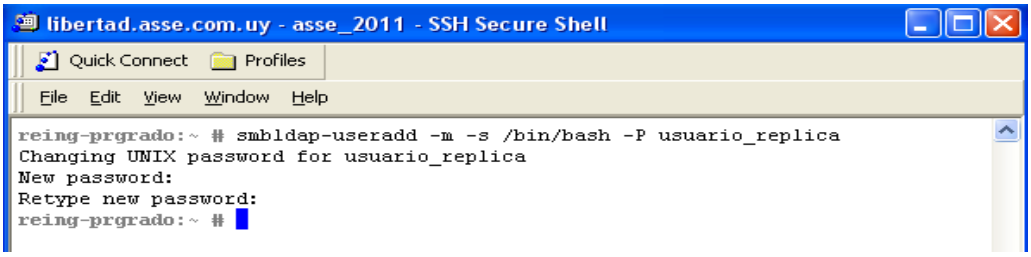

Figura 17: Comando para agregar el usuario "usuario_replica" en el Servidor LDAP Maestro.

A continuación, se busca en el servidor LDAP Esclavo el usuario “usuario_replica”, comprobando que el servidor esclavo se sincronizó con el servidor. Ver Figura 18

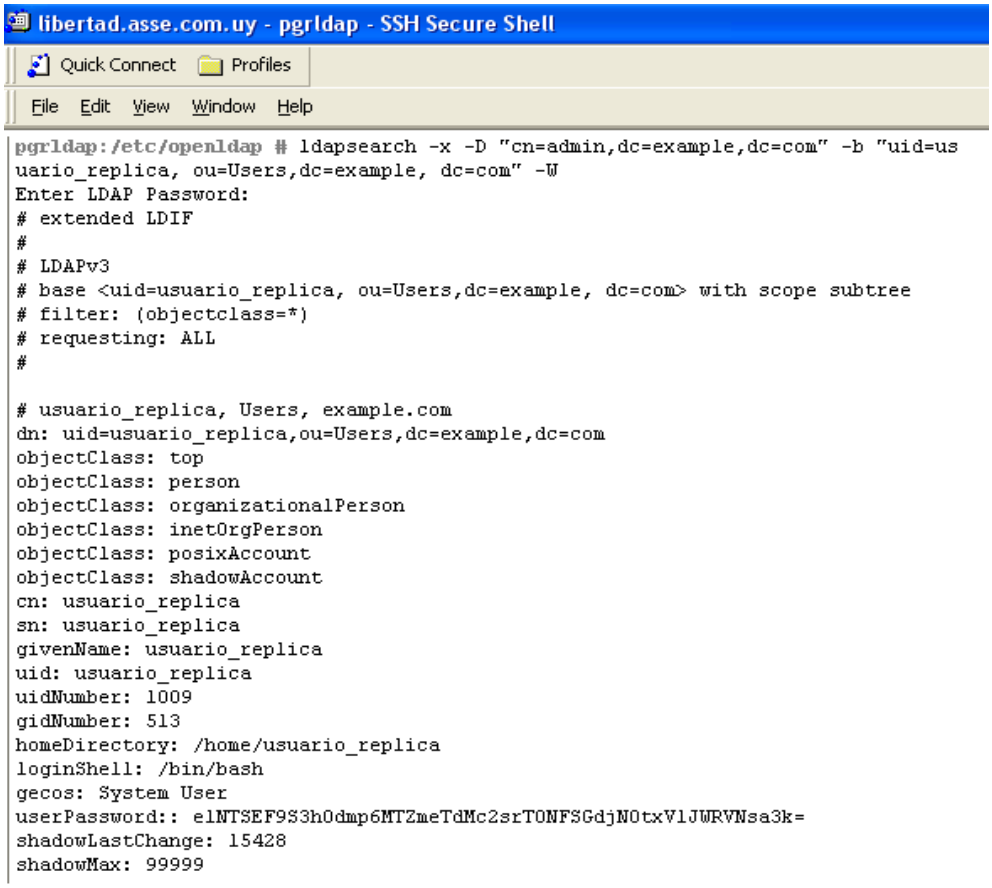

Figura 18: Buscar el usuario "usurario_replica" en el directorio LDAP Esclavo

1.6.3 Configuración del Esclavo RefreshOnly

La configuración del esclavo para el modo de sincronización refreshOnly es el detallado en la Figura 19.

A modo de prueba del método de replicación, se ejecuta una inserción en el servidor LDAP maestro, verificando que dicho cambio se replica en el servidor esclavo.

Se inserta el usuario “usuario_replica2”, en el servidor maestro. Utilizando el comando smbldap-useradd. Ver Figura 20.

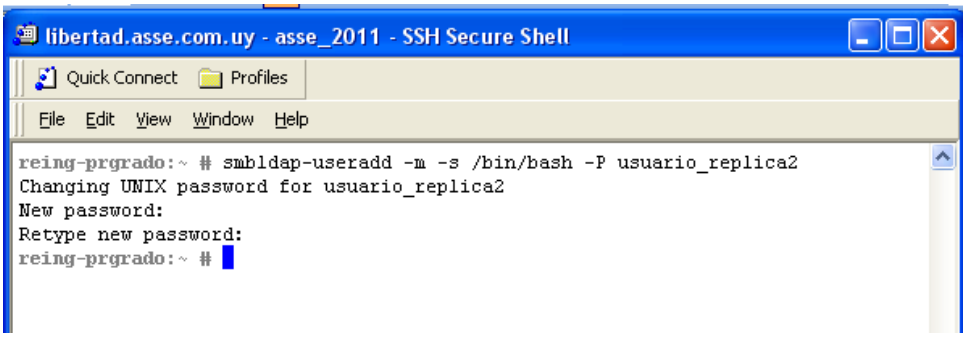

Figura 20: Comando para insertar el usuario usuario_replica2 en el servidor LDAP Maestro

Se consulta en el servidor esclavo que el usuario fue sincronizado desde el servidor maestro, pasado el tiempo establecido en la configuración (parámetro interval).Ver Figura 21.

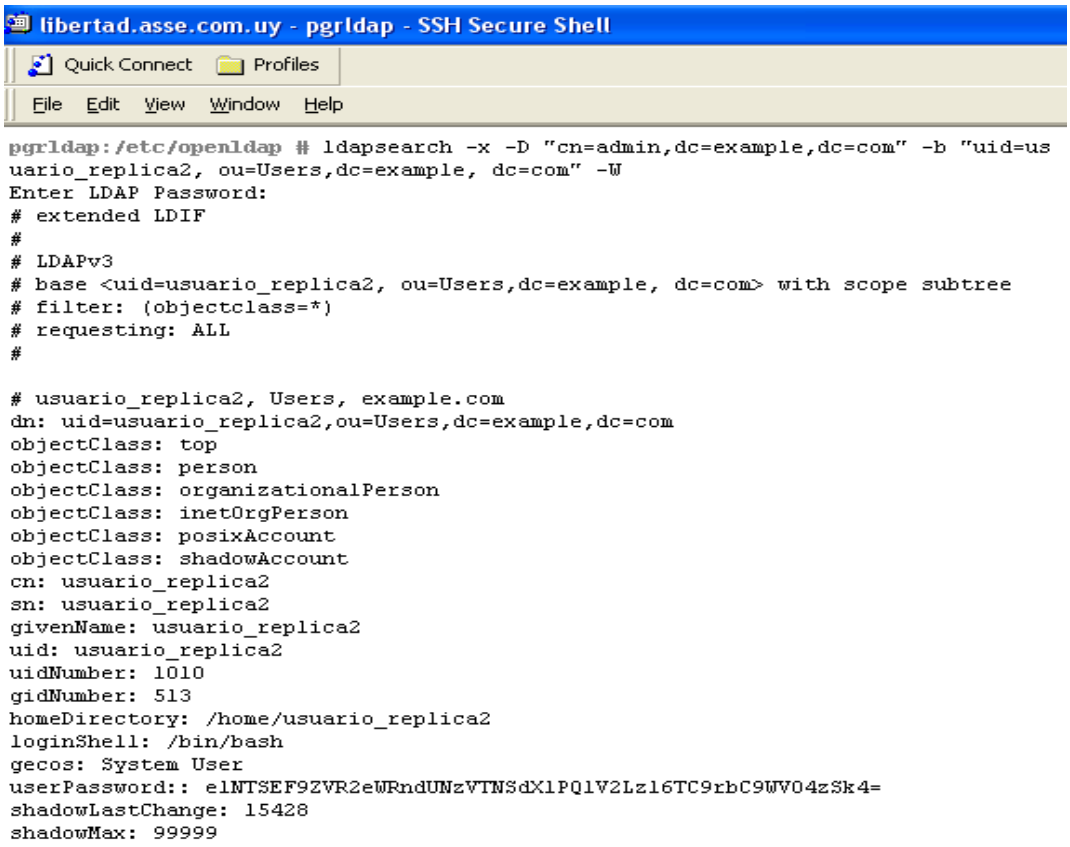

Figura 21: Comando para chequear que existe el usuario: "usuario_replica2" en el servidor LDAP Esclavo

