

UNIVERSIDAD CENTROCCIDENTAL
“LISANDRO ALVARADO”

**MODELO DE INTEGRACIÓN DE APLICACIONES ORIENTADO A
OBJETIVOS E INGENIERÍA DE MÉTODOS.**

JULIO CESAR TORRES

Barquisimeto, Noviembre 2013

UNIVERSIDAD CENTROCCIDENTAL “LISANDRO ALVARADO”
DECANATO DE CIENCIAS Y TECNOLOGIA
MAESTRIA EN CIENCIAS DE LA COMPUTACION
MENCION INGENIERIA DE SOFTWARE

**MODELO DE INTEGRACIÓN DE APLICACIONES ORIENTADO A
OBJETIVOS E INGENIERÍA DE MÉTODOS.**

Proyecto de trabajo presentado como requisito
parcial para optar al grado de Magister Scientiarum

Por: JULIO CESAR TORRES

Barquisimeto, Noviembre 2013

**MODELO DE INTEGRACIÓN DE APLICACIONES ORIENTADO A
OBJETIVOS E INGENIERÍA DE MÉTODOS.**

Por: JULIO CESAR TORRES GOMEZ

TRABAJO DE GRADO APROBADO

Rodolfo Canelón

Ramón Valera

Edgar González

Barquisimeto, ___ de _____ de 2013

DEDICATORIA

¡Pero no digas no puedo ni en broma, porque el inconsciente no tiene sentido del humor, lo tomará en serio, y te lo recordará cada vez que lo intentes.

Facundo Cabral

A la energía universal, creadora y omnipresente que me ayudo a canalizar mis fortalezas y sortear mis debilidades para alcanzar este objetivo.

A las lágrimas de preocupación de mi madre durante mi etapa de rebeldía e indefinición académica adolescente, no serán suficientes las excusas ofrecidas, le dedico con humildad este logro como muestra de “mejor tarde que nunca”.

A mi abuela Elena quien con su ejemplo tomando el pan de su plato para darlo a sus hijos y nietos, me enseñó la bondad y el compromiso con los seres queridos.

A mis hijas por brindarme la oportunidad de dar y recibir amor sin condiciones, sirva este esfuerzo como un ejemplo del camino a seguir.

A mi esposa por estar siempre a mi lado, brindándome apoyo y comprensión.

A mi amigo RAOH por su sinceridad y acompañamiento a pesar de no entender ni “J” de tecnología.

A los que creyeron y los que no...

Gracias por el impulso.

AGRADECIMIENTO

A mi familia por hacerme sentir miembro importante de un colectivo.

A nuestra casa de estudios, que posibilitó mi formación como parte del proceso de mejora profesional.

A mis compañeros de cohorte que me hicieron sentir el compañerismo y la solidaridad propias de un equipo en formación.

A mi tutor por su asertividad, profesionalismo y guía académica en el proceso de selección y construcción de este trabajo de investigación.

A los que estuvieron y estarán, gracias “totales”.

INDICE GENERAL

DEDICATORIA	iv
AGRADECIMIENTO	v
INDICE GENERAL	vi
INDICE DE TABLAS	ix
INDICE DE FIGURAS.....	x
RESUMEN	xi
INTRODUCCION	1
CAPITULO I	5
EL PROBLEMA.....	5
Objetivos de la Investigación.....	11
Objetivo General.....	11
Objetivos Específicos.....	11
Justificación e Importancia	12
Alcances y Limitaciones	13
CAPITULO II.....	14
MARCO TEORICO.....	14
Antecedentes de la investigación	14
Bases Teóricas	22
La Información.....	23
Los Sistemas.	24
Los Sistemas de Información	25
La Ingeniería de Métodos	26
La Orientación a Objetivos	27
NFR Framework.	29
i * Framework.	29
KAOS.....	30
Integración.	31

Estrategias de Integración.....	32
Data Warehouse.....	32
Enterprise Resource Planning (ERP).....	34
Enterprise Application Integration (EAI).....	35
Categorías de Integración.....	38
Niveles de Integración.....	40
Servicios.....	42
Arquitectura Orientada a Servicios.....	45
Servicios Web.....	47
XML.....	49
WSDL.....	50
UDDI.....	51
SOAP.....	51
Enterprise Service Bus.....	53
Patrones.....	57
Patrones Arquitecturales.....	58
Patrones de Integración.....	59
CAPITULO III.....	64
MARCO METODOLOGICO.....	64
Naturaleza de la Investigación.....	64
Diseño de la Investigación.....	65
Fase I. Diagnostico.....	66
Fase II. Factibilidad.....	66
Factibilidad Técnica.....	67
Factibilidad Operativa.....	67
Factibilidad Económica.....	68
Fase III Diseño.....	68
CAPITULO IV.....	70
PROPUESTA DEL ESTUDIO.....	70
Fase 1: Caracterizar el Modelo de Integración de Aplicaciones Orientado a Objetivos.....	70
El enfoque orientado a objetivos.....	71
i *.....	73

Casos de uso.....	77
Arquitectura Orientada a Servicios (SOA).	79
Enterprise Service Bus (ESB).	81
Fase 2: Diseñar un modelo de Integración de Aplicaciones Orientado a Objetivos empleando Ingeniería de Métodos.	86
Exploración.	88
Modelado.	89
Servicios.....	91
Fase 3: Emplear el modelo de integración resultante en un caso de estudio, en específico al Vicerrectorado Académico de la UCLA.	96
Exploración.	96
Iteración 1.	96
Iteración 2.	98
Modelado.	100
Casos de Uso.....	101
Iteración 1	101
Iteración 2	102
Iteración 3	103
Diagramas de Objetivos.....	108
Iteración 1	108
Iteración 2	110
Iteración 3	113
Servicios.....	115
Descubrimiento.....	115
Iteración 1	115
Iteración 2	116
Despliegue.....	119
CAPITULO V.....	122
CONCLUSIONES Y RECOMENDACIONES.....	122
Conclusiones.....	122
Recomendaciones.	124

INDICE DE TABLAS

Tabla N° 1. Categorización de Middleware.....	37
Tabla N° 2. Escenarios de Integración.....	39
Tabla N° 3. Niveles de Integración.....	40
Tabla N° 4. Matriz Nivel de Integración vs. Tecnología Middleware.....	41
Tabla N° 5 Comparación de modelos de desarrollo.	43
Tabla N° 6. Categorización de Patrones Arquitecturales.....	58
Tabla N° 7. Estilos de Integración.	59
Tabla N° 8. Patrones para servicios de Mensajería.....	62
Tabla N° 9. Patrones para manipulación de Mensajes.....	63
Tabla N° 10. Identificación – Caracterización de unidades organizacionales.	88
Tabla N° 11. Resumen de unidad organizacional en estudio.....	89
Tabla N° 12. Proceso de Modelado de Casos de Uso.....	90
Tabla N° 13. Proceso de Diagramado de Objetivos.....	91
Tabla N° 14. Correspondencia entre Modelos y Diagramas.....	91
Tabla N° 15. Identificación de Servicios.	92
Tabla N° 16. Catálogo de servicios.....	93
Tabla N° 17. Identificación – Caracterización Vicerrectorado Académico.....	98
Tabla N° 18. Identificación – Caracterización D.P.D.I.....	99
Tabla N° 19. Caso de uso: Consultar Docente.....	103
Tabla N° 20. Caso de uso: Consultar Cambios de Dedicación.....	103
Tabla N° 21. Caso de uso: Consultar experiencia laboral.....	104
Tabla N° 22. Caso de uso: Consultar ascensos.	104
Tabla N° 23. Caso de uso: Consultar contratos.	105
Tabla N° 24. Caso de uso: Actualizar datos personales.....	105
Tabla N° 25. Caso de uso: Gestionar solvencias.	106
Tabla N° 26. Caso de uso: Gestionar Constancias.....	106
Tabla N° 27. Caso de uso: Administrar unidad organizacional.....	107
Tabla N° 28. Caso de uso: Consultar docente en Unidad Organizacional.....	107
Tabla N° 29. Caso de uso: Generar solvencia en unidad organizacional.....	108
Tabla N° 30. Catálogo de Servicios (caso de estudio).....	119

INDICE DE FIGURAS

Figura N° 1. Vista de alto nivel de un sistema orientado a servicios.....	46
Figura N° 2. Estándares en Web Services.....	49
Figura N° 3. Interacción en una Arquitectura Orientada a Servicios Web.	53
Figura N° 4. Identificación de objetivos por medio de preguntas Why - What - How.....	72
Figura N° 5. Representación de Actores en i*.....	74
Figura N° 6. Representación de Objetivos en i*.....	75
Figura N° 7. Representación de Tareas en i*.....	75
Figura N° 8. Representación de Recursos en i*.....	75
Figura N° 9. Relaciones de Dependencia en i*.....	76
Figura N° 10. Links en i*.....	76
Figura N° 11. Notación de Diagramas de Caso de Uso.....	78
Figura N° 12. Principios de SOA.....	80
Figura N° 13. Evolución de Integración mediante ESB.	82
Figura N° 14. Estructura de Mensajes.	83
Figura N° 15. Rol de un ESB dentro de un proceso de Integración.....	84
Figura N° 16. Ingeniería de Métodos.....	86
Figura N° 17. Actividades del Modelo de Integración Orientado a Objetivos.	87
Figura N° 18. Superposición de diagramas.....	92
Figura N° 19. Secuencia de actividades Elicitacion de servicios.....	94
Figura N° 20. Aplicación ESB en modelo propuesto.	95
Figura N° 21. Organigrama de la UCLA.	97
Figura N° 22. Distribución de Vicerrectorado Académico y unidades relacionadas.....	97
Figura N° 23. Intercambio de información entre unidades organizacionales estudiadas.....	100
Figura N° 24. Casos de uso entre Vicerrectorado Académico y D.P.D.I.....	101
Figura N° 25. Diagrama de actores.....	101
Figura N° 26. Casos de Uso Vicerrectorado Académico.....	102
Figura N° 27. Diagrama de Objetivos Organizacionales.....	109
Figura N° 28. Diagrama Objetivos-Tareas-Usuarios en D.P.D.I.....	110
Figura N° 29. Diagrama Objetivos-Tareas-Recursos en C.D.C.H.T.....	111
Figura N° 30. Diagrama Objetivos-Tareas-Recursos D.F.P.A.....	112
Figura N° 31. Diagrama Objetivos-Tareas-Recursos D.E.U.....	113
Figura N° 32. Diagrama de Objetivos – Tareas- Recursos Vista General.....	114
Figura N° 33. Diagrama de Procesos de Unidades Organizacionales.....	116
Figura N° 34. Composición de Servicios Candidatos.....	117
Figura N° 35. Despliegue de Servicios en propuesta arquitectural.....	121

UNIVERSIDAD CENTROCCIDENTAL “LISANDRO ALVARADO”

DECANATO DE CIENCIAS Y TECNOLOGIA

MAESTRIA EN CIENCIAS DE LA COMPUTACION

MENCION INGENIERIA DE SOFTWARE

**MODELO DE INTEGRACIÓN DE APLICACIONES ORIENTADO A
OBJETIVOS E INGENIERIA DE METODOS.**

Autor: Julio Cesar Torres.

Tutor: Ramón Valera.

RESUMEN

Hoy más que nunca, comunicar es importante, y cobra una nueva dimensión si lo que se quiere relacionar son entidades encargadas de brindar información, históricamente dentro de las organizaciones se desarrollan soluciones a problemas particulares de algunas unidades, en muchos casos sin la adecuada planificación de los vínculos de comunicación necesarios entre ellas, condicionando practicas poco recomendables de intercambio de información así como la no correcta reutilización de recursos disponibles, desmejorando la eficiencia y no atendiendo a los objetivos estratégicos organizacionales. En atención a lo anterior la presente investigación propone un Modelo de Integración Orientado a Objetivos e Ingeniería de Métodos, cuyo propósito es identificar en el marco de una estructura de objetivos, los procesos de negocios interdepartamentales necesarios para alcanzarlos y los actores participantes, siendo identificables sus interacciones en casos de uso, que servirán como insumo necesario para la definición de servicios, dentro de un enfoque arquitectural que permita la integración de las aplicaciones presentes en los entornos organizacionales, todo lo anterior enmarcado en el ámbito de la Ingeniería de Métodos, mediante la cual se definen los procesos y los métodos necesarios para la obtención del fin de la investigación, representados como un conjunto artefactos entregables predefinidos y organizados, diagramados en modelos bajo las herramientas i* y UML, utilizando una metodología ágil de desarrollo como eXtreme Programming (XP), desarrollando un modelo útil a ser aplicado a casos de estudio en donde se identifiquen las falencias antes descritas. La propuesta se basa en prácticas probadas en el área de la Ingeniería de Software, respetando estándares de desarrollo así como elementos propios del dominio en estudio.

Palabras Clave: Integración de Aplicaciones, Orientación a Objetivos, Ingeniería de Métodos.

INTRODUCCION

En la actualidad, gran parte de las organizaciones soportan sus operaciones en los sistemas de información, estos han sido desarrollados según las necesidades particulares de las unidades presentes en estas, correspondiendo dicho desarrollo en su mayoría a las necesidades del momento histórico organizacional más que a la adecuada planificación y visión de futuro, estas prácticas han traído como consecuencia la proliferación de soluciones heterogéneas que favorecen el desarrollo de islas de información, descritas por De Pablos y otros (2011) como sistemas de información orientados a satisfacer necesidades específicas de la organización, en cuyo desarrollo se utilizan tecnologías diversas que dificultan su mantenimiento, interconexión y posterior comunicación.

En consecuencia, dichas soluciones frente a la constante necesidad de comunicación, apelan a prácticas no adecuadas de intercambio de datos, que provocan redundancia, falta de actualización y perjudican la exactitud de la información organizacional, no tomando en cuenta además la posibilidad de compartir los procesos por ellas ejecutados y que representan activos importantes en función de la economía de recursos, por lo tanto se requiere mejorar el proceso de comunicación entre unidades organizacionales y sus sistemas heterogéneos, evitando las prácticas antes descritas, apoyados en el uso de herramientas que facilitan estos procesos, logrando integrar los sistemas y sus recursos con la consecuente mejora en la generación de resultados.

En función de lo antes expuesto, se hace necesaria la correcta identificación de las unidades organizacionales, sus procesos de negocio y las entidades que participan en estas, con la finalidad de establecer con claridad cuáles son las interacciones presentes y necesarias entre ellas, y en función de estas establecer un

punto de partida para un esquema integrador, destacándose la necesidad de una visión amplia para obtener un esquema general de la organización. Algunos autores tales como Rolland y Salinesi (2005) comentan que entre el 60 y 70% de los proyectos de software fracasan por las limitaciones propias de la captura, validación y gestión de requisitos, ilustrando su importancia en el éxito de cualquier propuesta, a tal fin los mismos autores destacan que con la finalidad de paliar esta situación se hace necesaria la exploración de los grupos organizacionales y sus actividades en función del logro de los objetivos.

En este particular, Kavakli y Loucopoulos (2006) comentan que la definición de requisitos describe por qué se necesita un sistema basado en las condiciones presentes y futuras, describiendo una serie de características en función de satisfacer los objetivos, siendo en consecuencia los objetivos según Rolland y Salinesi (ob.cit) “la fuerza que conduce los procesos de la ingeniería de requisitos.”(p.1). En el mismo orden de ideas Raian (2010) comenta “...la ingeniería de requisitos orientada a objetivos tiene como razón principal capturar el porqué de las necesidades”(p.22), dando la posibilidad de generar software que se ocupa de resolver problemas de la organización, “superando las limitaciones de otros métodos, como el análisis orientado a objetos, en el que se comienza inmediatamente con el estudio de lo que el sistema tiene que hacer, sin justificar porque tiene que hacerlo”(p.22).

Es de destacar que el descubrimiento de los objetivos es un proceso iterativo, que permite abstraerlos en niveles que surgen en función de satisfacer los de nivel superior, generándose una jerarquía que clarifica la visión de la organización, permitiendo establecer relaciones entre ellos y los procesos de negocio que se ejecutan para su satisfacción (Markovic y Kowalkiewicz, 2008), posibilitando en consecuencia la identificación de los artefactos que participan en los procesos antes descritos y sus actores, haciéndose necesario a tal fin la aplicación de procesos sistemáticos normados por la Ingeniería de Métodos, descrita como la rama de la ingeniería para hacer frente a todas las actividades relacionadas con los métodos, técnicas y herramientas (Iribarren, 2012), en este caso definidas como un conjunto de

artefactos entregables para cuya obtención se da un proceso iterativo e incremental propio de las metodologías ágiles de desarrollo.

Por lo antes expuesto es importante aclarar que una solución de integración debe garantizar que a pesar que se requiere integrar datos y procesos, esto debe lograrse por medio de opciones que permitan mantener la independencia operativa de las unidades, por tal razón una alternativa viene dada por la transformación en servicios de aquellas funcionalidades descubiertas producto del refinamiento de objetivos, estos representan funcionalidades útiles, expuestas para ser consumidas, combinadas y reutilizadas (Quiroga, 2011) con la finalidad del logro de objetivos particulares, y cuyo desarrollo garantiza además según Konrad (2005), bajo acoplamiento y un ámbito de aplicación inter – empresarial.

Sin embargo y con la finalidad de hacer operativa una propuesta de integración cuyos componentes de interacción estén representados como servicios, y dado que estos son simplemente un medio para la construcción de aplicaciones distribuidas, se requiere una alternativa arquitectural en la que dichos recursos y sus funcionalidades sean organizados, en este contexto la Arquitectura Orientada a Servicios(SOA) emerge como alternativa, teniendo como finalidad según Keen y otros (2004) “...traer los beneficios de acoplamiento débil y la encapsulación a la integración en la empresa”.(p.37). Es definida por Konrad (ob.cit) como “un paradigma arquitectural que favorece la creación de aplicaciones vía la orquestación de los servicios que interactúan a través de una variedad de interfaces basados en estándares”, (p.8), siendo importante denotar que no representa un producto a ser comprado, adquirido o instalado, es un medio de pensar en función de servicios que encapsulan funcionalidades del negocio, y como tales deben ser reutilizadas.

Por lo antes expuesto el presente trabajo de investigación propondrá un Modelo de Integración de Aplicaciones Orientado a Objetivos e Ingeniería de Métodos. Para el cual se tomara como caso de estudio, al Vicerrectorado Académico

de la Universidad Centroccidental Lisandro Alvarado. A tal fin está dividida en capítulos, cuyo contenido se describe a continuación:

El Capítulo I, conformado por el planteamiento del problema, los objetivos de la investigación, la justificación e importancia de la misma y finalmente su alcance y limitaciones.

El Capítulo II, que contiene los antecedentes de la presente investigación, las bases teóricas que la sustentan, así como el sistema de variables a utilizar.

El Capítulo III, denominado Marco Metodológico cuyo contenido describe la metodología a emplear en el desarrollo de la investigación.

El Capítulo IV, corresponde al desarrollo formal de la propuesta, tomando en cuenta los elementos descritos en el Capítulo III atendiendo el logro de los objetivos propuestos.

Por último el Capítulo V, en el cual se detallan las conclusiones y recomendaciones resultantes de la investigación.

CAPITULO I

EL PROBLEMA

La información en las organizaciones, representa un activo importante en el desarrollo cotidiano de sus operaciones, en este particular De Pablos y otros(2011) comentan que “mas del 80% del tiempo total de trabajo de los directivos de una empresa esta dedicado al procesamiento de la información: buscándola, recibíendola, procesándola y utilizándola en una amplia variedad de tareas.(p. 21)” siendo conveniente el diseño de sistemas para “producirla y gestionarla”, con el objetivo de asegurar su fiabilidad, exactitud y disponibilidad para la toma de decisiones, representando los avances tecnológicos elementos condicionantes de su evolución natural, involucrando al software y su desarrollo.

Como resultado de estos avances, en la actualidad muchas organizaciones poseen sistemas de información organizados en mayor o menor medida, compuestos por sub-sistemas creados para dar respuesta a las necesidades particulares de las *unidades organizacionales* que la conforman, en este particular, Kendall y Kendall (2005), exponen que “Las organizaciones se componen de sistemas más pequeños e interrelacionados...que se encargan de funciones especializadas” (p.27), condicionando según De Pablos y otros(ob.cit) las *islas de información*, descritas por estos como “distintos sistemas de información construidos en las empresas en torno a areas productivas de una manera independiente, provocando falta de unidad e ineficacia en los flujos de información”(p.150), surgiendo así los *sistemas heterogéneos* en el entorno organizacional.

Con referencia a lo anterior, y frente a la variedad de operaciones presentes en las unidades organizacionales y cubiertas por los sistemas heterogéneos, se crea la necesidad de identificar con precisión dichos procesos de negocio y su interacción en

el entorno organizacional, proceso tradicionalmente administrado por la *Ingeniería de Requisitos*, definida por Sommerville(2005) como la descripción de los servicios y restricciones de un sistema de software, es decir, lo que el software debe hacer y bajo qué circunstancias debe hacerlo, más adelante señala que el propósito de la obtención de requisitos es “establecer un conjunto completo de objetivos que el sistema debe cumplir.” (p.25); sin embargo, a pesar de esta aseveración, en muchos casos en el proceso de elicitación de requisitos, estos se plantean en función de los objetivos de los usuarios o actores, y no a nivel de las unidades y la organización, generando como consecuencia algunas limitaciones a nivel de integrar procesos de negocio, sin garantías de reusabilidad.

En este particular, Markovic y Kowalkiewicz (2008) argumentan que a pesar de la necesidad de visión amplia en la definición y modelado de los procesos de negocio, muchas organizaciones no toman en cuenta los objetivos en el nivel adecuado, así mismo señalan que los procesos de negocio existen con la finalidad de alcanzar algún *objetivo organizacional*, describiendo los objetivos como un estado deseable, medible en cuanto a su satisfacción, alcanzables o mantenibles en tiempo determinado. En consecuencia, y en función de la mencionada identificación de objetivos, autores como Van Lamsweerde (2009) caracterizan la *Ingeniería de Requisitos Orientada a Objetivos*(Goal Oriented) como el uso de metas para la obtención, elaboración, estructuración, especificación, análisis, negociación, documentación y modificación de requisitos, ya que proporcionan los fundamentos de estos, y como tal los requisitos se definen en función de cumplir cierto o determinado objetivo establecido.

Sin embargo, y tal como se ha mencionado, un objetivo debe poder ser satisfecho y esto no es necesariamente implementable a niveles estratégicos, en este particular Van Lamsweerde (ob.cit) describe la necesidad de refinar objetivos de alto nivel irrealizables por si mismos, hasta que se alcancen sub-objetivos realizables que puedan ser asignados a agentes (compuestos por componentes de software, dispositivos de entrada / salida y agentes humanos), en este particular Markovic y

Kowalkiewicz (ob.cit) comentan que por medio del anterior refinamiento se pueden establecer vinculos entre los objetivos, los procesos de negocio y los artefactos tecnológicos para su ejecución, los cuales pueden ser recorridos en ambos sentidos, creandose una trazabilidad de arriba hacia abajo para los procesos de definición, descubrimiento y modelado, y en sentido contrario con la finalidad de pertinencia y comprobación.

Reforzando lo anterior, en investigaciones previas Van Lamsweerde (ob.cit) en torno al logro de los objetivos reflexiona sobre la necesidad de cooperación, requiriendose a diferencia de los requisitos en general combinaciones hibridas de multiples agentes para lograrlo, estos agentes son equiparables a *actores* y sus responsabilidades a *casos de uso*, los cuales según Pressman (2010) representan una descripción de como un actor interactua con un sistema, por tal razón frente a la creación de la jerarquía de objetivos, sus vínculos con los procesos de negocio y los actores que participan, es posible inferir que el enfoque orientado a objetivos representa una alternativa útil para clarificar la visión de alto nivel organizacional, su descomposición en procesos de negocio y la identificación de un conjunto hibrido de actores y casos de uso en su resolución, pudiendose extrapolar como parte del proceso de identificación de las interacciones entre las unidades en operación.

Tomando en cuenta los elementos antes descritos, se hace necesaria la implementación de un mecanismo o *modelo* que permita su obtención, clarificación y asignación de responsabilidades, además de formalizar sus procesos de obtención; en este particular es útil implementar actividades propias de disciplinas como la *Ingeniería de Métodos*, caracterizada por Brinkkemper en Iribarren(2012) como un enfoque para llevar a cabo el desarrollo de proyectos sobre la base de una manera específica de pensar, constituido por un conjunto de instrucciones y reglas estructurando actividades organizadas de forma sistemática, con el correspondiente desarrollo de *productos o artefactos* predefinidos en cada etapa, que en el caso del contexto en estudio, estarán representados por los modelos antes descritos, apoyados en herramientas especializadas para cada área particular.

Sin embargo, lo anteriormente expuesto por sí mismo no basta para mejorar la comunicación entre unidades con sistemas heterogéneos, sobre todo cuando los niveles de dependencia entre estos o *acoplamiento* son altos, esta característica definida por Pressman, (2010) como un indicador de carácter cualitativo referente al grado de conexión de un módulo con otros y con el mundo exterior, representa un elemento importante al diseñar soluciones de software, ya que según el mismo autor, la conectividad sencilla entre módulos genera como resultante un software más fácil de entender, con menor propensión al efecto “*ola*” propio de la propagación de errores entre estos, infiriéndose así que si bien los sistemas en muchos casos guardan relaciones funcionales entre sí, no por esto dichas relaciones deben interferir en el grado de libertad interna que tenga cada uno de los componentes interactuantes; por el contrario, lo que se busca es lograr un nivel aceptable de independencia que garantice que estos puedan ser modificados u optimizados, siempre y cuando mantengan u ofrezcan los *resultados* requeridos por otros.

De los resultados anteriormente expuestos surge el concepto de *servicio*, que según Aalbers (s.f) “representa una función de negocios claramente definida que puede ser invocada remotamente mediante protocolos de comunicación” (p.4), por su lado Boccalari (2010) la define como una unidad con funcionalidad definida, que un proveedor deja disponible en el ambiente a través de una interface; para que pueda ser utilizado por sus consumidores, siendo importante destacar su carácter reusable y su significado funcional completo, ya que sí bien puede internamente estar compuesto por varios procesos, busca la resolución de un problema o satisfacción de un objetivo particular, por tal razón pueden ser equiparables a los casos de uso previamente identificados con el enfoque orientado a objetivos.

Así, como paso siguiente al proceso de identificación de objetivos, procesos de negocio y artefactos para su satisfacción, actores y casos de uso a ser traducidos en servicios, se hace necesaria la aplicación de soluciones que permitan su uso para el adecuado intercambio entre unidades; esta *integración* o de forma más específica *Integración de Aplicaciones Empresariales* (EAI), definida por Vázquez (2010),

como “el intercambio sin restricciones de datos y procesos de negocio entre cualquier aplicación y fuente de datos existente en una empresa” (párr. 6), se ve enriquecida por *patrones de integración* definidos por Valera (2011), como “diseños comunes en el desarrollo de funcionalidades relacionadas con la integración de aplicaciones” (p.67), especificando según el mismo autor mecanismos estandarizados de ejecución de tareas, posibilitando mediante un lenguaje común determinadas funciones.

Sobre la base de las consideraciones anteriores, es fundamental delimitar como estructurar el conjunto de subsistemas interactuantes y su medio de comunicación, a la organización de componentes de software se le denomina *arquitectura*, la cual es definida por autores como Buschmann y otros (2007), como la organización estructural de los sistemas, que proporcionan un conjunto predefinido de subsistemas especificando sus responsabilidades e incluyendo las directrices para organizar sus relaciones, pudiendose aplicar *patrones arquitecturales* que según los mismos autores contribuyen a la construcción de arquitecturas de software factibles, tomando en cuenta algunos principios generales de estructuración para todo el sistema, con el consecuente impacto en sus subsistemas o componentes, ayudando al diseñador a alcanzar la solución en un contexto amplio de desarrollo considerando al sistema no como un ente aislado, sino como un componente interactuante generando soluciones integradoras.

De lo anterior, se hace evidente en el caso de integración de aplicaciones por medio de componentes de interacción predefinidos como servicios, la *Arquitectura Orientada a Servicios*(SOA), la cual según Espina (2007) es el resultado de décadas de evolución en busca de organizaciones más ágiles y efectivas, ya que según Papazoglou y Den Heuvel (2006) proporciona un conjunto de directrices, principios y técnicas que permiten la reorganización y redesplicue de la empresa, sus procesos de negocio y la información de manera eficaz, con la finalidad de apoyar y permitir los planes estratégicos y los niveles de productividad requeridos por los ambientes competitivos de negocios.

En el mismo orden de ideas, una SOA es definida por Lewis (2011) como una forma de diseño, desarrollo, implementación y administración de sistemas, en que los servicios proporcionan funcionalidad empresarial reutilizable, ya que su infraestructura permite el descubrimiento, composición e invocación de servicios, a través de interfaces bien definidas; sin embargo, parte de la correcta implementación de este enfoque arquitectónico, obliga según Papazoglou y Den Heuvel (ob.cit) frente a la existencia de conjuntos de procesos de negocio (existentes o previstos) con capacidades y funcionalidades en torno a objetivos comunes, a crear dominios de servicios, pudiendo elevar sus niveles de abstracción, simplificando su implementación y respondiendo a las necesidades antes planteadas como bajo acoplamiento, granularidad gruesa y reusabilidad, dando como resultado un entorno de computación distribuida con un alto grado de flexibilidad e interoperabilidad entre unidades.

De lo antes expuesto, es importante destacar que la creación y organización de dichos dominios de servicios, puede ser apoyada por la identificación y refinamiento de los objetivos y su relación con los procesos de negocio, por tal razón es posible inferir que el enfoque orientado a objetivos, puede representar un mecanismo útil en función del logro de la correcta implementación de una SOA como alternativa de integración, posibilitando la obtención de resultados de calidad propios de esta arquitectura; sin embargo y frente a estas aseveraciones se plantean un conjunto de interrogantes: ¿cuáles serán las características generales de un modelo de integración de aplicaciones orientado a objetivos.?, ¿cómo estaría diseñado el modelo de integración de aplicaciones orientado a objetivos, empleando a tal fin la Ingeniería de Métodos.?, y por ultimo pero no menos importante, ¿cómo sería aplicado dicho modelo de integración en un caso de estudio seleccionado.?

Por lo antes expuesto y tomando en consideración los beneficios de los enfoques, metodologías y tecnologías descritas, se propone un Modelo de Integración de Aplicaciones Orientado a Objetivos e Ingeniería de Métodos, tomando como caso

de estudio al Vicerrectorado Académico de la Universidad Centroccidental Lisandro Alvarado (UCLA).

Objetivos de la Investigación

Objetivo General

Proponer un Modelo de Integración de Aplicaciones Orientado a Objetivos e Ingeniería de Métodos.

Objetivos Específicos

- Definir las características generales de un modelo de integración de aplicaciones orientado a objetivos.
- Diseñar un modelo de integración de aplicaciones orientado a objetivos empleando ingeniería de métodos.
- Emplear el modelo de integración resultante en un caso de estudio, en específico al Vicerrectorado Académico de la UCLA.

Justificación e Importancia

La integración de aplicaciones heterogéneas en las organizaciones, permite hacer eficiente el uso de los recursos existentes, aunando esfuerzos en el logro de los objetivos, estos recursos en operación, cumplen funciones específicas dentro de cada unidad brindando soluciones adecuadas a los requerimientos por estas impuestos, sin embargo, la falta de comunicación efectiva entre estas aplicaciones limita el intercambio adecuado entre unidades, con el consecuente desperdicio de tiempo y recursos en la generación de resultados.

La propuesta de un modelo de integración orientado a objetivos e ingeniería de métodos, permitirá identificar por medio de su descomposición jerárquica, aquellos que por sus características pueden ser transformados en servicios útiles que servirán como canales de intercambio entre aplicaciones, este modelo representara un propuesta útil que servirá como referencia para desarrollar los mecanismos de interconexión eficaz de las aplicaciones legadas y futuras, generando como resultado la eliminación de la necesidad de duplicación de datos, el manejo de paquetes intermedios (hojas de cálculo, interfaces de comunicación a datos remotos, directorios compartidos, entre otros), con la consecuente mejora de los tiempos de respuesta, así como la fácil generación de reportes con datos integrados entre aplicaciones para dar respuesta efectiva a los requerimientos de información entre las unidades, con el consecuente beneficio en el aspecto técnico, administrativo y gerencial.

Alcances y Limitaciones

Para el desarrollo de la propuesta del Modelo de Integración de Aplicaciones Orientado a Objetivos e Ingeniería de Métodos, se tomara como caso de estudio al Vicerrectorado Académico UCLA, dentro del cual hacen vida unidades relacionadas a la administración de los recursos docentes y de investigación, entre las cuales se pueden mencionar: la Dirección de Personal Docente y de Investigación (DPDI), la Dirección de Formación de Personal Académico (DFPA), el Consejo de Desarrollo Científico Humanístico y Tecnológico (CDCHT), así como la Dirección de Extensión, todos en función del logro de los objetivos generales y específicos organizacionales.

La confidencialidad de los procesos e información por estas administradas así como por los sistemas analizados durante la presente investigación, representa una limitación a tomar en cuenta, siendo aplicable la referencia a dicha confidencialidad en los casos que así lo requiera.

CAPITULO II

MARCO TEORICO

Antecedentes de la investigación

En cualquier investigación como la presente se requiere por parte del autor, la revisión detallada de los estudios previos al tema de investigación, el propósito de esta revisión es identificar el camino recorrido por otros investigadores, catalogando sus alcances, particularidades así como sus aportes al área de investigación, todo esto con la finalidad de no desperdiciar recursos recorriendo “senderos” ya estudiados, y en caso de ser ese el objetivo, retomar líneas de investigación con enfoques que aporten novedades útiles al área de estudio, a tal fin, a continuación se mencionan aquellos trabajos que brindan antecedentes a la presente propuesta.