1.7 APÉNDICE LDAP

LDAP ("Lightweight Directory Acces Protocol", en español Protocolo Ligero de Acceso a Directorios) es un protocolo de tipo cliente-servidor para acceder a un servicio de directorio.

Un directorio es una base de datos, pero contiene mas información descriptiva y basada en atributos. En un directorio se lee mas de lo que se escribe. En un directorio las modificaciones son escasas, por tanto no se requieren grandes esquemas para transacciones, pues no se manejan actualizaciones de grandes volúmenes. El objetivo de un repositorio es retornar una respuesta rápida a operaciones de búsquedas o consultas.

Un directorio LDAP, no es una base de datos relacional, no está preparado para procesar miles o cientos de transacciones de cambios por minuto como lo está una base de datos relacional. El directorio LDAP es eficiente en las lecturas de datos.

La Figura 22 detalla la interacción entre el cliente y el servidor LDAP

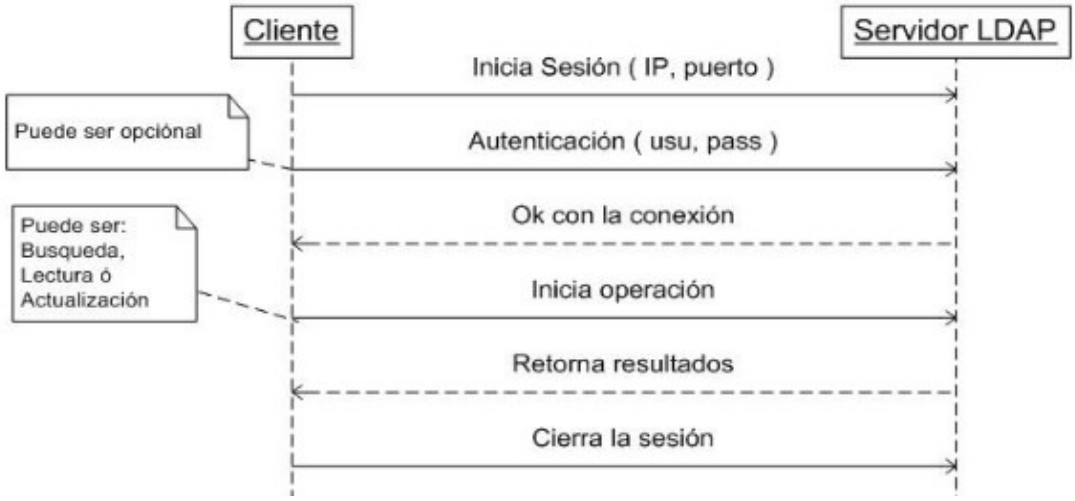

Figura 22: Interacción entre el cliente y el servidor LDAP

Al tratarse de una arquitectura cliente- servidor una aplicación que intenta acceder a un directorio de servicios no accede directamente a la base de datos, sino que invoca a una función de la API, quien se comunica con el servidor, dicho proceso es el que se comunica con el directorio y retorna a la operación el dato.

Puede darse el caso que un servidor actúe como cliente de otro Servidor para conseguir la información, a esto se le denomina servicio de directorio descentralizado, es decir los datos pueden estar fraccionados y/o replicados en varios servidores. Si la información se fracciona, cada servidor de directorio contiene un subconjunto único y no solapado de información, en caso de estar replicada la información se obtienen varias copias de la información en cada servidor. La Figura 23 detalla la interacción de un servicio de directorio descentralizado y no descentralizado.

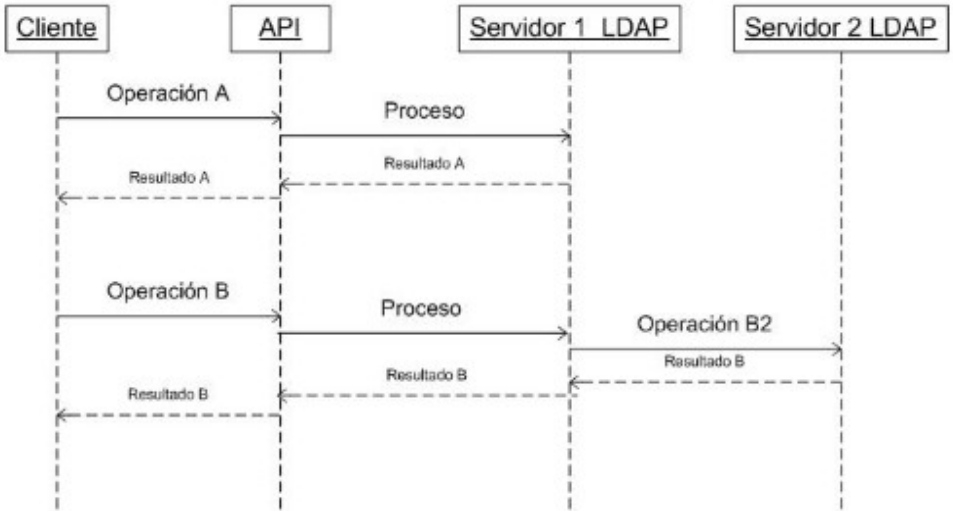

Figura 23: Servicio de directorio descentralizado y no descentralizado

1.7.1 Modelados de LDAP

LDAP además de ser un protocolo, ofrece el servicio de directorio, para entenderlo mejor se detallan los 4 modelos que maneja el estándar.

1. ***Modelo de información***, describe la estructura de la información que se almacena en el directorio.
2. ***Modelo de nombrado***, describe como se referencia y organiza la información en el directorio
3. ***Modelo funcional***, describe que operaciones se pueden realizarse sobre la información almacenada en el directorio.
4. ***Modelo de seguridad***, describe como se puede proteger la información almacenada en el directorio frente a accesos no autorizados.

1.7.1.1 Modelo de Información

El directorio organiza y mantiene la información en estructuras de datos llamadas entradas. Cada entrada describe a un objeto y tiene un nombre llamado *Distinguished Name* (DN). Cada DN consiste en una secuencia de partes más pequeñas llamadas *Relative Distinguished Name* (RDN), cada RDN se corresponde con una rama del DIT partiendo de la raíz hacia la entrada dentro del directorio.

La información que se almacena en el directorio se hace de forma jerárquica, formando el árbol de directorio (DIT) el cual organiza las entradas en forma de árbol basándose en los DN.

A continuación se detallan algunos conceptos a tener en cuenta:

- ***Esquema:*** Define las clases de objetos que se pueden almacenar en el directorio.
- ***Clase de objeto***: Es una descripción general de un tipo de objeto. Cada clase de objeto define los atributos que contiene, los atributos opcionales, los formatos de los atributos y define las subclases de objetos y en que puntos del DIT pueden aparecer.
- ***Entrada.*** Se compone de un conjunto de atributos, cada uno tiene un tipo y uno o varios valores.

Una entrada puede pertenecer a más de una clase de objetos. Por ejemplo, la entrada para personas se define mediante la clase de objetos person, pero también puede definirse mediante atributos en las clases de objetos inetOrgPerson, groupOfNames y organization.