Bazán (2009), desarrollo una investigación titulada “**Un modelo de integrabilidad con SOA y BPM**”, describiendo en esta el carácter originario de las ciencias de la computación, como simples desarrolladores de soluciones automatizadas a procesos históricamente manuales, que con el tiempo se enfrentan al problema de la poca comunicación entre ellas, definiendo como “desafío” alcanzar la obtención rápida de nuevos requerimientos, en un entorno actual en que los modelos son “pobres” en su integración y orientados a describir datos y transacciones.

Por lo antes expuesto, propone la necesidad de orientar los esfuerzos a lograr reducir la brecha existente entre los aspectos organizacionales y las tecnologías de información, proponiendo BPM como alternativa para “la modelización, ejecución y medida de rendimiento dentro de un ciclo de mejora continua” (p. 10), así mismo propone un enfoque arquitectural basado en SOA ya que “permite diseñar, construir,

desplegar e integrar los servicios independientes de los lenguajes en los que estén codificados y de las plataformas en las que se ejecutan”.

A tal fin, desarrolla un estudio evolutivo de las distintas estrategias orientadas a integración buscando “puntos de contacto entre ellas y eligiendo los elementos a componer” (p. 11), cubriendo todas las etapas del ciclo de vida de software, haciendo hincapié en modelar todos los procesos de negocio, identificándolos como consumidores de servicios, decantándose por BPM sobre UML a ser utilizado en la presente investigación por considerar al primero como más eficiente para modelar procesos de negocio, ya que si bien ambos diagraman a estos, UML se orienta a objetos en contraposición a BPM que lo hace a procesos, además de ser más cercano a los analistas de negocio.

De la investigación antes, descrita se toman las mismas motivaciones que justifican la presente investigación, así como el estudio evolutivo de las diversas estrategias de integración, diferenciándose en la metodología de modelado a implementar; así como, en la necesidad de demostrar en la presente un bus de servicio orientado a orquestar el flujo de servicios y mensajes entre las aplicaciones del dominio en estudio.

Raian (2010), en su artículo titulado “**Modeling and Reasoning about Contextual Requirements: Goal-based Framework**” analiza el como la Ingeniería de Requisitos en algunos casos, no toma en cuenta de manera apropiada el contexto en el que se desenvuelve el hecho a sistematizar, insistiendo constantemente en lograr un mecanismo útil que provea los elementos necesarios para obtener de forma eficiente la simbiosis requisitos – contexto; a tal fin, explora preguntas como la metodología necesaria para captar esta relación, como el contexto modifica los requisitos y como la aplicación de soluciones puede modificar a su vez el contexto en estudio, pudiendo además plantear un mecanismo de comprobación del modelo de requisitos contextual diseñado para detectar y evaluar la gravedad de los conflictos

propios del cambio constante del contexto en estudio, todo esto con economía de recursos.

Así, Raian(ob.cit), propone el diseño de un lenguaje contextual orientado a objetivos, que permita captar la variabilidad basada en el contexto de las necesidades a nivel de objetivos, inicialmente con TROPOS como herramienta de captura, para posteriormente proponer el prototipo de herramienta denominado RE-CONTEXT, que incorpora el formalismo y la forma de razonamiento estudiada a ser aplicada en casos de ejemplo que permiten incorporar la herramienta para el logro de los objetivos planteados.

Al igual que la presente investigación, la propuesta de Raian (ob.cit) estudia en el ámbito de la Ingeniería de Requisitos los objetivos como mecanismo de identificación y posterior verificación de la validez de los resultados obtenidos, por medio de su satisfacción, particularizando en el contexto como un elemento que influye en la variabilidad de la obtención de los requisitos y que de alguna forma impide aplicar soluciones tipo “todo en uno”, justificando así el desarrollo de una herramienta diseñada para aplicación de los principios estudiados en gran variedad de casos.

Por otro lado, Teruel, Navarro, López-Jaquero, Montero, y González (2011), en su investigación para la Universidad de Castilla la Mancha titulada: “**A comparative of Goal-Oriented approaches to modelling requirements for collaborative systems**”, estudian en el contexto de sistemas colaborativos, el enfoque orientado a objetivos como alternativa para la captura de requisitos principalmente de naturaleza no funcional (NFR), destacando el carácter no trivial de la especificación de estos requisitos por su fuerte impacto sobre la arquitectura final del sistema.

En este particular, destacan las técnicas orientadas a objetivos sobre otras tradicionales, como los casos de uso, referenciando a (Cockburn, 2000), ya que sólo se centran en el establecimiento de las características (es decir, actividades y entidades) que el sistema deberá apoyar, en comparación con la propuesta orientada a

objetivos en la que se especifica el por qué los sistemas se están construyendo, proporcionando la motivación y razón para justificar la especificación de requisitos del software, destacando su utilidad no solo en el análisis de los objetivos sino también para la elaboración y el perfeccionamiento de ellos.

Destacan además, que el modelo orientado a objetivos puede ser especificado en una variedad de formatos, mediante el uso de una notación más o menos definida, categorizando estas como: informales utilizando lenguaje natural para especificar los objetivos, semi-formales empleando cajas y diagramas de flechas, y por último formales donde los objetivos se expresan como afirmaciones lógicas en un lenguaje formal de especificación. Sin embargo, comentan que sin importar su formalidad, un modelo de objetivos se construye como un grafo dirigido por medio de un refinamiento de los objetivos de los sistemas, prolongándose hasta que los objetivos tienen suficiente granularidad y detalle con el fin de ser asignado a un agente (software o medio ambiente) de manera que sean verificables en el sistema a diseñar. Este proceso de refinamiento se lleva a cabo mediante el uso de relaciones lógicas de refinamiento AND / OR / XOR.

A tal fin, estudian tres enfoques orientados a objetivos: NFR Framework, i * Framework, y la metodología KAOS, comparándolos frente a un sistema colaborativo real, siendo utilizado a tal fin Google Docs., obedeciendo esta selección, al hecho de ser ampliamente conocido y por contar con un enfoque de colaboración claro como su principal objetivo. Como resultado de esta evaluación se determina que a pesar de presentar falencias y fortalezas particulares en distintas áreas de la evaluación, el menos adecuado según el estudio es NFR por su escaso soporte a requisitos funcionales, siendo el más adecuado i*, seguido de cerca por KAOS; sin embargo, es de destacar que dicha diferencia se ve representada por la contabilización de las categorías en estudio, repartiéndose las ventajas de forma alternativa entre estos enfoques, siendo importante destacar que ninguno de estos enfoques es capaz de destacar la importancia de los requisitos, algo que debe ser considerado en opinión de los investigadores en futuros trabajos.

En consecuencia y a modo de conclusión, como resultado del estudio se evidencia que ninguno de los enfoques estudiados representa una ventaja definitiva frente al resto, en el contexto de la investigación planteada por los autores, lo cual genera la necesidad de por medio de estudios posteriores, desarrollar una extensión útil a estos, con la finalidad de contribuir a mejorar la situación diagnosticada.

En el contexto de la presente investigación, lo anteriormente estudiado representa un gran avance en torno a reforzar la idea de la orientación a objetivos como un método útil para la elicitation de requisitos y su vinculación con procesos y actores, añadiendo además el estudio sistemático de algunas metodologías que aplican el modelado en orientación a objetivos, sus características y aplicabilidad, siendo este factor de gran importancia en la elección de la(s) metodología(s) de modelado para la presente investigación.

Continuando con la revisión de antecedentes, López (2011) en su investigación denominada “**Propuesta de Integración de Sistemas de Información basado en ROA. Caso: UNEXPO**”, al igual que las investigaciones anteriores, plantea un escenario en el que existen diversos sistemas de información, separados por barreras geográficas importantes, en el describe las características de estos sistemas, sus alcances así como necesidades de comunicación, plantea además el desarrollo de la integración como el resultado de un proceso continuo de estudio y adecuación de los elementos a fin de lograr el objetivo final, justifica su investigación resaltando el aspecto cambiante del quehacer organizacional, que condiciona las constantes variaciones requeridas para adaptarse a dicha dinámica institucional.

Es por esto que, resalta la cooperación entre unidades y sistemas como eje fundamental de la gestión operativa y estratégica, mas expone que dicha cooperación aunada al carácter dinámico de los requisitos de gestión, hace complicada la implementación de soluciones efectivas en cuanto a los recursos y su relación de rediseño – respuesta en el tiempo. Se desprende de esto que, las organizaciones deben

contar con arquitecturas flexibles que permitan cambios constantes a sus distintos módulos sin afectar su relación con su ecosistema.

A tal fin, propone la adopción de una Arquitectura orientada a Recursos (ROA), que representa un enfoque económico en cuanto a inversión, así como un conjunto de técnicas efectivas que orienten a los sistemas de información a realizar su labor de forma eficiente.

Para la caracterización del dominio en estudio, se utilizó el método InDoCaS, cuya finalidad es según Canelón (2010) en Iribarren (ob.cit) “obtener una arquitectura base para una familia de productos, las disciplinas principales de este proceso son el análisis y diseño del dominio.” Logrando así obtener los requisitos funcionales y no funcionales de las aplicaciones en estudio, clarificando así su interacción, permitiendo mediante la obtención de los artefactos generados la identificación de los recursos que proveerán, asociándolos posteriormente a URI's, describiendo además el curso típico de los eventos así como sus posibles condiciones de error.

Posterior a este proceso, para López (ob.cit) solo queda la transformación de estos recursos identificados con el lenguaje de programación deseado, en definitiva concluye que las ventajas de ROA con la utilización de procedimientos como RESTful, conlleva una reducción del tiempo de respuesta, control en el crecimiento de las aplicaciones, así como la consecuente mejora en los servicios prestados.

Si bien la investigación desarrollada por López (ob.cit), contiene lugares comunes con la presente investigación, en cuanto a su justificación y el objetivo final que es “integrar” los recursos a utilizar, el enfoque metodológico es distinto, y su adopción obedece a las características propias del dominio del caso de estudio particular a estudiar.

Mendez (2011), en su trabajo de investigación titulado “**Arquitectura Orientada a Servicios para la interoperabilidad con equipos médicos. Caso de estudio laboratorio clínico Mascia.**”, al igual que las investigaciones descritas con

anterioridad, justifica su investigación describiendo los aspectos negativos de los sistemas y/o equipos que generan información, trabajando de forma no integrada, describe los mecanismos de intercambio de información, tales como hojas de cálculo en formato privativo, con su consecuente sensibilidad al error así como el desperdicio de capital humano en la confección de dichos “documentos de transferencia”, sin olvidar el impacto en el tiempo de respuesta en un área tan sensible como es el área de la salud.

En ese mismo sentido describe las particularidades del dominio en estudio, en este caracteriza el proceso de integración entre equipos médicos y sistemas de información, como algo complejo ya que los primeros poseen líneas de comunicación bastante restringidas y cuya conexión en la actualidad, depende del uso de las herramientas de interconexión diseñadas a tal fin por la empresa comercializadora de los equipos médicos, estas herramientas que en general no se encuentran desarrolladas bajo estándares similares, dificultan su integración al entorno productivo del sistema de información, ya que su uso representa una erogación económica importante, y lo disímil de sus interfaces complican su coexistencia con el resto.

Por lo anteriormente descrito, propone una Arquitectura Orientada a Servicios (SOA), sustentada en un bus empresarial que permitirá la conexión transparente de los distintos dispositivos médicos, con los sistemas de información, siendo este bus el encargado de normar y orientar dicha intercomunicación. A tal fin, se diseña el producto haciendo uso de herramientas de estándares abiertos, que posibilitan el logro de los objetivos a bajo costo, concluyéndose que el producto final garantizaría la mejora sustancial en los tiempos de respuesta, en cuanto a los equipos de laboratorio con la entrega de resultados, para apoyar la labor médico-diagnóstica de las distintas patologías tratadas en esos centros de salud.

De la Vara y otros (2013) en su artículo “**On the Use of Goal Models and Business Process Models for Elicitation of System Requirements**” inicia su

análisis señalando que el modelado orientado a objetivos y orientado a procesos son dos técnicas de elicitation de requisitos que inicialmente se presentan como antagónicas, ambas han sido estudiadas por separado por diversos autores e incluso en conjunto en intentos de amalgamar su aplicación; por tal razón, no duda en señalarlos como equivalentes en algunos aspectos operativos, pudiendo ser aplicadas la una en apoyo a la otra, centrando su artículo en el estudio de los escenarios en que una técnica debe ser aplicada de preferencia sobre la otra y también cuando la aplicación de ambas representa la mejor opción.

Insiste además en el concepto de “objetivos operativos”, que son según su visión, aquellos que deben cubrir los procesos de negocio, ejemplificando esto incluso con herramientas como BPMN en donde los segundos son plasmados de forma implícita, describiendo la posibilidad de mediante el análisis, derivar los objetivos de los procesos de negocio mediante el uso de “árboles de objetivos” que se obtienen a partir de la ya estudiada aplicación de descomposiciones AND / OR describiendo la colaboración en caso de la primera o la alternabilidad de tareas en la segunda, así ejemplifica la elaboración de los árboles de objetivos tomando directrices que apoyan su construcción.

Continúa su análisis diferenciando la aplicabilidad de un modelado sobre el otro, comentando la orientación a objetivos como útil principalmente cuando por razones de poca claridad en la definición y obtención de los procesos de negocio, sea esto por discrepancias en su aplicación o por no definición (en caso de organizaciones noveles) representa una alternativa útil ya que contribuye a facilitar la definición de los procesos necesarios para su satisfacción, considerando la aplicación preferente del modelado de procesos en aquellos escenarios de organizaciones complejas en las que solo se requiere la automatización de procesos, siendo relevante el recurso a invertir en la elaboración de los árboles de objetivos.

Finaliza su artículo, señalando la aplicación de ambos enfoques en aquellas organizaciones donde el proceso de mejora continua exige la revisión de los procesos

en función de la eficiencia de recursos, para lo cual, la trazabilidad arboles de objetivos–procesos de negocio represente un elemento clarificador de las derivaciones y alternabilidades posibles para alcanzar la satisfacción de las entidades involucradas con economía de recursos. Así, la investigación efectuada por De la Vara(ob.cit) guarda lugares comunes con las anteriores referencias abordadas, al igual que con la presente investigación en torno a : trazabilidad de objetivos y procesos de negocio, herramientas para el modelado, insistiendo además en la utilidad del modelado de objetivos como herramienta clarificadora de los procesos, elemento a ser tomado por la presente investigación en la transformación de estos en servicios a ser apoyados en la arquitectura pertinente.

De los anteriores trabajos de investigación se deduce, que la falta de integración es un problema común en organizaciones de diversa índole, no siendo su razón social un catalizador particular que acelere o retrase sus consecuencias negativas, ya que no es un problema particular de un dominio, sino del enfoque dado a los procesos, la planificación y desarrollo de soluciones informáticas para esos dominios, así mismo, se demuestra que dicha problemática genera practicas poco recomendables en el intercambio de información entre aplicaciones, concluyéndose que la búsqueda y adopción de modelos, arquitecturas y esquemas orientados a la integración de aplicaciones, representan una alternativa de solución válida al problema fundamental descrito en las anteriores y la presente investigación.

Bases Teóricas

A continuación se lleva a cabo la revisión bibliográfica que permitirá la obtención de los conocimientos necesarios para sustentar la presente propuesta, es de destacar que el orden de los contenidos obedece a su relación y pertinencia con el tema en estudio, manteniéndose en lo posible su desarrollo de lo general a lo particular.

La Información.

Sin lugar a dudas, la humanidad depende de la información para su desarrollo, es por esto que en las distintas civilizaciones se dio especial importancia a su obtención, clasificación y preservación en el tiempo. Así, el hombre en su constante intento de preservar sus experiencias con fines evolutivos, científicos o incluso militares, ha ideado mecanismos diversos para su preservación, siendo el primero la pintura, seguido de la escritura y registros históricos preservados en distintos medios como papiros, cueros, papel, y su agrupación natural, los libros, así la consecuente creación de bibliotecas, no es más que un proceso de organización natural de la información sin la cual el desarrollo de la humanidad tanto en el aspecto humanístico como científico y tecnológico no sería lo que es en la actualidad.

A pesar que el concepto de información como tal es tan antiguo como la propia humanidad, su definición y delimitación es común de los tiempos modernos, algunos autores como De Pablos y otros (2011) como “todo lo capturado, almacenado, procesado y distribuido por el sistema.”(p. 21) de lo cual se infiere que según su contexto poseen un significado y propósito que pueden ser la reducción de la incertidumbre o el incremento del conocimiento sobre algo. De lo anterior se desprenden términos fuertemente vinculados al concepto que de una u otra manera lo describen y sin los cuales pierde su esencia, tales como “datos” o conjunto de elementos en “bruto” sobre temas particulares, y de cuya “organización” en un “contexto” particular depende el nivel de “conocimiento” sobre un tema, para reducir la “incertidumbre”, y es este elemento, la razón de ser de esta búsqueda, no se puede aprender sin información, ni planificar frente a la incertidumbre, mostrándose así una relación inversamente proporcional entre los términos que de una u otra forma han condicionado el desarrollo de civilizaciones cuyo legado hoy prevalece en el tiempo.

Sobre la base de las consideraciones anteriores, se observa que la obtención de la información debe ser descrita como un proceso acumulativo de carácter incremental y de cuya correcta organización depende la velocidad de recuperación

para su uso posterior, siendo este carácter incremental el que genera la necesidad de crear mecanismos dinámicos de clasificación que garanticen en el tiempo de alguna manera su vinculación con los intereses de sus usuarios.

En consecuencia y dada la relación conocimiento - decisión – acción, y frente a la descrita amplia, y en aumento constante cantidad de información generada, se crea la necesidad de estudiar medios que permitan su adecuada clasificación y ulterior recuperación, por tal razón surgen áreas del saber directamente vinculadas al manejo de la información que estudian sus particularidades para dar respuesta a estas necesidades, surgiendo así la idea de los sistemas para el manejo de la información.

Los Sistemas.

El universo es un gran sistema, hasta la más pequeña de las criaturas forma parte de alguno y dentro de sí, estas son contenedoras de varios; por tal razón, la visión sistémica es natural, y tomarlo para el desarrollo intelectual humano es de evidente utilidad, la real academia de la lengua define un sistema como “Conjunto de reglas o principios sobre una materia racionalmente enlazados entre sí”, algunos autores como Sommerville (2005) los definen como, “una colección de componentes interrelacionados que trabajan conjuntamente para lograr algún objetivo” (p. 20).

De lo anteriormente planteado resaltan los conceptos de interrelación y objetivos, siendo el primero sin duda un elemento primordial ya que define el *cómo*, a razón de que todo sistema está compuesto por partes o componentes, y estos no trabajan de forma aislada, el *para que* de esta relación es el logro de los objetivos que pueden ir desde la manipulación de elementos simples a otros más complejos, sin embargo el resultado general buscado es común a cada una de las partes. Así el hombre crea sistemas compuestos por partes relacionadas que permiten lograr los objetivos propios, que en el caso del manejo de la información son la búsqueda, organización, almacenamiento y recuperación, surgiendo los sistemas especializados en el manejo de la información.

Los Sistemas de Información.

Definidos por De Pablos (2011) como “un conjunto de recursos técnicos, humanos y económicos, interrelacionados dinámicamente y organizados en torno al objetivo de satisfacer las necesidades de información...para la gestión y la correcta adopción de decisiones.” (p. 21), representan pues un valor fundamental en el que hacer organizacional, a todos los niveles. Un error común es considerar los sistemas de información como conjuntos estrechamente vinculados a elementos tecnológicos, aunque es indudable que con la evolución de la tecnología su uso en conjunto está íntimamente relacionado, un sistema de información puede funcionar por sí mismo sin estar apoyado necesariamente en estos recursos.

Todo sistema contiene entidades que lo constituyen, que como se ha señalado con anterioridad, cumplen una función en busca de un objetivo particular orientado a objetivos generales, estos elementos son los recursos, los usuarios y la información, siendo importante destacar, que los usuarios pueden ser personas así como otros sistemas interactuantes, y los recursos, están representados por los medios utilizados para facilitar el logro de las operaciones, es allí donde entran en juego los elementos tecnológicos, que divididos en dos categorías, a saber Hardware o plataforma física y Software compuesto por la lógica y rutinas para la interacción entre los usuarios y el hardware, estos participan activamente en los procesos aportando sus capacidades y haciendo eficientes los tiempos de respuesta para el logro de los objetivos organizacionales.

Cada organización desarrolla sus sistemas según sus necesidades, en este orden de ideas, Kendall y Kendall (2005) comentan “Los sistemas de información se desarrollan con diversos propósitos, según las necesidades de la empresa” (p. 2), infiriéndose de esto que los sistemas de información no nacen o crean de forma espontánea, producto de una moda o tendencia actual, se crean para cumplir un objetivo dentro de la organización, esto determina que existan gran variedad, y por tal

puedan ser agrupados desde distintos puntos de vista o niveles según su área de influencia.

De lo anteriormente planteado, se infiere que los diversos sistemas se diferencian más que por sus elementos constitutivos por sus objetivos u orientación, que si bien obedecen a un fin común organizacional, desempeñan su labor en función de uno o más objetivos particulares, es allí donde dentro de las organizaciones se observa la presencia de sistemas heterogéneos, que apoyan las funciones de unidades particulares, estos sistemas heterogéneos en su mayoría desarrollados sin la adecuada planificación, comúnmente no cuentan con los mecanismos adecuados de comunicación entre sí; por tanto, se hace necesario el concepto comunicación inter-sistemas. Sin embargo, este proceso debe ser precedido por la metodología necesaria para alcanzar productos de calidad, haciéndose necesarias ramas del saber encargadas del diseño de estas metodologías y su aplicación práctica, entre estas es destacable la ingeniería de métodos.

La Ingeniería de Métodos

Definida por Brinkkemper (1996) como "...la disciplina de la ingeniería para diseñar, construir y adaptar métodos, técnicas y herramientas para el desarrollo de sistemas de información" (p. 276), tiene como finalidad entre otras permitir establecer la clara separación entre el producto elaborado por el método y el proceso que elabora el producto.

En este particular desde el punto de vista conceptual, un método está compuesto por un modelo de proceso que describe los pasos y actividades descritas por el modelo, y un conjunto de descripciones de producto en donde están predefinidos los contenidos que serán entregados en los pasos descritos por el modelo (Harmsen, 1997), pudiéndose concluir que un modelo de producto representa una especificación de un producto entregable dentro del método.

En el mismo orden y dirección, según Brinkkemper (ob.cit) el modelo es la estructura de los productos, categorizados como: entregables, tablas, diagramas, documentos, entre otros producidos en un método de desarrollo y el modelo de procesos, corresponde a la descripción de los pasos a realizar en el método, pudiendo ser: determinación, análisis, diseño de requisitos, modelado por destacar algunos.

En función del objetivo en estudio, y frente a los métodos y procesos antes descritos, se hace necesaria la implementación de técnicas descubrimiento de los vínculos de comunicación que la harán posible la comunicación en entornos heterogéneos, históricamente este procedimiento se lleva a cabo por técnicas propias de la ingeniería de requisitos, sin embargo con la finalidad de lograr una visualización macro de aquello que se pretende integrar, se plantea la posibilidad de realizar dicha aproximación por medio del estudio de los objetivos organizacionales, pudiendo utilizar a tal fin un enfoque orientado a objetivos (Goal Oriented (G.O)).

La Orientación a Objetivos

En principio, es importante destacar que este enfoque no es para nada nuevo, diversos autores han realizado investigaciones que resaltan sus beneficios en el proceso de análisis, descubrimiento y administración de requisitos, en este particular Letier y Van Lamsweerde (2002) lo destacan como un paradigma cada vez más reconocido para la obtención, estructuración, análisis y documentación de requisitos, así mismo, Rolland y Salinesi (2005) comentan que la orientación a objetivos se ha desarrollado y probado en función de apoyar la captura de requisitos en las actividades de ingeniería, abarcando su especificación, validación, modificación, estructuración y negociación.

En la misma dirección, los objetivos son definidos como aquello que una organización persigue alcanzar (Letier y Van Lamsweerde, ob.cit) siendo en general de carácter invariable al menos en periodos cortos de tiempo, así Markovic y Kowalkiewicz (2008) describen los objetivos como un estado deseable, medible en cuanto a su satisfacción, alcanzables o mantenibles en tiempo determinado, y para

esta satisfacción se desarrollan los procesos de negocio, destacándose que frente a esta necesidad de satisfacción pueden y deben en general ser descompuestos en jerarquías que van desde los objetivos estratégicos a los objetivos técnicos en franca colaboración para el logro de los objetivos superiores.

Como ya se ha destacado, algunos autores han descrito los beneficios del enfoque orientado a objetivos, entre ellos Van Lamsweerde (2009) destaca:

- A alto nivel, proporcionan una mejor visión abstracta de la gestión organizacional.
- El logro de los requisitos manteniendo su relevancia, ya que proporcionan un criterio preciso de especificación, haciéndose evidente aquellos que no colaboran en la resolución de algún objetivo.
- El refinamiento de objetivos proporciona un mecanismo natural de estructuración que permite reducir la complejidad, mejorando la legibilidad de la documentación resultante.
- Así mismo, el refinamiento permite identificar alternativas de satisfacción de objetivos distintas a las ya implementadas.
- Favorecen la identificación de conflictos entre los requisitos.

Sin embargo, la obtención de objetivos no es tarea fácil, si bien en algunos casos se identifican con la revisión de documentos de la organización (la declaración de empresa, misión, políticas, procedimientos) estos se refieren a los objetivos estratégicos que reflejan lo que se quiere de la organización desde la visión macro, mas no describen las actividades de negocio en termino de sus actores (Kavakli y Loucopoulos, 2006), requiriendose la aplicación de tecnicas de investigación con el apoyo de los actores de negocio permitiendose luego mediante refinamiento y abstracción la obtencion del conjunto requerido de objetivos en cuya jerarquia se alcanzan su plena satisfacción.

Asi y con la finalidad de apoyar el análisis y descubrimiento de los objetivos y su documentación, se han desarrollado algunas herramientas que pemiten el modelado

de los mismos, utilizando técnicas que permiten lograr la aproximación a estos manteniendo un proceso formal, entre estos es importante destacar:

NFR Framework.

Siendo su objetivo hacer frente a requisitos no funcionales (NFR), también conocidos como requisitos de calidad, en este particular Teruel y otros (2011) señalan que a diferencia de los requisitos funcionales, los requisitos abordados por NFR permiten especificar las limitaciones del sistema así como nociones particulares de los factores de calidad de un sistema debe cumplir, generando como consecuencia que los requisitos funcionales describen aquello que el sistema va a hacer, los no funcionales describen el “como” va a lograr el “que”, manteniendo por ende la vinculación de los NFR a algún requisito funcional.

Más adelante, Teruel y otros (ob.cit) señalan que de acuerdo con este framework, los objetivos pueden presentar conflictos siendo necesaria su representación como softgoals, la cual es descompuesta en sub-objetivos a ser representada por una estructura gráfica inspirada en arboles con relaciones AND / OR para su resolución, siendo necesario a tal fin la utilización de enlaces de contribución que definen los aportes de estos, continuándose este proceso hasta el punto que se pueda afirmar que se han cumplido los operacionalizaciones del NFR relacionadas.

i * Framework.

Consiste en un enfoque para hacer frente a las necesidades en las diversas fases del proceso de desarrollo de software, descritas como fases temprana y tardía de análisis de requisitos y diseño arquitectónico, siendo durante las etapas tempranas la recolección y análisis de las intenciones de las partes interesadas, siendo modelados como objetivos, que por medio del análisis y sus pasos definidos conducen a los requisitos funcionales y no funcionales del sistema a ser construido, destacando en los primeros requisitos la identificación de los actores y sus dependencias para el logro de los objetivos, y las tareas a realizar a tal fin.

Así en su fase posterior, resulta en una especificación de requisitos en que se describen todos los requisitos funcionales y no funcionales del sistema, representándose como uno o más actores que participan en un modelo de dependencia estratégica, junto con otros actores del entorno operativo del sistema. En otras palabras, el sistema entra en escena como uno o más actores que contribuyen al cumplimiento de los objetivos de las partes interesadas, siendo necesario además en el diseño arquitectónico la selección de los estilos alternativos tomando como criterio de selección las cualidades deseadas previamente descritas en el proceso, destacándose el perfeccionamiento de estas cualidades a ser representadas como softgoals, descompuestas como subobjetivos de carácter más específico.

KAOS

Como parte del Framework del mismo nombre desarrollado para la obtención, indicación y análisis de los objetivos, requisitos, escenarios y asignación de responsabilidades, está compuesto varios elementos descritos a continuación:

- **Meta:** describe un conjunto de comportamientos admisibles del sistema, siendo destacable la necesidad de definición clara de los objetivos para la correcta verificación de cumplimiento de la meta.
- **Softgoals:** son utilizados para documentar las preferencias entre los comportamientos de sistemas alternativos, destacándose al igual que i^* en la no existencia de un criterio claro para la definición de la satisfacción de este tipo de objetivos, esperándose por tanto su satisfacción en límites aceptables.
- **Agentes:** se define como un componente del sistema activo que tiene un papel específico para satisfacer un objetivo. Un agente puede ser un agente humano, un dispositivo o un componente de software.
- **AND / OR:** corresponde al mecanismo de descomposición de una meta con un conjunto de sub-objetivos, según esta relación la meta se cumple si todos (AND) los sub-objetivos se cumplen, o (OR) al menos alguno de ellos es satisfecho.

- **Conflicto potencial:** definido como un enlace que permite determinar el no cumplimiento de la meta frente a la satisfacción de alguno (una relación contraste) en determinadas condiciones.
- **Asignación de Responsabilidad:** representa la definición de una relación de cumplimiento entre un agente y una meta a alcanzar.

Es de destacar de lo anteriormente expuesto, que la aplicación de una metodología sobre la otra depende necesariamente de las características particulares de cada una en el contexto en estudio, según el estudio comparativo realizado por Teruel (ob.cit), i * resulta el más apropiado para sistemas colaborativos, lo cual lo hace su aplicación apropiada para la presente investigación. Así con los requisitos y relaciones de los sistemas de información identificados, es posible continuar con los esfuerzos en función de lograr el fin deseado, que es alcanzar un modelo que permita establecer elementos requeridos para lograr mejorar la comunicación entre las aplicaciones, pudiéndose establecer esta relación como un proceso de integración.

Integración.

A razón de las necesidades de información en constante aumento, se requiere contar con los mecanismos necesarios para la toma de decisiones en el menor tiempo posible, a tal fin, es importante alcanzar una visión unificada de los datos y las aplicaciones en la organización al igual que con sus entidades vinculadas, en este orden de ideas, Sommerville (2005) agrega que "...las complejas relaciones entre los componentes de un sistema, indican que el sistema es más que simplemente la suma de sus partes...(p.21), siendo cada parte, vista como un módulo, un componente o incluso un sistema en sí mismo.

Si bien el concepto de integración de componentes de software, representa una solución que habilita a las diferentes aplicaciones a trabajar conjuntamente en forma eficiente, con un rendimiento cercano al tiempo real, no por esto está exento de dificultades en su aplicación, las mismas vienen de la diversidad de tecnologías con las que los sistemas son construidos, la escasa o incluso inexistente documentación

sobre su diseño y alcance, la diversidad de formatos de datos en uso, así como las barreras que se dan al intentar integrar sistemas pertenecientes a ecosistemas organizacionales distintos.

Por tal razón, el objetivo final del proceso de integración es salvar las anteriormente descritas dificultades, permitiendo compartir datos y procesos de negocio a través de diferentes sistemas, en forma independiente del lenguaje de programación o el soporte de datos en uso. Este proceso, es llevado a cabo, según las características particulares del entorno organizacional, apoyándose en enfoques o estrategias suficientemente exploradas, contentivas de características específicas que condicionan su aplicabilidad, permitiendo así, reducir el margen de error producto de la improvisación en cuanto a la implementación de soluciones al problema fundamental de comunicación efectiva, entre soluciones tecnológicas de información.

Estrategias de Integración.

Según Mendoza (s.f), existen tres enfoques arquitectónicos a considerar en cuanto al proceso de integración se refiere:

- Data Warehouse.
- Enterprise Resource Planning (ERP).
- Enterprise Application Integration (EAI).

Cada uno de ellos con características particulares que los hacen útiles a determinadas realidades organizacionales en cuanto a integración se refiere.

Data Warehouse.

Definido por Inmon (2012) como “un conjunto de datos integrados, históricos, variantes en el tiempo y unidos alrededor de un tema específico, que es usado por la gerencia para la toma de decisiones”(texto en línea), este concepto se representa como el almacenamiento de información homogénea y fiable, en una estructura basada en la consulta y el tratamiento jerarquizado de la misma, y en un entorno diferenciado de los sistemas operacionales; sin embargo, no debe ser visto como un

simple almacén de datos, ya que de ser así, no resolvería los problemas que originaron su creación. En ese orden de ideas, Inmon (ob.cit) caracteriza los Data Warehouse como:

Integrados: ya que los datos almacenados deben integrarse en una estructura consistente, eliminando las inconsistencias existentes entre los diversos sistemas operacionales. Siendo estructurada en distintos niveles de detalle para adecuarse a las distintas necesidades de los usuarios.

Temáticos: jerarquizado los datos con la finalidad de almacenar solo lo necesario para el proceso de generación del conocimiento del negocio, organizándose por temas para facilitar su acceso y entendimiento por parte de los usuarios finales.

Históricos: guardando una relación de las variables organizacionales en el tiempo a fin de permitir comparaciones y análisis de tendencias.

No volátiles: como el concepto anterior, la información no es modificada es permanente, solo se actualiza con la incorporación de los últimos valores que tomaron las distintas variables contenidas en él, sin ningún tipo de acción sobre lo que ya existía.

A juicio de Mendoza (s.f), este enfoque tiene algunas ventajas entre las que destacan la integración de datos sin cambios en los sistemas de origen, su adaptabilidad a datos internos y externos, proporcionando un entorno integrado para informes, análisis y minería de datos; sin embargo, como desventaja destaca la necesidad de agregación de datos, el esfuerzo necesario para la adecuación y transformación de los datos a ser integrados y el no soporte para la integración de procesos limitando la reusabilidad.