- ***Tipo del atributo***. Define restricciones (como ser el tamaño total del atributo), la sintaxis que describe el tipo de información que proporciona el atributo y además define como se realizará las comparaciones.

Como ejemplo de sintaxis se expone la Tabla 34 la cual enumera una lista de atributos y se detalla su sintaxis.

Atributo	Sintaxis
tel	Número de teléfono, tratado como texto, pero se ignoran los espacios en blanco y guiones
bin	Información binaria
ces	Cadena con mayúsculas y minúsculas exactas (las mayúsculas y minúsculas son significativas durante las comparaciones)
cis	Cadena con mayúsculas y minúsculas ignoradas (las mayúsculas y minúsculas no son significativas durante las comparaciones)
dn	"distinguished name" (nombre distintivo)

Tabla 34: Sintaxis de atributos de LDAP.

Los tipos de atributos en el directorio forman un árbol de clases. Se puede ver en la Figura 24 un ejemplo de esto, en donde el tipo de atributo "commonName" es una subclase del tipo de atributo "name”

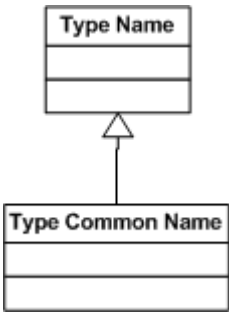

Figura 24: Ejemplo de tipo de atributo en el directorio.

La Figura 25 expone un DIT en donde detalla el contenido de una entrada

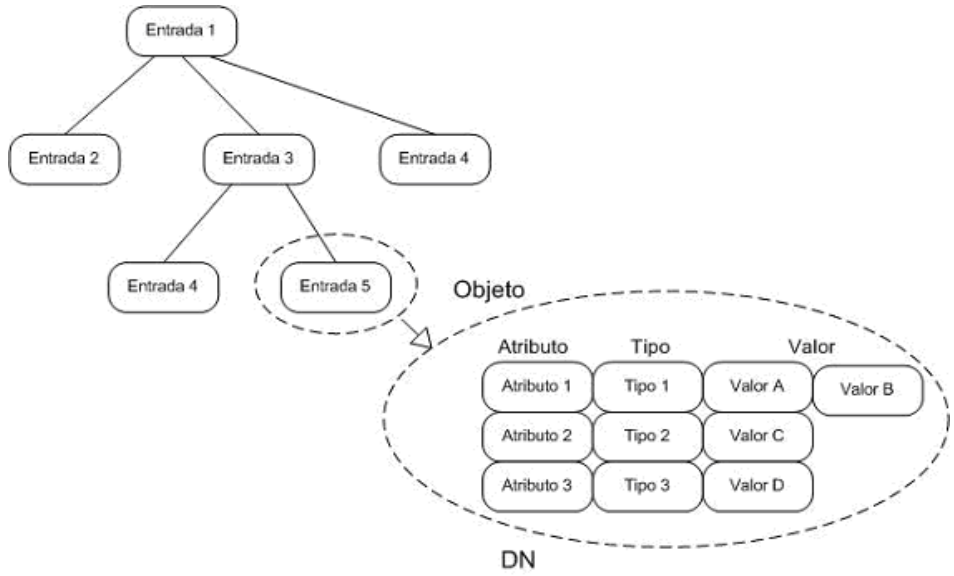

Figura 25: Ejemplo de un DIT con detalle de la entrada 5.

1.7.1.2 Modelado de Nombrado

Como se mencionó antes las entradas son organizadas dentro del DIT en base a su DN, y cada DN son secuencias de RDNs y a su vez cada RDN se corresponde con una rama del DIT partiendo de la raíz hasta llegar a la entrada dentro del directorio. De lo expuesto se deduce que la entrada raíz del DIT es el único que no tiene entrada padre.

El diseño del DIT es flexible, se puede crear un grupo que contenga a todos los usuarios y otro con los grupos que existen en la compañía, ó se puede optar por una estructura que respete la estructura jerárquica de la compañía.

La Figura 26 ejemplifica la jerarquía del DIT y se compara con el servicio de nombre de dominio (DNS), y la estructura de un sistema de archivo.

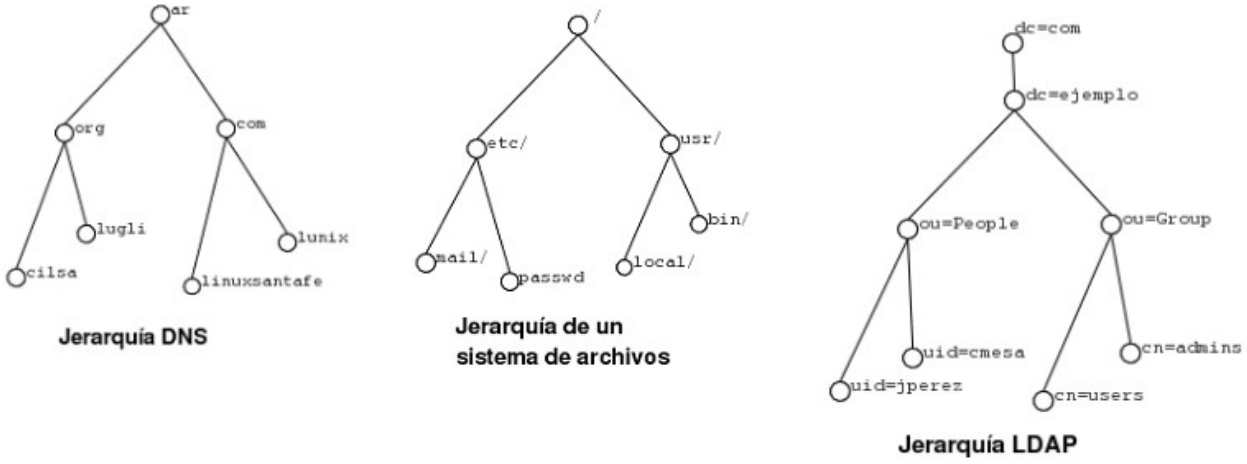

Figura 26: Comparación entre estructuras jerárquicas y LDAP

En el caso de la estructura de la jerarquía LDAP, ou se refiere a Unidad Organizativa y como ejemplo se puede inidcar que el DN del usuario jperez es uid=jperez ou=People dc=ejemplo dc=com.

LDAP permite el uso de alias. Los alias son similares a los accesos directos en windows o un enlace simbólico en unix. Los alias son útiles para enlazar una entrada hacia otra que por la estructura en si no sería facil relacionar.
El uso de alias puede ocasionar problemas de rendimiento, debido a que apunta a una entrada cualquiera sin importar en que servidor LDAP se encuentra. Puede ocurrir que el tiempo en resolver una búsqueda lleve mas tiempo de lo estipulado.
El objetivo del alias se puede conseguir también con el uso de referral el cual se usa cuando tenemos un directorio distribuido (es decir no todo el DIT se encuentra en el mismo servidor LDAP). El referral permite integrar el directorio formado por partes de distintos servidores LDAP.

A modo de ejemplo del uso del Referral, se puede ver la Figura 27 en donde People se encuentra en un servidor y Group en otro servidor.

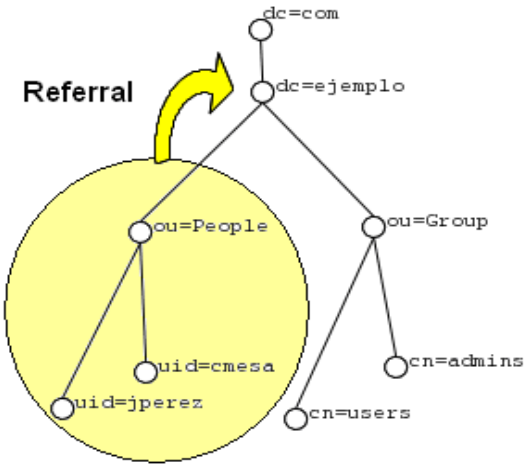

Figura 27: Ejemplo de uso de Referral

El API de LDAP permite especificar si se desea que se devuelvan las entradas de tipo referral o no. Éste tipo de entradas permite particionar y distribuir el servicio de directorio entre varios servidores, aumentando el rendimiento he incluso la tolerancia a fallas.