Enterprise Resource Planning (ERP).

Constituye un enfoque para integrar las operaciones informáticas dentro de las organizaciones, Chiesa (2004) las define como una aplicación que permite la gestión de todos los procesos de una compañía en forma integrada, compuesta por diversos módulos para dar respuesta a cada una de las necesidades propias del negocio, brindando información cruzada e integrada de todos ellos, otros autores como Kumar y Hillegersberg (2000) lo definen como un conjunto de paquetes de sistemas de información configurables, dentro de las cuales se integra la información de la empresa por medio de sus áreas funcionales.

Por lo anteriormente expuesto, se puede vislumbrar a un ERP como un gran macro sistema, que engloba las operaciones organizacionales, permitiendo integrar sus distintas áreas, posibilitando la obtención de información consolidada de todos ellos. Entre los principales beneficios que justifican la inversión organizacional en estas tecnologías destacan:

- La reducción de la incertidumbre con respecto a la veracidad de la información, ya que todas las áreas de la empresa manejan la misma información integrada, no existiendo diferentes sistemas en cuyo seno se manejen datos redundantes, que comprometan la integridad de la información.
- Mejora considerable en la comunicación interdepartamental, permitiendo tener un mayor conocimiento de lo que está sucediendo en la empresa.
- Integración de los procesos comerciales.
- El aumento de la productividad es inminente gracias a la automatización de los procesos y la integración del negocio.

Por tal razón, se asume que la aplicación de esta tecnología implica la revisión de la metodología y procesos existentes en la organización, por lo cual las aplicaciones en uso deberán ser migradas a fin de formar parte de la nueva plataforma integrada, esta particularidad dificulta su aplicación en organizaciones que poseen sistemas de

mediana a gran envergadura que apoyan sus operaciones; por otro lado, en el aspecto económico algunas alternativas tienen un costo bastante elevado, en cuanto a implantación y soporte se refiere.

Es importante destacar, la incapacidad de este enfoque de cubrir el total de operaciones de algunas organizaciones con características muy particulares, lo cual condiciona la permanencia de algunas aplicaciones legadas, que de alguna manera cohabitan dentro de la organización sin integrarse al resto; por tal razón se podría inferir que en algunos casos la integración con esta solución no es total, comprometiendo la veracidad de los resultados del proceso integrador.

Enterprise Application Integration (EAI).

Este enfoque para abordar el problema de integración, es definido de forma genérica por McKeen referenciado por (Mendoza, ob.cit) como "...los planes, métodos y herramientas que propician la modernización, consolidación y coordinación de las aplicaciones basadas en el computador dentro de una empresa", su objetivo es lograr la interoperabilidad y organización del flujo de información entre aplicaciones heterogéneas, para formar un sistema de información empresarial consolidado, a tal fin se implementa una arquitectura bajo la cual las distintas aplicaciones se comuniquen entre sí, sin las limitaciones propias de las características inherentes al desarrollo particular de cada aplicación.

En el enfoque tradicional de comunicación entre aplicaciones y recursos, las interacciones se dan entre estas de forma directa, conectándose un elemento a otro formando una red, que tiene como particularidad el incremento de complejidad al momento del aumento sustancial de estos recursos, impactando negativamente los procesos de revisión y mantenimiento de las aplicaciones, por tal razón se hace necesaria una opción alternativa que permita independientemente del número de recursos (almacenes de datos o aplicaciones) su comunicación, revisión y optimización futura atendiendo a atributos de calidad como la mantenibilidad y escalabilidad entre otros. A tal fin, dentro del enfoque EAI se plantea la necesidad de

crear un componente de software que permita lograr la descrita interoperabilidad, denominado middleware.

Middleware

Caracterizado por Wikipedia en español, como un elemento que “...nos abstrae de la complejidad y heterogeneidad de las redes de comunicaciones subyacentes, así como de los sistemas operativos y lenguajes de programación, proporcionando una API para la fácil programación y manejo de aplicaciones distribuidas”.(texto en línea), es un software separado, intermedio, de conectividad que se encarga principalmente de comunicar las aplicaciones entre si actuando como un intermediario que actúa entre estas, evitando su comunicación directa, reduciendo la complejidad al limitar la comunicaciones punto a punto entre aplicaciones.

Por lo antes planteado, se visualiza el intermediario como un recurso cuya responsabilidad está vinculada a la comunicación entre elementos, saltando a la vista el beneficio en cuanto a la conectividad futura de recursos (sin importar su número) en la dinámica organizacional. Sin embargo, es importante destacar que la correcta implementación de este software implica alterar las aplicaciones con la finalidad de redistribuir sus solicitudes comúnmente orientadas hacia los almacenes de datos o hacia las aplicaciones vinculadas, y redirigirlas hacia el intermediario (middleware), siendo este el encargado de dicha comunicación, y de la resultante entrega de los resultados solicitados.

Tipos de Middleware

Muchas son las posibilidades de categorización del software middleware, algunos autores como Pérez y otros (2003) basados en estudios previos, y tomando en cuenta las clasificaciones de otros autores, los clasifican de la siguiente manera:

MIDDLEWARE	DESCRIPCIÓN
Distributed Tuples (DT)	Es el <i>Middleware</i> para acceso a base de datos que permite desarrollar sistemas independientes del manejador de base de datos que lo soporta.
Remote Procedure Call (RPC)	Es el <i>Middleware</i> diseñado como servicio síncrono para permitir la gestión remota de redes. Esconde las operaciones de envío y recepción bajo el aspecto de una llamada convencional a una rutina o procedimiento.
Messaging Oriented Middleware (MOM)	<i>Middleware</i> orientado a mensajes; está diseñado para el servicio de mensajes con tecnología asíncrona. Permite el envío de mensajes entre aplicaciones, las aplicaciones sólo ponen y sacan mensajes de las colas, no se conectan.
Distributed Object Middleware (DOM)	<i>Middleware</i> para tecnologías orientadas a objetos; los objetos piden servicio a otros objetos que se encuentran en la red. Se encarga de establecer comunicación entre los clientes y los objetos de forma transparente respecto a la distribución. Ejemplos: <ul style="list-style-type: none"> • Common Object Request Broker Architecture (CORBA). • Remote Method Invocation (RMI). • Distributed Component Object Model.
Transaction Processing Monitors (TP Monitors)	El <i>Middleware</i> para Procesamiento de Transacciones ya que facilita la conectividad y el acceso a un gran número de usuarios con servicios de <i>back-end</i> limitados.
Database Access Technology (DBAT)	Son las Application Programming Interface (API) creando una capa transparente para el acceso a base de datos, ocultando la complejidad dada por el manejador de base de datos.
Component Oriented Framework (COF)	Este <i>Middleware</i> está soportado en el modelo de desarrollo de aplicaciones basado en componentes, permite la creación de una aplicación como conjunto de componentes reusables.
Directory Services (DS)	Es el <i>Middleware</i> que se basa en directorios que permiten reducir costos administrativos, simplifican y/o distribuyen tareas administrativas, reducen el número de <i>passwords</i> para un usuario mediante una única combinación de login /password, aumentan la seguridad y proporcionan una única localización para la información de los usuarios.
Application Servers (AS)	Es el <i>Middleware</i> que se enfoca en la parte de la de aplicación o lógica del negocio. Conceptualmente un sistema puede tener muchas capas; sin embargo, la arquitectura más popular es de tres capas: interfaz, aplicación y base de datos.


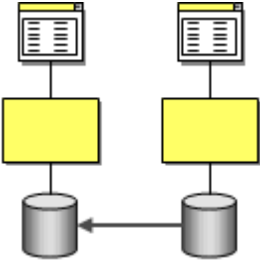
Tabla N° 1. Categorización de *Middleware*.

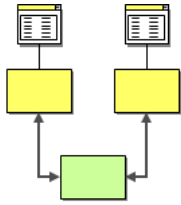
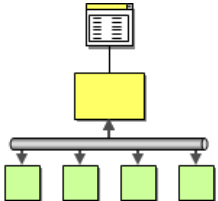
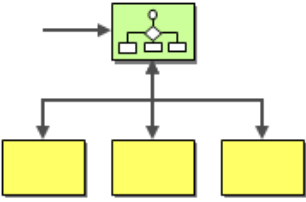
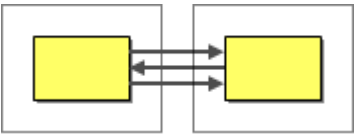
Fuente: Pérez y otros (2003).

De lo anteriormente planteado, se aprecia que todas las tecnologías Middleware permiten comunicación, siendo su diferencia el énfasis que presentan como apoyo en aspectos como datos y procesos, por tal razón la selección de la tecnología apropiada representa un reto importante, y exige la revisión y el alcance a dar al proyecto de integración. Así mismo destaca la necesidad de modificación de los recursos a ser intercomunicados; sin embargo, dicha alteración no implica cambios sustanciales dentro de las aplicaciones que limiten la factibilidad técnica de implementación.

Categorías de Integración

Algunos autores como Hohpe y Woolf (2003), categorizan los proyectos de integración en seis(6) grupos, descritos a continuación:

<p>Information Portals</p> 	<p>Corresponden a portales que buscan centralizar sistemas, en estos los usuarios pueden acceder a múltiples aplicaciones vinculadas, facilitando su utilización, dividiendo en algunos casos la pantalla en sectores en los que se presentan resultados tomados de la interacción de datos o procesos entre aplicaciones.</p>
<p>Data Replication</p> 	<p>Se aplica en casos en los N aplicaciones comparten datos, manteniéndose estos como propios de cada aplicación, y replicándose en caso de ser necesario hacia aquellos almacenes que así lo requieran, dicha replicación puede síncrona o asíncrona según las necesidades organizacionales, implementándose tecnologías apropiadas a cada caso.</p>

<p>Shared Business Functions</p> 	<p>Al igual que el escenario anterior, se trata intercambio entre aplicaciones, en este caso son rutinas de uso común. La implementación de este mecanismo permite la reutilización de recursos que brindan soluciones a problemas ya resueltos por otras unidades a ser aplicados en diversos escenarios según las necesidades del negocio.</p>
<p>Service-Oriented Architectures</p> 	<p>En este escenario las aplicaciones comparten un catálogo de servicios expuestos que pueden estar compuestos por la combinación de datos y procesos, en los que la interacción consiste en la publicación, búsqueda y utilización de recursos con la finalidad de integrar las aplicaciones.</p>
<p>Distributed Business Processes</p> 	<p>Consiste en la implementación de un componente de gestión que coordina procesos entre negocios con la finalidad de tomar aquellos elementos por estos administrados, reutilizando recursos en función de integrarlos.</p>
<p>Business-to-Business Integration</p> 	<p>Considerado el nivel más alto de integración ya que a diferencia de los escenarios anteriores no está limitado a las aplicaciones dentro entorno organizacional, haciéndose extensivo a su ecosistema, a tal fin pueden ser utilizados algunos de los mecanismos descritos con anterioridad, teniendo en cuenta que la comunicación es principalmente por red global requiriéndose por tanto mecanismos de seguridad más rigurosos.</p>

*Tabla N° 2. Escenarios de Integración.
Fuente: adaptado de Hohpe y Woolf (ob.cit) p.34 – 37.*

Niveles de Integración

Estos niveles permiten según Pérez y otros (ob.cit), establecer proyectos de integración de sistemas al definir como está la empresa y hasta dónde quiere llegar, identificándose así la estrategia a utilizar, expone así cuatro(4) niveles a saber :

NIVEL	DESCRIPCION
1	Integración Punto a Punto. Representa el nivel más pobre de integración, ya que las tecnologías utilizadas se comunican a través de interfaces intermedias donde no se contempla la visión del negocio; es decir, establece una infraestructura básica para el intercambio de datos entre aplicaciones. La interrelación entre los sistemas es baja, por lo que tienen un alto grado de independencia.
2	Integración Estructural. A este nivel, la empresa usa tecnología Middleware para estandarizar y controlar el intercambio de información. Deben tomarse en cuenta dos aspectos: (1) existe una estructura central que controla el intercambio de información; y (2) a nivel de Middleware se consolidan las reglas de negocio y las transacciones entre las aplicaciones. En este nivel, las organizaciones cuentan con interfaces integradas y fuentes de datos comunes, pero no se integra con componentes externos del negocio.
3	Integración de Procesos. A este nivel, la organización ya ha logrado la transición de compartir la información entre aplicaciones para manejar en realidad el flujo de información entre las aplicaciones. Para conseguirlo, han desarrollado un modelo de negocio común que cubre la totalidad de la empresa. Utiliza tecnología sofisticada para poner en práctica el modelo de negocio en la capa Middleware. Existe gran integración en este nivel, debido a que se toman en cuenta las reglas de negocio y el flujo de procesos.
4	Integración Externa. Las compañías en este nivel están considerando las verdaderas aplicaciones empresariales, la influencia tecnológica, la transformación de los procesos de negocio y las nuevas estructuras, para redefinir la organización desde el punto de vista de servir a los clientes. Estas organizaciones usan tecnologías de EAI para transformar el negocio, a menudo uniendo directamente a clientes y proveedores para operaciones internas. Estas compañías contemplan nuevas capacidades para crear innovadoras ofertas en línea, nuevos productos y servicios, y pueden mejorar una marca registrada o crear una nueva identidad en Internet.

Tabla N° 3. Niveles de Integración.

Fuente: Pérez y otros (ob.cit).

De lo anteriormente expuesto, es importante destacar que el paso de un nivel a otro significa una evolución por parte de la empresa en lo que a integración se refiere, por tanto implica un esfuerzo importante a implementar y de ninguna manera representa una tarea fácil y rápida; por otro lado en opinión del autor de la presente investigación sería necesario incluir dentro de esta clasificación el nivel (0) con la finalidad de representar aquellas organizaciones donde no existe intercambio alguno entre aplicaciones, no por falta de necesidad sino por falta de las tecnologías intermedias descritas en el nivel superior, o por falencias en su manejo por parte del personal operativo.

Es importante destacar que en las investigaciones realizadas por Pérez (ob.cit), se establecen una metodología por medio de la cual es posible identificar los niveles de integración haciendo un cruce con la categorización del Middleware descrito en apartados anteriores, dicha matriz se presenta a continuación:

NIVEL	TECNOLOGÍAS								
	DT	RPC	MOM	DOM	TPM	DBAT	COF	DS	AS
4									
3									
2									
1									

Tabla N° 4. Matriz Nivel de Integración vs. Tecnología Middleware.

Fuente: Pérez y otros (ob.cit)

Por lo antes expuesto se toman las tecnologías middleware y su aplicación como un elemento definitorio del nivel de integración propio de cada organización en particular, siendo útil para su revisión, identificación y planificación a futuro en el contexto de integración de aplicaciones.

En los apartados anteriores, se ha descrito la comunicación entre aplicaciones como una necesidad para lograr la integración de datos y recursos entre estas, con el

consecuente cumplimiento de los objetivos organizacionales, está comunicación va más allá de la simple compartición de recursos e información, tal como se plantea en el enfoque del manejo básico de los sistemas operativos y administración de archivos, por tal es útil cambiar el simple concepto de comunicación entre recursos a compartición de servicios.

Servicios

Quiroga (2011), describe un servicio como “...una forma de exponer una visión externa de un sistema, con reutilización interna y una composición tradicional basada en el diseño de componentes”, (p.25), así mismo, servicio es algo que una aplicación puede compartir o consumir, sean datos, procesos, u otros elementos útiles, y es este adjetivo de importancia capital para la correcta comprensión y definición de un servicio, no tendría sentido crear un servicio que no será consumido, sea por su trivialidad o su no adecuación a las necesidades particulares de la organización; por tal razón, la creación y posterior publicación y consumo de servicios está condicionada por las necesidades, ya que un servicio en general viene a atender un requisito u objetivo dentro del entorno operativo.

Es posible hacer un paralelismo entre el desarrollo basado en objetos, componentes y servicios, en este particular Quiroga (ob.cit), establece algunos lugares comunes entre estos modelos de desarrollo:

- Como los objetos y los componentes, los servicios son un bloque fundamental que combina información y comportamiento, esconde el trabajo interno para que esté fuera de intrusos (encapsulamiento) y presenta un interfaz simple para el resto del organismo.
- Los servicios también representan bloques de construcción naturales que nos permiten organizar capacidades de forma que nos sean familiares.
- Al igual que objetos y componentes, los servicios pueden ser organizados en clases y jerarquías heredando los comportamientos y pudiendo ser utilizados después de forma única, como jerarquías o colaboraciones.

Sin embargo, en cuanto al desarrollo de software en entornos orientados a servicios, existen algunas particularidades que los diferencian del desarrollo estructurado, orientado a objetos o componentes, algunos de los cuales son representados a continuación:

	PROGRAMACION ESTRUCTURADA	OBJETOS	COMPONENTES	SERVICIOS
GRANULARIDAD	Muy fina	Fina	Intermedia	Gruesa
CONTRATO	Definido	Privado/Público	Público	Publicado
REUSABILIDAD	Baja		Intermedia	Alta
ACOPLAMIENTO	Fuerte		Débil	Muy débil
DEPENDENCIAS	Tiempo de compilación.			Run-time
AMBITO DE APLICACIÓN	Intra-aplicación		Inter-aplicación	Inter-empresas

Tabla N° 5 Comparación de modelos de desarrollo.

Fuente: Konrad (2005 p.18)

De lo anteriormente expuesto, destacan conceptos como la granularidad, caracterizada por Papazoglou y Den Heuvel (2006), en el contexto del diseño de servicios y su desarrollo como el proceso de identificación de los servicios adecuados, y su organización en jerarquías manejables de servicios compuestos, por tal el concepto granularidad puede tener sinonimia con el concepto de agrupación en este caso de servicios, otros conceptos que destacan son la alta reusabilidad de los servicios, que permite ahorrar costos operativos así como el bajo acoplamiento, y su disponibilidad inter-empresas, siendo este último una característica que convierte a este método de desarrollo en un aliado para el logro de los procesos de integración.

Sin embargo, a pesar de su utilidad, no todo dentro de la organización puede o debe ser considerado un servicio, para obtener esta denominación debe cumplir con ciertas características a saber:

- **Reusabilidad:** su construcción y diseño debe ser efectuada para ser usado en el entorno intra e inter aplicaciones del mismo dominio, e incluso en el dominio público para su uso masivo.
- **Contrato formal:** todo servicio desarrollado, debe proporcionar un contrato en el cual figuren: el nombre del servicio, su forma de acceso, las funcionales que ofrece, los datos de entrada de cada una de las funcionalidades y los datos de salida, garantizándose así que su acceso será solo por este contrato.
- **Bajo acoplamiento:** característica descrita en el cuadro comparativo anterior y es representado como independencia entre servicios, proveedores y consumidores.
- **Posibilidad de composición:** los servicios deben ser construidos de tal manera que puedan ser utilizados para construir servicios genéricos de más alto nivel, el cual estarán compuesto de servicios de más bajo nivel.
- **Autonomía:** cada servicio debe tener su propio entorno de ejecución. De esta manera el servicio es totalmente independiente y se puede asegurar que podrá ser reutilizable desde el punto de vista de la plataforma de ejecución.
- **Sin estado:** un servicio no debe guardar ningún tipo de información, sólo contendrá lógica, y toda la información estará almacenada en algún sistema de información sea del tipo que sea.
- **Publicabilidad:** todo servicio debe poder ser descubierto de alguna forma para que pueda ser utilizado, consiguiendo así evitar la creación accidental de servicios que proporcionen las mismas funcionalidades.

Para el aprovechamiento de los servicios con el consecuente beneficio antes descrito, no basta con el simple descubrimiento de un elemento útil a ser convertido en servicio, se hace necesario dotar al entorno de la organización de todos y cada uno de los elementos para su correcta generación – compartición y consumo, a estos componentes generadores y consumidores, y su organización funcional se les denomina arquitectura y en el caso particular de integración, arquitectura orientada a servicios.

Arquitectura Orientada a Servicios

En base a consideraciones anteriores, una arquitectura de software puede ser definida como “La organización fundamental de un sistema, tomando sus componentes, las relaciones entre ellos, su entorno y los principios que gobiernan su diseño y Evolución” (ANSI/IEEE, 2000), así todo se refiere a organización de componentes y su interrelación entendiéndose esta como el resultado de principios generales que se basan en prácticas probadas y por supuesto se adecuan a las necesidades del negocio.

En un sentido amplio, una Arquitectura Orientada a Servicios(SOA) es definida por Bazán (2009) como: “un modelo de referencia para entender las relaciones más significativas dentro del dominio de un problema concreto y facilitar el desarrollo de estándares o especificaciones” (p.32), vislumbrándose como un modelo, mas no una tecnología que pueda ser adquirida y aplicada como un paquete de implementación, otros autores como Ruiz (2010) caracterizan a una SOA como “...un estilo de arquitectura cuyo objetivo es conseguir perder el acoplamiento entre servicios que interactúan” (p.15), destacándose de esto la importancia que tiene el bajo acoplamiento(característico de SOA) como un aspecto deseable a todo producto de software a ser desarrollado.

Así, una SOA puede ser definida también como una arquitectura de software para la construcción de aplicaciones que implementan procesos de negocio o servicios, mediante el uso de un conjunto de componentes orquestados para brindar un nivel de servicio bien definido, esta arquitectura permite que la funcionalidad de la aplicación se exponga y consuma, como un conjunto de servicios usando una forma estándar de interacción que les permiten ser invocados, publicados y descubiertos, con el objetivo, según Papazoglou y Den Heuvel (ob.cit) de posibilitar y apoyar los planes estratégicos y los niveles de productividad que son requeridos por los ambientes de negocio competitivos.

En este contexto, SOA permite separar funciones en distintas unidades o servicios, que los desarrolladores hacen accesibles dentro de una red, con la finalidad de que los usuarios puedan combinarlas y reutilizarlas en la producción de aplicaciones, estos servicios se comunican entre sí pasando información de uno a otro o coordinando actividades entre dos o más de ellos, así en una SOA, un servicio mapea una función u objetivo identificado durante un proceso de análisis del negocio; dependiendo de la función que se trate, la granularidad del mismo puede ser más o menos fina o gruesa según su grado de agrupación.

Una visión general de cómo se distribuyen los componentes dentro de esta arquitectura se presenta a continuación:

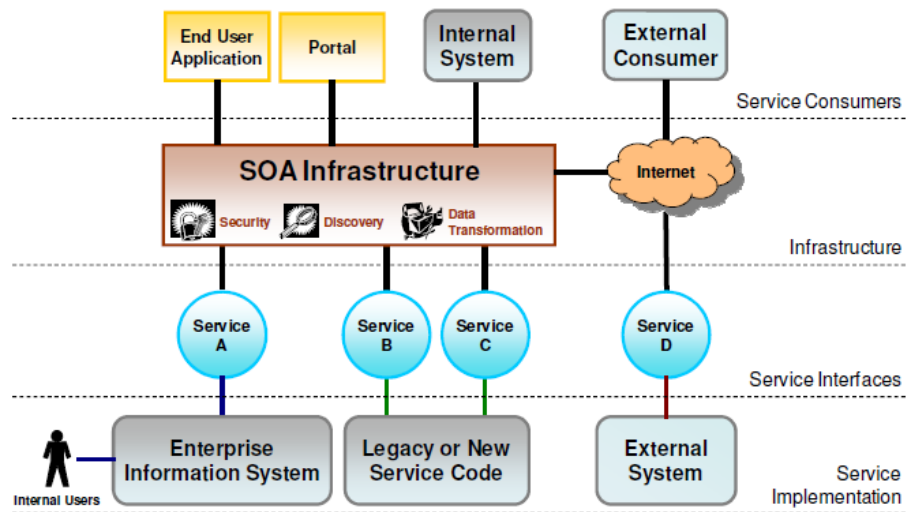


Figura N° 1. Vista de alto nivel de un sistema orientado a servicios.

Fuente: Lewis (2010 p.2).

Reforzando lo anterior, según Ruiz (ob.cit) dentro de una SOA existen varios elementos fundamentales, siendo el principal el servicio, y entre ellos las colaboraciones posibles son publicar, descubrir e interactuar. Por tal razón, dentro de este estilo arquitectural existen roles presentes en los elementos participantes:

- **Publicar servicio:** hacer accesible el servicio a sus consumidores.
- **Buscar servicio:** localizar servicios útiles a ser utilizados.

- **Interacción con servicio:** definido por la comunicación entre el consumidor de servicios sea este una aplicación u otro servicio y su proveedor.

De lo anterior, es importante destacar que cada elemento puede jugar uno o varios roles dentro de esta arquitectura, pudiendo ser consumidor o proveedor al mismo tiempo de servicios garantizando constante interacción. Si bien no es la única forma de utilización de servicios, en la actualidad dentro de una SOA es común su implementación dentro de plataformas amplias de red que superan las limitaciones de las redes locales, a esta implementación de servicios se les conoce como Web Services (Servicios Web).

Servicios Web

Los Servicios Web (WS) del Inglés Web Services, se pueden definir como una tecnología que permite que las aplicaciones se comuniquen en una forma que no depende de la plataforma ni del lenguaje de programación. Representan una interfaz de software que describe un conjunto de operaciones a las cuales se puede acceder por la red a través de mensajería XML estandarizada.

Algunas organizaciones como el World Wide Web Consortium (W3C) los define como:

Conjunto de aplicaciones o de tecnologías con capacidad para interoperar en la Web. Estas aplicaciones o tecnologías intercambian datos entre sí con el objetivo de ofrecer unos servicios. Los proveedores ofrecen sus servicios como procedimientos remotos y los usuarios solicitan un servicio llamando a estos procedimientos a través de la Web.

Utilizando una visión estratificada, es posible ver los WS como unidades de trabajo, donde cada una maneja una tarea funcional específica, este viene a ser el nivel más bajo de operación, en el escalón siguiente, las tareas se pueden combinar en

otras orientadas a negocios para manejar tareas operacionales empresariales específicas, esto permite que el personal técnico y operativo tenga una visión de aplicaciones que puedan manejar temas empresariales en conjunto, en un flujo de trabajo de aplicaciones soportadas por los WS, para posteriormente en el siguiente nivel, los arquitectos de procesos pueden agregarlos para resolver problemas en el ámbito empresarial

En el mismo orden de ideas, se pueden mencionar algunos beneficios de la aplicación de tecnologías vinculadas a WS, a saber:

- Permiten la interacción entre servicios desarrollados en cualquier lenguaje y funcionando en cualquier plataforma.
- Permiten conceptualizar funciones de aplicaciones en tareas, lo que lleva al desarrollo y a flujos de trabajo orientados a tareas.
- Permite el bajo acoplamiento, lo que hace las aplicaciones independientes de los cambios aplicados en el funcionamiento interno de los servicios en niveles inferiores.
- Adaptar las aplicaciones ya existentes a las cambiantes condiciones empresariales y necesidades de clientes.
- Proporcionar aplicaciones de software ya existentes o legadas con interfaces de servicio sin cambiar las aplicaciones originales, lo que permite operar totalmente en el entorno de servicios. Este beneficio es fundamental para la presente investigación, ya que como se expuso en el capítulo correspondiente el dominio en estudio posee aplicaciones en funcionamiento que brindan recursos consumidos por distintas áreas organizacionales.
- Introducir otras funciones administrativas o de gestión de operaciones como confiabilidad, rendición de cuentas, seguridad entre otros atributos de calidad, independientemente de la función original, lo que aumenta su versatilidad y utilidad en el entorno de computación empresarial.

Para la correcta comunicación de los servicios, y según lo anteriormente descrito en que los servicios deben ser definidos, publicados y descubiertos para su utilización, los WS utilizan para su funcionamiento una serie de elementos estandarizados que posibilitan la interacción e intercambio, reforzando esta aseveración, Espina (2007), describe a un WS como un tipo de interfaz de un sistema de información, concretamente una interfaz que ofrece las funcionalidades del sistema en forma de servicios accesibles a través de los estándares abiertos utilizados en Internet como SOAP, XML, WSDL, UDDI o HTTP. Estos estándares categorizados y su utilización dentro de un WS se mencionan y describen a continuación:

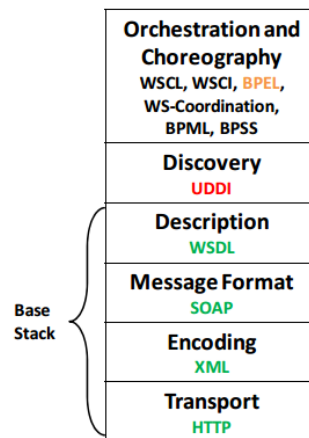


Figura N° 2. Estándares en Web Services.
Fuente: Lewis (2011 p.5).

XML

Valera (2011), lo define como un Lenguaje de etiquetado extensible que destaca por su simplicidad, pero estricto en su estructura, cuyo papel es fundamental en el intercambio de una gran variedad de datos, este lenguaje al igual que el HyperText Markup Language (HTML) tiene como origen al lenguaje Standard Generalized Markup Language (SGML) con el cual comparte la capacidad de separar el contenido y la presentación, representa en sí mismo la evolución lógica del

lenguaje HTML superando las limitaciones propias de este con un formato más simple que el SGML.

Para comprender su funcionamiento la W3C recomienda algunos puntos importantes a tener en cuenta: representa un estándar, con datos escritos en archivos de texto, si bien es similar a HTML en cuanto a marcas y atributos, estas no representan un conjunto fijo como en HTML, si bien está escrito en formato texto no es para ser leído, las marcas y atributos permiten delimitar conjuntos de datos a ser capturados e interpretados, no requiere licencias, es independiente de la plataforma y tiene un amplio soporte.

WSDL

El Lenguaje de Descripción de Servicios Web o Web Services Description Language (WSDL) es una especificación para describir y publicar los formatos y protocolos de un servicio Web de manera estandarizada, usualmente basada en gramática XML, fue desarrollado por Microsoft, Ariba e IBM y fue aceptada la v1.1 de la especificación WSDL por el W3C y publicada en su sitio Web.

El mismo describe todas las operaciones que permitan la interacción de un cliente con el servicio Web:

- Elementos de entrada y salida junto con los tipos asociados. Para invocar cada operación, el cliente necesita conocer los parámetros esperados por cada operación de la entrada y la salida esperada de cada operación.
- Ligación e información de la dirección para cada servicio. Para permitir el acoplamiento entre un solicitante y un servicio, especifica donde el servicio puede encontrarse y el protocolo de transporte que debe usarse para invocar el servicio.

En esencia, WSDL define un contrato que compromete al proveedor soportar la separación de la aplicación de la interfaz, no especifica cómo cada servicio se implementa. Proporciona el formato común para codificar y descifrar mensajes hacia

o desde prácticamente cualquier aplicación back-end, como CORBA, COM, EJB, JMS, MQ Series, sistemas ERP y así sucesivamente.

UDDI

Pudiendo ser descrito como un directorio de servicios Web, en el que es posible su publicación, caracterización y posterior recuperación, por su parte Espina (2007) considera que UDDI juega un papel clave dentro de la SOA y su implementación a través de WS, representando la fuente de los servicios, el lugar al que hay que acudir para conocer qué servicios ofrece cada organización, su tipo y como pueden ser utilizados, cumpliendo con el principio de publicabilidad de servicios llevando este concepto más allá creándose la idea de un repositorio, entendiéndose este no como un depósito que proporciona recursos, sino como un registro que contiene direcciones, por tal se puede vislumbrar como un registro público diseñado para almacenar de forma estructurada información sobre empresas y los servicios que éstas ofrecen.

A tal fin, Espina (ob.cit) describe su objetivo como el de definir un conjunto de servicios que describan y permitan el descubrimiento de proveedores de servicios (conocida como sección blanca), los servicios web que proporcionan (conocida como sección amarilla) y de la información técnica que permite el acceso a los mismos (conocida como sección verde).

Como principales beneficios de UDDI están las facilidades de reutilización y ahorro en tiempo y costos de implementación de servicios web, pues mediante el descubrimiento de servicios ya existentes y publicados por terceros, es posible mantener procesos de negocio de forma ágil y lograr interoperabilidad no solo a nivel interno de la organización, también entre diferentes empresas.

SOAP

Son las siglas de Simple Object Access Protocol, su razón de ser según Ruiz (2011) es proporcionar un mecanismo estándar de empaquetar mensajes abiertos y

extensibles para que las aplicaciones se comuniquen a través de la Web usando mensajes basados en XML, con independencia de sistemas operativos, modelos de objetos o lenguajes de programación.

A tal fin SOAP proporciona un envoltorio dentro del cual se define una cabecera y un cuerpo correctamente diferenciados en que cada elemento define una particularidad del mensaje, así la cabecera contiene datos de identificación del mensaje como remitente y destinatario y el cuerpo la carga de datos del mensaje, en este particular se puede hacer una analogía entre un mensaje SOAP y un paquete de red bajo el protocolo TCP. Entre las ventajas de SOAP se pueden mencionar:

- **Independencia de lenguaje de programación:** no proporciona una API por lo que su desarrollo permite la selección del lenguaje de programación a gusto de los desarrolladores según sus requerimientos o limitaciones propias del contexto de aplicación.
- **Disociación del protocolo de transporte:** en la especificación de SOAP no describe con un protocolo particular de comunicación ya que un mensaje SOAP no es más que un documento XML por tanto su envío puede ser realizado por cualquier protocolo capaz de transmitir texto.
- **Aprovechamiento de estándares:** sus principales desarrolladores se basaron de la reutilización y extensión de los estándares existentes para el logro de sus objetivos.
- **Interoperabilidad entre múltiples entornos:** su desarrollo basado sobre estándares existentes garantiza su interoperabilidad entre estos estándares y SOAP.