A continuación en se detalla Table 35 con los atributos LDAP mas utilizados[20].

Abreviatura	Nombre Atributo	Clase de Objeto	Descripción
c	countryName	Country	2 caracteres del código del país definido en la ISO 3166
cn	commonName	person organizationalPerson organizationalRole grupOfNames applicationProcess aplicationEntity posixAccount device	
dc	domainComponent	dcObject	Cualquier parte de un nombre de dominio
-	facsimileTelephoneNumber	residentialPerson organizationalRole organizationalPerson	
co	friendlyCountryName	friendlyCounty	Nombre completo del país
gn	givenName	inetOrgPerson	
homePhone	homeTelephoneNumber	inetOrgPerson	
-	jpegPhoto	inetOrgPerson	Foto en formato jpg
-	localityName	locality organizationalPerson	
mail	rfc822Mailbox	inetOrgPerson	Dirección de e-mail
o	organizationName	organization	Nombre de la organization
ou	organisationalUnitName	organizationUnit	Nombre de la organización en general
-	owner	groupOfNames device grouOfUniqueNames	
pager	pagerTelephoneNumber	inetOrgPerson	
-	postalAddress	organizationalPerson	
postalCode	postalCode	organizationalPerson	Codigo postal
sn	surname	person	Apellido
st	stateOrProvinceName	organizationalPerson	
street	streetAddress	organizationalPerson	
-	telephoneNumber	organizationalPerson	
userPassword	-	organization organizationalUnit personalizacionesdmd simpleSecurityObject domain posixAccount	Contraseña de usuario de algún tipo de control de acceso
uid	userid	account inetOrgPerson posixAccount	Valor único. En general nombre de usuario.

Table 35: Atributos LDAP más usados

1.7.1.3 Modelo Funcional

Define el grupo de operaciones que se pueden realizar sobre la estructura y los datos almacenados en el directorio DIT.

Existen 3 tipos de operaciones que son soportadas por el directorio LDAP:

- ***Consulta***. Incluye búsquedas y recuperación.
- ***Actualización***. Incluye agregar, eliminar, modificar.
- ***Autenticación y Control.*** Incluye identificación usuarios y control de sesión.

Una vez instalado el servidor LDAP, se puede operar en la estructura del directorio, para ésto existen primitivas las cuales se detallan en la Table 36 :

Primitiva	Descripción
ldapsearch	Búsqueda, dado un criterio por el usuario
ldapadd	Agregar una entrada
ldapdelete	Eliminar una entrada
ldapmodify	Modificar una entrada
ldappasswd	Cambiar la contraseña de una entrada del directorio
bind	Permite autenticar al cliente frente al directorio (sesión anónima, sesión autenticada mediante texto en claro, sesiones autenticadas mediante claves cifradas)
unbind	Cierra la conexión con el servidor LDAP

Table 36: Primitivas LDAP

Los parámetros de las primitivas mostradas anteriormente son los siguientes:

-x: autenticación en modo simple
-c: no detenerse frente a errores
-f: leer la información de un archivo
-v: modo detallado
-k: usar la autenticación kerberos
-H: utilizar un determinado servidor ldap.
Un ejemplo puede ser : -H ldap: / / ldap.mycompany.com.
-W: preguntar por el password en el bind
-D: especifica el DN para el bind

1.7.1.4 Modelo de seguridad

Respecto a la seguridad en el directorio, algunos permiten el acceso público pero limitan el acceso a las operaciones.

Una política de seguridad, define **quien** tiene **que** tipo de accesos sobre **qué** información.

El directorio debe cubrir con las capacidades básicas para implementar la política de seguridad. Puede que el directorio propiamente no lo cumpla, pero debe estar ligado a un servicio de red fiable que proporcione los servicios básicos de seguridad. Los pasos serían:

- **Autenticar:** Identificando al usuario.
- **Autorizar:** Una vez que se identificó al usuario se chequea si tiene permisos para operar.

Para realizar la autorización, se usan las listas de control de acceso (ACL), en donde las listas unen los objetos y/o atributos contenidos en el directorio. En general para facilitar el manejo de listas lo que se hace es agrupar en grupos a los usuarios con los mismos permisos. Las ACL pueden controlar el acceso dependiendo de quien está solicitando los datos, que datos están siendo solicitados, dónde están los datos almacenados, todo ésto realizado directamente a través del directorio LDAP del lado del servidor, ésto permite deslindar al nivel de aplicación de usuarios de hacer comprobaciones de seguridad. Algunos ejemplos que se logra al usar ACL:

- Impedir que un usuario consulte la contraseña de otra persona.
- Permitir que un tipo de persona (por ejemplo un gerente) acceda al determinada información de otra persona (por ejemplo el número de teléfono).
- Crear restricciones de acceso según la ip.

Para configurar el control de acceso al directorio LDAP en el servidor, se agrega el mismo en una sección global del archivo slapd.conf que tiene una validez mientras no se especifiquen otras reglas de control de acceso, en la sección de las bases de datos. A continuación se detallan la sintaxis de las reglas que son necesarias para especificar el control de acceso:

```
access to <que> by <quien> <acceso>
```

<que> Representa al objeto o al atributo al cual se quiere definir el acceso. Se pueden restringir ramas enteras, o áreas entera del árbol (por medio de expresiones regulares). LDAP evalúa las reglas en el orden expuesto. Con lo que se requiere colocar primero la mas restrictiva a la menos.

<quien> Representa quien tiene acceso al objeto o atributo expuesto en el campo <que>. Se puede utilizar los valores detallados en la Table 37.

Identificador	Significado
*	Todos los usuarios.
anonymous	Usuarios anónimos (No autentificados)
users	Usuarios autentificados
self	Usuarios unidos al objeto destino
dn=<regex>	Todos los usuarios a los que puede aplicarse esta expresión regular

Table 37: Valores posibles para el parámetro "quien" en la configuración de acceso al DIT LDAP

<acceso> Representa el tipo de acceso. Se puede utilizar los valores detallados en la Table 38
.

Identificador	Significado
none	Acceso prohibido
auth	Para contactar con el servidor
compare	Para acceso comparables a objetos
search	Para utilizar filtros de búsqueda
read	Permiso de lectura
write	Permiso de escritura

Table 38: Valores posibles para el parámetro "acceso" en la configuración de acceso al DIT LDAP

Para ejemplificar el control de acceso, se detalla en la Figura 28 una configuración en donde todos los usuarios tienen permiso de lectura para el directorio y que el administrador (rootdn) es el único que tiene permisos de escritura.

Nota: Si no se puede aplicar ninguna regla el permiso será denegado. Sólo se conceden aquellos permisos autorizados explícitamente. En caso de no existir ninguna regla, se aplica el siguiente principio: permiso de escritura para el administrador y permiso de lectura para todos los demás.

A modo informativo la versión 2 de LDAP, solo permite sesiones anónimas y autenticación mediante texto en claro. A partir de la versión 3 de LDAP, se permite sesiones cifradas tiene soporte para SASL (Simple Authenticacion Security Layer) y Además se han definido operaciones extendidas como ser TSL (Extension for Transport Layer Security).

1.7.2 LDIF (formato de intercambio de datos)

Se refiere al estándar para representar entradas del directorio LDAP en formato de texto.
Se utiliza para importar y exportar información del directorio entre distintos servidores basados en LDAP, o para describir una serie de cambios que se impactarán en el directorio.

Un archivo LDIF almacena información en jerarquía de entradas orientado a objetos. Todos los servidores LDAP incluyen una utilidad para convertir archivos LDIF a formato orientado a objeto. Normalmente es un fichero ASCII.

La entrada en un archivo LDIF se representa por dos partes:

* El DN (nombre distinguido) el cual se debe especificar en la primera línea de la entrada.
* Los atributos de la entrada. Que se representa por el nombre del atributo seguido de dos puntos y el valor. Si existen atributos multivaluados se colocan seguidos.

Para ejemplificar el uso del archivo LDIF, se detalla la estructura de un DIT particular en la Figura 29, sobre el cual se explica a continuación el uso de las operaciones que soporta el formato LDIF..