Por lo antes descrito se puede visualizar los servicios web como una aplicación del desarrollo orientado a servicios apalancados en la Web, y para su implementación se hacen necesarios estándares y protocolos que buscan cumplir con los requisitos antes descritos a los servicios, a saber definición, publicabilidad y uso,

estos elementos y su relación se pueden ejemplificar por medio de la siguiente ilustración:



*Figura N° 3. Interacción en una Arquitectura Orientada a Servicios Web.
Fuente: Ruiz (ob.cit) p.23*

En lo anteriormente expuesto, se visualiza la relación bidireccional entre proveedores y consumidores de servicios y sus repositorios, siendo en el contexto de los Servicios Web, instrumentados por los descritos WSDL, UDDI, y SOAP, todos con el lenguaje común XML.

Enterprise Service Bus

En apartados anteriores se definieron algunas alternativas para el logro de los objetivos de integración, en estas se estudiaron tecnologías monolíticas que integran (o pretenden integrar) las operaciones organizacionales, otras que incluyen la transformación e intercambio de datos para ser ubicados en repositorios integrados, así como la implementación de software intermedio (Middleware) para el logro de los objetivos de integración, la aplicabilidad de estas soluciones dependen del contexto de integración en el cual se requiera su uso, sin embargo y tomando en cuenta algunas

de sus características, se hace necesario implementar alternativas que tomen sus elementos positivos evitando en lo posible sus limitaciones.

En este particular, Vázquez (2010) describe nuevas respuestas al problema de la integración necesariamente más sofisticadas y ambiciosas, entre estas destacan:

- **Hub-and-Spoke:** en esta alternativa se conectan todos los sistemas a un punto central, diferenciándose así de la integración punto a punto, la conexión a este punto central se logra por medio de conectores ligeros e independientes de su tecnología particular, sin embargo la conexión a este punto central hace las operaciones en extremo dependientes de este, por tanto se toma como una limitación de esta solución.
- **Enterprise Message Bus:** para esta solución, no existe un punto central de conexión sino que los conectores están desacoplados gracias al intercambio de mensajes a través de un broker.

Tomando en consideración lo antes descrito, para la presente investigación se utilizara como alternativa un Enterprise Service Bus (ESB), definido por Hitpass (2013) como “...una herramienta de software para una arquitectura de integración corporativa y como lo indica su nombre, su principal funcionalidad es transportar y orquestar servicios.” (p.277), por su parte Bazán (ob.cit) lo define como “...una plataforma de software que da soporte a muchas funcionalidades resueltas a nivel de la capa de aplicación en los enfoques tradicionales de construcción de aplicaciones.” (p.39). Entre estas funcionalidades se pueden mencionar:

- **La comunicación:** el ESB se ocupa del ruteo de los mensajes entre los servicios.
- **La conectividad:** el ESB resuelve la conectividad entre extremos mediante la conversión de protocolos entre solicitante y servicio.
- **La transformación:** es responsabilidad del ESB resolver la transformación de formatos de mensajes entre solicitante y servicio.
- **La portabilidad:** los servicios serán distribuidos independientemente del lenguaje de programación en el que estén escritos y del sistema operativo subyacente.

- **La seguridad:** el ESB posee la capacidad de incorporar los niveles de seguridad necesarios para garantizar servicios que puedan autenticarse, autorizarse y auditarse.

Reforzando lo anterior, Caponi y otros referenciados por Valera (ob.cit) se refieren a esta tecnología como una plataforma de integración de aplicaciones basada en estándares, que combina mensajería, servicios ruteo y transformación de mensajes con la finalidad de conectar con confiabilidad un número significativo de aplicaciones. Por lo antes expuesto, se puede concluir que un ESB es en principio una herramienta de software, destinado a la integración de aplicaciones que tiene como función facilitar la oferta y demanda de servicios, gracias a la creación y la gestión de distintos flujos de datos, de manera totalmente transparente para los desarrolladores de aplicaciones. Actualmente se ha convertido en la forma de integrar la multitud de aplicaciones utilizadas a nivel corporativo, y se puede considerar la siguiente generación de herramientas EAI disponibles en el mercado.

En la configuración del ESB se detalla la lógica que orquesta la arquitectura, que básicamente se activa a partir de eventos automatizados o que son lanzados por las propias aplicaciones. La configuración puede interpretarse como una capa de abstracción, que facilita la creación de servicios a partir de esta, sin necesariamente conocer su funcionamiento interno.

Las características más destacadas de un ESB son las siguientes:

- Es una plataforma de integración basada en estándares de comunicación abiertos, con el objetivo de ofrecer una integración sencilla y eficiente, soporta los protocolos más utilizados (HTTP, FTP, JMS, SMTP entre otras.).
- Combina los paradigmas SOA y EDA, integrando la potencia de la arquitectura orientada a servicios con la versatilidad de la arquitectura dirigida por eventos.
- Está basado en la naturaleza síncrona de los servicios y asíncrona de los eventos, provee una capa de abstracción que facilita la asíncronia de la que

carecen las aplicaciones o servicios, de esta forma se añade una gran versatilidad a los servicios.

- Contiene herramientas para configurar el flujo de los mensajes, de forma que es posible transformar, duplicar, filtrar, y añadir seguridad y control de acceso a todas las transacciones, de esta manera se potencian las posibilidades a la hora de complementar los flujos de datos, enriqueciendo y automatizando los procesos de negocio.
- Permite incorporar módulos personalizados, esta funcionalidad se requiere ya que cada organización dispone de servicios totalmente diferentes, y cada uno de estos servicios tiene sus propios procedimientos, con el uso de ESB se pueden personalizar estos procedimientos mediante la adición de módulos con nuevos procesos y protocolos.

Entre las alternativas de ESB presentes en el mercado en la actualidad se tienen algunas de carácter comercial, tales como: Oracle ESB, IBM WebSphere, así como algunas alternativas Open Source como Mule, Apache ServiceMix, WSO2 entre los más destacados, para la presente investigación se tomaran como alternativas las que se encuentran en la categoría Open Source, ya que disponen de licencia de uso sin gastos económicos asociados.

Con base a los aspectos anteriormente expuestos, en el proceso de diseño y desarrollo, es común que a lo largo de multitud de experiencias acumuladas hay problemas que se repiten o que son análogos, es decir, que responden a un cierto tipo de patrón de aparición o comportamiento, no tiene mucho sentido por parte de un equipo de trabajo, abordar cualquiera de estas problemáticas comenzando desde cero, frente a estas problemáticas y la aplicación de soluciones comúnmente aplicadas, en las que se da la adopción de técnicas comunes que faciliten y optimicen su resolución, se desarrollan los patrones como herramientas de desarrollo.

Patrones

De forma general Junco (2008), describe un patrón como una manera estándar de realizar ciertas tareas y ayudan a conocer con un lenguaje común determinadas cosas que desarrollamos habitualmente para resolver un problema, siempre que su contexto lo haga relevante, resaltando así, que cada patrón es una regla de 3 partes, que expresa una relación entre un contexto, un problema y una solución, en este orden de ideas Buschmann y otros (2007) comentan:

“Los patrones le ayudan a construir sobre la experiencia colectiva de ingenieros de software experimentados. Estos capturan la experiencia existente y que ha demostrado ser exitosa en el desarrollo de software, y ayudan a promover las buenas prácticas de diseño. Cada patrón aborda un problema específico y recurrente en el diseño o implementación de un sistema de software.” (p.1).

De los conceptos anteriores es importante aclarar, según Welicki (2013) que un patron “...no es una solución en sí misma, sino la documentación de la forma en que construyeron soluciones a problemas similares en el pasado, lo cual permite una mejor gestión de la experiencia y transferencia de conocimientos. [Texto en línea]. Por lo antes expuesto se puede inferir que el carácter incremental de la acumulación de conocimientos permite a los especialistas agrupar soluciones a problemas comunes, que sirven como guías de aplicación a ser adaptadas a los detalles particulares de cada problemática planteada, convirtiéndose así en un patrón o molde para la acción a tomar.

La variedad de escenarios de aplicación de patrones, generan la necesidad de categorización de estos, entre estas categorías compiladas de autores como Buschmann y otros (2007), es importante mencionar:

- Patrones Arquitecturales.
- Patrones de Diseño.
- Idiomas.

Siendo resaltante para la presente investigación la primera categoría referente a los patrones arquitecturales.

Patrones Arquitecturales

Descritos por Buschmann y otros (ob.cit) como una plantilla de la estructura de sistemas de software, predefiniendo sus subsistemas y normando las relaciones entre ellos, representan el nivel más alto del sistema de patrones ya que ayudan a especificar la estructura fundamental de las aplicaciones, estos pueden ser sub clasificados de la siguiente manera:

Categoría	Descripción	Patrones
De estructura	Descomposición de tareas en sub-tareas cooperativas.	<ul style="list-style-type: none"> • Layers. • Pipes and Filtres. • Blackboard.
Sistemas Distribuidos	Proporcionan infraestructuras cuyos componentes se encuentran distribuidos en sub-sistemas o procesos diferentes.	<ul style="list-style-type: none"> • Broker.
Sistemas Interactivos	Permiten definir las estructuras de sistemas con interacción hombre-máquina.	<ul style="list-style-type: none"> • Model-View-Controller. • Presentation-Abstraction-Control
Sistemas Adaptables.	Proporcionan la infraestructura para sistemas que evolucionan como respuesta a las necesidades del negocio.	<ul style="list-style-type: none"> • Microkernel. • Reflection.

*Tabla N° 6. Categorización de Patrones Arquitecturales.
Fuente: adaptado de Buschmann y otros (ob.cit) y Valera (2011)*

Patrones de Integración

Al igual que algunos de los contextos ya estudiados en los que el uso de patrones posibilita la aplicación de soluciones a problemas comunes, en el estudio de procesos de integración pueden ser aplicados con el consecuente beneficio, a tal fin Hohpe y Woolf (2003), proponen algunos enfoques que permiten agrupar patrones según estilos integración:

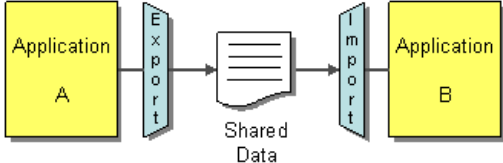
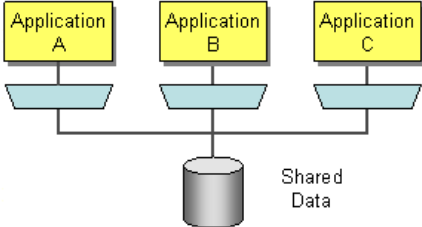
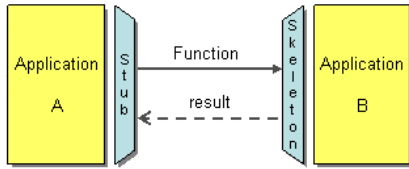
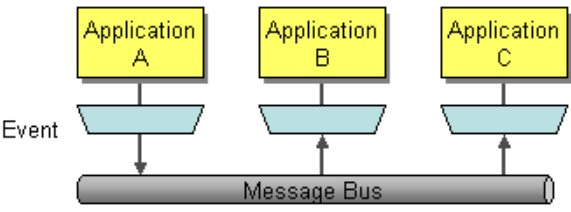
<p style="text-align: center;">File Transfer.</p> 	<p>Cada aplicación produce archivos a ser consumidos por otras aplicaciones y consume los producidos por otras aplicaciones.</p>
<p style="text-align: center;">Shared Database.</p> 	<p>Diversas aplicaciones se conectan a bases de datos en común.</p>
<p style="text-align: center;">Remote Procedure Invocation.</p> 	<p>Consiste en la publicación por parte de las aplicaciones de algunos de sus procedimientos a ser invocados remotamente por otras.</p>
<p style="text-align: center;">Messaging.</p> 	<p>El intercambio entre aplicaciones se establece mediante un sistema de mensajería donde se envían y reciben los recursos.</p>

Tabla N° 7. Estilos de Integración.
Fuente: Hohpe y Woolf (2003) p.64-73.

De los estilos anteriormente expuestos, se pueden resaltar que en el estilo File Transfer se hace necesaria la negociación de las partes en cuanto al contenido y formato del archivo de intercambio, siendo de carácter interno sus procedimientos de uso o transformación de los contenidos intercambiados, por tal razón si las aplicaciones mantienen el formato preestablecido, internamente pueden efectuar cambios en sus procedimientos lo cual garantiza el bajo acoplamiento entre aplicaciones, sin embargo en algunos escenarios lo que se quiere compartir son los datos en tiempo real, a tal fin surge el estilo Shared Data, que permite la interacción sobre datos comunes garantizándose así su vigencia en el tiempo, por otra parte en algunos casos no son solo los datos lo que se requiere compartir, algunos procesos efectuados por aplicaciones pueden ser insumo de otros procesos que se llevan a cabo por otras, en este caso el estilo Remote Procedure Invocation permite publicar – compartir dichos procedimientos para ser reutilizados por su entorno de integración.

Frente a las características de los estilos antes descritos en cuanto a la interacción frente a los datos o la funcionalidad (excluyentes entre sí), se hace necesaria una alternativa que aglutine estas opciones, a tal fin surge la mensajería como estilo de integración, en la que la colaboración se realiza en tiempo real con garantía de desacoplamiento, garantizándose que aun frente al escenario donde no existe disponibilidad de respuesta por diferencias en el rendimiento entre aplicaciones o por factores externos(energía, sobrecarga de procesos), los mensajes al ser asíncronos no requieren para su funcionamiento que el destinatario del mensaje emita respuesta inmediata, de la entrega del mensaje se ocupa el intermediario permitiendo así según Valera (ob.cit) que las aplicaciones se centren en lo que deben compartir y no en como compartirlo, por tal razón se establece un alto grado de cohesión de las aplicaciones con bajo nivel de acoplamiento entre ellas.

En este mismo orden de ideas, y tomando aspectos comparativos proporcionados por Hohpe y Woolf (ob.cit), se puede concluir que además de lo anteriormente descrito en cuanto a disponibilidad no exclusiva de datos y funcionalidades a compartir, el estilo mensaje proporciona mayor disponibilidad que

File Transfer, mayor encapsulamiento que Shared Data y mayor nivel de seguridad que Remote Procedure Invocation, por tal razón se considera que para la presente investigación es la alternativa más idónea.

Con referencia a lo anterior, Junco (2008) categoriza los patrones de integración referentes a mensajería en tres grupos de los cuales se tomarán 2 para la presente investigación:

- Patrones para servicios de mensajería.
- Patrones para manipulación de mensajes.

Patrones para servicios de Mensajería	
	<p>Message Channel: Consiste en conectar sistemas por medio de un sistema de mensajería en la cual las aplicaciones envían y reciben los mensajes hacia y desde el canal.</p>
	<p>Pipe & Filters: Comúnmente utilizado para dividir tareas grandes en más pequeñas favoreciendo la modularidad aplicando filtros y tuberías para la interconexión.</p>
	<p>Content-Based Router: Examina el contenido de un mensaje para distribuirlo por diferentes canales en función de los datos que componen dicho mensaje.</p>
	<p>Message Filter : Su objetivo es evitar la recepción de mensajes no deseados, a tal fin crea un conjunto de criterios (filtros) de manera que si el mensaje contiene propiedades que coinciden con los filtros especificados, este será enviado por un canal de salida, en caso contrario será descartado.</p>

	<p>Recipient List: Tiene como función re direccionar un mensaje a una lista de suscriptores del mismo.</p>
	<p>Splitter: Descompone un mensaje contenido de elementos repetitivos, con la finalidad de procesarlos por separado.</p>
	<p>Aggregator: Al contrario del anterior, su propósito es combinar varios mensajes con la finalidad de procesarlos como uno solo.</p>
	<p>Scatter Gatter: Permite frente a una petición combinar la respuesta de varias fuentes con la finalidad de seleccionar entre ellas.</p>

Tabla N° 8. Patrones para servicios de Mensajería
Fuente: Junco (ob.cit)

Patrones para manipulación de Mensajes.	
	<p>Message Translator: Frecuentemente utilizada junto con filtros, con la finalidad de transformar mensajes de un formato a otro.</p>
	<p>Envelope Wrapper: Utilizado para contener datos dentro del mensaje con compatibilidad de servicios de mensajería.</p>
	<p>Content-Enricher: Utilizado en casos en que el mensaje de origen requiere elementos que no dispone el emisor, por tal razón, recibe de otras fuentes los elementos faltantes.</p>

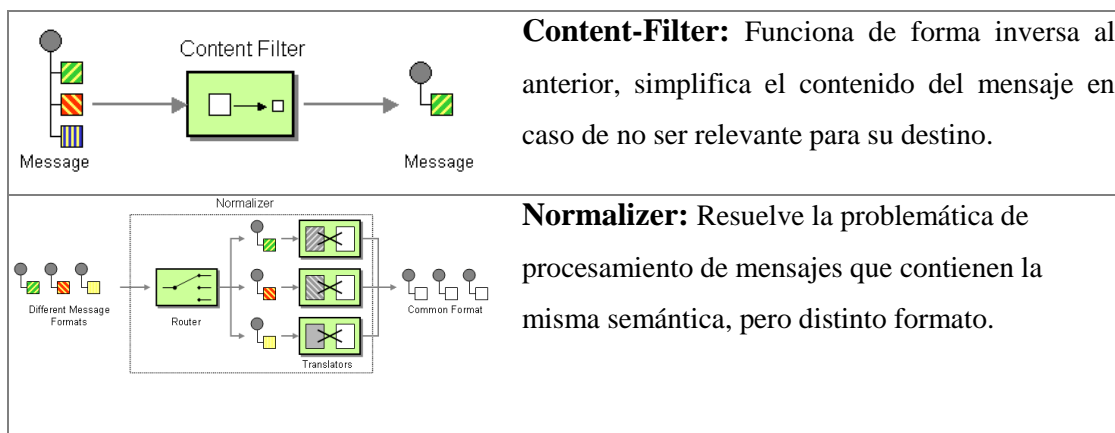


Tabla N° 9. Patrones para manipulación de Mensajes.

Fuente: Junco (ob.cit)

Es importante destacar que la categorización de patrones antes descrita por Junco (ob.cit), representa una visión particular donde resalta a su juicio, los de mayor importancia, y por ende son tomados en la presente investigación, no sin antes aclarar la existencia de muchos otros relacionados.

Dada la revisión teórica anterior, en la que destacan temas generales referentes a los sistemas, la información, las organizaciones sus necesidades de comunicación, el enfoque para descubrir sus elementos constitutivos, y de dichos elementos definir cuales representan vínculos de comunicación a ser definidos como servicios, la visión de integración organizacional, el enfoque arquitectural para valerse de los servicios en función del logro de los objetivos, y los medios tecnológicos para hacer posible dicha implementación, se plantea el presente trabajo de investigación que pretende mediante una propuesta de un Modelo de Integración de Aplicaciones utilizando la orientación a objetivos para la identificación de servicios y la ingeniería de métodos para normar la metodología a aplicar en estos procesos, brindar una alternativa para el logro de los objetivos de integración en entornos organizacionales, todo a ser aplicado en un área de estudio, que en este caso particular estará delimitado al Vicerrectorado Académico de la Universidad Centroccidental Lisandro Alvarado (UCLA) y sus unidades vinculadas.

CAPITULO III

MARCO METODOLOGICO

El presente capítulo, dentro del proceso de investigación denominado marco metodológico, corresponde al espacio reservado para el análisis y descripción de los métodos, técnicas y procedimientos a aplicar en el proceso de investigación, en este particular, Balestrini (2006) señala que el fin esencial del Marco Metodológico es:

Situar, en el lenguaje de investigación, los métodos e instrumentos que se emplearán en la investigación planteada, desde la ubicación acerca al tipo de estudio y el diseño de la investigación; su universo o población; su muestra; los instrumentos y técnicas de recolección de los datos; la medición; hasta la codificación, análisis y presentación de los datos. De esta manera se proporcionará al lector una información detallada acerca de cómo se realizará la investigación. (p. 126).

Tomando en cuenta lo anteriormente planteado, se presenta a continuación el contenido detallado del presente capítulo siguiendo para esto, la estructura de contenidos sugeridos por el autor citado.

Naturaleza de la Investigación

El hombre como individuo y sociedad, en su constante búsqueda de mejorar sus condiciones de vida y frente al carácter incierto de su destino, intenta identificar su entorno pretendiendo al conocerlo de antemano, manipularlo a sus necesidades y ambiciones, esta constante búsqueda de información de carácter acumulativo que incrementa constantemente su acervo, no solo se limita a las áreas particulares de estudio, también a los procesos inherentes a su obtención y organización, es por esto que algunos autores en clara intención de formalizar estos esfuerzos los definen como investigación, y cuando su realización es cuidadosa, rigurosa y planificada,

investigación científica, la cual partiendo de la realidad, la investiga, analiza, y formula hipótesis con la finalidad de fundamentar nuevas teorías (Tamayo, 1999).

En este mismo orden y dirección, la investigación según su propósito se ha dividido en dos vertientes, la pura cuyo objetivo es plantear teorías mediante la generalización de realidades, y la aplicada referente al contraste de esas teorías con la realidad palpable, así y frente a la variedad de enfoques a aplicar a los problemas de investigación, se subdividen en tipos tales como, histórica describiendo el pasado, descriptiva caracterizando lo actual y experimental pretendiendo definir el futuro. (Tamayo, ob.cit).

Frente a lo antes planteado, es evidente que una investigación como la presente donde se plantea la propuesta de un modelo que pretende brindar una solución a un problema identificable en distintos contextos de la realidad, se ubica dentro de la categoría de investigación aplicada de tipo descriptivo, a su vez enmarcada dentro los proyectos de investigación en estudios superiores, dentro de la modalidad de proyecto factible, definida por la Universidad Pedagógica Experimental Libertador (2010) en su libro titulado Manual de Trabajos de Grado de Especialización y Maestría y Tesis Doctorales como, “la investigación, elaboración y desarrollo de una propuesta de un modelo operativo viable para solucionar problemas, requerimientos o necesidades de organizaciones o grupos sociales” (p. 21).

Diseño de la Investigación

Definido el tipo y modalidad de la investigación, es pertinente continuar sus fases de estudio, a saber:

Fase I. Diagnostico

Con la finalidad de realizar cualquier investigación, el responsable de la misma debe en principio realizar una revisión exhaustiva de la documentación formal sobre el tema, adquiriendo los conocimientos necesarios e incluso contrastándolos con sus bases pre adquiridas, este proceso de revisión documental, sirve de base en el establecimiento del estado actual del arte, así como los distintos enfoques en que el tema ha sido estudiado por otros investigadores, estimulando su juicio crítico y permitiendo delimitar el alcance de su aporte al área en estudio.

Posteriormente, utilizando diversos medios de recolección de información, el investigador toma del área en estudio aquellos elementos que sirven de realidad a estudiar, identificando su estado actual, el cual permitirá evaluar inicialmente su pertinencia al área de estudio planteada. En este sentido, frente a la presente propuesta de un Modelo de Integración orientado a objetivos, el área de estudio propuesta está delimitada al Vicerrectorado Académico de la UCLA, presenta los problemas descritos en apartados anteriores propios de procesos de intercambio inadecuados entre aplicaciones, que condicionan en la actualidad practicas reñidas con conceptos básicos propios de la administración de los recursos, gerencia organizacional y desde el punto de vista técnico, la Ingeniería de Software.

Para lo antes descrito y como herramientas de recolección de datos, el investigador se valió de la observación directa, así como de la revisión documental de la organización y documentos técnicos de las unidades es estudio, que le permitieron establecer una visión global del ecosistema, muy útil en un enfoque orientado a objetivos, concluyéndose la pertinencia al tema en estudio, así como un esbozo inicial de los beneficios que generaría la propuesta.

Fase II. Factibilidad

Todo estudio de factibilidad, requiere ser presentado con todas las posibles ventajas para la organización, sin descuidar ninguno de los elementos necesarios para

la funcionalidad del proyecto. A tal fin, dentro de los estudios de factibilidad se establecen dos niveles en la presentación del estudio, a saber:

- **Requisitos Óptimos:** estos elementos deberán ser los necesarios para que las actividades y resultados del proyecto sean obtenidos con la máxima eficacia.
- **Requisitos Mínimos:** comprende los requisitos mínimos necesarios con que el proyecto debe contar para obtener las metas y objetivos, este paso trata de hacer uso de los recursos disponibles de la empresa para minimizar cualquier gasto o adquisición adicional.

Factibilidad Técnica

En el contexto de la investigación, la factibilidad técnica se refiere a la disponibilidad del recurso técnico para llevar a cabo la propuesta, dichos recursos se encuentran presentes en la unidad en investigación ya que la institución cuenta con herramientas de Hardware relativamente actuales en el tiempo, compuesta por Servidores, equipos de escritorio, así como la plataforma de red y comunicaciones que soporta las operaciones actuales. En cuanto a software, la organización dispone de licencia de los sistemas operativos y paquetes de software en funcionamiento, además de existir planes de migración para cumplir con las regulaciones gubernamentales propias del decreto gubernamental 3390, que obligan a la adhesión a software libre en las organizaciones públicas, por tal razón dentro del marco de la presente investigación se dará preferencia a aquellas herramientas que cumplan con la licencia requerida por el descrito decreto.

Factibilidad Operativa

La factibilidad Operativa se refiere a si la propuesta una vez desarrollada, representara beneficios prácticos para la organización, en tal sentido, se hace necesario que sus entidades consumidoras estén vinculados a la misma y consideren que su desarrollo se justifica y brinda valor agregado al negocio, a tal fin se toman en cuenta los factores referentes al rechazo de la misma, sea por resistencia al cambio o

por su complejidad implícita, en este particular, el personal en primera instancia, consumidor del producto está formado por desarrolladores del departamento de Sistemas, conocedores del dominio y futuros consumidores de servicios y beneficiarios de las mejoras planteadas, lo cual garantiza la operatividad de la propuesta.

Factibilidad Económica

Dado el carácter libre de las herramientas a utilizar en el desarrollo de la propuesta, así como la existencia de equipos de hardware necesario, para el desarrollo de la misma, no se requiere la adquisición de componentes de hardware adicionales ni tampoco licencias de uso de software, lo cual redundará en un ahorro cuantificable para la organización, siendo considerada la propuesta, económicamente factible.

En conclusión, se considera que la organización cuenta con más recursos que los mínimos requeridos para el desarrollo de la propuesta, razón por la cual se concluye que el estudio de factibilidad avala el desarrollo de la misma.

Fase III Diseño

Con la finalidad de cumplir con la presente propuesta de investigación, a continuación se describe cómo será abordada su resolución en función de los objetivos planteados:

Fase 1: Caracterizar el Modelo de Integración de Aplicaciones Orientado a Objetivos.

El objetivo fundamental de esta fase, es por medio del estudio, investigación y profundo proceso de análisis, caracterizar de forma general el modelo de integración orientado a objetivos deseable, destacando sus partes constitutivas, los objetivos y su vínculo con los procesos de negocio, así como su relación con los actores que involucrados en su ejecución participan en casos de uso, destacándose en este proceso de descomposición, la identificación de las relaciones entre estos y su posible

vinculación al concepto de servicios como elementos de integración, los aspectos arquitecturales que apoyen la integración orientada a servicios, así como las herramientas tecnológicas necesarias para su implementación, definiendo una alternativa útil para los procesos de integración planteados.

Fase 2: Diseñar un modelo de Integración de Aplicaciones Orientado a Objetivos empleando Ingeniería de Métodos.

Tomando en cuenta el objetivo alcanzado en la fase anterior, se plantea el diseño del modelo de integración orientado a objetivos, en el cual mediante los procesos definidos en el contexto de la ingeniería de métodos y la aplicación de una metodología ágil de desarrollo como eXtreme Programming (XP), se plantea la elaboración iterativa de artefactos entregables (Brinkkemper, 1996), en cuyo proceso de estructuración incremental se garantice el logro de los objetivos. Siendo destacable dentro de estos artefactos, el modelado de objetivos por medio del framework i* y su resultante definición de procesos de negocio a ser abordados por actores en casos de uso modelados en lenguaje UML, que permitirán por medio de su vinculación la identificación de servicios candidatos a ser administrados dentro de una SOA, por un Bus de Servicios Empresariales (ESB).

Fase 3: Emplear el modelo de integración resultante en un caso de estudio, en específico al Vicerrectorado Académico de la UCLA

Teniendo como insumo el modelo de integración definido en la etapa anterior, se procederá a aplicarlo en el área en estudio seleccionada, en este caso el Vicerrectorado Académico de la UCLA y sus unidades vinculadas, definiendo con claridad el alcance del análisis, delimitando dichas unidades en cantidad y pertinencia al objeto en estudio, elaborando en el enfoque de Ingeniería de Métodos, el conjunto de artefactos predefinidos, que posibilitaran la propuesta del modelo de integración en estudio.

CAPITULO IV

PROPUESTA DEL ESTUDIO

Posterior a lo desarrollado en capítulos anteriores en los que se definió el problema en estudio, la exploración de antecedentes de investigación, la revisión bibliográfica necesaria para la correcta comprensión del objeto en estudio, así como la metodología necesaria para su implementación, el presente capítulo tiene como finalidad explicar los elementos constitutivos de la propuesta, a tal fin, se tomara en cuenta el objetivo general de la misma, así como sus objetivos específicos generatrices de las fases de estudio descritas en el capítulo anterior.

Fase 1: Caracterizar el Modelo de Integración de Aplicaciones Orientado a Objetivos.

En el transcurso del planteamiento del problema en estudio, se mencionó la necesidad de clarificar aquellos elementos que cualquier solución de software a implementar debe cubrir para justificar su desarrollo, estos aspectos traducidos a requisitos comúnmente categorizados en funcionales (asociados a aspectos pragmáticos) y no funcionales (asociados a atributos de calidad), son generalmente definidos en las etapas iniciales de los proyectos de software, y administrados por disciplinas como la ingeniería de requisitos, la cual abarca los aspectos concernientes a los objetivos a satisfacer (Somerville, 2005), sin embargo como se mencionó con anterioridad y frente a la existencia de aplicaciones heterogéneas en el entorno organizacional, es común la elicitación de dichos requisitos en función de actores, o procesos propios de cada unidad, sin una visión integradora que permita su obtención en función de la organización por encima de sus unidades constitutivas.

El enfoque orientado a objetivos.

Es importante resaltar que la elicitación de requisitos es un proceso dinámico, que si bien se explora tradicionalmente en las etapas iniciales, está presente durante las etapas siguientes de desarrollo y mantenimiento de software, lo cual obedece a la necesidad de validación de la obtención de resultados requeridos y la adaptación constante a nuevas funcionalidades propias del dinamismo organizacional, en este particular, un enfoque orientado a objetivos permite modelar las necesidades de la organización con una visión más duradera, ya que los objetivos son más consistentes e invariables en el tiempo, al menos a corto plazo (Letier y Van Lamsweerde, 2002) lo cual representa un elemento de valor en el entorno operativo.

Por lo antes expuesto, en la presente investigación se plantea la orientación a objetivos como una alternativa que permita la obtención de los requisitos en función de objetivos organizacionales, de los cuales derivan objetivos operativos paliando así los efectos adversos propios del aislamiento de unidades intra-organizacionales antes descrito; sin embargo, es conveniente aclarar que la elicitación de requisitos planteada va más allá de la simple identificación de aquello que el sistema debe hacer, mediante la orientación a objetivos se plantea dicha identificación en función de integrar las unidades en interacción, posibilitando así el desarrollo de un entorno de producción en que las unidades brinden procesos definidos traducidos en servicios a ser consumidos por otras, sin desperdicio de recursos y con la adecuada planificación de responsabilidades.

Así, se requiere en principio el modelado de los objetivos en función de la organización y sus unidades constitutivas, tomando a tal fin algunos conceptos ya estudiados en las bases teóricas de la presente investigación, tales como el *refinamiento de objetivos*, en cuyo desarrollo se destacan como beneficios: la estructuración funcional y jerárquica organizacional, la reducción de la complejidad, el reconocimiento de alternativas de satisfacción de objetivos e identificación de conflictos de resolución (Van Lamsweerde, 2009).

En función de lo anterior, en el proceso de refinado de objetivos es común utilizar técnicas de descubrimiento, entre las cuales destaca el empleo de interrogantes que permiten clarificar el panorama a investigar, un ejemplo de este metodo se ilustra a continuación:

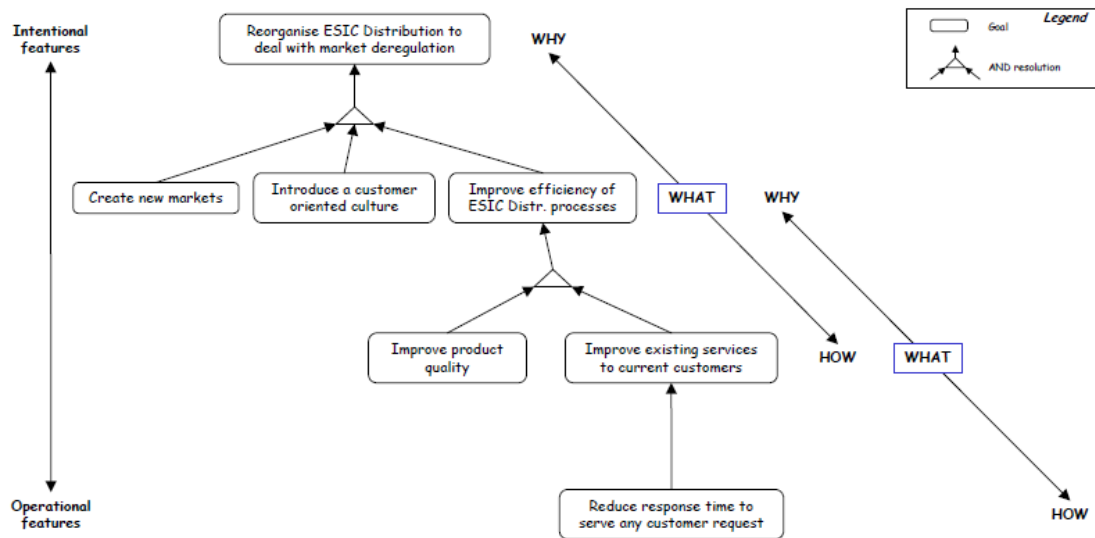


Figura N° 4. Identificación de objetivos por medio de preguntas Why - What - How.
Fuente: Kavakli (2006)

En el anterior modelo, el proceso de descubrimiento de objetivos se efectúa por medio de la ejecución iterativa de preguntas *Why – What – How* (Porqué - Qué - Cómo), donde el *Why* se plantea como clarificador de objetivos a nivel superior, el *What* para niveles intermedios y el *How* para el nivel inferior o entorno operativo, mediante este proceso es posible elaborar un diagrama inicial de objetivos que permite visualizar las jerarquías entre objetivos y sus relaciones desde los estratégicos a los operacionales.