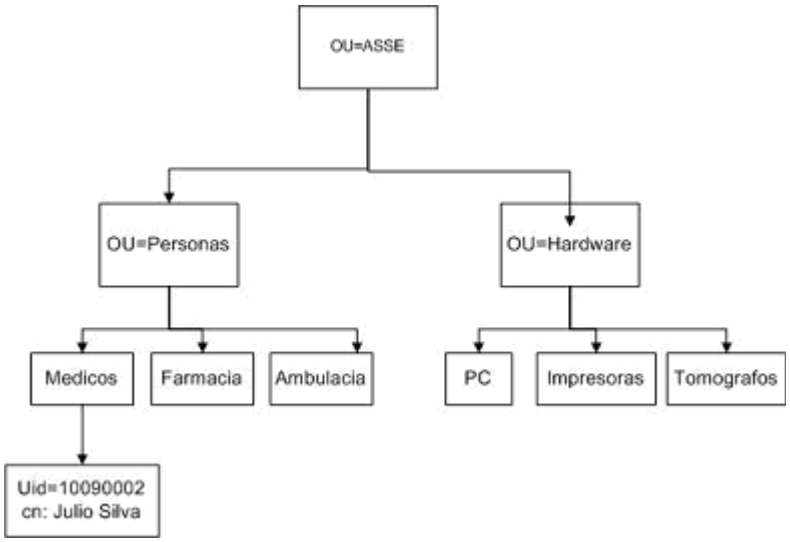

Figura 29: Ejemplo de DIT

Una entrada del DIT de la Figura 29en formato LDIF se representa como se indica en la Figura 30. Allí se expone las clases de objetos contenidos en el DIT, así como también los atributos que se almacenan en una entrada.

Si se quisiera representar el valor de un atributo que no es ASCII, como puede ser una imagen. Se representa en codificación base64.

1.7.2.1 Sintaxis de operaciones en formato LDIF

Con el formato LDIF se pueden hacer operaciones sobre el directorio LDAP como ser actualización, inserción y borrado de entradas.

Los puntos a tener en cuenta para definir la estructura del archivo LDIF son:

- Cualquier línea que comience con el carácter # se considera un comentario y es ignorada cuando se procesa el archivo.
- La primera línea no comentada del archivo debe ser el número de versión.
- Luego de la versión hay un conjunto de uno o más registros.
- Cada registro está compuesto por campos y hay un único campo por línea.
- Cada línea está separada por un fin de línea.
- Los registros están separados por una o más líneas en blanco dentro del archivo.

El formato LDIF para cada registro es el expuesto en la Figura 31:

Para insertar una nueva entrada al directorio LDAP, se debe de respetar el formato según se muestra en la Figura 32.

Para eliminar una entrada del directorio LDAP , se debe de respetar el formato según expresa la Figura 33.

Para Modificar una entrada en el directorio LDAP, se debe de respetar el formato según se muestra en la Figura 34.

(*) TipoCambio puede ser: add, delete, replace.

1.7.2.2 Ejemplos de como operar con archivos en formato LDAP

Teniendo en cuenta la estructura del DIT detallada en la Figura 29 se expone a continuación ejemplos del archivo LDIF, que permiten realizar operaraciones sobre el directorio LDAP.

Eliminar la entrada identificada por el uid=:10090002 la cual pertenece al grupo de los médico. Ver Figura 35.

Agregar un nuevo médico con uid:111111111 al directorio LDAP, ver Figura 36.

Agregar el atributo “telephoneNumber” a la entrada identificada con el uid:111111111 la cual ya existente en el directorio LDAP, ver Figura 37.

Eliminar todos los valores del atributo “telephoneNumber” de la entrada identificada con uid:111111111, ver Figura 38.

Eliminar el valore 099877665 del atributo “telephoneNumber” de la entrada identificada con el uid:111111111, ver Figura 39.

Modificar el “telephoneNumber” de la entrada identificada por el uid:111111111 con el valor 099123456, ver Figura 40.

Realizar varias operaciones sobre el directorio LDAP en un mismo archivo LDIF, para realizar esto se debe separar cada sentencia por un guión, como se muestra en la Figura 41 en donde se borra el valor del atributo “telephoneNumber” se agrega el atributo mail y se borra los valores del atributo description de la entrada identificada por el uid:111111111.

```
dn: uid=11111111,ou=Medicos,ou=Personas,ou=ASSE
        changetype: modify
delete: telephoneNumber
telephoneNumber: 916249500
-
add: mail
mail: pepe@pepe.com
-
delete: description
```

Figura 41: Realizar varias sentencias en el directorio LDAP utilizando formato LDIF

1.7.3 Exploradores LDAP en el Mercado

Los exploradores (LDAP browser) [21] que existen en el mercado, permiten visualizar la estructura de árbol LDAP. Para un usuario principiante consultar los tipos de datos y conocer las distintas formas de búsqueda en el árbol LDAP puede ser bastante engorroso. Para facilitar al usuario el acceso al directorio LDAP y realizar búsquedas sobre el DIT, se presentan varias alternativas junto con sus características. Entre las aplicaciones libres destacamos Gq, Phpldapadmin (aplicación web) y JXplorer.

1.7.3.1 Exploradores basados en JAVA

En la Table 39 se mencionaran algunos de los exploradores que están desarrollados en java y se indica las características principales de cada uno. Los más utilizados son Apache Directory Studio y Jxplorer.

Browser LDAP	Significado
Apache Directory Studio	Es una herramienta libre que se puede utilizar con cualquier servidor LDAP pero está especialmente diseñado para utilizar con ApacheDS. Es una aplicación Eclipse RCP (Rich Client Platform) compuesto de varios plugins. Tiene una interfaz bastante limpia e intuitiva y lo mejor interesantes es que tiene un desarrollo bastante acelerado. Apache Directory Studio puede ser utilizado como una herramienta independiente ó como un plugin para Eclipse. No sólo permite leer y mostrar el árbol de su servidor LDAP sino que también le permite modificar mediante la creación, edición o eliminación de entradas. [22]
Gawor's LDAP Browser	Es difícil encontrar, y no ha sido actualizado desde hace años. Esta basado en java y funciona en cualquier sistema operativo.
Jxplorer	JXplorer es un navegador de código abierto de LDAP, originalmente desarrollado por Computer Associates' eTrust. Es compatible con las normas de un propósito general ldap navegador que puede utilizarse para leer y buscar cualquier directorio LDAP. Está disponible para descarga gratuita bajo licencia Open Source. Ha sido probado y se ejecutan en Windows, Solaris, Linux y OS390, y debería funcionar en cualquier sistema operativo de apoyo java . Algunas de sus características son: • Operaciones LDAP normales: añadir, eliminar, copiar, modificar. • Operaciones complejas: árbol de copiar y borrar árbol • Opcional GUI basado en la construcción de filtros de búsqueda • Apoyo DSML • Arquitectura extensible basada en clase con objeto de Java plugins [23]

Table 39: Browser LDAP Basados en JAVA

1.7.3.2 Exploradores para Windows

En la Table 40 se nombran los exploradores que son utilizados en el sistema operativo Windows.

Browser LDAP	Significado
Active Directory Explorer (AD Explorer)	Es un visor y editor avanzado de Active Directory (AD). Es utilizado para navegar fácilmente una base de datos AD, definir ubicaciones favoritas, ver las propiedades y atributos de objetos sin tener que abrir los cuadros de diálogos, ver del esquema de un objeto, y ejecutar búsquedas sofisticadas que se pueden guardar y volver a ejecutar. AD Explorer también incluye la posibilidad de guardar consultas de una base de datos de AD para la visión fuera de línea y las comparaciones. El tipo de software es Freeware.[24]
Softerra Browser	Es una versión liviana de Softerra LDAP Administrator. Es compatible con operaciones de lectura únicamente y no se permite la modificación de los datos de directorios LDAP. Para la gestión completa de los directorios LDAP se utiliza Softerra LDAP Administrator.[25]
Cygsoft LDAP Browser	Cygsoft LDAP Browser es un explorador basado en C++. Proporciona al usuario la facilidad de buscar, crear, listar y editar registros en los directorios. Facilita la modificación del contenido de las bases de datos LDAP. Cygsoft LDAP Browser muestra la información del directorio LDAP en una estructura de árbol. Al hacer clic en un objeto los atributos y sus valores correspondientes se muestran en forma de tabla. Con este navegador es posible acceder a los datos en cuestión de segundos, ahorrando tiempo y el aumento de la productividad
LDAP Explorer	Navegador LDAP multi-plataforma desarrollado en .Net con C#. Es utilizado en plataformas Windows y Linux (Debian, Red Hat, Mandriva). Se ha testeado en los siguientes servidores de directorio: Active Directory (win2000 y win2003), Novell E-Directory y OpenLDAP 2.x. Las características principales son las siguientes: • SSL / TLS • Soporte completo de UNICODE • Crear / editar / eliminar objetos LDAP • Tiene licencia BSD (Berkeley Software Distribution). [26]
LDAP Admin	Herramienta gratuita de administración de Win32 para la gestión de directorios LDAP. Permite navegar, buscar, modificar, crear y eliminar objetos en el servidor LDAP. También es compatible con las operaciones más complejas, como copiar el directorio. Amplía las funciones comunes de edición para apoyar a determinados tipos de objetos. Es el software libre de código abierto distribuido bajo la licencia GNU (General Public License). Las caracteristicas principales son: • Navegación y edición de directorios LDAP • Operaciones recursivas en árboles de directorios (copiar, mover y borrar) • Modificar las operaciones sobre conjuntos de datos • Cambiar el nombre de las entradas LDAP • El soporte LDAP SSL (usando la API de Windows).[27]

Table 40: Browser LDAP soportados por Windows

1.7.3.3 Exploradores para Linux/Unix

En la Table 41 se nombran los exploradores que son utilizados en el sistema operativo Linux/Unix.

Browser LDAP	Significado
KdirAdm	Herramienta de administración de directorios LDAP desarrollada para KDE Environment versión 2 o posterior. Su objetivo es ofrecer toda la funcionalidad de la mayoría de las herramientas de administración de directorios comerciales, creación, modificación, eliminación de las entidades de directorio de todo tipo, navegación de todas las entidades, incluyendo una búsqueda exhaustiva.[28]
Directory Administrator	Utilizado en GNOME. Este explorador es utilizado para la gestión de los usuarios de UNIX y de los grupos de servidores de directorio LDAP.
GQ	Es un cliente basado en GTK. Algunas características clave de GQ cliente LDAP son: • Navegador LDAP • LDAP V3 navegador de esquema • Plantilla de constructor • Exportación de servidor subárbol o la totalidad de LDIF • Buscar en base a argumentos simples o filtro LDAP • Editar y eliminar las entradas • Añadir las entradas con una entrada existente, o en base a su propia plantilla [29].
ldapvi	Es un cliente LDAP interactivo para terminales Unix. Con esta herramienta es posible actualizar las entradas de LDAP con un editor de texto. [30]

Table 41: Browser LDAP soportados por Linux/Unix

1.7.3.4 Exploradores basados en la web

En la Table 42 se listan los exploradores basados en aplicaciones web.

Browser LDAP	Significado
PhpLDAPadmin	Es un cliente LDAP basado en la web. Proporciona facilidad de acceso desde cualquier lugar. Ya que es una aplicación web este navegador LDAP funciona en muchas plataformas, por lo que el servidor LDAP es fácilmente manejable desde cualquier lugar. Las principales características son: • Navegador árbol LDAP (ver imágenes) • Copiar recursivamente el árbol entero • Eliminar las entradas LDAP • Recursivamente eliminar árboles enteros • Ver y editar los atributos de imagen (como jpegPhoto) • Avanzada LDAP navegador esquema • Plantilla basada en la creación de la entrada • Búsqueda de LDAP (simple y avanzada) • LDIF y exportación DSML • Importación LDIF • Cambiar el nombre de LDAP las entradas • Autenticación configurable (inicio de sesión anónimo, web, o estática) • Configurable de sólo lectura y lectura / escritura de los modos. • Disponible en 10 idiomas [31]
Web2ldap	Es un proyecto escrito en Python que permite ver la información de un directorio LDAP desde la web.

Table 42: Browser LDAP basados en la web

1.7.4 Trabajando con JLDAP

En el proceso de investigación del uso de LDAP, se realizó pruebas de conexión con los servidores OpenLdap y Apache DS [32], intentando acceder y operar desde el código java desarrollado en Eclipse con la librería JLDAP [33] al servidor LDAP.

En la Figura 42 se muestra la estructura del DIT presente en el servidor, sobre el cual se realizaron las pruebas .

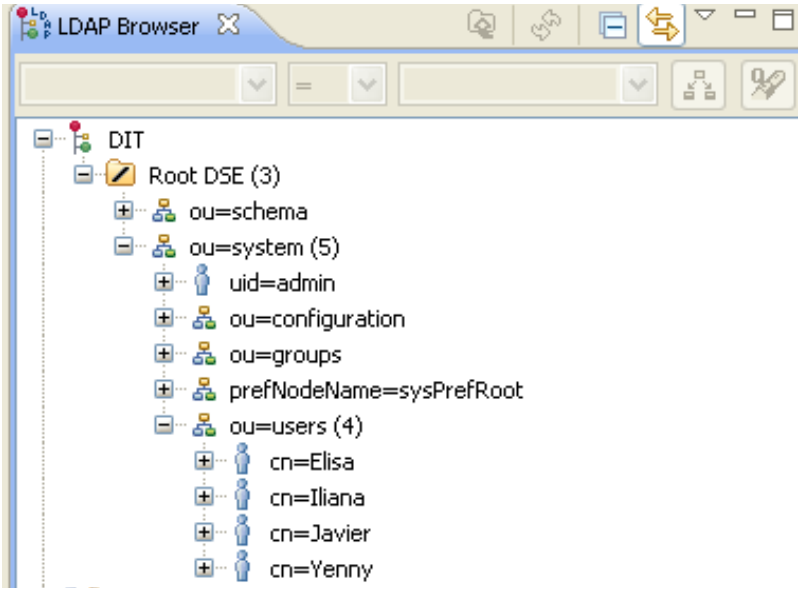

Figura 42: Estructura del DIT para probar librería JLDAP

Las pruebas realizadas consistieron en los siguientes puntos sobre el directorio LDAP:

- Conectarse al directorio LDAP. El código se expone en la Table 43.
- Agregar una Entrada al DIT. El código se expone en la Table 44.
- Borrar una Entrada del DIT. El código se expone en la Table 45.
- Buscar una Entrada en el DIT. El código se expone en la Table 46.

A continuación se detalla el código de cada punto.

```
import com.novell.ldap.LDAPConnection;
import com.novell.ldap.LDAPException;
import java.io.UnsupportedEncodingException;
import java.util.logging.Level;
import java.util.logging.Logger;

public class conexion_JLDAP_Novell {

          private int ldapPort;
            private int ldapVersion;
          private LDAPConnection lc;
           private String login;
           private String ldapHost = "localhost";

   public conexion_JLDAP_Novell() {}

 // ********************************************************************************************************
// Este método permite realizar la conexión al servidor de LDAP (ApacheDS),  para el usuario manager.
// Parametros:
//         strManager:  Usuario
//                    strPassword: Password
//                    rama: Rama del arbol a la que accedo
// ********************************************************************************************************
   public LDAPConnection ConexionManager(String strManager, String strPassword, String rama) {


          // PARA APACHEDS.....

       login= "uid="+strManager+ ","+ rama;
       ldapPort = LDAPConnection.DEFAULT_PORT;
       ldapVersion = LDAPConnection.LDAP_V3;


       try {
             lc = new LDAPConnection();
             lc.connect(ldapHost, ldapPort);
             System.out.println("====Conectado al Servidor LDAP====");
             lc.bind(ldapVersion, login, new String(strPassword));
             System.out.println("Autenticado en el servidor....");

         }
          catch (LDAPException ex) {
               System.out.println("Ocurrio error de LOGUEO");
                           Logger.getLogger(conexion_JLDAP_Novell.class.getName()).log(Level.SEVERE,null,
ex);}
          return lc;
     }
 // ********************************************************************************
// Este codigo permite cerrar una conexión LDAP
// Parametro: Conexión
//********************************************************************************
     public void CerrarConLDAP(LDAPConnection lc) {

    try {
             lc.disconnect();
             System.out.println("Conexion Cerrada Correctamente...");
        } catch (LDAPException ex) {
                   Logger.getLogger(conexion_JLDAP_Novell.class.getName()).log(Level.SEVERE,null, ex);}
     }}
```