De lo anterior, es importante destacar que tal como se mencionó con anterioridad, el enfoque orientado a objetivos permite entre otras cosas la identificación de alternativas de satisfacción (OR), por tal razón un enfoque como el planteado sirve como elemento clarificador de objetivos, siendo necesaria la

implementación de alternativas que permitan no solo plantear las relaciones entre los objetivos como simples relaciones excluyentes (OR), sino como relaciones complejas AND / OR enriqueciendo el modelado (Teruel y otros, 2011).

Por lo antes expuesto, se toma en consideración para el modelado de objetivos la utilización de algunos frameworks mencionados convenientemente en las bases teoricas, que permiten mediante su simbología ampliada el graficado de las distintas relaciones entre objetivos, posibilitando su refinamiento con sus respectivos beneficios.

Posterior al proceso de analisis, apoyados en enfoques comparativos tales el de Teruel (ob.cit), entre las alternativas estudiadas se tomara el framewok *i** para la presente investigación, no solo por su riqueza en simbología representativa, sino por su mejor adecuación a entornos colaborativos tales como el planteado en la presente investigación.

***i* *.**

Siendo una alternativa para el modelado basado en objetivos, representa en sí mismo un enfoque para afrontar el análisis de requisitos en las distintas etapas de desarrollo, con gran alcance en su aplicación en variedad de dominios, siendo una de las técnicas de modelado organizacional mejor fundamentadas en la actualidad (Estrada, 2008), enfocandose su representación en dos aspectos principales:

- Las relaciones entre la red de actores del negocio.
- Comportamiento interno requerido para satisfacer las dependencias entre los actores.

En consecuencia, permite la descripción de la organización como una red de actores con libertad de acción, pero con franca inter dependencia para el logro de los objetivos. Contiene en su especificación, un conjutno definido de representaciones graficas que permiten ilustrar partes principales del modelado de objetivos planteado.

Dentro de esta especificación según la i* Wiki destacan :

- **Actores:** representan una entidad que tiene objetivos estratégicos e intencionalidad dentro del negocio, lleva a cabo acciones para el logro de los objetivos, por ende representa una unidad a la que pueden ser atribuidas dependencias intencionales. Es destacable además la posibilidad de especialización de los actores en :
 - **Roles:** caracterización del comportamiento de un actor dentro de un contexto de dominio especializado, representa un estado de generalización.
 - **Agente:** representa un actor con características físicas concretas, indistintamente usado para representar personas, hardware, software con características concretas que lo diferencian de los demás, siendo un estado de especialización.
 - **Posición:** representa un punto intermedio entre Roles y Agentes, comúnmente utilizada para representar un conjunto de funciones desempeñadas por agentes para cubrir ciertos o determinados roles.

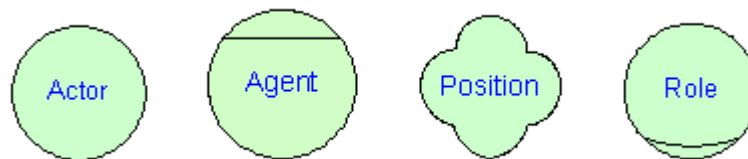


Figura N° 5. Representación de Actores en i*.

Fuente: http://istar.rwth-aachen.de/tiki-view_articles.php

- **Objetivos:** representa el interés estratégico de agentes de negocio, en i* pueden ser visualizados como HardGoals y SoftGoals, difiriendo en cuanto a la poca claridad entre los criterios de satisfacción de los SoftGoals, siendo comúnmente asociados a requisitos no-funcionales.

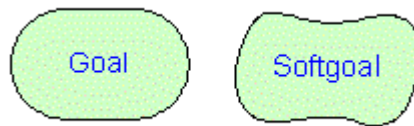


Figura N° 6. Representación de Objetivos en i*.
 Fuente: http://istar.rwth-aachen.de/tiki-view_articles.php

- **Tareas:** especifica un determinado curso de acción que permite parcial o totalmente el cumplimiento de una meta, representan operaciones, procesos entre otras.

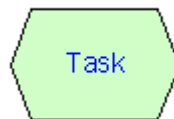


Figura N° 7. Representación de Tareas en i*.
 Fuente: http://istar.rwth-aachen.de/tiki-view_articles.php

- **Recursos:** representa una entidad física o información.

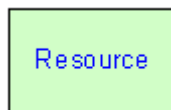


Figura N° 8. Representación de Recursos en i*.
 Fuente: http://istar.rwth-aachen.de/tiki-view_articles.php

- **Dependencia:** representa la relación entre N actores que interactúan para el logro de algún objetivo sin cuya participación conjunta, la resolución no es factible. En esta relación se plantean roles tales como:
 - **Dependee:** actor que depende en una relación de dependencia.
 - **Depender:** actor de quien se depende en la relación.

- **Dependum:** elemento intermedio de la relación de dependencia. Es destacable que este elemento puede estar representado por variedad de elementos dentro del modelo.

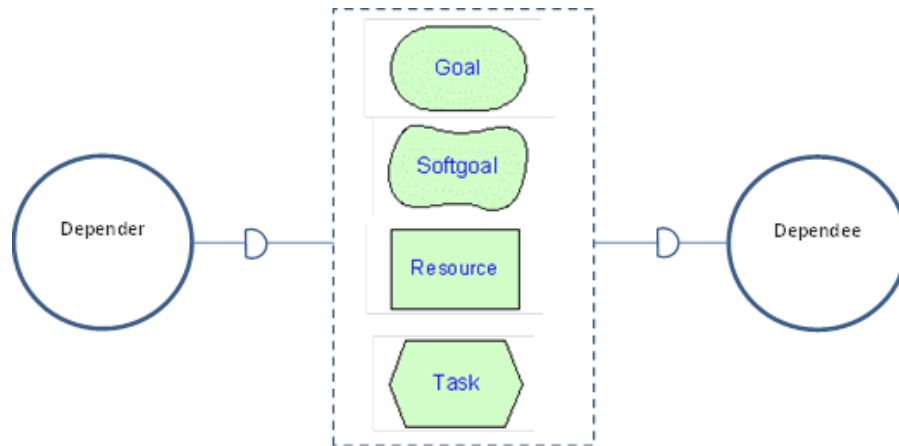


Figura N° 9. Relaciones de Dependencia en i*

Fuente: adaptado de http://istar.rwth-aachen.de/tiki-view_articles.php

- **Links:**

- **Meat-end links:** enlaces que indican la relación entre un extremo y un medio para alcanzarlo.
- **Decomposition links:** brindan enlaces a una tarea que puede ser descompuesta en N elementos.
- **Contribution links:** representa elementos que colaboran de forma positiva o negativa con la resolución de una tarea o proceso.

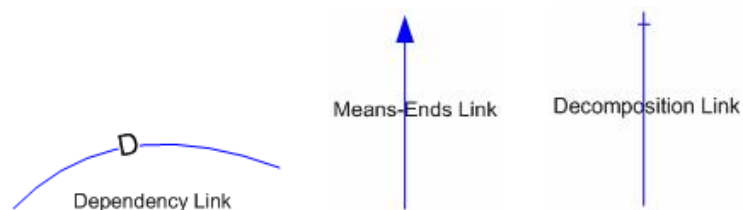


Figura N° 10. Links en i*

Fuente: Fuente: http://istar.rwth-aachen.de/tiki-view_articles.php

Los elementos antes descritos, contribuyen al modelado del negocio facilitando su elaboración y comprensión, representando un valor agregado en el entorno de desarrollo, por tal razón el framework i* y sus elementos constitutivos representan una herramienta importante a ser utilizada activamente en la presente investigación; sin embargo, y según lo expuesto por Teruel (ob.cit), en función de simplificar el diagramado puede ser utilizado intensivamente (combinando Figuras N° 7-11) a partir del nivel operativo organizacional previamente identificado. En este mismo orden y dirección, es importante destacar que existen otras herramientas que pueden ser utilizadas para identificar e ilustrar los actores y los procesos a los cuales están asociados, entre estos destaca por su validez y aplicabilidad los diagramas de caso de uso.

Casos de uso.

Siendo definido en líneas generales, como una descripción de los pasos y actividades que deberán realizarse para llevar a cabo algún proceso, un caso de uso sintetiza un evento y tiene un formato similar para la especificación de un proceso (Kendall y Kendall, 2005), en el contexto de la Ingeniería de Software, representa una secuencia de interacciones que se desarrollarán entre un sistema y sus actores, como respuesta a eventos definidos, son especialmente utilizados en el ámbito de la Ingeniería de Requisitos para ilustrar los requerimientos del sistema, al plasmar sus reacciones a eventos endógenos y exógenos.

En el mismo orden y dirección, para diagramar lo antes expuesto se requiere el uso de herramientas especializadas y sobre todo estandarizadas en función de uniformizar su uso e interpretación, a tal fin se cuenta con los *Diagramas de Caso de Uso*, como parte del *Lenguaje de Modelado Unificado* (UML) el cual es definido por Pressman (2010) como “un lenguaje estándar para escribir diseños de software... que puede usarse para visualizar, especificar, construir y documentar los artefactos de un sistema de software intensivo” (p.725), los diagramas de caso de uso representan una herramienta útil para ilustrar las interacciones de los actores con el sistema ya que

“ayudan a determinar la funcionalidad y características del software desde la perspectiva del usuario” (p.730).

Así, en función de la ya mencionada ilustración, se cuenta con un conjunto de simbologías, presentadas a continuación:

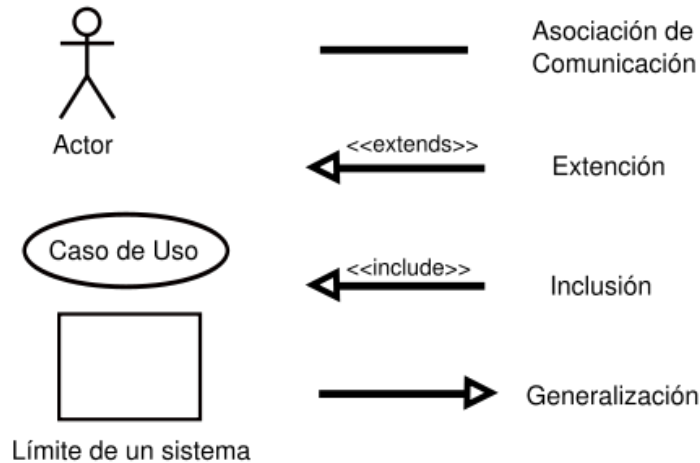


Figura N° 11. Notación de Diagramas de Caso de Uso.
Fuente: http://es.wikipedia.org/wiki/Diagrama_de_casos_de_uso

Entre los beneficios de esta técnica de modelado, destaca que permite especificar los requisitos en función de las necesidades de los usuarios y no en base a preconcepciones tecnológicas, siendo particularmente útiles en discriminar las interacciones entre los usuarios y procesos, permitiendo por ende visualizar colaboraciones y posibles colisiones con el consecuente desperdicio de recursos; sin embargo, es destacable su poca capacidad para ilustrar requisitos no funcionales, por tal razón para la presente investigación se plantea su uso en conjunto con el ya descrito modelado de objetivos con i*.

En función de lo anterior, se prevé el levantamiento de la información concerniente a aquello que se pretende integrar (unidades, sistemas, módulos), bajo la orientación a objetivos con el uso de:

- Framework i*.
- Diagramas de Caso de Uso.

Ambos, con la finalidad de clarificar el panorama que permita definir las interacciones entre las unidades organizacionales a integrar, pudiendo definir sus interacciones en función a servicios a ser posteriormente consumidos bajo un enfoque arquitectural propio de este paradigma.

Arquitectura Orientada a Servicios (SOA).

Es relevante recordar el concepto de servicio previamente expuesto por autores como Boccalari (2010), Aalbers (s.f) y Bazan (2009), como una funcionalidad de un sistema con características particulares que es publicada para su utilización bajo ciertas normas predefinidas, siendo particularmente resaltante su adaptabilidad, agilidad y flexibilidad, lo cual permite la planificación del desarrollo de soluciones con reutilización de recursos en ambientes organizacionales amplios, en comparación con otros enfoques de desarrollo (ver Tabla N° 5), sin embargo, en el proceso de elicitation de funcionalidades (servicios candidatos), no se trata de generar y exponer servicios de forma descoordinada o en exceso; por el contrario, lo deseable es crear conjuntos o dominios de servicios compuestos cuya exposición obedezca a las necesidades reales de aquello que se pretenda integrar (Papazoglou y Den Heuvel, 2006), por lo cual se refuerza la necesidad de un proceso de identificación de servicios como el propuesto en la presente investigación, como elementos principales para el desarrollo de soluciones.

En el mismo orden de ideas, existen algunos lineamientos o principios que permiten vislumbrar la adecuación o no del enfoque a un proceso integrador, posibilitando así el establecimiento de relaciones entre estos principios:

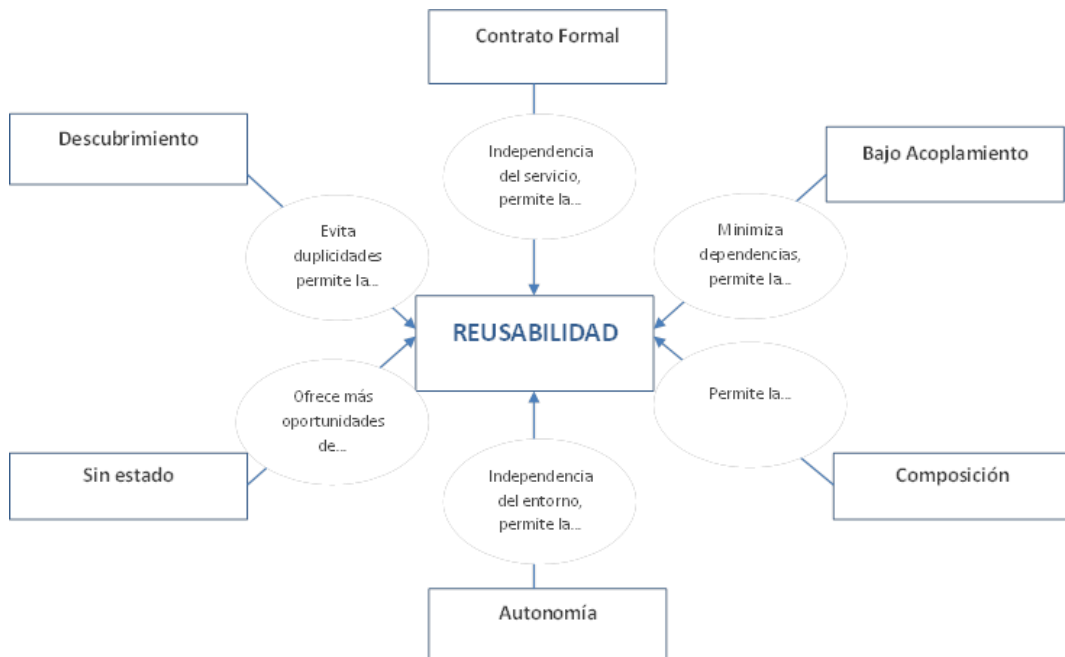


Figura N° 12. Principios de SOA.

Fuente: Adaptado de <http://arquitecturaorientadaaservicios.blogspot.com>

Siendo destacable en la relación expuesta, la preponderancia del concepto reusabilidad como integrador y elemento promotor de esta arquitectura de software, sin que esto implique el acoplamiento entre aplicaciones, descrito como una de los grandes enemigos de la integración, que ocurre cuando es difícil o imposible la modificación de alguna de sus partes sin afectar al resto (Valera, 2011).

Según se ha citado con anterioridad, la definición de una SOA implica analizar globalmente las necesidades de la organización, a fin de dar una respuesta tecnológica coordinada y reutilizable, por ende involucra pensar en la evolución de la organización a mediano y largo plazo (10 – 20 años), evaluando las necesidades actuales y futuras, con la finalidad de ajustar costes de desarrollo en la selección del paradigma a utilizar (Vázquez, 2010).

Así, una SOA puede ser finalmente definida como un marco de trabajo conceptual, que posibilita a las organizaciones desarrollar conglomerados de los objetivos organizacionales con su infraestructura de tecnología y de información,

permitiendo integrar los datos y la lógica de sus sistemas en operación, posibilitando la reusabilidad, escalabilidad, bajo acoplamiento y flexibilidad entre otros beneficios.

Sin embargo, frente a la competitividad propia de tiempos modernos y la necesidad de constante innovación tecnológica para responder a dichos retos, se plantea el estudio formal de los problemas inherentes a la integración o Enterprise Application Integration (EAI), las cuales fueron afrontadas inicialmente con sistemas punto a punto o mediante bases de datos, las cuales perduran en el tiempo, probablemente por su economía y simplicidad (Valera, ob.cit), siendo inconsistentes con el aumento de la complejidad tecnológica, lo cual condiciona el desarrollo de métodos de integración con mayor nivel de sofisticación y alcance.

Entre estas alternativas, es destacable el Hub-and-Spoke, que consiste en un punto central de conexión (en contraposición de punto a punto) por medio de conectores ligeros e independientes; sin embargo, con fuerte dependencia funcional de este Hub para las operaciones, por tal razón se desarrollan alternativas con estos y otros beneficios paliando sus debilidades, surgiendo los Enterprise Message Bus.

Enterprise Service Bus (ESB).

Es definida por Caponi, Rodriguez, y Zamudio (2008) en Valera (ob.cit) como “una plataforma de integración de aplicaciones basada en estándares, que combina entre otras cosas: mensajería, servicios, ruteo y transformación de mensajes, con el objetivo de coordinar y conectar de manera confiable un número significativo de aplicaciones”(p.76), a fin de ilustrar la evolución que representa esta tecnología, a continuación se compara frente a un enfoque tradicional de comunicación:

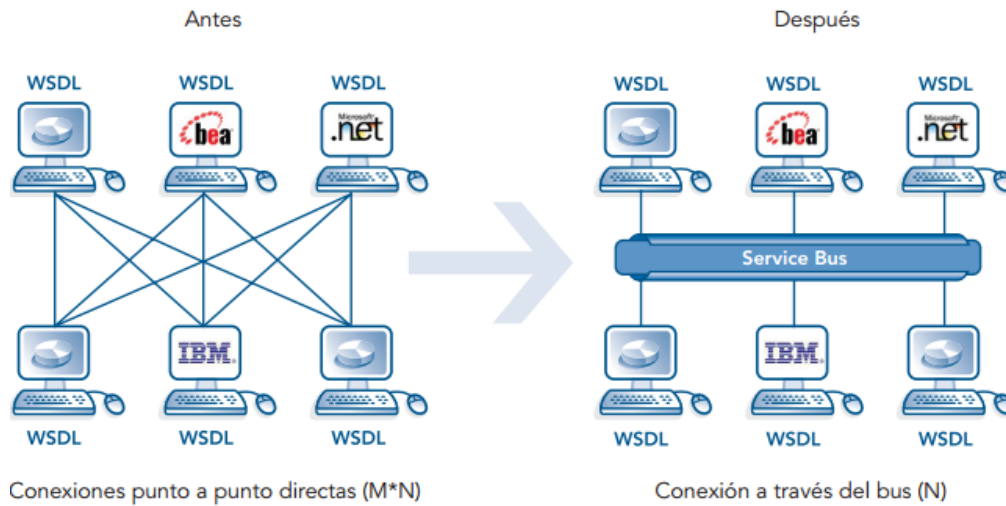


Figura N° 13. Evolución de Integración mediante ESB.

Fuente: <http://temariotic.wikidot.com/tema52aes>

Algunos autores como Valera (ob.cit), categorizan los componentes de un ESB:

- **Invocación:** proporcionando apoyo a los protocolos de transporte de forma síncrona o asíncrona.
- **Routeo:** cuya función radica en el envío de mensaje del origen al destino.
- **Transformación:** posibilitando el cambio de protocolos según requerimientos.
- **Mensaje:** unidad básica de información.
- **Orquestación / Coreografía:** Se refieren a procesos complejos de BPMN/BPEL.

De lo cual, es de gran importancia resaltar la mensajería como elemento tecnológico fundamental, siendo mencionada anteriormente en las bases teóricas entre los estilos de integración por Hohpe y Woolf (2003) e ilustrado en la Tabla N° 7, pudiéndose definir por los anteriores autores, como una tecnología que permite a las aplicaciones comunicarse de manera confiable, asíncrona y con alto grado de rendimiento, posibilitándose dicha comunicación por medio de canales dentro de los cuales viajan los paquetes de datos o mensajes, que son enviados por emisores y

recibidos por receptores en constante intercambio de roles, siendo a su vez convenientemente agrupados en patrones de mensajería antes descritas por Junco (2008) e ilustradas en las tablas N° 8 - 9, destacándose la importancia de los sistemas de mensajería que permiten administrar los mensajes posibilitando entre otras cosas, la ya mencionada asincronía entre emisor y receptor.

Así, es conveniente caracterizar los mensajes como algún tipo de estructura de datos con tres partes constitutivas:

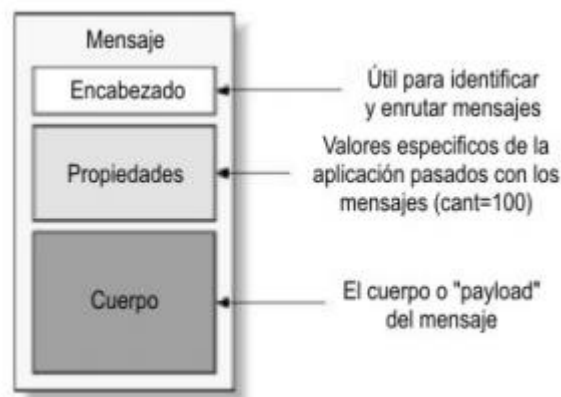


Figura N° 14. Estructura de Mensajes.

Fuente: Caponi, Rodriguez y Zamudio (2008) en Valera(ob.cit)

Siendo evidente su simplicidad y el rol de cada una de las partes dentro del proceso de comunicación; sin embargo, en el contexto de un ESB es destacable delimitar su uso como una herramienta para posibilitar una SOA; ya que por medio de estos, su tecnología se ocupa de la integración a nivel de plataformas, protocolos y formas de acceso, y la SOA propiamente a nivel de contratos y reutilización, (Valera, ob.cit).

Una visión del rol del un ESB dentro de una SOA y su impacto, se presenta a continuación:

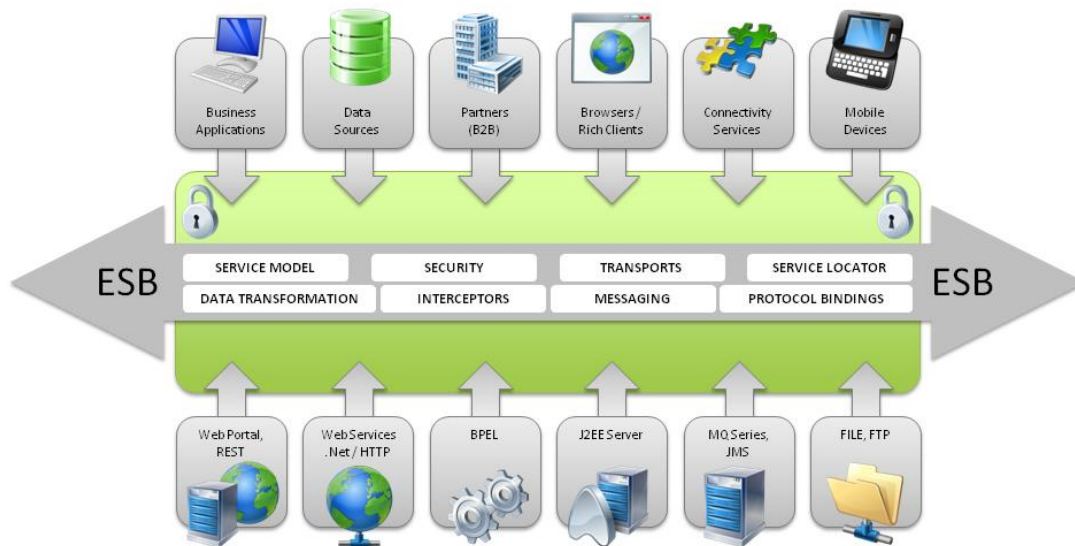


Figura N° 15. Rol de un ESB dentro de un proceso de Integración.
Fuente: <http://www.yenlo.nl/nl/enterprise-service-bus-in-soa-2/>.

Por lo antes expuesto, es importante precisar la importancia de un ESB dentro de una SOA, ya que permite la administración integral de los servicios en función del logro de los objetivos de integración planteados. Sin embargo, es importante mencionar que al igual que la variedad de alternativas para ilustrar los objetivos organizacionales, en el contexto de los ESB existen diversas alternativas comerciales a ser evaluadas para su adopción. En la presente investigación, en aras de lo anterior se tomaran estudios como el de Ciurana (2007) en Valera (ob.cit), en el que evalúa algunas herramientas que implementan un ESB, concluyendo como alternativas recomendadas Mule y WebSphere ESB.

Sin embargo, dadas las regulaciones propias de la adopción de alternativas de software presentes en la legislación nacional, superficialmente descrito en el estudio de factibilidad técnica, es importante destacar el licenciamiento de cada una de estas herramientas como factor decisorio; en este particular, Mule al ser Open Source se eleva como alternativa por adaptarse a las regulaciones antes descritas.

Además de lo anterior, Mule según Ramos (2012), contiene los beneficios propios de las tecnologías ESB siendo ligero y escalable, multiplataforma, con compatibilidad para distintos formatos de mensaje (SOAP, REST), disponible para diferentes proveedores, en torno a autenticación y autorización (Spring Security, JAAS, LDAP, CAS...), compatible para múltiples servidores (Tomcat, JBoss, WebSphere, WebLogic, Geronimo...) y de bases de datos (Oracle, MySQL, DB2, SQL Server, Derby), así como contenedores (Spring, EJB3), soportando diversos lenguajes de programación (Java, Groovy, Javascript, JRuby...), pudiendo ser además integrable con otros servicios como Facebook, Twitter, Amazon, Paypal entre otros, y por último, pero no menos importante su soporte y adhesión a los patrones de integración empresariales propuestos por Hohpe y Woolf (ob.cit).

Así mismo, es destacable identificar la metodología de desarrollo a aplicar en la presente investigación, en este particular y previa revisión de las metodologías empleadas en las investigaciones referenciadas en los antecedentes, se toma como alternativa general una metodología de desarrollo ágil, entre las cuales destaca eXtreme Programming (XP), siendo caracterizada como una metodología que permite el desarrollo de proyectos en grupos de trabajo pequeños, posibilitando el progreso iterativo e incremental, que permite la entrega continua de avances o mejoras para el logro del objetivo, siendo particularmente adaptable en su filosofía enfatizando la adaptabilidad sobre la previsibilidad de los requisitos; lo cual, constituye una visión útil en el desarrollo de un modelo como el estudiado que pretenda ser adaptable o implementable en distintos contextos de investigación.

Así, y precisando de una vez, el modelo propuesto está caracterizado por proveer un conjunto de elementos orientados al logro de la integración en entornos organizacionales, con predominancia en aplicaciones heterogéneas, para cuyo proceso está fundamentado en la elicitación de funcionalidades a ser traducidas a servicios utilizando el enfoque orientado a objetivos como elemento clarificador del panorama organizacional, para posteriormente proveer dentro de una arquitectura específica dichos servicios como elementos de integración, haciendo uso de un ESB

basado en mensajería como Mule funcionando como entidad administradora con todos los beneficios descritos en cuanto a flexibilidad, escalabilidad y eficiencia, todo lo anterior normado con buenas prácticas propias de la aplicación de patrones, así como metodologías de desarrollo como XP y la ingeniería de métodos para la ejecución de los pasos propios de la elaboración de un modelo de integración como el planteado.

Fase 2: Diseñar un modelo de Integración de Aplicaciones Orientado a Objetivos empleando Ingeniería de Métodos.

Con plena conciencia de lo planteado en las bases teóricas, en las que se describió someramente la Ingeniería de Métodos, a continuación se presenta una aproximación gráfica a sus partes constitutivas:

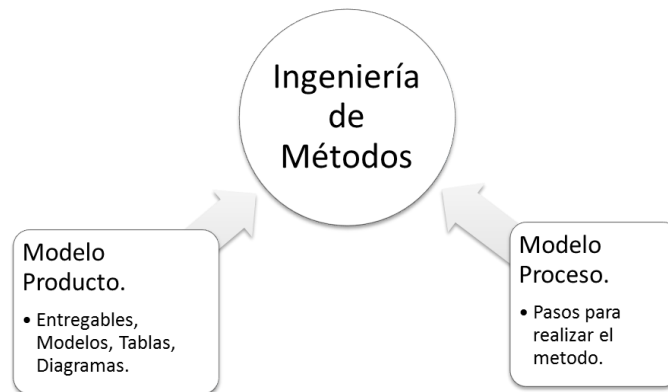


Figura N° 16. Ingeniería de Métodos.

Fuente: Adaptado de Iribarren (2012).

Destacandose de lo anterior, que la aplicación de esta disciplina no solo se ocupa de los productos, también de sus procesos de producción, tomándose para la presente investigación los recursos utilizados en estos procesos; así a continuación se presenta un diagrama que contiene la idea general del modelo, sus procesos y productos resultantes a ser descritos exhaustivamente en etapas siguientes.

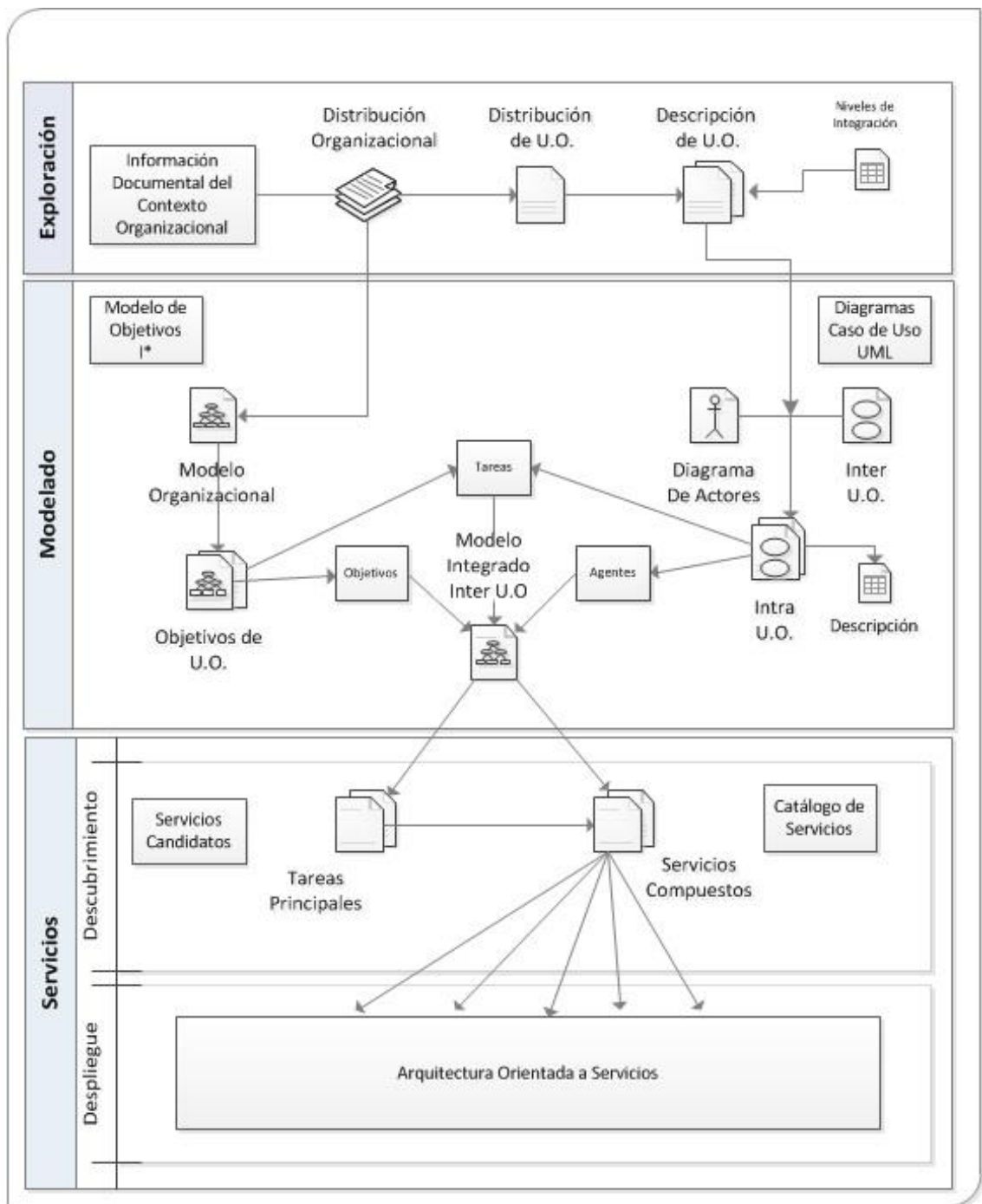


Figura N° 17. Actividades del Modelo de Integración Orientado a Objetivos.

Fuente: Autor de la investigación.

De lo anterior se desprende que el modelo contiene tres grandes etapas a saber: Exploración, Modelado y Servicios, siendo esta ultima contentiva de dos

etapas denominadas Descubrimiento y Despliegue, todas a ser definidas y detalladas a continuación.

Exploración.