Table 43: Código para realizar una conexión con el servidor LDAP

```
import com.novell.ldap.LDAPAttribute;
import com.novell.ldap.LDAPAttributeSet;
import com.novell.ldap.LDAPConnection;
import com.novell.ldap.LDAPEntry;
import com.novell.ldap.LDAPException;
import com.novell.ldap.LDAPModification;
import com.novell.ldap.LDAPSearchResults;
import java.io.UnsupportedEncodingException;
import java.util.Enumeration;
import java.util.Iterator;
import java.util.logging.Level;
import java.util.logging.Logger;

public class Agregar {

//*************************************************************************
// Procedimiento que permite agregar los datos a una persona.
// Parametros: Datos a completar (nombre, apellido, telefono, password) y
//                          RamaConex: Rama del directorio Ldap para realizar la conexión
//                          RamaAdd: La rama del directorio en donde voy a insertar la entrada
//*************************************************************************
 public void Add_EntradaPerson( String nom, String ape, String pass, String tel,  String usu, String
clave, String ramaConex,String ramaAdd) {
                    conexion_JLDAP_Novell  con= new  conexion_JLDAP_Novell();
                    LDAPConnection lc= con.ConexionManager("admin","secret",ramaConex);

                  try {
                      LDAPEntry usuario = Datos(nom, ape, pass, tel,ramaAdd);
                      lc.add(usuario);
                      con.CerrarConLDAP(lc);
                      System.out.println("Usuario Ingresado Correctamente ...");

                  } catch(LDAPException ex) {
                            if (ex.getResultCode() == 68) {
                                    System.err.println("ERROR:El Usuario ya se encuentra ingresado...");
                            }
                            Logger.getLogger(Agregar.class.getName()).log(Level.SEVERE,null, ex);
         }}
//*************************************************************************************************
// Procedimiento auxiliar para armar la estructura de la entrada para luego ser insertada
// Parametros: Datos a insertar (nombre, apellido, password, telefono)
                          Rama en donde se inserta la entrada
//*************************************************************************************************
public LDAPEntry Datos(String nom, String ape, String pass, String tel, String ramaAdd) {

        LDAPAttributeSet setAtr = new LDAPAttributeSet();
        setAtr.add(new LDAPAttribute("objectclass", new String("person")));
        setAtr.add(new LDAPAttribute("objectclass", new String("top")));

        setAtr.add(new LDAPAttribute("sn", new String(ape)));
        setAtr.add(new LDAPAttribute("cn", new String(nom)));
        setAtr.add(new LDAPAttribute("telephonenumber", new String(tel)));

       String dn = "cn="+nom+ ","+ramaAdd; // ou=users, ou=system";
       LDAPEntry newEntry = new LDAPEntry(dn, setAtr);
       return newEntry;
   }
}
```

Table 44: Código para agregar una entrada en el servidor LDAP

```
mport com.novell.ldap.*;
import java.io.UnsupportedEncodingException;
import java.util.Enumeration;
import java.util.Iterator;
import java.util.logging.Level;
import java.util.logging.Logger;


public class Buscar {

                private int searchScope = LDAPConnection.SCOPE_SUB;
                private String filtro;
                private LDAPSearchResults searchResults;

                public Buscar(){
                }

//*****************************************************************************************
//Procedimiento  para buscar un usuario dentro del servidor LDAP
// Parámetros:  strFiltro: Filtro a aplicar
//                              ramaBuscar: Rama en donde buscar dentro del directorio
//*****************************************************************************************
public void BuscarEntrada(LDAPConnection lc, String strFiltro,String ramaBuscar) {
        filtro = "(commonName="+ strFiltro + ")";
        try {
         searchResults = lc.search(ramaBuscar, searchScope, filtro, null, false);
                //Recorre Todos los Usuarios de la Base
            while (searchResults.hasMore()) {
                LDAPEntry nextEntry = null;
                try {
                                nextEntry = searchResults.next();
                } catch (LDAPException e) {
                                System.out.println("Error: " + e.toString());
                }
                LDAPAttributeSet attributeSet = nextEntry.getAttributeSet();
                    Iterator allAttributes = attributeSet.iterator();

                 //Recore los atributos del usuario
                        while (allAttributes.hasNext()) {
                                LDAPAttribute attribute = (LDAPAttribute) allAttributes.next();
                                String attributeName = attribute.getName();  //NOMBRE
                                        Enumeration allValues = attribute.getStringValues(); //VALOR
                                        if (allValues != null) {
                                                while (allValues.hasMoreElements()) {
                                  String value = (String) allValues.nextElement();
                                                        System.out.println(attributeName + ":  " + value);
                                                }
                                        }
                         }
                        lc.disconnect();
                    }
         } catch (LDAPException ex) {
             Logger.getLogger(Buscar.class.getName()).log(Level.SEVERE,null, ex);
        }
    }
 }
```

Table 45: Código para buscar una entrada por el atributo “commonName”en el servidor LDAP

```
import com.novell.ldap.LDAPAttribute;
import com.novell.ldap.LDAPAttributeSet;
import com.novell.ldap.LDAPConnection;
import com.novell.ldap.LDAPEntry;
import com.novell.ldap.LDAPException;
import com.novell.ldap.LDAPModification;
import com.novell.ldap.LDAPSearchResults;
import java.io.UnsupportedEncodingException;
import java.util.Enumeration;
import java.util.Iterator;
import java.util.logging.Level;
import java.util.logging.Logger;

public class Eliminar {

// ********************************************************************************
// Procedimiento para Borrar un entrada dado el atributo identificatorio
// Parámetros:  idAtributo: Atributo que identifica a la entrada
//                          strUser:  Usuario autorizado a acceder al directorios
//                          strPass:  Password
//                          ramaConx:  Rama del directorio para la conexión
//                          ramaRemove: Rama del direcotrio para eliminar
// ********************************************************************************
 public void EliminarPorUID(String idAtributo, String strUser, String strPass, String ramaConx, String
ramaRemove){
                    try {
                                conexion_JLDAP_Novell  con= new  conexion_JLDAP_Novell();
                                LDAPConnection lc= con.ConexionManager(strUser,strPass,ramaConx);

                                String dn = "cn="+ idAtributo +","+ramaRemove; //",ou=users, ou=system";

            lc.delete(dn);
            System.out.println("\nEntry: " + dn + " Fue Eliminado Correctamente...");
            con.CerrarConLDAP(lc);

                    } catch (LDAPException e) {
             if (e.getResultCode() == LDAPException.NO_SUCH_OBJECT) {
                 System.err.println("Error: NO existe ese usuario...");
             } else if (e.getResultCode() == LDAPException.INSUFFICIENT_ACCESS_RIGHTS) {
                System.err.println("Error: NO tiene permisos suficientes para realizar esta transaccion...");
            } else {

                 System.err.println("Error: " + e.toString());
             }
          }
     }
}
```

Table 46: Código para borrar una entrada en el servidor LDAP

```
mport com.novell.ldap.LDAPAttribute;
import com.novell.ldap.LDAPAttributeSet;
import com.novell.ldap.LDAPConnection;
import com.novell.ldap.LDAPEntry;
import com.novell.ldap.LDAPException;
import com.novell.ldap.LDAPModification;
import com.novell.ldap.LDAPSearchResults;
import java.io.UnsupportedEncodingException;
import java.util.Enumeration;
import java.util.Iterator;
import java.util.logging.Level;
import java.util.logging.Logger;


public class Modificar {

// ********************************************************************************************
// Procedimiento que permite modificar el la entrada según el atributo identificatorio.
// Parametros:
                TipoCambio: Indica a futuro que cambio quiero hacer
        AtrId: Atributo que identifica la entrada que quiero cambiar
        AtrCambio : El valor que quiero asignar
        Usr: Usuario autorizado
        Pas: Password
        RamaCon: Rama del servidor para realizar la conexión
        Rama Mod: Rama del servidor para realizar el cambio

// ********************************************************************************************
  public void Modificar(int tipoCambio, String atrId, String atrCambio, String usr, String pas,String
ramaCon,
 String ramaMod) {

                try {
            LDAPAttribute atrubuto;

            conexion_JLDAP_Novell  con= new  conexion_JLDAP_Novell();
                    LDAPConnection lc= con.ConexionManager(usr,pas,ramaCon);

                    atrubuto = new LDAPAttribute("telephonenumber", atrCambio);
            String dn = "cn="+atrId+","+ ramaMod;  //"ou=users, ou=system";

                       lc.modify(dn, new LDAPModification (LDAPModification.REPLACE, atrubuto));
                       System.out.println("Atributo Modificado OK...");

                   } catch (LDAPException ex) {
                         if (ex.getResultCode() ==  LDAPException.INSUFFICIENT_ACCESS_RIGHTS) {
                                 System.err.println("Error: NO tiene permisos suficientes para realizar esta
transaccion...");
                    }
        }
    }
 }
```

Table 47: Código para modificar una entrada en el servidor LDAP (en este caso particular cambia el teléfono de un usuario).

```
import com.novell.ldap.LDAPConnection;

public class Main {
	public static void main(String[] argv) {

		conexion_JLDAP_Novell con = new conexion_JLDAP_Novell();
	Buscar b= new Buscar();
	Agregar a= new Agregar();
	Modificar m= new Modificar();
		Eliminar e= new Eliminar();


		//ApacheDS.
		 LDAPConnection conLdap=con.ConexionManager("admin","secret","ou=system");

	// ____________________________________________________________________________
	//Agrego entrada  Javier
		// ramaConx= "ou=system",
	// ramaAdd=  "ou=users, ou=system"
		// Parametros (valores de los atributos, usuario ,clave, ramaConx,  ramaAdd )

		  a.Add_EntradaPerson("Javier", "Oliva", "javier_pass", "61090087", "admin", "secret",
"ou=system", "ou=users, ou=system");


//____________________________________________________________________________
		//Busco la entrada Javier
		//  ramaBuscar = "ou=users, ou=system"
	// Parametros (conexion, idBuscar, ramaBusqueda)

	 b.BuscarEntrada(conLdap, "Javier","ou=users, ou=system");

	// ____________________________________________________________________________
	//Modificar entrada, el telefono de Yenny
	// ramaCon= "ou=system" ,
		//ramaMod="ou=users, ou=system"
		//    Parametros (tipoCambio, idAtributo, valorCambio, usuario, clave, ramaConex,
ramaModif)
	//	tipoCambio enumerado 0- Tel, 1-Nombre, 2-Apellido


			m.Modificar(0,  "Yenny",  "1111111","admin","secret","ou=system","ou=users,
ou=system");

	//____________________________________________________________________________
	//Eliminar entrada Mauro
	// Parametros (idAtributo, usu, clave, ramaConx, ramaEliminar)
	// ramaCon= "ou=system" ,
	// ramaEliminar="ou=users, ou=system"

		 e.EliminarPorUID("Mauro","admin", "secret", "ou=system","ou=users, ou=system");

	//____________________________________________________________________________
	//Cierro la conexion LDAP
	con.CerrarConLDAP(conLdap);

	}
}
```

Table 48: Código del Main, que hace uso de las funcionalidad descriptas anteriormente

Bibliografía

[1] Página del MIT - http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-rg-es-4/ch-ssh.html - Último acceso <04/04/12>

[2] SSL Protocol - http://technet.microsoft.com/en-us/library/cc785811.aspx - Último acceso <04/04/12>

[3] NTLM - https://msmvps.com/blogs/juansa/archive/2008/12/26/seguridad-autenticarse-ntlm-ntlmv2-kerberos.aspx - Último acceso <04/04/12>

[4] Autenticación Implicita - http://technet.microsoft.com/es-es/library/cc754104(v=ws.10).aspx - Último acceso <04/04/12>

[5] PASPORT - http://msdn.microsoft.com/es-es/library/f8e50t0f(v=vs.80).aspx - Último acceso <04/04/12>

[6] IPSec - http://es.wikipedia.org/wiki/IPsec - Último acceso <04/04/12>

[7] CHAP - http://technet.microsoft.com/es-es/library/cc775567(v=ws.10).aspx - Último acceso <04/04/12>

[8] RADIUS - http://es.wikipedia.org/wiki/RADIUS - Último acceso <04/04/12>

[9] Autenticación en Linux - http://www.linuxtotal.com.mx/index.php?cont=info_admon_008 - Último acceso <04/04/12>

[11] Autenticación Windows - http://es.kioskea.net/contents/winnt/ntusers.php3 - Último acceso <04/04/12>

[11] Conceptos PAM - http://sopa.dis.ulpgc.es/ii-aso/portal_aso/leclinux/seguridad/pam/pam_doc.pdf - Último acceso <04/04/12>

[12] Controlador dominio+Samba+OpenLdap - http://www.tuxjm.net/docs/samba+ldap-como/html-onechunk/ - Último acceso <04/04/12>

[14] Tipos de replicas - http://www.zytrax.com/books/ldap/ch7/ - Último acceso <04/04/12>

[15] - http://www.zytrax.com/books/ldap/ch7/#ol-syncrepl-ro - Último acceso <04/04/12>

[16] Replicas OpenLdap - [http://www.openldap.org/doc/admin24/replication.html] - Último acceso <04/04/12>

[17] - http://www.openldap.org/doc/admin24/replication.html - Último acceso <04/04/12>

[18] Respaldo y Recuperación LDAP - http://tuxjm.net/2010/01/11/como-respaldar-y-restaurar-una-base-de-datos-de-openldap/ - Último acceso <04/04/12>

[19] Replicas- syncprov-checkpoint - http://www.zytrax.com/books/ldap/ch6/ - Último acceso <04/04/12>

[20] Replicas- syncprov-sessionlog - http://www.openldap.org/lists/openldap-software/200811/msg00050.html - Último acceso <04/04/12>

[21] LDAP Tabla de Atributos más usados - http://www.zytrax.com/books/ldap/ape/#attributes - Último acceso <04/04/12>

[22] Exploradores LDAP en el Mercado - http://ldapwiki.willeke.com/wiki/LDAP%20Browsers - Último acceso <04/04/12>

[23] Apache Directory Studio - http://directory.apache.org/studio/ - Último acceso <04/04/12>

[23] Jxplorer - http://www.jxplorer.org/ - Último acceso <04/04/12>

[25] Active Directory Explorer - http://en.wikipedia.org/wiki/Active_Directory_Explorer - Último acceso <04/04/12>

[26] Softerra Browser - http://www.ldapbrowser.com/info_softerra-ldap-browser.htm - Último acceso <04/04/12>

[26] LDAP Exporter - http://www.ldapexplorer.com/ - Último acceso <04/04/12>

[27] Ldap Admin - http://ldapadmin.sourceforge.net/ - Último acceso <04/04/12>

[29] KdirAdm - http://www.carillon.ca/kdiradm/ - Último acceso <04/04/12>

[30] GQ - http://laaad.sourceforge.net/en/gq-browser.htm - Último acceso <04/04/12>

[31] Ldapvi - http://www.lichteblau.com/ldapvi/ - Último acceso <04/04/12>

[31] PHPLDAP Admin - http://phpldapadmin.sourceforge.net/wiki/index.php/Main_Page - Último acceso <04/04/12>

[33] JLDAP - http://www.openldap.org/jldap/ - Último acceso <04/04/12>

[34] Libreria JLDAP - http://mvnrepository.com/artifact/com.novell.ldap/jldap/4.3 - Último acceso <04/04/12>