El proceso de exploración, identificación y definición de la organización y sus *Unidades Organizacionales (UO)* a estudiar para el proceso de integración, se elabora a partir del siguiente conjunto de actividades en correspondencia con la Ingeniería de Métodos:

Actividad	Iteración	Proceso		Producto
		Tareas	Recursos	
Identificación, Caracterización Organizacional.	1	Describir la organización en su contexto operativo.	<ul style="list-style-type: none"> • Revisión Documental. • Observacion Directa 	<ul style="list-style-type: none"> • Descripción Organizacional. (Diagramas).
Identificar (UO).	2	Selección e identificación de la entidad a estudiar.	<ul style="list-style-type: none"> • Descripción Organizacional. • Observación directa del contexto en estudio. 	<ul style="list-style-type: none"> • Resumen de UO en estudio: • Identificador de UO (id). • Resumen de características.
Caracterizar (UO).		Descripción general de entidad en estudio.	<ul style="list-style-type: none"> • Revisión documental. • Entrevista no estructurada. 	
Determinar nivel de integración (UO).		Revisión de tecnologías middleware en uso.	<ul style="list-style-type: none"> • Matriz Nivel de Integración vs. Tecnología Middleware. (Tabla N° 4.) 	<ul style="list-style-type: none"> • Nivel de integración. (0 – 5).

Tabla N° 10. Identificación – Caracterización de unidades organizacionales.

Fuente: Autor de la investigación.

De lo anterior, es importante resaltar que las actividades se realizan de forma iterativa, en todas aquellas UO a incorporar en el proceso de integración, plasmándose sus resultados en el formato siguiente como producto resultante:

Identificador	
Nombre	
Ubicación	
Descripción	
Unidades organizacionales relacionadas	Nivel de Integración

*Tabla N° 11. Resumen de unidad organizacional en estudio.
Fuente: Autor de la investigación.*

Siendo destacable el identificador para su vinculación a artefactos posteriores, la descripción de la unidad como insumo inicial para la ulterior diagramación y modelado, las unidades organizacionales relacionadas en función de identificar inicialmente las distintas interacciones o necesidades de integración, así mismo el nivel de integración actual, como medida relativa de la importancia de la entidad dentro del entorno operativo que condicionó la implementación de algunas alternativas middleware caracterizadas con anterioridad.

Modelado.

Posterior al proceso de elaboración de los productos antes descritos, para cada una de las unidades organizacionales en estudio se elabora el modelo de objetivos y casos de uso según lo descrito en la figura N°17 y expandido en la siguiente secuencia de actividades a ser realizadas en forma secuencial:

Actividad	Iteración	Proceso		Producto
		Tareas	Recursos	
Diagramar Casos de Uso.	1	<ul style="list-style-type: none"> Identificar caso de uso principal entre UO. Identificación de Actores principales. 	<ul style="list-style-type: none"> Resumen de UO. Histórias de usuario UML 	<ul style="list-style-type: none"> Diagramas de casos de uso principal ente UO. Diccionario de Actores.
	2	<ul style="list-style-type: none"> Elaboración de diagramas de Casos de Uso 	<ul style="list-style-type: none"> Histórias de usuario UML 	<ul style="list-style-type: none"> Diagramas de casos de uso de UO en UML.
	3	<ul style="list-style-type: none"> Describir de forma detallada los casos de uso. 	<ul style="list-style-type: none"> Diagramas de Casos de Uso Diccionario de Actores. UML 	<ul style="list-style-type: none"> Cuadros descriptivos de Casos de Uso.

Tabla N° 12. Proceso de Modelado de Casos de Uso.

Fuente: Autor de la investigación.

Actividad	Iteración	Proceso		Producto
		Tareas	Recursos	
Modelar Objetivos.	1	<ul style="list-style-type: none"> Diagramar la organización 	<ul style="list-style-type: none"> Descripción Organizacional. Preguntas : <ul style="list-style-type: none"> Why – What - How I* 	<ul style="list-style-type: none"> Modelo General Organizacional (G.O)
	2	<ul style="list-style-type: none"> Diagramar las U.O preseleccionadas identificando: Objetivos, Tareas, Actores y Recursos. 	<ul style="list-style-type: none"> Resumen de UO. Diagramas de Caso de Uso. I* 	<ul style="list-style-type: none"> Modelo UO (G.O).

	3	<ul style="list-style-type: none"> • Diagramar en forma general los UO en interacción. 	<ul style="list-style-type: none"> • Diagramas de general organizacional. • Diagramas UO. • I* 	<ul style="list-style-type: none"> • Modelo Final de Interacciones. (G.O)
--	---	---	---	--

Tabla N° 13. Proceso de Diagramado de Objetivos.

Fuente: autor de la investigación.

Siendo conveniente una vez más, aclarar el carácter iterativo e incremental en la elaboración de lo anterior, sobre todo tomando en cuenta la depuración progresiva producto del análisis y la superposición de los resultados con la finalidad de lograr la correspondencia entre elementos constitutivos de cada lenguaje, a saber:

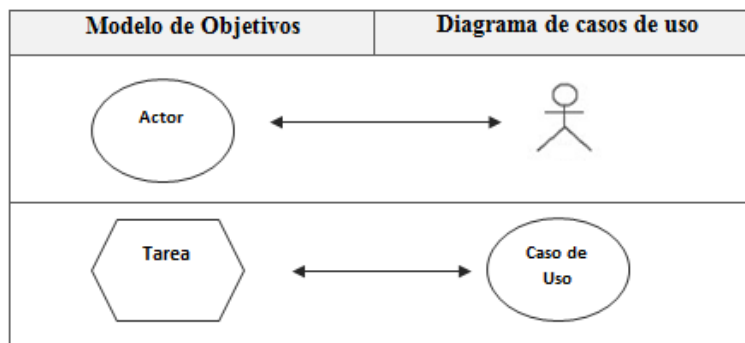


Tabla N° 14. Correspondencia entre Modelos y Diagramas.

Fuente: Autor de la investigación.

Resultando en la identificación de conflictos y alternativas de resolución a los procesos o casos de uso, en función de la satisfacción de los objetivos de cada unidad organizacional en estudio.

Servicios.

Ahora bien, hasta este punto se han tratado a las unidades organizacionales como elementos aislados, en torno a los cuales se clarifico su funcionamiento interno, es el momento de evaluarlos en función de sus interacciones; a tal fin, con los productos anteriormente generados y mediante la superposición de diagramas, principalmente a nivel de objetivos técnicos u operacionales, se efectúa el proceso de

identificación de procesos o casos de uso en cuya resolución participen actores presentes en diversas unidad organizacionales:

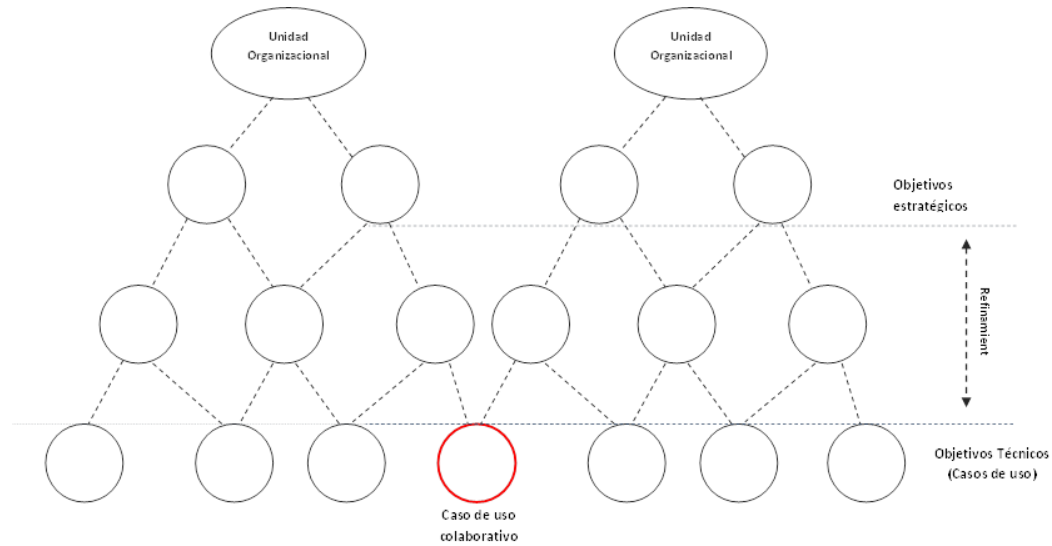


Figura N° 18. Superposición de diagramas.
Fuente: autor de la investigación.

Lo anterior con la finalidad de evidenciar su colaboración en la satisfacción de objetivos particulares, a este proceso se le denomina *Identificación de servicios* y su ejecución se lleva a cabo siguiendo la siguiente secuencia:

Actividad	Iteración	Proceso		Producto
		Tareas	Recursos	
Identificar servicios candidatos.	1	<ul style="list-style-type: none"> Identificación de servicios candidatos observando objetivos generales y las tareas asociadas para su satisfacción. 	<ul style="list-style-type: none"> Modelos GO. Diagramas de Casos de Uso. 	<ul style="list-style-type: none"> Conjunto de Servicios candidatos.
Elaborar catalogo de servicios.	2	<ul style="list-style-type: none"> Depurar servicios candidatos en función de combinación, agregación y simplificación. 	<ul style="list-style-type: none"> Servicios Candidatos. 	<ul style="list-style-type: none"> Catalogo de Servicios.

Tabla N° 15. Identificación de Servicios.
Fuente: autor de la investigación.

De lo cual es resaltante agregar según Earl (2005) en Valera (ob.cit) que las tareas principales servirán a priori como servicios candidatos. Así y posterior a estos procesos y con el correspondiente producto resultante, a saber conjunto de servicios candidatos y en correspondencia a lo planteado por Papazoglou y Den Heuvel (2006), en torno a la construcción de dominios de servicios y su combinación para obtenerlos con granularidad gruesa, una vez mas se analizan en función de identificar lugares comunes, colaboraciones, correpondencias o conflictos que permitan depurar el catalogo de servicios final a ser expuesto en el siguiente artefacto como resumen del producto generado:

Objetivo	Interacciones			Servicio
	Actores	Unidades Involucradas	Casos de uso	

*Tabla N° 16. Catálogo de servicios
Fuente: autor de la investigación*

El producto anterior, mantiene su relación con la estructura planteada en la Ingeniería de Metodos, contiene no solo los servicios resultantes (productos), también los objetivos a satisfacer con estos servicios así como los elementos involucrados en su satisfacción (procesos), tales como: actores y casos de uso que reflejan las interacciones de las unidades organizacionales estudiadas, para dar respuesta inicial a las necesidades de integración del contexto en estudio; lo anteriormente expuesto puede ser resumido en el siguiente diagrama de actividades:

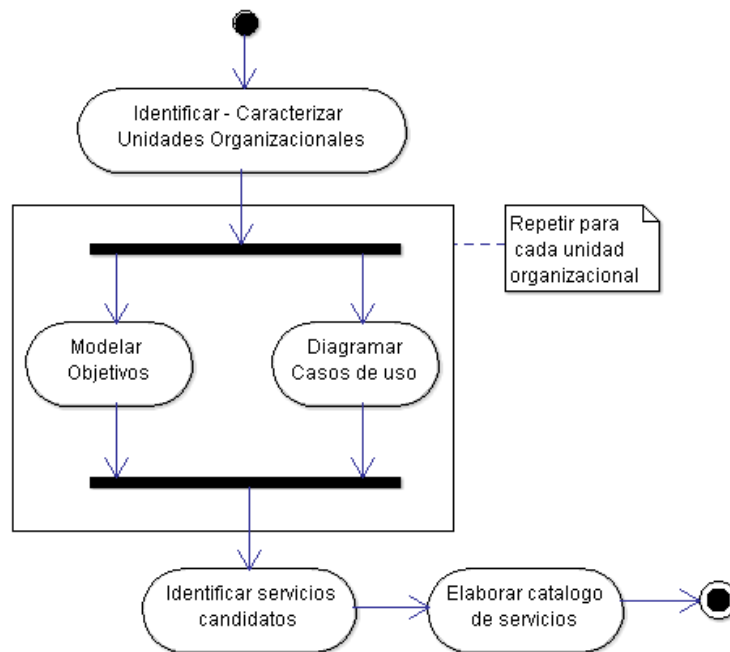
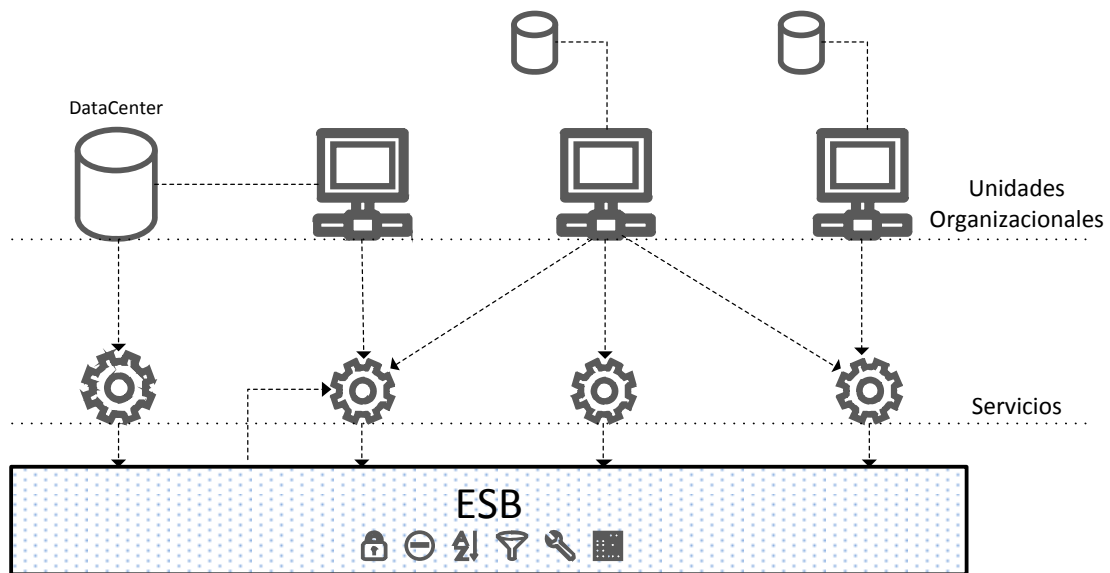


Figura N° 19. Secuencia de actividades Elicitacion de servicios.
 Fuente: autor de la investigación.

A este nivel y con los artefactos generados, es tiempo de establecer el modelo arquitectural que posibilite la integración de las unidades organizacionales en estudio, siendo la arquitectura orientada a servicios (SOA), el enfoque preseleccionado a tal fin. En este particular, y por lo antes estudiado dentro del contexto de una SOA, su aplicación dentro del modelo propuesto se lleva a cabo mediante la implementación de un ESB, dicho componente orquestador de servicios a ser desplegado dentro del ambiente en estudio, se puede derivar de su visión ampliada ilustrada en la figura N° 16, dando como resultado la siguiente aproximación que contiene los elementos antes identificados:



*Figura N° 20. Aplicación ESB en modelo propuesto.
Fuente: autor de la investigación.*

De lo anteriormente ilustrado, es destacable el origen heterogeneo de las unidades organizacionales a integrar, las cuales pueden ser sistemas en operación, fuentes de datos, servidores de contenidos por mencionar algunos, manteniendo su independencia funcional, así como los servicios que pueden provenir de unidad organizacionales individuales y/o compuestas, todas a ser gestionadas por el ESB como componente encargado de las dinámicas concernientes a su funcionalidad en torno a patrones de mensajería antes descritos, tales como servicios, manipulación y orquestación de mensajes (ver Tablas N° 8 y 9), lo cual permite un ambiente de integración escalable con mínimo impacto funcional de implantación en el entorno de aplicación.

Fase 3: Emplear el modelo de integración resultante en un caso de estudio, en específico al Vicerrectorado Académico de la UCLA.

Posterior a las etapas anteriores en las que se caracteriza en primer lugar y diseña en segunda instancia el modelo propuesto, y en correspondencia a las fases de elaboración definidas en el capítulo 3, corresponde la aplicación del modelo resultante en un caso de estudio, siendo seleccionado a tal fin el Vicerrectorado Académico de la Universidad Centroccidental Lisandro Alvarado (UCLA) y su entorno de operación; por tal razón, a modo de introducción es conveniente definir y describir de forma general la institución y su estructura departamental, dentro de la cual están insertas las unidades organizacionales en estudio.

Exploración.

Iteración 1.

Identificación, caracterización Organizacional: Según la misión de la organización publicada en el sitio web institucional, la UCLA es una organización educativa que se destaca por ser una institución con calidad académica y pertinencia social, reconocida por su aporte al desarrollo local y nacional, fundamentada en una cultura institucional orientada hacia el progreso y la eficiencia (UCLA, 2008), por tal razón y para el correcto desarrollo de sus operaciones, internamente se encuentra estructurada en unidades claramente diferenciadas, presentando al igual que otras organizaciones fuertes vínculos con su entorno, que garantiza el constante intercambio intra y extra institucional.

Lo anteriormente expuesto, genera necesidades en función de los objetivos organizacionales, resueltas en su mayoría aplicando técnicas poco adecuadas u optimizadas, condicionando los efectos negativos descritos en la presente investigación, tales como Sistemas Heterogeneos, Islas de Información y falta de confiabilidad en la información. A continuación se presenta el organigrama institucional en el que se ilustran sus unidades constitutivas:

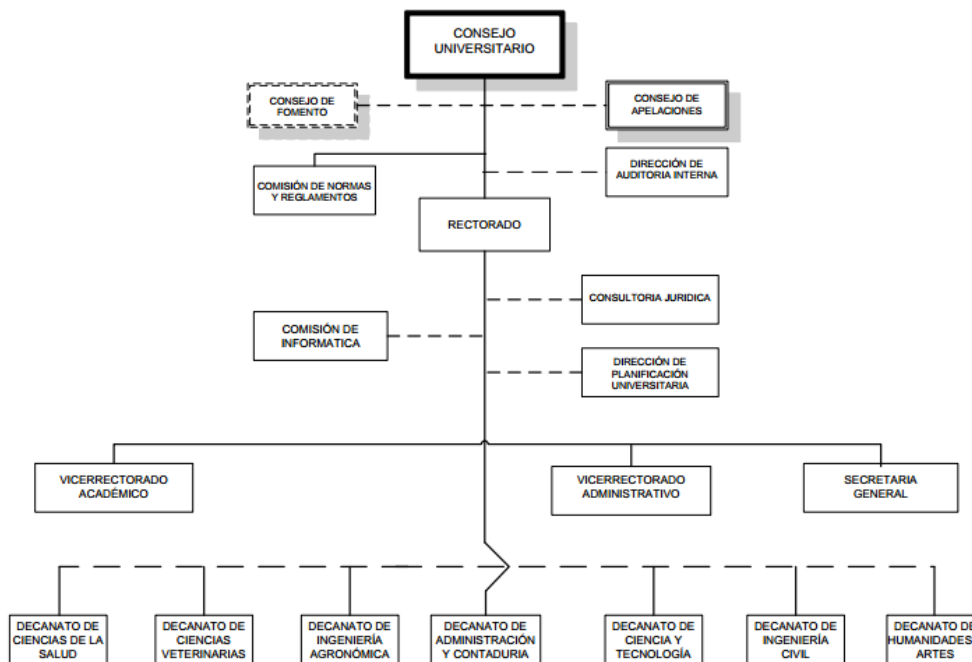


Figura N° 21. Organigrama de la UCLA.

Fuente: <http://www.ucla.edu.ve/valores/organigramas/Basica%20oct%202010.pdf>

Resaltando para la presente investigación, el Vicerrectorado Académico dentro del contexto organizacional, por lo cual es importante destacar sus partes constitutivas así como su relación cercana con otras unidades en función de sus operaciones regulares:

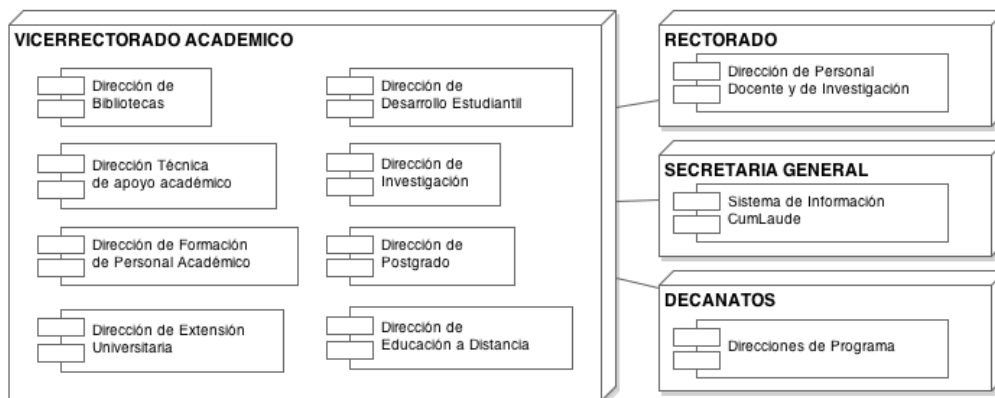


Figura N° 22. Distribución de Vicerrectorado Académico y unidades relacionadas.

Fuente: autor de la investigación.

Iteración 2.

Selección e Identificación de las Unidades Organizacionales: En función de la aplicación del modelo propuesto, se tomarán un sub conjunto de estas unidades organizacionales que representen a su vez un conglomerado de funciones y relaciones que permitan ejemplificarlo, así y aplicando de una vez el modelo ajustado a los pasos descritos en su fase de exploración (ver Tabla N° 10), se ejecuta la actividad correspondiente a la identificación y descripción de las unidades organizacionales en estudio.

Identificador	UO-01
Nombre	Vicerrectorado Académico
Ubicación	Sede Rectoral
Descripción	
Es la autoridad responsable de supervisar y coordinar las actividades académicas de docencia, investigación y extensión de la Universidad en los niveles de postgrado y pregrado, formulando, planificando y evaluando las políticas y estrategias dirigidas hacia el desarrollo permanente del conocimiento y a la búsqueda de la excelencia de dichas actividades (UCLA, ob.cit).	
Funciones:	
<ul style="list-style-type: none"> • Impulsar políticas orientadas a las labores propias del personal docente y de investigación. • Contribuir con dinamismo en la formación integral del estudiante para responder a las necesidades educativas regionales, nacionales e internacionales. 	
Unidades organizacionales relacionadas	Nivel de Integración
Vicerrectorado Académico	
Dirección de Extensión. (D.E.U)	1
Dirección de Investigación. (C.D.C.H.T)	0
Dirección de Formación de Personal Docente y de Investigación (D.F.P.A)	1
Rectorado	
Dirección de Personal Docente y de Investigación (D.P.D.I).	1

Tabla N° 17. Identificación – Caracterización Vicerrectorado Académico

Fuente: autor de la investigación.

Dada la relación entre unidades organizacionales antes descrita se hace necesario el análisis de la Dirección de Personal Docente y de Investigación.

Identificador	UO-02
Nombre	Dirección de Personal Docente y de Investigación (D.P.D.I.)
Ubicación	Rectorado (antiguamente Vicerrectorado Académico)
Descripción	
Es la unidad encargada de asesorar la correcta aplicación y cumplimiento del marco legal que rige la educación superior, como ente asesor para las autoridades universitarias y el personal docente, haciendo seguimiento y custodia de los datos y su movimiento académico- administrativo.	
Funciones:	
<ul style="list-style-type: none"> • Elaborar y proponer políticas y procedimientos para los mecanismos de ingreso del personal docente. • Registro y actualización de los datos inherentes a los expedientes docentes. • Supervisar el control, registro y la correcta clasificación del personal docente, ajustados a las normas y procedimientos establecidos. • Autorizar y registrar el préstamo de expedientes y elaboración de documentos a solicitud de las partes interesadas. 	
Unidades organizacionales relacionadas	Nivel de Integración
Vicerrectorado Académico	
D.F.P.A	1
D.E.U	1
C.D.C.H.T	0
Secretaría General	
Sistema de Registro Académico (CUM LAUDE)	0
Vicerrectorado Administrativo	
Recursos Humanos	0

Tabla N° 18. Identificación – Caracterización D.P.D.I.

Fuente: autor de la investigación.

De lo anteriormente expuesto se evidencian algunas relaciones entre unidades, destacando el nivel de integración que se encuentra entre los valores:

- 0: intercambio por formatos intermedios (Batch) determinando falta de actualización.
- 1: orígenes de datos (DBAT)

Por tal razón, se infiere que los intentos de integración a la fecha se centran en el intercambio de datos más no de procesos, con el consecuente desperdicio de recursos. Lo anteriormente expuesto se ilustra a continuación:

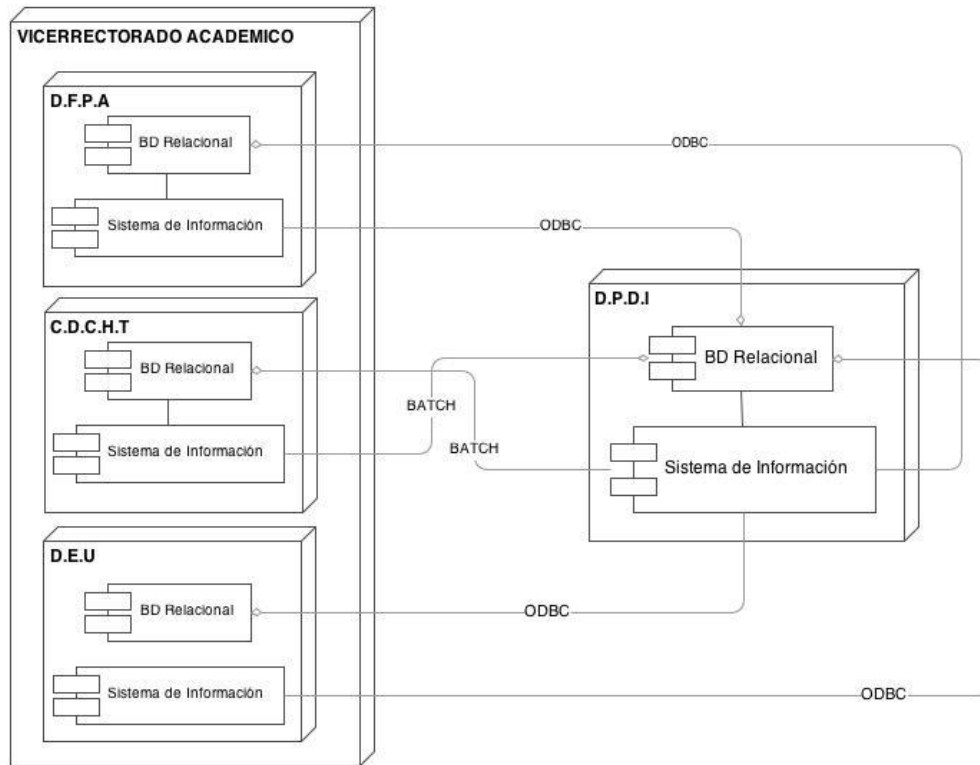


Figura N° 23. Intercambio de información entre unidades organizacionales estudiadas.
Fuente: autor de la investigación.

Sin embargo, es importante destacar que el nivel 1 denota la necesidad de comunicación y el esfuerzo en lograrla, lo cual se toma como un aspecto clarificador de las interacciones actuales entre unidades y su precedencia para el presente modelado.

Modelado.

Así y continuando con el proceso, corresponde la aplicación de las tareas descritas en el Cuadro N° 13, correspondientes a la elaboración de Diagramas de Casos de Uso y Modelado de Objetivos.

Casos de Uso

Iteración 1

Inicialmente se evidencia la interacción entre las unidades organizacionales UO-01 – UO-02, previamente caracterizadas, pudiendo ilustrarse sus casos de uso de forma simplificada a continuación:

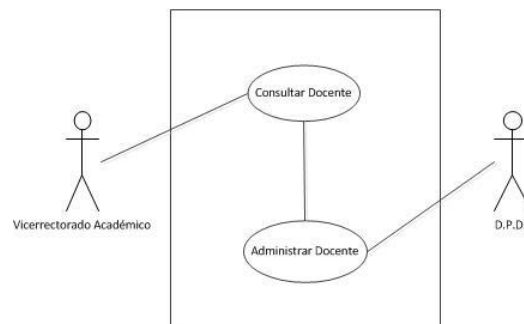


Figura N° 24. Casos de uso entre Vicerrectorado Académico y D.P.D.I.
Fuente: autor de la investigación.

Así como el diccionario de actores, agrupado por sus áreas de influencia, lo cual determina sus privilegios o alcance:

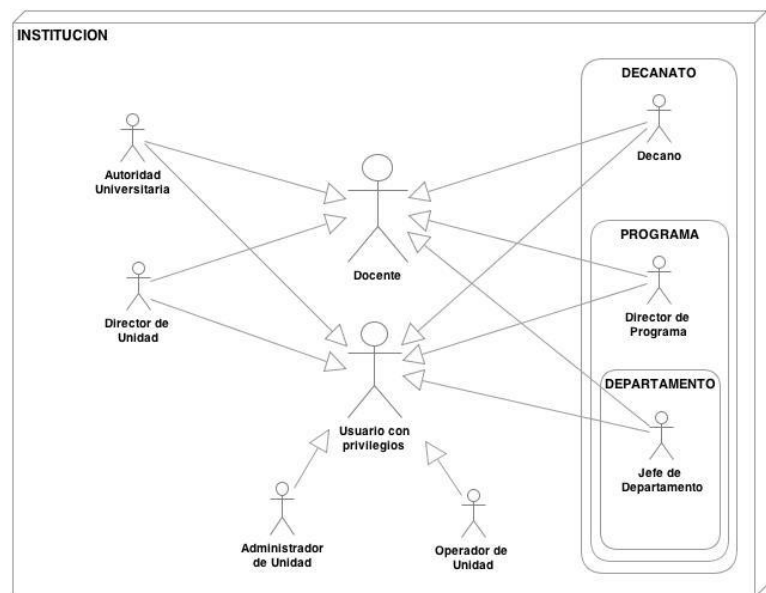


Figura N° 25. Diagrama de actores.
Fuente: autor de la investigación.

De lo anterior, es importante aclarar que el alcance de cada usuario depende del contexto en el cual este incluido, por ejemplo el Rector al extender de usuario administrador puede participar en sus casos de uso con alcance a todos los docentes de la institución; así mismo, un Director de programa solo puede consultar lo atinente a su programa y así sucesivamente, hasta llegar al Docente sin privilegios que solo posee alcance para sus propios datos.

Iteración 2

Con los subproductos anteriores se elabora el diagrama de Casos de Uso propios del Vicerrectorado Académico y algunas de sus unidades constitutivas:

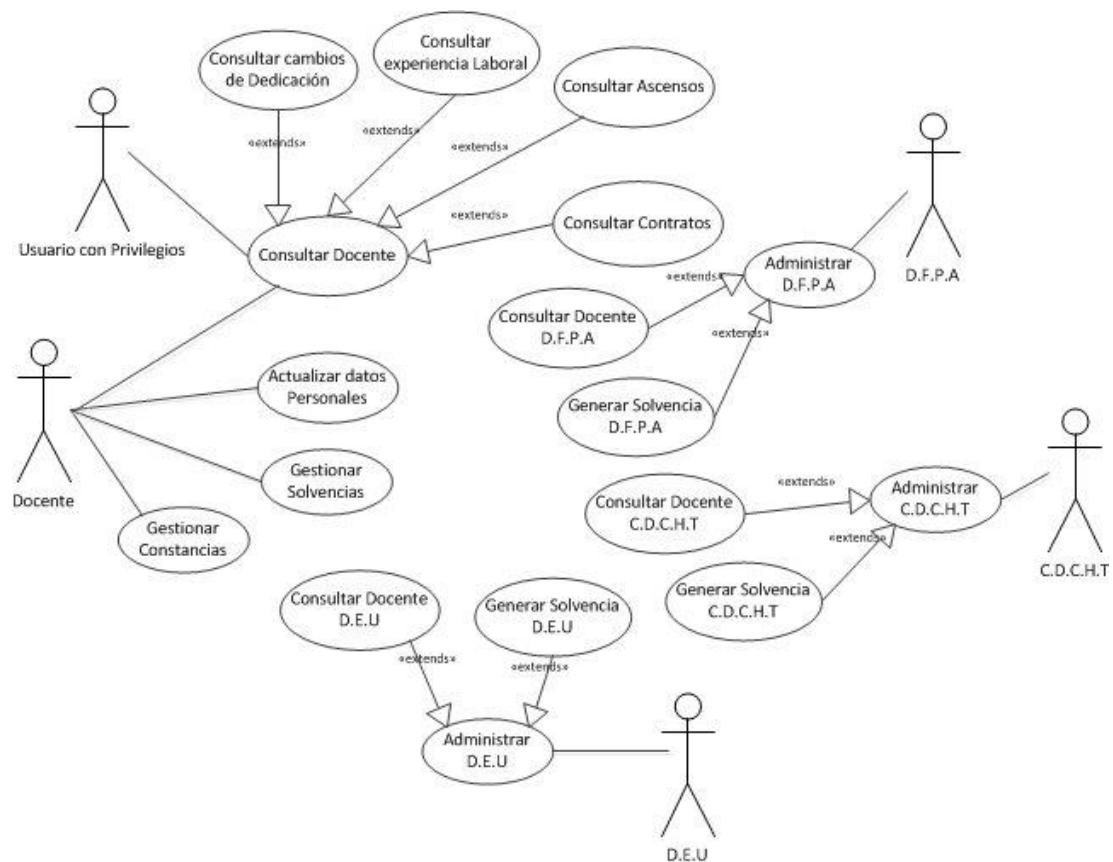


Figura N° 26. Casos de Uso Vicerrectorado Académico.

Fuente: autor de la investigación.

Iteración 3

A continuación se elabora la descripción detallada de los casos de uso hasta ahora expuestos:

Ref.:	UC-1
Caso de Uso:	Consultar docente.
Descripción :	Obtención de los atributos propios en una instancia particular de la clase Docente.
Pre-condición:	Actor con privilegios para acceder a la consulta.
Curso Normal	
<ol style="list-style-type: none"> 1. El actor ingresa criterio de búsqueda del docente. 2. El buscador localiza el docente correspondiente al criterio de búsqueda en el almacén de datos (D.P.D.I). 3. El sistema devuelve una instancia de la clase Docente contentivo del resultado de la búsqueda. 4. El menú del sistema muestra opciones para ampliar el alcance de la búsqueda. 	
Curso Alterno	
<ol style="list-style-type: none"> 2. El buscador NO localiza el docente correspondiente al criterio de búsqueda en el almacén de datos. 3. El sistema devuelve un mensaje al actor informando el resultado anterior. 	
Post-condición:	El actor dispone de los datos (0..1) correspondientes a la búsqueda y menú de opciones que permite obtener más información.

Tabla N° 19. Caso de uso: Consultar Docente.

Fuente: autor de la investigación.

Ref.:	UC-1-1
Caso de Uso:	Consultar cambios de Dedicación.
Descripción :	Obtención del conjunto de cambios de Dedicación del docente en el tiempo.
Pre-condición:	UC-1
Curso Normal	
<ol style="list-style-type: none"> 1. El usuario activa la opción correspondiente a “ver cambios de Dedicación”. 2. El buscador localiza el set de cambios de Dedicación correspondientes al docente en el almacén de datos (D.P.D.I). 3. El sistema devuelve el set de instancias de los cambios de dedicación del Docente. 	
Curso Alterno	
<ol style="list-style-type: none"> 2. El buscador NO localiza cambios de dedicación del docente en el almacén de datos. 3. El sistema devuelve mensaje correspondiente al resultado anterior. 	
Post-condición:	El actor dispone del set de datos (0..N) correspondiente a los cambios de dedicación del docente.

Tabla N° 20. Caso de uso: Consultar Cambios de Dedicación.

Fuente: autor de la investigación.

Ref.:	UC-1-2
Caso de Uso:	Consultar experiencia laboral.
Descripción :	Obtención de conjunto de datos (0...N) correspondientes a la experiencia laboral docente intra y extra universitaria.
Pre-condición:	UC-1.
Curso Normal	
<ol style="list-style-type: none"> 1. El usuario activa la opción correspondiente a “ver experiencia laboral”. 2. El buscador localiza el set de experiencia laboral correspondientes al docente en el almacén de datos (D.P.D.I). 3. El sistema devuelve el set de instancias de experiencia laboral del Docente. 	
Curso Alterno	
<ol style="list-style-type: none"> 2. El buscador NO localiza experiencia laboral del docente en el almacén de datos. 3. El sistema devuelve mensaje correspondiente al resultado anterior. 	
Post-condición:	El actor dispone del set de datos (0..N) correspondiente a la experiencia laboral del docente.

Tabla N° 21. Caso de uso: Consultar experiencia laboral.

Fuente: el autor de la investigación.

Ref.:	UC-1-3
Caso de Uso:	Consultar ascensos.
Descripción :	Obtención de conjunto de datos correspondientes a los ascensos del docente a lo largo de su carrera académica.
Pre-condición:	UC-1.
Curso Normal	
<ol style="list-style-type: none"> 1. El usuario activa la opción correspondiente a “ver ascensos”. 2. El buscador localiza el set de ascensos correspondientes al docente en el almacén de datos (D.P.D.I). 3. El sistema devuelve el set de instancias de ascensos del Docente. 	
Curso Alterno	
<ol style="list-style-type: none"> 2. El buscador NO localiza ascensos del docente en el almacén de datos. 3. El sistema devuelve mensaje correspondiente al resultado anterior. 	
Post-condición:	El actor dispone del set de datos (0...N) correspondiente a los ascensos del docente.

Tabla N° 22. Caso de uso: Consultar ascensos.

Fuentes: autor de la investigación.

Ref.:	UC-1-4
Caso de Uso:	Consultar contratos.
Descripción :	Obtención de conjunto de datos correspondientes a los contratos del docente a lo largo de su carrera académica.
Pre-condición:	UC-1.
Curso Normal	
<ol style="list-style-type: none"> 1. El usuario activa la opción correspondiente a “ver Contratos”. 2. El buscador localiza el set de contratos correspondientes al docente en el almacén de datos (D.P.D.I). 3. El sistema devuelve el set de instancias de contratos del Docente. 	
Curso Alterno	
<ol style="list-style-type: none"> 2. El buscador NO localiza contratos del docente en el almacén de datos. 3. El sistema devuelve mensaje correspondiente al resultado anterior. 	
Post-condición:	El actor dispone del set de datos (0...N) correspondiente a los contratos del docente.

Tabla N° 23. Caso de uso: Consultar contratos.

Fuente: autor de la investigación.

Ref.:	UC-2
Caso de Uso:	Actualizar datos personales.
Descripción :	Actualización de la información personal del docente.
Pre-condición:	Docente con privilegios para acceder a su cuenta.
Curso Normal	
<ol style="list-style-type: none"> 1. El usuario activa la opción correspondiente a “actualizar datos personales”. 2. El sistema despliega atributos modificables del docente. 3. El docente modifica sus datos y guarda los cambios. 4. El sistema ejecuta la transacción de actualización. 5. El sistema devuelve el mensaje de actualización satisfactoria. 	
Curso Alterno	
<ol style="list-style-type: none"> 4. El sistema NO ejecuta la transacción de actualización. 5. El sistema devuelve mensaje correspondiente al resultado anterior. 	
Post-condición:	El docente dispone de herramientas necesarias para la actualización de sus datos personales.

Tabla N° 24. Caso de uso: Actualizar datos personales.

Fuente: autor de la investigación.

Ref.:	UC-3
Caso de Uso:	Gestionar solvencias.
Descripción :	Solicitud de solvencias de Formación, Investigación y Extensión del Docente.
Pre-condición:	Docente en ejercicio.
Curso Normal	
<ol style="list-style-type: none"> 1. El docente solicita su solvencia en las distintas instancias correspondientes: <ol style="list-style-type: none"> a. Formación. b. Investigación. c. Extensión. 2. La unidad organizacional ejecuta su caso de uso generar solvencia. 3. El docente solvente retira de las unidades anteriores el resultado del caso de uso anterior, para cada unidad organizacional. 	
Curso Alterno	
<ol style="list-style-type: none"> 3. El docente NO esta solvente en la unidad organizacional. 	
Post-condición:	El docente obtiene resultados actualizados de su estatus en cada unidad organizacional.

Tabla N° 25. Caso de uso: Gestionar solvencias.

Fuente: autor de la investigación.

Ref.:	UC-4
Caso de Uso:	Gestionar constancias
Descripción :	Solicitud de constancias a Unidad Organizacional
Pre-condición:	Docente en ejercicio.
Curso Normal	
<ol style="list-style-type: none"> 1. El docente solicita su constancia a la D.P.D.I. 2. La unidad organizacional ejecuta su caso de uso generar constancia. 3. El docente retira de la unidad el documento solicitado. 	
Curso Alterno	
<ol style="list-style-type: none"> 	
Post-condición:	El docente obtiene constancia de la unidad organizacional.

Tabla N° 26. Caso de uso: Gestionar Constancias.

Fuente: autor de la investigación.

Ref.:	UC-5
Caso de Uso:	Administrar Unidad Organizacional.
Descripción :	Administración de la información referente a la actividad docente propia de cada unidad organizacional.
Pre-condición:	Usuario con privilegios de administración en la unidad organizacional.
Curso Normal	
<ol style="list-style-type: none"> 1. El usuario activa el caso de uso “Consultar docente” (UC-1). 2. Si la consulta devuelve resultado, el sistema despliega información del docente y menú de opciones de administración. 3. El usuario selecciona la opción de administración a utilizar. <ul style="list-style-type: none"> • Registrar – Modificar – Eliminar. 4. El usuario confirma la operación anterior en el control correspondiente. 5. El sistema ejecuta la transacción en el almacén de datos de: <ul style="list-style-type: none"> • D.F.P.A en caso de administración de actividades de formación. • C.D.C.H.T en caso de administración de actividades de investigación. • D.E.U en caso de administración de actividades de extensión. 6. El sistema devuelve el mensaje de transacción satisfactoria. 	
Curso Alterno	
<ol style="list-style-type: none"> 2. La consulta no devuelve resultados. 5. El sistema NO ejecuta la transacción en el almacén de datos. 6. El sistema devuelve mensaje correspondiente al resultado anterior. 	
Post-condición:	El usuario realiza la administración de la unidad organizacional.

Tabla N° 27. Caso de uso: Administrar unidad organizacional.

Fuente: autor de la investigación.

Ref.:	UC-5-1
Caso de Uso:	Consultar Docente en Unidad Organizacional.
Descripción :	Obtención de conjunto de datos correspondientes a la actividad del docente en la unidad organizacional a lo largo de su carrera.
Pre-condición:	UC-1.
Curso Normal	
<ol style="list-style-type: none"> 1. El usuario activa la opción correspondiente a “ver registros” para la unidad organizacional. 2. El buscador localiza el set de registros en el almacén de datos correspondiente a: <ul style="list-style-type: none"> • D.F.P.A en caso de administración de actividades de formación. • C.D.C.H.T en caso de administración de actividades de investigación. • D.E.U en caso de administración de actividades de extensión. 3. El sistema devuelve el set de instancias correspondientes a la actividad del docente en la unidad académica particular. 	
Curso Alterno	
<ol style="list-style-type: none"> 2. El buscador NO localiza registros de actividad del docente en la unidad académica particular. 3. El sistema devuelve mensaje correspondiente al resultado anterior. 	
Post-condición:	El usuario dispone del set de datos (0...N) correspondiente a la actividad del docente propia de cada unidad organizacional del docente.

Tabla N° 28. Caso de uso: Consultar docente en Unidad Organizacional.

Fuente: autor de la investigación.

Ref.:	UC-5-2
Caso de Uso:	Generar solvencia de unidad organizacional.
Descripción :	Obtención de status actual en cuanto a solvencia académica de D.F.P.A.
Pre-condición:	UC-1.
Curso Normal	
<ol style="list-style-type: none"> 1. El docente activa el caso de uso “Gestionar Solvencias” (UC-3) en cada unidad organizacional. 2. El usuario receptor de solicitudes, activa el caso de uso (UC-1). 3. El usuario gestor consulta según las reglas de negocio particulares de su unidad organizacional, el estatus actual del docente. 4. El usuario emite el resultado de lo anterior al docente solicitante. 	
Curso Alterno	
Post-condición:	El docente solicitante obtiene su estatus actual en la unidad organizacional pudiendo estar: SOLVENTE O INSOLVENTE.
Observaciones:	<p>Es importante resaltar que para la consulta de solvencia se evalúan dos aspectos:</p> <ul style="list-style-type: none"> • Académico – Administrativo. <p>Siendo el segundo NO tomado en cuenta ya que compete al Vicerrectorado Administrativo lo cual está fuera del alcance planteado para la presente investigación.</p>

*Tabla N° 29. Caso de uso: Generar solvencia en unidad organizacional.
Fuente: autor de la investigación.*

Diagramas de Objetivos

Iteración 1

En correspondencia con lo planteado por Kavakli (2006) y Teruel (2011), con referencia al refinamiento de los objetivos estratégicos a operacionales y ajustados a los pasos definidos del Modelo planteado, a continuación se presenta un diagrama superficial de objetivos de la organización:

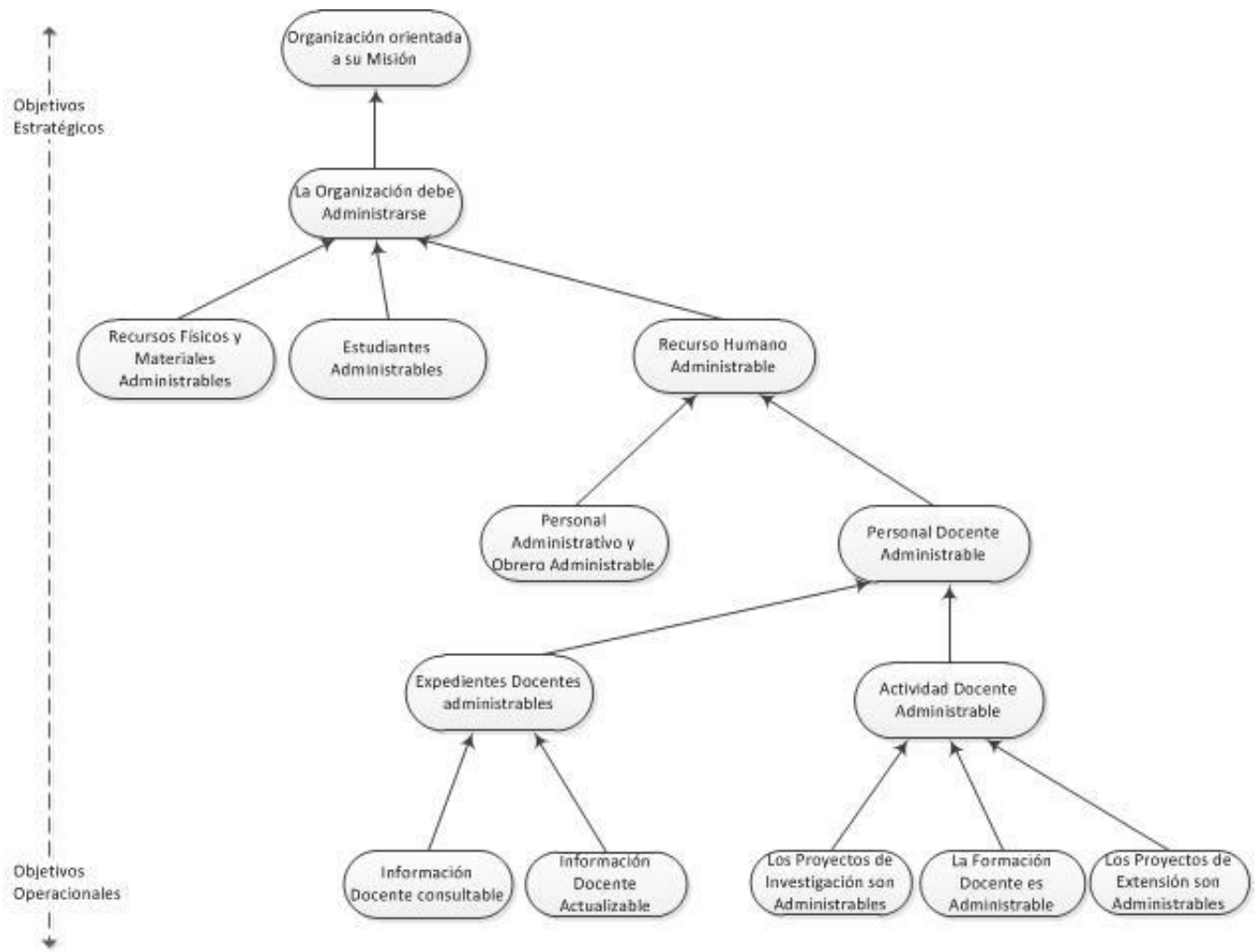


Figura N° 27. Diagrama de Objetivos Organizacionales.
Fuente: autor de la investigación.

De lo anterior, es destacable aclarar que en función del alcance de la presente investigación se tomarán exclusivamente los procesos orientados a satisfacer la administración docente, a saber administración de expedientes docente, y actividades docentes de investigación, formación y extensión, en este particular en la siguiente iteración se analizan las unidades organizacionales en función de sus objetivos, tareas y usuarios involucrados.

Iteración 2

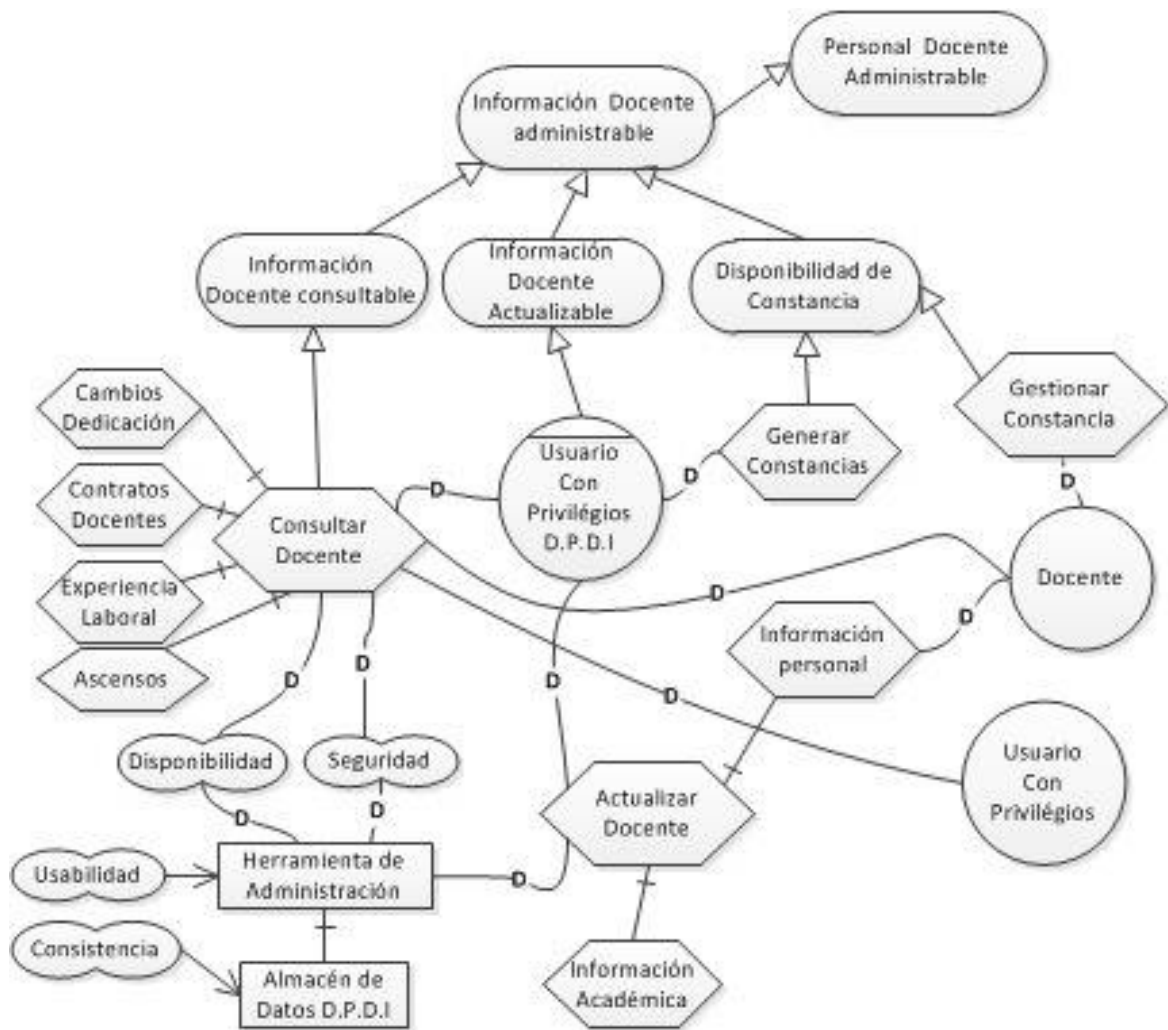


Figura N° 28. Diagrama Objetivos-Tareas-Usuarios en D.P.D.I

Fuente: autor de la investigación

Destacando de lo anterior, para satisfacer los objetivos de Consulta y Actualización docente, los procesos Consultar Docente, Actualizar Docente, con sus procesos internos involucrados, activados por los usuarios Docente, Usuario con privilegios de la unidad organizacional(Director, administrador, operador), respectivamente.

Así, y continuando con el proceso se analizan algunas unidades constitutivas de la unidad organizacional UO-01 (Vicerrectorado Académico) y su interacción con el diagrama anterior en los procesos de consulta docente administrados por D.P.D.I.

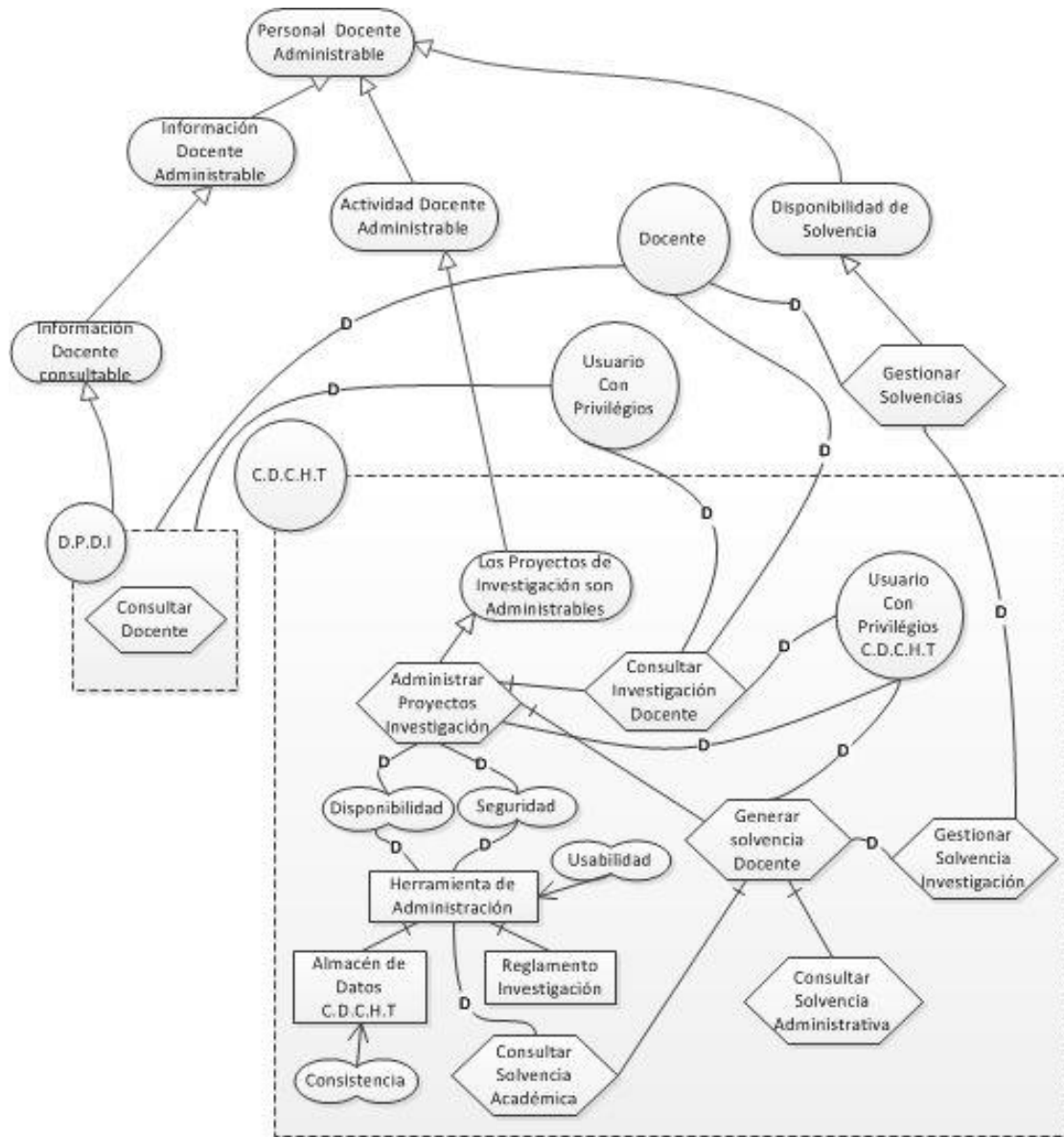


Figura N° 29. Diagrama Objetivos-Tareas-Recursos en C.D.C.H.T
Fuente: autor de la investigación.

Ilustrando lo anterior las tareas procesos y recursos propios de la satisfacción de objetivo(s) de las actividades de investigación docente.

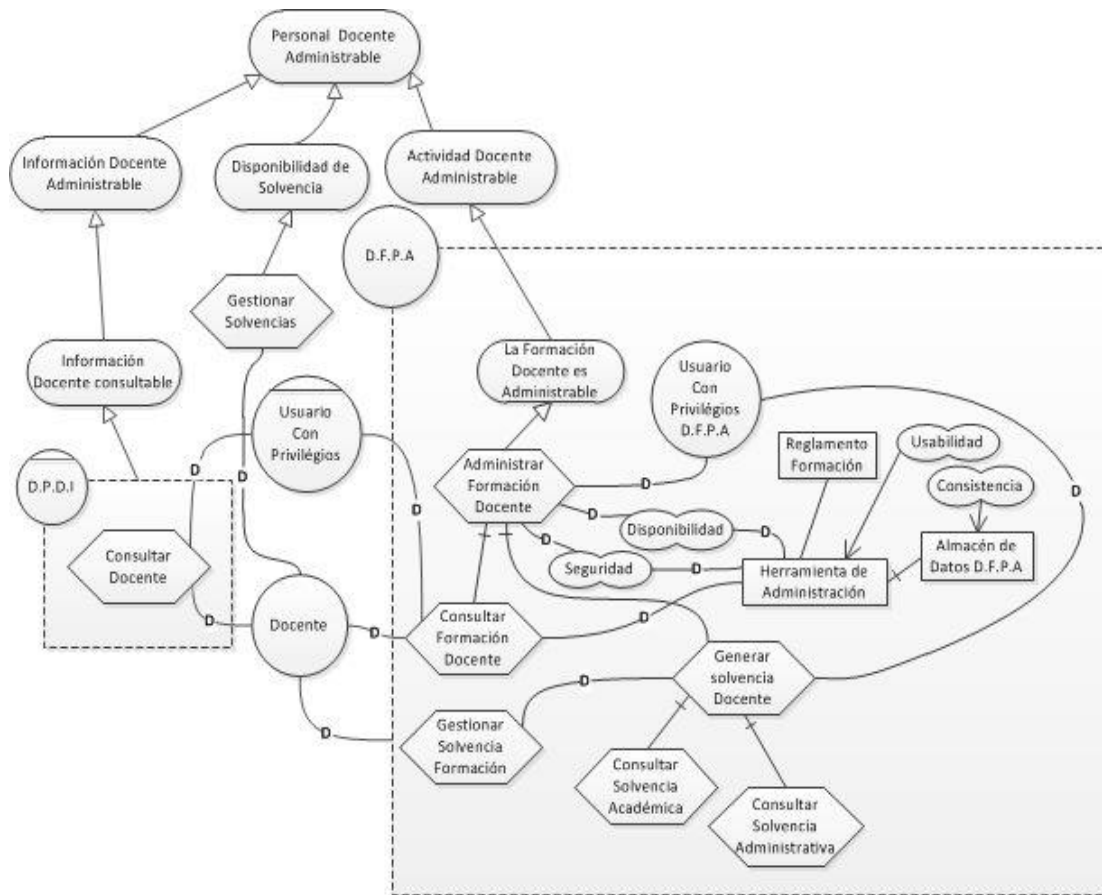


Figura N° 30. Diagrama Objetivos-Tareas-Recursos D.F.P.A
Fuente: autor de la investigación.

Ilustrando lo anterior las tareas, procesos y recursos propios de la satisfacción de objetivo(s) de las actividades de formación docente.

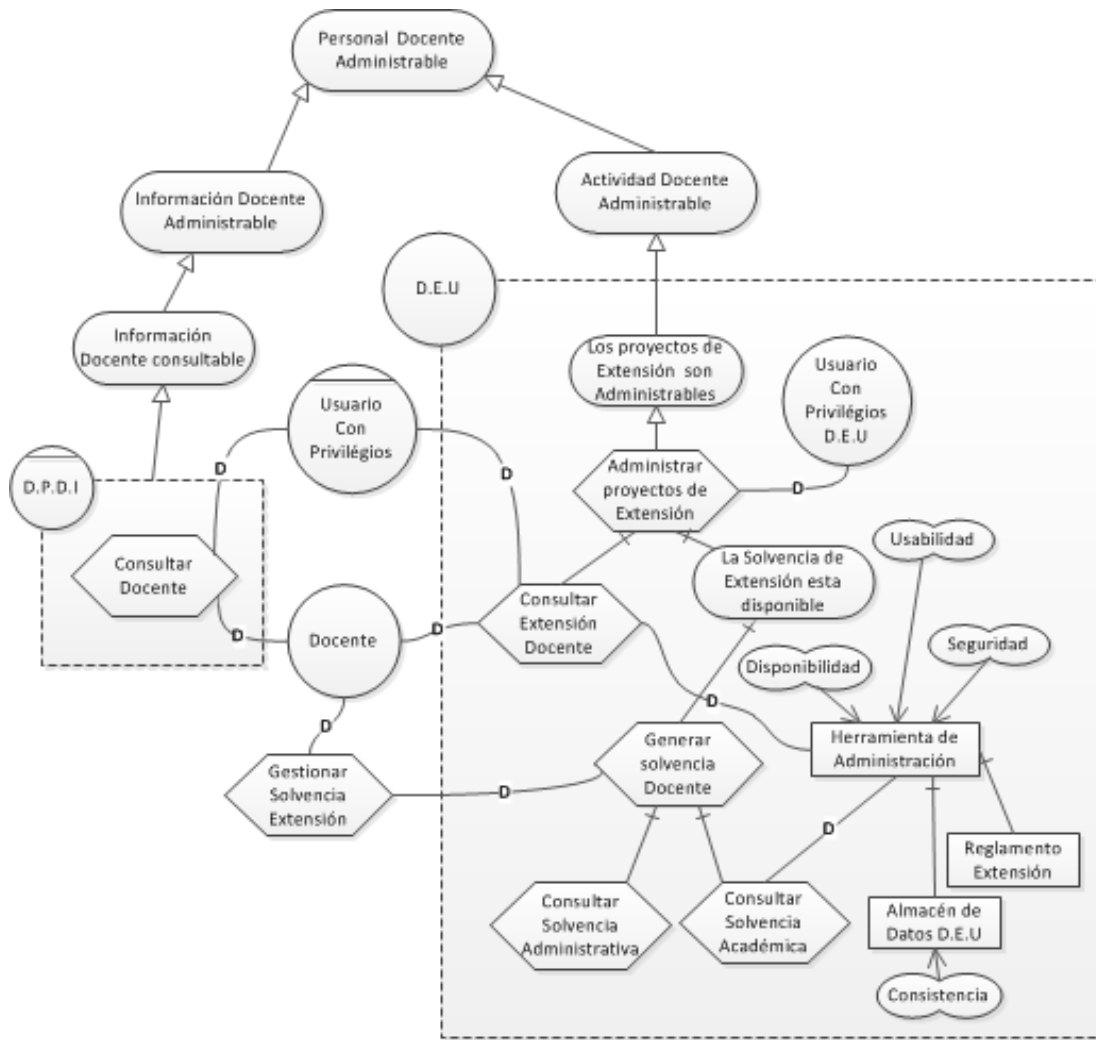


Figura N° 31. Diagrama Objetivos-Tareas-Recursos D.E.U
Fuente: autor de la investigación.

Ilustrando lo anterior las tareas, procesos y recursos propios de la satisfacción de objetivo(s) de las actividades de extensión en que participan docentes.

Iteración 3

A continuación a modo de resumen se presenta el diagrama de objetivos integrado que involucra las tareas, procesos y actores que participan en su satisfacción, contenido de las unidades organizacionales (UO-01, UO-02) antes estudiadas:

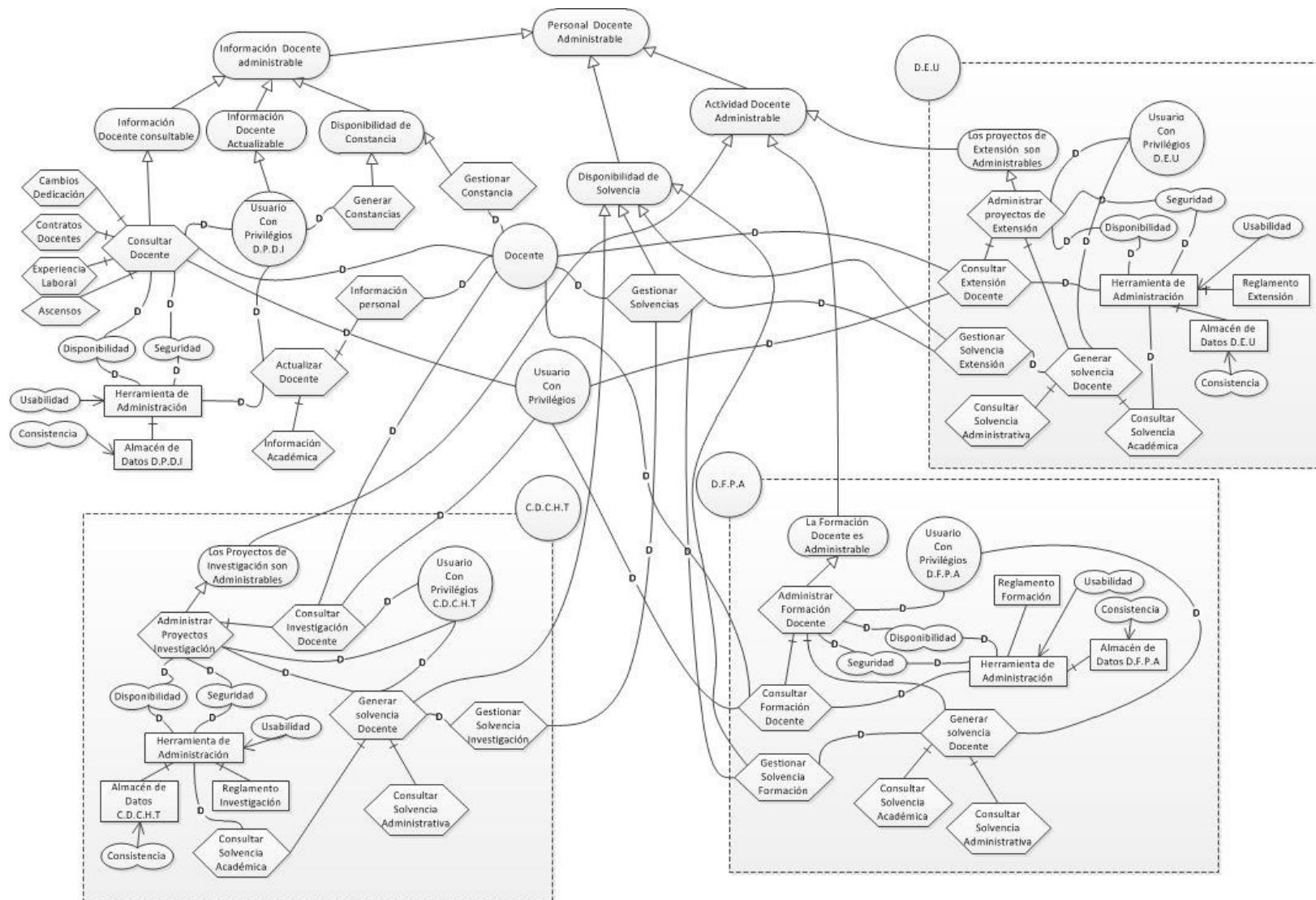


Figura N° 32. Diagrama de Objetivos – Tareas- Recursos Vista General.

Fuente: autor de la investigación.

Servicios.

Descubrimiento

En la presente etapa y según lo descrito en el diseño del modelo, corresponde realizar las actividades inherentes a la identificación de servicios que posibilitaran insumos necesarios para el proceso de integración de la organización, a tal fin, y en correspondencia con lo descrito en la Tabla N° 14 se procede al análisis de los productos anteriores en función de elaborar el conjunto de servicios candidatos.

Iteración 1

Identificación de servicios candidatos: Tomando en cuenta los diagramas generales de Objetivos y casos de uso ilustrados en pasos anteriores (ver Figuras N° 26-31), y su descripción de casos de uso (ver Tablas N° 19-29) se identifican objetivos principales entre unidades organizacionales, los cuales están satisfechos por una serie de tareas o procesos manteniendo la trazabilidad objetivos – tareas, estas a su vez son efectuadas por actores propios de las UO estudiadas y otros externos que fungen como solicitantes según el caso.

De lo anterior, se puede inferir que en principio es fundamental definir estas cadenas de responsabilidad permitiendo a cada unidad efectuar estrictamente “aquello que le corresponde”, por ejemplo en correspondencia a la satisfacción del objetivo *Docente Consultable*, según su contexto dicha tarea es realizada por la UO-2 en cuanto a información personal e histórico de movimientos, y por las presentes en la UO-1 en cuanto a lo referente a las actividades propias del docente para cada unidad, por tal razón a este nivel se toman todas las tareas expuestas en cada unidad organizacional vinculadas estrictamente a su origen, resultando a continuación el conjunto de tareas o casos de uso efectuadas por cada unidad organizacional particular:

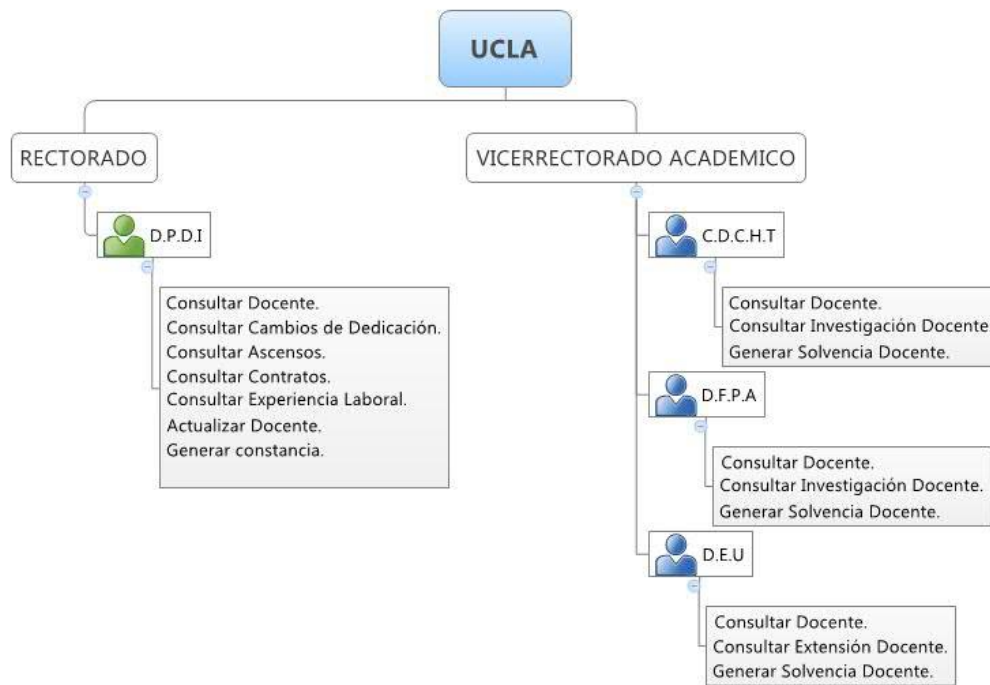


Figura N° 33. Diagrama de Procesos de Unidades Organizacionales.
Fuente: autor de la investigación.

Iteración 2

Elaborar Catálogo de Servicios: en este paso, tomando en cuenta lo planteado por Earl (2005) en Valera (ob.cit) en que las tareas principales servirán a priori como servicios candidatos y extrapolando esta premisa a la orientación a objetivos, se puede inferir que la identificación de objetivos principales, puede ilustrar las combinaciones naturales de tareas o servicios candidatos con la finalidad de alcanzar la ya descrita granularidad de servicios, pudiéndose ilustrar con la siguiente figura:

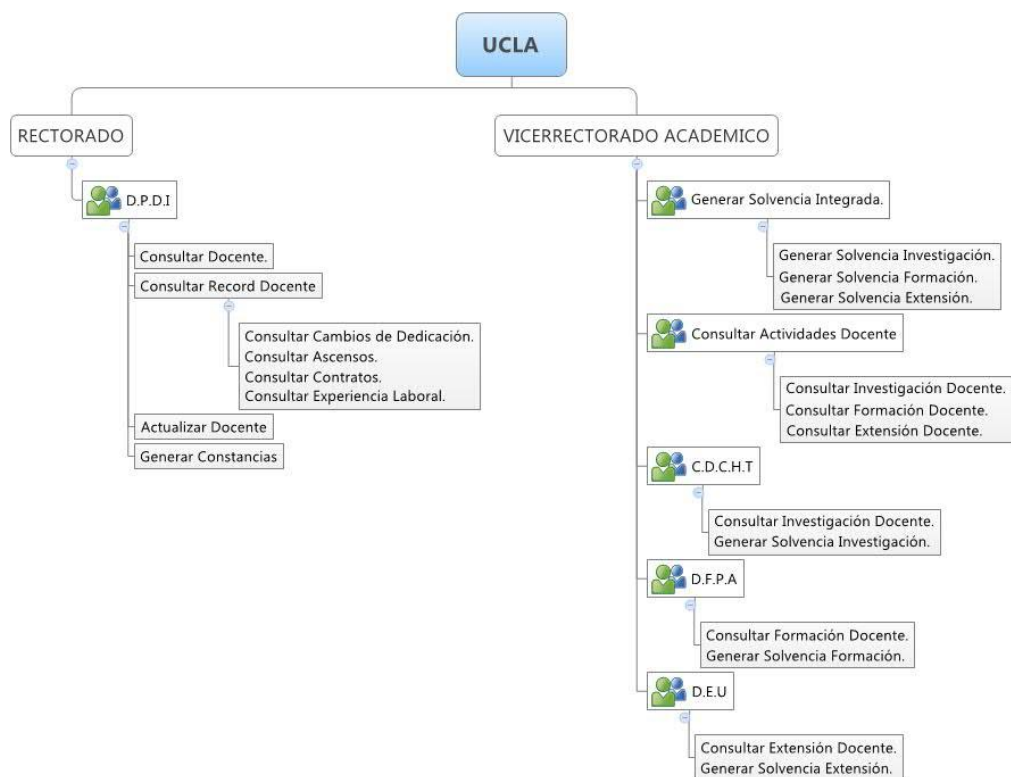


Figura N° 34. Composición de Servicios Candidatos.

Fuente: autor de la investigación.

Desprendiéndose de lo anterior, la especialización de los servicios y su composición, en cuanto al primer aspecto, las tareas inherentes a *Consultar Docente*, si bien son ejecutadas por distintos actores previamente identificados, incluso los del dominio de las unidades adscritas a V.R.A, pertenecen a D.P.D.I por tal razón dicho servicio estará a cargo de esta unidad que brindara no solo el acceso a los datos como hasta ahora vía DBAT u Batch (niveles de integración 1-0) sino ejecutara el proceso observando atributos de calidad propios de los SoftGoals involucrados(ver Figura N° 27).

En cuanto a composición de servicios, las tareas correspondientes a consultas de cambios de dedicación, ascensos, contratos y experiencia laboral, se pueden componer en un solo servicio que puede ser denominado *Consultar Record Docente*, al igual las actividades docentes de Investigación, Formación y Extensión, pueden ser

compuestas en el servicios *Consultar Actividades Docente*, así mismo el *Servicio Generar Solvencia Integrada* estará compuesto por los servicios de solvencia propios de cada unidad organizacional que se plantean también como servicios individuales (a cargo de cada unidad) ya que según la dinámica del negocio estudiada, para algunos procesos internos se requieren juntos o por separado en distintos escenarios.

Así, y en correspondencia con lo planteado en el modelo, corresponde elaborar el artefacto final de esta etapa, a saber Catálogo de Servicios en el formato definido (ver Tabla N° 16), lo cual se expone a continuación:

Objetivo	Interacciones			Servicio
	Actores	Unidades Involucradas	Casos de uso	
Información Docente Consultable	<ul style="list-style-type: none"> • Docente. • Usuario con Privilegios. • Usuarios (U.O). 	<ul style="list-style-type: none"> • D.P.D.I 	<ul style="list-style-type: none"> • Consultar Docente. 	S-1: Consultar Docente.
			<ul style="list-style-type: none"> • Consultar cambios de Dedicación • Consultar Ascensos. • Consultar Contratos. • Consultar Experiencia Laboral. 	S-2: Consultar Record Docente.
Información Docente Consultable	<ul style="list-style-type: none"> • Docente. • Usuario con Privilegios. • Usuarios (U.O). 	<ul style="list-style-type: none"> • D.F.P.A • C.D.C.H.T • D.E.U 	<ul style="list-style-type: none"> • Consultar Docente en unidad organizacional: <ul style="list-style-type: none"> ▪ Investigación. ▪ Extensión. ▪ Formación. 	S-3: Consultar Actividades de Formación Docente.
				S-4: Consultar Actividades de Investigación Docente.
				S-5: Consultar Actividades de Extensión Docente.

				S-6: Consultar Actividades Docente.
Información Docente Actualizable.	<ul style="list-style-type: none"> • Docente. • Usuarios con privilegios (D.P.D.I) 	<ul style="list-style-type: none"> • D.P.D.I 	<ul style="list-style-type: none"> • Actualizar Datos Personales 	S-7: Actualizar Datos Docente.
Generar Constancias.			<ul style="list-style-type: none"> • Generar Constancias 	S-8: Generar Constancias
Generar Solvencia.	<ul style="list-style-type: none"> • Docente. • Usuarios con privilegios (U.O) 	<ul style="list-style-type: none"> • D.F.P.A • C.D.C.H.T • D.E.U 	<ul style="list-style-type: none"> • Generar Solvencia de Unidad Organizacional. 	S-9: Generar Solvencia Formación.
				S-10: Generar Solvencia Investigación
				S-11: Generar Solvencia Extensión.
				S-12: Generar Solvencia Integrada.

Tabla N° 30. Catálogo de Servicios (caso de estudio).

Fuente: autor de la investigación.

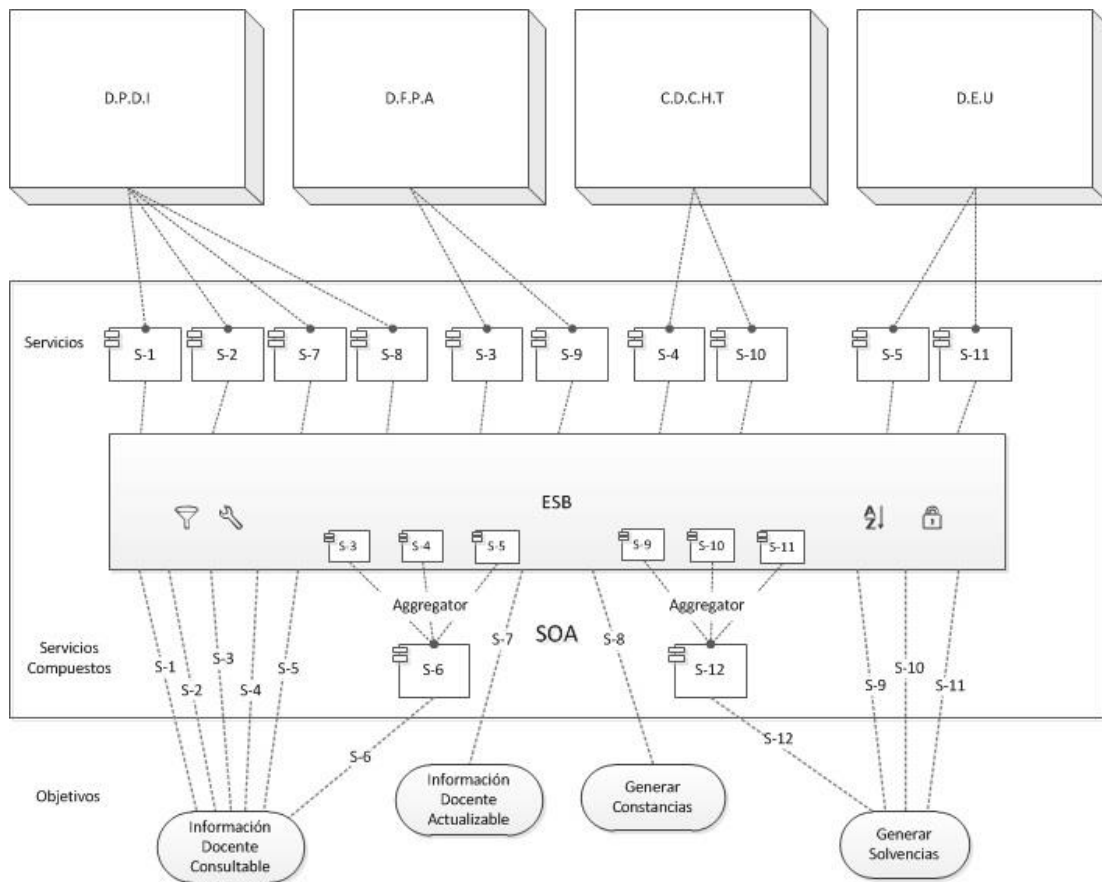
Despliegue.

Con el catálogo de servicios generado en la fase anterior, se procede a la definición arquitectural que posibilite la integración, en este caso según lo anteriormente descrito se efectúa por medio de una arquitectura orientada a servicios, implementando un bus de servicios empresariales basado en mensajes, lo anterior como resultado sensato del estudio previo de los estilos de integración y en consecuencia de estos, los patrones de mensajería (ver Tablas N° 7,8 y 9).

Por lo anteriormente expuesto es conveniente aclarar, algunos de los patrones presentes en el bus de servicios planteado, referenciándose algunos a continuación sin orden jerárquico particular:

- Message Bus: define el bus de servicios.
- Content Base Router: direccionamiento y selección de ruta.
- Aggregator: combina mensajes para procesarlos como un todo.
- Content Enricher: agrega partes faltantes para consolidar el mensaje.

Así, con los elementos tecnológicos descritos se procede finalmente a la elaboración del diagrama que resume la plataforma de integración planteada, en esta se encontraran desplegados los servicios elicitados mediante la combinación de orientación a objetivos y casos de uso, en un entorno distribuido, escalable y cuyo principal fin es la integración mediante componentes independientes, que garanticen la reusabilidad y por ende posibiliten la integración por medio de la compartición de recursos sea esta como simples consumidores o como compositores de servicios, a continuación se expone el diagrama que resume lo anterior:



*Figura N° 35. Despliegue de Servicios en propuesta arquitectural.
Fuente: autor de la investigación.*

Por último, pero no menos importante es fundamental aclarar los servicios como entes independientes, cuya combinación es posible dentro del concepto de reusabilidad deseado, por tal razón el catálogo de servicios actualmente elicitados obedecen a la satisfacción de los objetivos definidos, mas pueden y (en el contexto de lo deseable) deben ser utilizados de ahora en más como parte de conglomerados de servicios en función de la satisfacción de objetivos futuros, solo así el concepto de SOA deja de ser algo abstracto y quimérico, se convierte en una realidad palpable para la organización.

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

El presente trabajo de investigación, cuyo desarrollo se fundamenta en la búsqueda de alternativas para mejorar los procesos de comunicación organizacional definido como integración, se elaboró en función de las etapas tempranas de este proceso, a saber elicitación de combinaciones de datos y procesos a ser traducidos en servicios, en función de lo cual se estudiaron alternativas de descubrimiento, en cuya combinación efectiva pudo ser identificado aquello que podría ser convertido a servicios, con su consecuente funcionalidad orientándose principalmente a la reusabilidad. Por tal razón, y posterior a los procesos rigurosos propios de la elaboración de un proyecto de investigación de esta modalidad, se llega a una serie de conclusiones a ser descritas a continuación.

Conclusiones

- Un estudio como el planteado se realiza en torno a un tema de investigación, evaluando su pertinencia y valor al área de estudio; por tal razón, para el proceso de elaboración o desarrollo, se requieren enfoques metodológicos que orienten los esfuerzos en función del logro de los objetivos, surgiendo así metodologías ágiles entre las cuales se encuentra: eXtreme Programing, definiéndose como un proceso iterativo e incremental que permite entre otras cosas, el logro de los objetivos con economía de recursos y adaptabilidad a los cambios, enriqueciéndose lo anterior, con la aplicación de la Ingeniería de Métodos como disciplina, que aporta rigurosidad en la obtención de los productos resultantes de los procesos efectuados en cada etapa, añadiendo valor agregado al resultado obtenido.

- Así, y en consonancia con lo planteado como problema en el capítulo I, en la presente investigación se evidenció que el dinamismo propio de las organizaciones, ha condicionado en el tiempo el desarrollo de soluciones de forma descoordinada, generando algunos inconvenientes de comunicación que atentan contra la sinergia organizacional; en consecuencia, las organizaciones y sus unidades constitutivas deben buscar puntos de encuentro con la finalidad de lograr ante todo la economía de recursos y un buen punto de partida es compartir.
- Sin embargo, compartir no se limita a datos, es importante permitir la especialización de algunas tareas por medio de la definición de cadenas de responsabilidad, que permita a cada quien “hacer lo que mejor sabe hacer” y compartir estos procesos y resultados.
- Un mecanismo poco atendido a la fecha, consiste en modelar las organizaciones y los procesos que en ellas se ejecutan en torno a sus objetivos, definiendo así tareas y participantes para su satisfacción, siendo además definidos en función de la razón de ser organizacional y en sentido secuencial, discriminados hasta los niveles cercanos a funciones procedimentales u operativas, resueltas tradicionalmente como casos de uso, planteándose su combinación como principal clarificador de aquellos servicios que surgirán como consecuencia de su necesidad de satisfacción.
- Por lo antes expuesto, se utilizaron herramientas de diagramado tales como i* que destaca por su riqueza gráfica y adaptabilidad, así como UML para el diagramado de casos de uso en función de la elicitación de procesos y actores que interactúan en torno a la satisfacción de objetivos organizacionales.
- En congruencia a todo lo anterior, se elaboró un modelo de integración Orientado a Objetivos e Ingeniería de Métodos que presenta por su carácter genérico características que lo hacen aplicable a diversidad de contextos organizacionales.

- Así, en algunos pasos se llega al descubrimiento de un conjunto de funcionalidades traducibles a servicios, contentivas de atributos de calidad propios de la Ingeniería de Software, que permiten establecer los bloques operativos necesarios para lograr la integración organizacional.
- Finalmente se establece la arquitectura de software necesaria para la correcta implementación de las funcionalidades antes descritas, contentiva de elementos tecnológicos de calidad comprobada como los Enterprise Service Bus (ESB), superando las limitaciones propias de enfoques anteriores.
- Todo lo anterior, se aplicó a un caso de estudio en el que se ejemplificó el modelo planteado obteniendo los resultados esperados.

Por otra parte, en torno al proceso de elaboración de la presente investigación se identificaron algunas particularidades y/o necesidades que de momento excedían los límites definidos de la presente, por tal razón se exponen como recomendaciones.

Recomendaciones.

- Promover que el proceso integrador se desarrolle como un fenómeno viral que contagie a todos por la elocuencia de sus beneficios.
- Plantear el presente aporte metodológico como un clarificador de servicios en contextos de organizacionales heterogéneos.
- Aplicar los resultados de la presente investigación y su aporte en el caso de estudio en su contexto operativo, con la finalidad de colocar la primera piedra en el proceso integrador institucional.
- De los enfoques de modelado GO estudiados, destacándose para la presente i* y en congruencia con lo planteado por Teruel y otros (2011), ninguno destaca la importancia de los requisitos, lo cual debe ser considerado en función del desarrollo de una extensión o mejora a estas herramientas.
- Plantear la combinación de metodologías y tecnologías descritas en el desarrollo y formalización de un Framework que permita sumar resultados y beneficios.

BIBLIOGRAFIA

- Aalbers, H. (s.f). Introducción a SOA (II). [Documento en Linea] Disponible en:
[http://www.huibert-aalbers.com/IT_Insight/Spanish/PDF/ITI005Sp-SOA\(II\).pdf](http://www.huibert-aalbers.com/IT_Insight/Spanish/PDF/ITI005Sp-SOA(II).pdf).
[Consultado, Abril 2013].
- Arias, E. (2012). Análisis de Calidad en Enterprise Service Bus, propietarios y de. [Documento en Linea] Disponible en
:http://sedici.unlp.edu.ar/bitstream/handle/10915/18879/Documento_completo.pdf?sequence=1. [Consultado, Abril 2013].
- Arias, F. (2006). *El Proyecto de Investigación. Introducción a la metodología científica.(5a.ed.)*. Caracas, Venezuela: Episteme.
- Avila, R. (2001). *Metodología de la Investigación*. Lima, Peru: Estudios y Ediciones R.A.
- Balestrini, M. (2006). *Como se elabora el proyecto de Investigación (7a.ed.)*. Caracas, Venezuela: Consultores Asociados OBL.
- Bazán, P. (2009). Un modelo de integrabilidad con SOA y BPM. *Tesis de grado presentado para optar al título de Magister Scientiarum en Redes de Datos*. Buenos Aires, Argentina: Universidad Nacional de la Plata.
- Boccalari, C. (2010). Las tecnologías SOA y ESB como herramientas integradoras para el acceso unificado a servicios colaborativos heterogeneos. [Documento en linea] Disponible en :
<http://sedici.unlp.edu.ar/bitstream/handle/10915/4024/Presentaci%C3%B3n.pdf?sequence=1>. [Consultado, Marzo 2013].
- Brinkkemper, S. (1996). Method engineering: engineering of information systems development methods and tools. University of Twente . Information and Software Technology. [Articulo en Linea] Disponible en :
<http://doc.utwente.nl/18012/1/Brinkkemper96method.pdf>. [Consultado, Mayo 2013].
- Buschmann, F., Henney, K., & Schimdt, D. (2007). *Pattern-Oriented Software Architecture: On Patterns And Pattern Language, Volumen 5*. Wiley & Sons.

- Caponi, M., Rodriguez, P., & Zamudio, P. (2008). Mensajería en Sistemas de Información. *Trabajo de grado de Licenciatura, Universidad de la Republica, Facultad de Ingenieria*. Montevideo, Uruguay.
- Chiesa, F. (2004). Metodología para selección de sistemas ERP. *[Documento en Linea]* Disponible en: <http://www.ucla.edu.ve/dac/departamentos/informatica-ii/metodologia-para-seleccion-de-sistemas-erp.PDF>. [Consultado, Marzo 2013].
- De la Vara, J. L., Sanchez, J., & Pastor, O. (2013). On the Use of Goal Models and Business Process Models for Elicitation of System Requirements. *[Documento en Linea]* Disponible en: http://link.springer.com/chapter/10.1007/978-3-642-38484-4_13. Valencia, España: [Consultado: Agosto, 2013].
- De Pablos Heredero, C., López-Hermoso Agius, J. J., & Martín-Romo Romero, S. (2011). Organización y transformación de los sistemas de información en la empresa. *[Libro en Linea]* Disponible en : <http://books.google.co.ve/books?id=2pqwKkqxxosC&printsec=frontcover&hl=es#v=onepage&q&f=false>. [Consultado: Agosto, 2013].
- Espina, P. (2007). Extensibilidad de UDDI. *[Documento en Linea]* Disponible en: [http://www.lsi.us.es/docs/doctorado/memorias/Pedro%20Espina%20\(Periodo%20Investigacion\).pdf](http://www.lsi.us.es/docs/doctorado/memorias/Pedro%20Espina%20(Periodo%20Investigacion).pdf). Andalucía, España: [Consultado, Marzo 2013].
- Estrada Esquivel, H. (2008). A service-oriented approach for the i* framework. *[Documento en Linea]* Disponible en : <http://riunet.upv.es/handle/10251/3305>. Valencia, España: [Consultado: Agosto, 2013].
- Harmsen, A. (. (1997). Situational Method Engineering. University of Twente. *[Articulo en Linea]* Disponible en: [http://eprints.eemcs.utwente.nl/17266/01/af_harmsen\[1\].pdf](http://eprints.eemcs.utwente.nl/17266/01/af_harmsen[1].pdf). [Consutado, Mayo 2013].
- Hitpass, B. (2013). *BPM, Fundamentos y conceptos de integración*. Santiago de Chile: BHH-Ltda.
- Hohpe, G., & Woolf, B. (2003). *Enterprise Integration Patterns*. Boston, Estados Unidos: Addison-Wesley.
- Huhns, M. N. (2005). *Service-Oriented Computing: Key Concepts and Principles*. University of South Carolina Scholar Commons.
- Inmon, B. (2012). Bill Inmon el padre del Data Warehousing (DWH) en México. *[Texto en Linea]* Disponible en : <http://www.intellego.com.mx/es/noticias/bill-inmon-el-padre-del-data-warehousing-dwh-en-mexico>. [Consultado, Marzo 2013].

- Iribarren, O. (2012). Modelo de Mejora Continua para la gestión de Requisitos usando Métodos Ágiles. *Trabajo de grado presentado para optar al título de Magister Scientiarum en Ciencias de la Computación*. Barquisimeto, Lara: Universidad Centroccidental Lisandro Alvarado.
- Junco, P. (2008). EAI World. Patrones de Integración. [Texto en Línea] Disponible en : <http://blogs.technet.com/b/pablojunco/archive/2008/05/21/patrones-de-integracion.aspx>. [Consultado, Marzo 2013].
- Kavakli, E., & Loucopoulos, P. (2006). Experiences With Goal-Oriented Modeling of Organizational Change. *IEEE Transactions On Systems, Man, And Cybernetics Part C: Applications And Reviews Vol. 36, No. 2*. [Consultado: Agosto, 2013].
- Keen, M., Acharya, A., Bishop, S., Hopkins, A., Milinski, S., Nott, C., y otros. (2004). *Patterns: Implementing an SOA Using an Enterprise Service Bus*. IBM RedBooks.
- Kendall, K., & Kendall, J. (2005). *Análisis y Diseño de Sistemas (6a.ed.)*. Mexico: Pearson Education.
- Konrad, F. (2005). Arquitectura Orientada a Servicios(SOA). [Documento en Línea] Disponible en : http://ldc.usb.ve/~yudith/docencia/UCV/Exposiciones/SOA_UCVV0.2.pdf. [Consultado, Febrero 2013].
- Kumar, K., & Hillegersberg, J. (2000). Enterprise resource planning: Introduction. Communications of the ACM. [Texto en Línea] Disponible en : <http://0-delivery.acm.org.millenium.itesm.mx/10.1145/340000/332063/p22-kumar.pdf?key1=332063&key2=8889879211&coll=portal&dl=ACM&CFID=55975134&CFTOKEN=20158783>. [Consultado, Marzo 2013].
- Letier, E., & Van Lamsweerde, A. (2002). Agent-Based Tactics for Goal-Oriented Requirements Elaboration. [Documento en Línea] Disponible en : <http://www.info.ucl.ac.be/Research/Publication/2002/lcse02.pdf>. [Consultado, Abril 2013].
- Lewis, G. (2011). Getting Started with Service-Oriented Architecture (SOA) Terminology. [Documento en Línea]. Disponible en : http://www.sei.cmu.edu/library/assets/whitepapers/SOA_Terminology.pdf. [Consultado, Abril 2013].
- López, J. (2011). Propuesta de Integración de Sistemas de Información basado en ROA. Caso: UNEXPO. *Trabajo de grado presentado para optar al título de Magister Scientiarum en Ciencias de la Computación*. Barquisimeto, Lara: Universidad Centroccidental Lisandro Alvarado.

- Markovic, I., & Kowalkiewicz, M. (2008). Linking Business Goals to Process Models in Semantic Business. [Documento en Linea] Disponible en : <http://140.118.1.157/cht/Research%20Pro/Business%20Process%20Management/Linking%20business%20goals%20to%20process%20models%20in%20semantic%20business%20process%20modeling.pdf>. [Consultado, Abril 2013].
- Mendez, C. (2011). Arquitectura Orientada a Servicios para la interoperabilidad con equipos médicos. Caso de estudio laboratorio clínico Mascia. *Trabajo de grado presentado para optar al título de Magister Scientiarum en Ciencias de la Computación*. Barquisimeto, Lara: Universidad Centroccidental Lisandro Alvarado [Consultado: Marzo 2013].
- Mendoza, L. (s.f.). Sistemas de Información III. [Documento en Linea] Disponible en : [http://prof.usb.ve/lmendoza/Documentos/PS-6117%20\(Teor%EDa\)/Teor%EDa%20PS6117%20Integraci%F3n%20de%20Sistemas.pdf](http://prof.usb.ve/lmendoza/Documentos/PS-6117%20(Teor%EDa)/Teor%EDa%20PS6117%20Integraci%F3n%20de%20Sistemas.pdf). [Consultado, Marzo 2013].
- Papazoglou, M., & Den Heuvel, W. (2006). *Service-oriented design and development methodology*. *Int. J of Web Engineering and Technology (IJWET)* 2(4). 412 - 442.
- Pérez, M., Mendoza, L. E., & Carvajal, Y. (2003). *Orientaciones para la Selección de Tecnologías de Integración de Sistemas de Software. III Workshop de Ingeniería de Software*. Chillán, Chile.
- Pressman, R. (2010). *Ingeniería del Software, un enfoque práctico*. (7a.ed). New York, USA: McGraw-Hill.
- Quiroga, M. (2011). Estudio de arquitecturas de redes orientadas a servicios. *Proyecto final de carrera*. Barcelona, España: Universidad Politecnica de Catalunya.
- Raian, A. (2010). Modeling and Reasoning about Contextual Requirements: Goal-based Framework. *PhD Dissertation*. University of Trento Department of Information Engineering and Computer Science.
- Ramos, J. A. (2012). Primeros Pasos con Mule ESB. [Documento en línea] Disponible en : http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=primeros_pasos_mule. [Consultado, Agosto 2013].
- Rolland, C., & Salinesi, C. (2005). Modeling Goals and Reasoning with them. [Documento en Linea] Disponible en: <https://www.google.co.ve/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0CD8QFjAB&url=http%3A%2F%2Fepi.univ->

paris1.fr%2Fservlet%2Fcom.univ.collaboratif.utils.LectureFichiergw%3FCODE_FICHIER%3D1289202608241%26ID_FICHE%3D20. [Consultado, Abril 2013].

- Ruiz, M. (2010). Generación automática de servicios web a partir de modelos conceptuales. *Memoria para optar al grado de Doctor en Informática*. Valencia, España.: Universidad Politécnica de Valencia.
- Sommerville, I. (2005). *Ingeniería del Software (7a.ed.)*. Madrid, España: Pearson Education.
- Tamayo, M. (1999). *El Proyecto de Investigación*. Bogotá, Colombia: ICFES.
- Teruel, M. A., Navarro, E., López-Jaquero, V., Montero, F., & González, P. (2011). A comparative of Goal-Oriented approaches to modelling requirements for collaborative systems. *Technical Report # DIAB-11-03-1*. Universidad de Castilla La Mancha.
- UPEL. (2010). Universidad Pedagógica Experimental Libertador. *Manual de trabajo de grado y tesis doctorales*. Caracas, Venezuela: Fondo editorial de la Universidad Pedagógica Experimental Libertador.
- Valera, E. (2011). Construcción de un motor de búsqueda de contenidos en repositorios confiables, basado en Crawlers, enmarcado en una arquitectura orientada a servicio. *Trabajo de grado presentado para optar al título de Magister Scientiarum en Ciencias de la Computación*. Barquisimeto, Lara: Universidad Centroccidental Lisandro Alvarado.
- Van Lamsweerde, A. (2009). *Requirements Engineering - From System Goals to UML Models to Software Specifications*. Wiley 2009.
- Vázquez, J. (2010). ¿Por qué un Enterprise Service Bus (ESB)? [Texto en Línea] Disponible en : <http://blogs.tecsisa.com/articulos-tecnicos/por-que-un-enterprise-service-bus/>. [Consultado, Marzo 2013].
- Welicki, L. (2013). Patrones y Antipatrones: una Introducción - Parte I. [Texto en Línea] Disponible en : <http://msdn.microsoft.com/es-es/library/bb972242.aspx>. [Consultado: Marzo, 2013].